

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

UMI[®]



Université d'Ottawa • University of Ottawa

**Topoclimatic modeling of
summer surface air temperature
in the Canadian Arctic Archipelago**

David E. Atkinson

Department of Geography
University of Ottawa

A dissertation submitted to the school of Graduate and PostDoctoral Studies
in partial fulfillment for the degree of
Doctor of Philosophy in Geography
August 25th, 2000



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file *Votre référence*

Our file *Notre référence*

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-58261-2

Canada

A b s t r a c t

In the Canadian High Arctic general patterns of temperature are poorly resolved at the meso-scale. This project addressed this issue in three stages.

In the first stage a data set of non-standard weather observations was assembled and quality controlled. The data set possessed approximately 58000 observations, including dry-bulb temperature, wind, visibility and cloud cover, from the spring and summer seasons of the years 1974–1993. Up to 10% of the data were unusable due to erroneous station information.

The second part of the project consisted of a principal components analysis (PCA) of daily temperature data in the Canadian Arctic Archipelago (CAA). The PCA 1) demonstrated how the timing and extent of synoptic events could be tracked, 2) identified the major regional controls of temperature in the CAA, and 3) showed that the non-standard data exhibited general coherency with regional patterns yet were able to reveal zones of coherency at the meso-scale in temperature patterns.

In the third stage of the project a model to estimate surface air temperature at the meso-scale was constructed. It was based on a 1 km resolution digital elevation model of the CAA. The effects on temperature due to site elevation and coastal proximity were selected for parameterization. The change in temperature with elevation was implemented in the model using derived environmental lapse rates. Advection effects were handled using resultant winds combined with air temperature above the ocean. Lapse rates and resultant wind estimates were obtained from upper air ascents. Model results for 14-day runs were compared to observed data. Residuals ($n=385$) possessed a mean absolute error of 1.5°C . The model was sensitive to steep surface inversions and to low-level warming. Sensitivity analyses were performed on the model to determine response to alterations in lapse rate calculation, sea surface temperature, and wind field generation. The model was most sensitive to lapse rate calculation. The lowest mean absolute error (0.2) was obtained using a moderate lapse rate calculation, moderate wind field and variable sea-surface temperature.

Acknowledgements

Ten years ago several important things happened to me. I got married. I met a very good friend, Paul Wolfe. And I went to the High Arctic for the first time. These things changed my life and set me on a path that has led here. But I will follow the thread a bit farther back. There are many people who have helped me to move along this path in one way or another, and it began the summer before 1990.

I owe a lot to the Geological Survey of Canada and the summer student work program operated by the Canadian federal government. In the summer of 1989 I went to work in the lab of Dr. Bob Mott, a palynologist, who introduced me to the workaday aspects of paleoenvironmental reconstruction. It was with Bob, Hélène Jetté, and Sue Wilson (then Prinzen, now with a PhD of her own from Queen's and with whom I will be sharing a post-doc office!) that I first did field work. After the field season I also had the privilege to work the rest of the summer with Dr. Roger McNeely, director of the GSC Radiocarbon Dating Laboratory. Both then, and during another summer work-term in 1995, in working with Roger I was exposed to one of the highest standards of scientific endeavour I have encountered. I continue to value Roger's friendship and example today.

Later, in February of 1990, an unexpected thing happened to me during the GSC Public Forum. Dr. Sylvia Edlund, an arctic botanist and a colleague of Roger and Bob in the Terrain Sciences Division of the GSC, asked what I was doing that summer, and whether the work for my Master's, begun in September of 1989, was of a nature that would benefit from a summer field season in the High Arctic. My original Master's project had nothing to do with the Arctic, but was only in the initial stages, so I told her Of Course! I was incredibly happy to have a chance to get up there. Before the summer work began, Sylvia introduced me to her good friend and colleague, Dr. Bea Alt, a glacier meteorologist and arctic climatologist. By the time the summer had come to a close, I was completely in love with the arctic. As well, the process had been initiated that would lead eventually to this document.

I owe a lot to Sylvia and her dedication to the arctic, her research, and to her stewardship of emerging arctic researchers. The opportunity she offered me in the spring of 1990 literally placed me on the path of my working career. She approached her work with patience and thoroughness, and it diminishes arctic research that she has retired and moved on to other pursuits. If and when I ever run my own arctic camps, I will adopt her non-generator approach – clean, non-mechanized, and above all, *quiet*, as it should be in the High Arctic.

And then there is Bea. If Sylvia faced me in the right direction, it was Bea who took my hand and began walking with me. The earliest manifestation of this project was a desire to organize, once and for all, the Polar Continental Shelf Project weather observation data base, and also to continue on and figure out ways to employ it to meaningfully augment arctic climate understanding. To that end I worked with Bea and the Glaciology section in the summers of 1991 – 1994. Much of our work concerned the PCSP data base, both archived and on-going data gathering. Bea also arranged for me to work for a month at the Meteorology Service of Canada (formerly Atmospheric Environment Service) main data archive unit in Downsview, Ontario. From my time at AES I must list Tom Agnew, Barrie Maxwell, and Brian Peters, and more so Dave Etkin and the irrepressible Angus Headly for all the assistance they lent (Angus. why would you trade Barbados for 3 years at

Mould Bay?). I had to learn their mainframe and TSO system pretty quickly! To this day I still include JCL on my resumé.

In three of those four summers with Glaciology Bea sent me to the High Arctic, to assist in the process of gathering and recording data, to get to know the sites whenever possible, and to get to know the people – researchers and support staff. And thus I must offer my thanks to many of the highly competent people working at the Polar Continental Shelf Project. The director of PCSP, Bonni Hrycyk, never had any problem letting me stay at the PCSP base in Resolute Bay or using PCSP resources in Ottawa. The people with whom I had the most direct working contact were the base managers: Dave Maloley, Jim Godden, Barry Hough, Claude Brunet, and Jerry Maceachern. These guys gave me the freedom I needed to do what was required. And then there were the PCSP representatives from The Rock of Newfoundland: The Benois, George and Wally, Rick Reed and Tim Norman. When you needed something, they knew how to do it/get it. The pilots from Canadian Helicopters and Ken Borek Aviation were always great and never minded me coming along.

As the data work progressed we were also turning our attention to ways in which these data could be used in a climatology. During my introductory GIS course with the very supportive Dr. Graeme Bonham-Carter I put together a preliminary idea of how GIS might be used in conjunction with the PCSP data. This appealed enormously to Konrad and Bea and we began to run with it. After much evolution it has arrived in the form that follows. Bea has continued to be an unswerving, creative and incredibly energetic and optimistic ally in this project, and I hope it is something she is satisfied with as a new chapter in arctic climatology. Furthermore, regarding the PCSP data base, I hope this work gives her a sense of closure on a project she initiated with determination and considerable vision almost 30 years ago. I am very pleased to be now closely involved with her on several new projects, and to be able to continue working with her.

Another thing Sylvia did for me was to recommend someone I might wish to consider as a supervisor for a PhD: Prof. Konrad Gajewski, at the time recently arrived at the University of Ottawa. I went to see him – and found someone who is very smart and possessed of a wide breadth of experience, who is available to his students like no one else, and who is completely committed to his students *and their financial support*. Above all, though, I found someone who believed in the value and potential of the PCSP data, who was interested in new approaches to their incorporation, and who was simply completely enthusiastic about the project. Konrad is also fundamentally patient, such that even after I took off for two years in the middle of the PhD on a misguided business foray, he was waiting when I came back without question "We've got 2 and a half years – Let's go!". Konrad is also very pragmatic and has a strong sense of how to go about succeeding in the research environment – publishing, research grants, conferences – that he conveys to us at every opportunity. This is an invaluable, indeed essential, yet untaught aspect the professional researcher must deal with, and I am grateful that he is as committed to our success as he is to his own. He tackles all that he does with passion and energy – during the time I have known him he has built up several lab areas complete with equipment for computing and paleoecological work. He now has a number of graduate students who are working on very diverse and demanding projects, all of which are moving out into uncharted methodological terrain. Despite the diversity, though, he shifts gears amongst us with a breathtaking ease, and displays his competence time and again with continued insightful observations about our work. I was fortunate to have Prof. Gajewski as my supervisor, and I will consider myself lucky to have continued opportunities to work with him. I will carry his motto with me: "Don't waste your time!".

I also wish to acknowledge the members of my thesis committee. It was not always easy – the concepts sometimes proved difficult to get into a functional state – but my board remained open-minded and their comments were always constructive and intelligent. They included the following: Prof. Antoni Lewkowicz, who can be a fearsome taskmaster, but who, despite that front, is an excellent and meticulous research scientist, and someone who really cares about arctic research. Prof. Peter Johnson, a leader in the arctic research community for years and someone who can ask some interesting questions! Prof. Daniel Lagarec, challenging my math in his thoughtful manner, and Prof. André Dabrowski, a professor of mathematics and statistics, who patiently put up with my lack of mathematical sophistication. I would also like to thank my external reviewer, Prof. Wayne Rouse of McMaster University, for his positive and very complimentary review of my thesis.

Elsewhere in the school environment, there are Konrad's other PhD students: Dr. Jocelyne Bourgeois, who beat me to the finish fair and square, and Mike Sawada and André Viau, with whom I have shared the lab now for three years. We have had a lot of good talks, learned from and helped each other technically in a freely giving manner that I know I will miss, and I am continually impressed with their dedication and knowledge of computers and of the theoretical underpinnings of the methods they are programming. These guys will go far. Mike also read through an early version of Chapter 3 – a significant undertaking, he will recall. I would also like to thank my good friend Bert Catt for SQL expertise – he saved me a lot of frustration.

My family. As I mentioned, many things happened in the summer of 1990 – the most important of which was my marriage to my wife, Catherine. She is not simply a friendly face to return to, someone to share my less-busy time with, someone to raise our children with, she is also a model of academic excellence a match for which I have yet to encounter. She has a quick intelligence that makes all conversations meaningful, even those I might consider 'shop'. And she has been all-supportive in this project, giving time whenever it was necessary, especially the onerous summer of 1999. I love her and I thank her for her patience. And then there are my kids! Kenny was born during the academic year I began the PhD; in a sense, his life has defined my PhD tenure. He is happy and very bright, has an irrepressible energy and yet can focus for hours – I love his outlook. Emma, the younger, is at once sweet in friendship yet fearsome when crossed – I love her spirit. They show me the future, and move me along when my own spirit flags. And my parents! Most people feel a general debt to their parents, but in my case they have between them a combination of creativity and strong work ethic, not to mention a pervasive honour and integrity in their approach to people and life, setting an example that has taken me and my three siblings far.

Finally, I would like to return to my friend, Paul Wolfe. I first met him at Sylvia's camp in the High Arctic, Hot Weather Creek, in 1990. It was apparent that we were 'kindred spirits', especially in terms of our love of the arctic and interest in physical sciences. In 1993 I spent three of the best weeks I have known at his glacier camp, helping him set up equipment for his Master's work. I was honoured to be his best man at his wedding in 1994 to Maria Shantz. After his Master's he began the PhD program at the University of Alberta, under Prof. Martin Sharpe. I was looking forward to the day when we might work together in the same university department. Unfortunately it was not to be; he left too soon. Instead now I reflect upon the example he set, and I dedicate the spirit of this arctic research to his memory.

Table of Contents

ABSTRACT	II
ACKNOWLEDGEMENTS.....	III
TABLE OF CONTENTS.....	VI
APPENDICES	VIII
LIST OF FIGURES	IX
LIST OF TABLES	XV
CHAPTER 1: INTRODUCTION - SURFACE AIR TEMPERATURE IN THE HIGH ARCTIC.....	1
1.1 BACKGROUND AND JUSTIFICATION	1
1.2 GENERAL LITERATURE REVIEW	6
CHAPTER 2: A NEW DATABASE OF HIGH ARCTIC CLIMATE DATA FROM THE POLAR CONTINENTAL SHELF PROJECT ARCHIVES	10
2.1 INTRODUCTION	10
2.2 HISTORY OF THE DATA SET	12
2.2.1 <i>1974–1983.....</i>	<i>15</i>
2.2.2 <i>1984–1986.....</i>	<i>16</i>
2.2.3 <i>1987–1993.....</i>	<i>17</i>
2.2.4 <i>Station information and original records.....</i>	<i>17</i>
2.3 METHODS	17
2.3.1 <i>Reading and processing the original data files.....</i>	<i>17</i>
2.3.2 <i>Error checking</i>	<i>20</i>
2.4 DATA SET SUMMARY	31
2.5 USES OF THE DATA.....	34
2.6 CONCLUSION	35

CHAPTER 3: A PRINCIPAL COMPONENTS ANALYSIS OF THE JULY TEMPERATURE CLIMATE OF THE CANADIAN ARCTIC ARCHIPELAGO	37
3.1 INTRODUCTION	37
3.2 METHOD	44
3.2.1 <i>Analyses</i>	44
3.2.2 <i>Principal components analysis: methodological review</i>	40
3.3 RESULTS	47
3.3.1 <i>Comparing PCA and traditional synoptic analyses</i>	47
3.3.2 <i>July PCA analyses – AES data only</i>	53
3.3.3 <i>July PCA – AES and PCSP data</i>	55
3.4 DISCUSSION	57
CHAPTER 4: A MODEL TO ESTIMATE SURFACE AIR TEMPERATURES IN THE CANADIAN ARCTIC ARCHIPELAGO	59
4.1 INTRODUCTION	59
4.2 MODEL DESCRIPTION AND OPERATION	63
4.2.1 <i>Initial estimate of surface air temperature</i>	65
4.2.2 <i>Coastal advection</i>	70
4.2.2 <i>Sensitivity analysis</i>	75
4.3 RESULTS	77
4.3.1 <i>Run periods</i>	78
4.3.1 <i>Surface air temperature estimates</i>	79
4.3.2 <i>Residual analysis, original model runs</i>	79
4.3.2 <i>Residual analysis, sensitivity runs</i>	85
4.4 DISCUSSION	88
4.4.1 <i>Original model</i>	89
4.4.2 <i>Sensitivity analysis</i>	95
4.5 FUTURE WORK	97

CHAPTER 5 SUMMARY 100
REFERENCES..... 105

APPENDICES

A: PCSP database statistics, original formats, and computer code 175
B: SAS package to conduct, plot, and map the principal components analyses 223
C: SAS modules for topoclimate model 247

List of Figures

	page
Figure 2.1: The Canadian Arctic Archipelago. Stations of the Atmospheric Environment Service of Canada (AES) are indicated. AES stations with upper air facilities are underlined.	112
Figure 2.2a: Comparisons of AES and PCSP mean daily temperature, summer 1977: Plots of selected stations. Vertical dashed lines indicate the beginning of the month. Horizontal dashed lines are placed for reference at +5°C, 0°C, and -5°C.	113
Figure 2.2b: Comparisons of AES and PCSP mean daily temperature, summer 1977: locations of stations presented in the comparison. AES stations are indicated with a diamond, PCSP stations with a dot.	114
Figure 2.3a: Comparisons of AES and PCSP mean daily temperature, summer 1979: Plots of selected stations. Vertical dashed lines indicate the beginning of the month. Horizontal dashed lines are placed for reference at +5°C, 0°C, and -5°C.	115
Figure 2.3b: Comparisons of AES and PCSP mean daily temperature, summer 1979: locations of stations presented in the comparison. AES stations are indicated with a diamond, PCSP stations with a dot.	116
Figure 2.4a: Comparisons of AES and PCSP mean daily temperature, summer 1981: Plots of selected stations. Vertical dashed lines indicate the beginning of the month. Horizontal dashed lines are placed for reference at +5°C, 0°C, and -5°C.	117
Figure 2.4b: Comparisons of AES and PCSP mean daily temperature, summer 1981: locations of stations presented in the comparison. AES stations are indicated with a diamond, PCSP stations with a dot.	118
Figure 2.5a: PCSP station densities using 100 km search radius. This pattern is indicative of logistical constraints, such as aircraft operating distances from the Resolute Bay – Eureka corridor.	119
Figure 2.5b: PCSP station densities using 50 km search radius. This pattern reflects some of the most active research areas over the 1974 – 1993 period.	120
Figure 2.6: Number of stations in the PCSP database per year, broken down by duration.	121
Figure 2.7: Breakdown of PCSP stations by distance from coast (km).	122
Figure 2.8: Breakdown of PCSP stations by elevation (m).	122

Figure 2.9:	Mean and median number of observations per station, PCSP data set.	123
Figure 2.10:	July mean temperature normals, 1941-1970 (after Maxwell, 1980). Solid lines are 0 and 5°C isotherms , dashed lines are 2.5° C isotherms.	124
Figure 3.1:	The Canadian Arctic Archipelago. Stations of the Atmospheric Environment Service of Canada (AES) are indicated. AES stations with upper air facilities are underlined.	125
Figure 3.2:	Record lengths (daily mean temperature) from AES stations in the Canadian Arctic Archipelago. Starred columns indicate stations with one or more years of missing data.	126
Figure 3.3:	Principal components analysis results for a five-month period (a) and a 12-month period (b). Note the amount of variance that is explained by the first components. Performed on mean daily temperature data.	127
Figure 3.4a-l:	Daily precipitation totals and mean daily temperature data for the period June 1 – August 31, 1962. Stations are indicated on the plots. Vertical dashed lines indicate the beginning of a month; horizontal dashed line is a reference line placed at 0°C.	128
Figure 3.5a-h:	Principal components analysis results for summer, 1962. a), c), e), g) and i): Maps of component loadings. b), d), f), h) and j): Time series plots of component scores. k): Plots of mean temperature for the period. l): Plot showing percent explained variance by each component. Vertical lines in time series plots indicate the beginning of a month; horizontal lines are reference lines placed at component score values of +1, 0, and –1.	129
Figure 3.6a-h:	Daily precipitation totals and mean daily temperature data for the period June 1 – August 31, 1964. Stations are indicated on the plots. Vertical dashed lines indicate the beginning of a month; horizontal dashed line is a reference line placed as 0°C.	131
Figure 3.7a-d:	Principal components analysis results for summer, 1964. a), c), e), g) and i): Maps of component loadings. b), d), f), h) and j): Time series plots of component scores. k): Plots of mean temperature for the period. l): Plot showing percent explained variance by each component. Vertical lines in time series plots indicate the beginning of a month; horizontal lines are reference lines placed at component score values of +1, 0, and –1.	132
Figure 3.8:	Principal components analysis on AES July mean daily data, 1951 – 1974. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.	134

Figure 3.9: Principal components analysis on AES July mean daily data, 1957 – 1977. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.	135
Figure 3.10: Principal components analysis on AES July mean daily data, 1969 – 1985. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.	136
Figure 3.11: Principal components analysis on AES July mean daily data, 1977 – 1988. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.	137
Figure 3.12: Principal components analysis of AES July mean daily data for summer, 1975. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.	138
Figure 3.13: Principal components analysis of AES and PCSP July mean daily data for summer, 1975. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined.	139
Figure 3.14: Principal components analysis of AES July mean daily data for summer, 1983. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.	140
Figure 3.15: Principal components analysis of AES and PCSP July mean daily data for summer, 1983. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined.	141
Figure 3.16: Principal components analysis of AES July mean daily data for summer, 1984. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component 1scores.	142
Figure 3.17: Principal components analysis of AES and PCSP July mean daily data for summer, 1984. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined.	143
Figure 3.18: Principal components analysis of AES July mean daily data for summer, 1990. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.	144
Figure 3.19: Principal components analysis of AES and PCSP July mean daily data for summer, 1990. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined.	145

Figure 3.20: Principal components analysis of AES July mean daily data for summer, 1991. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.	146
Figure 3.21: Principal components analysis of AES and PCSP July mean daily data for summer, 1991. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined.	147
Figure 4.1: The Canadian Arctic Archipelago. Stations of the Atmospheric Environment Service of Canada (AES) are indicated. AES stations with upper air facilities are underlined.	148
Figure 4.2a: Examples of upper air temperature profiles derived from individual rawinsonde ascents at AES Eureka. a): July 02 1987, 0000 UCT; b): July 23 1987, 1200 UCT. Thin line with black dots represents original data points; thicker line is the polynomial fit. c) Mean rawinsonde ascent profiles for July generated by averaging all polynomial estimates for each ascent over the month of July for a given year. Coastal effect is estimated by taking the difference between the surface temperature (solid line) and the extrapolation of the straight portion of the curve (dashed line) to the surface.	149
Figure 4.2b: Comparison showing difference between the three new algorithms for inversion removal introduced for sensitivity testing. The first column shows the "low-slope" removal, the second column shows the "peak-point" removal, and the third column shows no removal. The first row are upper air temperature data from AES Eureka, 16-29 July, 1988, the second row, AES Alert, 29 June-12 July, 1981, the third row, AES Eureka, 9-22 July, 1979. Heavy line represents mean temperature profile curve, light line represents result of removal algorithm.	150
Figure 4.3: Schematic representation depicting how the base temperature estimate is obtained from the polynomial coefficient images and the DEM. Dark pixels indicate the locations of upper air stations on the image. The same coefficient from each station is interpolated to obtain values to form a new image in which each pixel contains an interpolated value for that coefficient. This is repeated for each coefficient. Resultant images are then multiplied with the DEM and these results are totaled, as per the equation.	151
Figure 4.4: The operation of a wind effect filter.	152
Figure 4.5: Sea surface temperatures used in the variable SST sensitivity analysis. Source is Maxwell (1982). Plot represents mean late-summer conditions. Original data come from US Navy data and other sources.	153

Figure 4.6:	Temperature estimates for the period 4 July – 17 July 1976. Values are rounded to the nearest degree Celsius. Ocean is a uniform 2°C. Warmer areas are light, cooler areas are dark. Values range from a maximum of 7°C to a minimum of –13°C.	154
Figure 4.7:	Temperature estimates for the period 16 July – 29 July 1988. Values are rounded to the nearest degree Celsius. Ocean is a uniform 2°C. Warmer areas are light, cooler areas are dark. Values range from a maximum of 13°C to a minimum of –5°C.	155
Figure 4.8:	Temperature estimates for the period 9 July – 22 July 1990. Values are rounded to the nearest degree Celsius. Ocean is a uniform 2°C. Warmer areas are light, cooler areas are dark. Values range from a maximum of 13°C to a minimum of –13°C.	156
Figure 4.9a-p:	Plots of residuals for each of the 14-day model runs using original inversion removal algorithm. Contours are at 6, 4.5, 3, 1.5, -1.5, -3, -4.5, -6; units are degrees Celsius. Heavy shading are zones of negative residual less than –1.5°C (negative residuals); light shading are zones of positive residual more than +1.5°C (positive residuals). Negative residuals occur where observed values are less than 1.5°C less than the model estimate; positive residuals occur where observed values are more than 1.5°C more than the model estimate (see text for elaboration). Periods of the runs are indicated on the plots.	157
Figure 4.10a-f:	Comparisons of residuals plots from 14-day model runs and from runs conducted for the month of July of the same year. Contours are at 6, 3, 1.5, -1.5, -3, -6; units are degrees Celsius. Heavy shading are zones of negative residual less than –1.5°C (negative residual); light shading are zones of positive residual more than +1.5°C (positive residuals). Negative residuals occur where observed values are less than 1.5°C less than the model estimate; positive residuals occur where observed values are more than 1.5°C more than the model estimate. Periods of runs are indicated on the plots.	159
Figure 4.11a-h:	Comparisons between different sensitivity runs for the 1974 run period (9 – 22 July, 1974). Sources of the results is as follows: A - original (high slope algorithm) model, B – low-slope algorithm, C – peak-point algorithm, D – no curve modification, E – low-slope algorithm with variable SST, F – peak-point algorithm with variable SST, G – low-slope algorithm, constant SST and maximum wind, H – low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates).	160
Figure 4.12a-h:	Comparisons between different sensitivity runs for the 1978 run period (21 July – 3 August, 1978). Sources of the results is as follows: A - original (high slope algorithm) model, B – low-slope algorithm, C – peak-point algorithm, D – no curve modification, E – low-slope algorithm with variable SST, F – peak-point algorithm with variable SST, G – low-slope algorithm, constant SST and maximum wind, H – low-slope algorithm, variable SST and max wind. Shading indicates negative	

residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates). 161

Figure 4.13a-h: Comparisons between different sensitivity runs for the 1983 run period (28 June – 11 July, 1983). Sources of the results is as follows: A - original (high slope algorithm) model, B - low-slope algorithm, C - peak-point algorithm, D - no curve modification, E - low-slope algorithm with variable SST, F - peak-point algorithm with variable SST, G - low-slope algorithm, constant SST and maximum wind, H - low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates). 162

Figure 4.14a-h: Comparisons between different sensitivity runs for the 1984 run period (16 – 29 July, 1984). Sources of the results is as follows: A - original (high slope algorithm) model, B - low-slope algorithm, C - peak-point algorithm, D - no curve modification, E - low-slope algorithm with variable SST, F - peak-point algorithm with variable SST, G - low-slope algorithm, constant SST and maximum wind, H - low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates). 163

Figure 4.15a-h: Comparisons between different sensitivity runs for the 1988 run period (16 – 29 July, 1988). Sources of the results is as follows: A - original (high slope algorithm) model, B - low-slope algorithm, C - peak-point algorithm, D - no curve modification, E - low-slope algorithm with variable SST, F - peak-point algorithm with variable SST, G - low-slope algorithm, constant SST and maximum wind, H - low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates). 164

Figure 4.16: Example of net change with introduction of variable SST (Variable SST minus constant SST). Dark shading represent areas in which an increase in temperature was observed, light shading represent areas in which a decrease in temperature was observed. 165

Figure 4.17a-p: Plots of inversion heights at the upper air stations for the periods of the model runs. 'Inversion height' is the distance from the ground to the elevation of maximum temperature on the vertical temperature profile. Elevations are in meters; contour interval is 100 m. 166

Figure 4.18: Mean temperatures for the period 16 July – 29 July 1984. Model parameters include "peak-point" inversion removal, variable SST, and normal wind field. 167

Figure 4.19: Mean temperatures for the period 16 July – 29 July 1988. Model parameters include "peak-point" inversion removal, variable SST, and normal wind field. 168

List of Tables

Table 2.1: Equipment used to gather climatic elements.	14
Table 2.2: Station identifiers used in the different data storage formats.	19
Table 2.3: Number of errors by type.	21
Table 2.4: Percentages of errors and total numbers of observations by year.	22
Table 2.5: Station database 'confirmation' field flag codes.	25
Table 2.6: Observational database 'confirmation' field flag codes.	26
Table 2.7: Results from principal component analyses of July temperature data showing percent variance explained by first three components for stations used in figures 2.2, 2.3 and 2.4.	29
Table 2.8: PCA component loadings for 1977. Loading values range from +1 to -1. A large positive or negative value is indicative of a relationship to the factor.	30
Table 2.9: PCA component loadings for 1979. Loading values range from +1 to -1. A large positive or negative value is indicative of a relationship to the factor.	30
Table 2.10: PCA component loadings for 1981. Loading values range from +1 to -1. A large positive or negative value is indicative of a relationship to the factor.	30
Table 3.1: Time periods covered in principal components analyses of AES data.	45

Table 4.1: Upper air stations used to generate regional estimates of environmental lapse rate. All stations operated by the Atmospheric Environment Service Canada unless located in Alaska.....	66
Table 4.2: Frequency of inversions observed in mean ascent curves for the periods of model runs. Value class is height of the inversion maximum in meters above the ground. GT refers to "greater than 700 meters" (observed only at the Alaska stations).	68
Table 4.3: Wind direction and velocity classification categories.	71
Table 4.4: Nature of alterations to runs for sensitivity analyses.....	77
Table 4.5: Periods for which sensitivity analyses were run.....	77
Table 4.6: Model run dates for 14-day periods.	79
Table 4.7: Additional run periods	79
Table 4.8: Residual totals by residual classification.....	81
Table 4.9: Mean, standard deviation, and sample size of observed values, model estimates, and residuals for each year of model run and for all years.....	81
Table 4.10: Frequency of occurrence of residual zones possessing certain number of stations.....	82
Table 4.11: Frequency of occurrence of residual zones possessing certain number of stations, by residual type.	83
Table 4.12: Residual totals by residual classification.....	85
Table 4.13: Mean, standard deviation, and sample size of observed values, model estimates, and residuals for each year of model run and for all years.....	87

Chapter I

INTRODUCTION:

SURFACE AIR TEMPERATURE IN THE HIGH ARCTIC

1.1 Background and justification

Observations of weather are made daily at thousands of sites around the world. From these data, in large measure, has been built our understanding of climate. This understanding is adequate at the largest spatial scales – hemispheric, continental, regional – yet when climate at the meso-scale, that range between several 10's to several 100's of kilometers, is considered an important weakness in the modern world network of weather observing stations becomes apparent: the density of observing stations is very uneven. For that reason, even today, a detailed understanding of the meso-scale climate of some areas of the world is not available.

The High Arctic of Canada is one such place. This area is an archipelago with islands of varying size and topography, resulting in a complex surface. The presence of perennial or seasonal sea-ice between the islands adds further complexity to the underlying surface. The network of weather stations in the archipelago is sparse. It was designed to capture the major features of the large-scale (1000's of kilometers) climate which, in general, it does – the synoptic climatology (1000-2000 kilometers) for this region is relatively well understood and its portrayal at the continental scale exists in the literature in adequate detail (Bryson and Hare, 1974). However, the paucity and coastal location bias of these stations compromises their capacity to investigate climate at the meso-scale, especially in the summer season, when differences amongst the various juxtaposed surfaces are at their most extreme.

Orvig (1970) sums these ideas up:

Another problem is caused by the paucity of observing stations in the polar regions. A similar number of stations in a continental area would make it impossible to prepare a climatology. ... The details of any analysis of climatic elements in the polar regions, as far as they exist, remain somewhat uncertain. However, it is quite safe to state that the large scale distributions are reasonably well-known, as are the general atmospheric conditions and processes.

- S. Orvig, 1970

Thus when considering features of the temperature regime at scales smaller than the average distance between the stations, which is well within the range of meso-scale consideration for this region, the lack of observing stations is problematic (Cogley and McCann, 1976; Jackson, 1969; King and Turner, 1997). Indicative of this situation is the fact that there are a number of basic, unanswered questions about the climate of this region: What is the extent of the warm zone around the Atmospheric Environment

Service Canada (AES)* station at Eureka? How much farther east and south past AES Resolute Bay does the cool northwest flow off the Arctic Ocean penetrate? What is the nature of the July temperature in the north Lancaster Sound / south Ellesmere Island area? How does the mean temperature on the interior of Banks or Victoria Island compare with their coastal AES stations, Sachs Harbour and Cambridge Bay/Holman? The existing AES network cannot provide answers to such questions.

Processes operating at the synoptic-scale establish the basic climatic characteristics for a given location. Examples of synoptic-scale activity include the pressure situation of the atmosphere, which determines the dominant wind flow regime and which in turn determines the nature of the air mass being driven into the region, or the occurrence of major weather events like cyclonic systems, which can determine whether it is sunny or cloudy over large areas.

Processes operating at the meso-scale, however, act on and modify the climatic characteristics established by synoptic-scale processes. Examples of meso-scale processes include cooling effects due to down-valley (katabatic) cooling caused by proximity to icefields and large glaciers, cooling of warm land surfaces due to cool air advection caused by proximity to the ocean and favorable winds, or temperature changes caused by differences in elevation. Little is known about the extents of influence of such meso-scale processes. In response to this shortcoming, there have been several projects aimed at increasing the meso-scale detail of surface temperature

* As of January 2000 the Atmospheric Environment Service Canada has been renamed and is now known as the Meteorological Service of Canada (MSC).

plots by increasing station density using short-term, historical or automatic weather station data. These will be reviewed in detail later in the document.

This work addresses issues in our lack of knowledge about the meso-scale climate. The general goal is to gain a better understand of the summer, meso-scale temperature climate of the Canadian Arctic Archipelago. More specifically, the challenge is to determine what aspects of the meso-scale exert the most pervasive influence on surface air temperature. With this in mind what are considered to be two of the most prevalent processes acting on temperature at the meso-scale, in terms of the size of the area affected, have been selected for specific attention. The underlying assumption for this project may be stated as follows:

For a given land location, two of the most important factors operating at the meso-scale to modify the synoptic-scale, summer-time temperature are elevation and proximity to the ocean.

Site elevation is a major influence on the surface climate. Distance from the coast determines the strength and frequency with which air advected from the ocean can modify the air temperature regime at the site. This factor is an important consideration in the Canadian Arctic Archipelago (CAA) because the often ice-covered ocean can have surface energy characteristics that are very different from adjacent land surfaces. These two factors modify temperature patterns that have been determined by synoptic-scale processes and form, it is assumed, the two most important meso-scale factors. To evaluate this assumption a model was developed that parameterized the action of these

factors, that is, elevation and distance from the coast, to generate estimates of surface air temperature. Differences between the temperature predicted by the model and observed temperatures – residuals – were used to indicate model shortcomings and to suggest avenues for future work. The development of the model, its results and assessment is presented in detail in Chapter 4.

The observational data used to assess model performance were gathered independently of the data used to develop the model. Temperature data recorded at surface weather stations (as opposed to the upper-air stations) of the Atmospheric Environment Service Canada (AES) were used to gauge model performance. However, because most of these stations are situated on the coast, additional temperature data from the Polar Continental Shelf Project (PCSP) database of non-standard weather observations, some of which are interior stations, were also used.

The PCSP data set is a non-standard set of data that has not been organized or sufficiently quality controlled. 'Non-standard' refers to data that have been gathered by personnel other than those from AES and which have not been subjected to the AES quality control process. Consequently, a first step was to assemble these data and perform some quality control in order to prepare these data for use in climate research. Details of the data base assembly and quality control process is presented in Chapter 2.

To further establish the veracity of the PCSP data, and to pursue the goal of better understanding the meso-scale temperature climatology of the CAA, a principal components analysis (PCA) of surface air temperature was performed. The analysis had

three main goals: assess the ability of PCA to record synoptic-scale events, identify synoptic-scale patterns in air temperature, and demonstrate how PCSP data can capture meso-scale detail in surface temperature patterns. This work is presented in Chapter 3.

Chapters 2, 3, and 4 of this thesis were written as separate manuscripts and the appropriate literature is reviewed within the chapters. Technical details are presented in the Appendices and referred to in the appropriate chapters. In general, the term 'surface air temperature' is understood to refer to measurements of air temperature made near the surface, at a height of approximately 1.5 m. (Shea et al., 1994).

1.2 General literature review

Within climate research efforts that have been conducted at the largest spatial scales, that is, hemispheric or continental, the Canadian High Arctic has often been treated as a single zone and contrasted with other large zones in the circumpolar arctic, or it has been grouped with the rest of the circumpolar arctic and contrasted with temperate, tropical or south polar regions. One example is the analysis by Jones *et al.* (1982) of broad trends and patterns in surface air temperature over the northern hemisphere. A region-specific companion study focussed on patterns of surface air temperature in the circumpolar region (Kelly *et al.*, 1982). Broad patterns of the circumpolar arctic circulation were the subject of several studies, including Serreze and Barry (1988), who investigated synoptic activity over a period of a few years, Walsh and Chapman (1990), who examined circulation variability, and Power and Mysak (1992) who examined patterns and trends in sea-level pressure and linked these to sea ice extent. LeDrew (1988) examined the life cycles of several low pressure systems traversing the

circumpolar basin. Nakamura and Oort (1988) presented a comparative study of atmospheric heat budgets of the north and south polar regions. In all of these studies, the Canadian arctic was considered as one zone, with only broad scale trends in this zone being treated.

At the next smaller scale, the synoptic-scale, there are several studies focussing on the region of the Canadian High Arctic. Alt and Maxwell (submitted) have recently completed a study of the overall arctic climate. Higuchi (1988) studied trends in surface air temperature variability. The spatial pattern of low atmosphere vertical structure across this region was the subject of two studies, one by Kahl *et al.* (1992) and the other by Bradley and Keimig (1992). Bradley and England (1979) performed an objective classification of synoptic weather regimes. For the Canadian arctic, however, no study is as comprehensive as that produced by Maxwell (1980, 1982). In these two large volumes, data summaries were presented in tabular, graphical and mapped form along with comprehensive analyses of all available climatic elements in this region.

At the meso-scale, there are almost no studies available for the Canadian High Arctic. One reason for this is a lack of data. Since there are few stations in the High Arctic, analyses can be conducted only at large (regional, synoptic) and small (local, micro) spatial scales. There are two ways to increase the resolution of analysis at the meso-scale: increase station density or model the factors that influence climate.

There are three exceptions in which investigators attempted direct analyses of meso-scale. In each of these cases resolution was increased by utilizing a greater

density of data points. Each study focussed on surface air temperature. Jacobs (1990) and Jacobs and Grondin (1988) conducted meso-scale analyses of surface air temperature on Baffin Island, augmenting AES data by using data recorded by automatic weather stations at several sites. Parks Canada (1994) presented a detailed meso-scale plot of surface air temperature for north Ellesmere Island. Here, the density of data was increased by using transient and historical AES station records and by incorporating non-standard data collected by Parks Canada. Alt and Maxwell (1990) presented a detailed surface air temperature climatology for the Queen Elizabeth Islands. In this case, additional data came from the Polar Continental Shelf Project (PCSP) data set. The PCSP data set is described in appendix A; its assembly and assessment constituted one of the foci of this thesis. Jacobs' work was based on objective analysis techniques; the studies of Parks Canada and Alt and Maxwell were based on the subjective, manual integration of disparate data sets.

At spatial scales smaller than the meso-scale (the local- and micro-scale), climate analyses have been conducted in support of various studies, ranging from the very small to large, multi-year research projects. Examples from glaciology are numerous and include work on Axel Heiberg Island (Muller and Roskin-Sharlin, 1967; Havens et al. 1965; Havens, 1964; Andrews, 1964), Devon Island (Koerner, 1979), Meighen Island (Alt, 1975, 1979), Ellesmere Island (Wolfe, 1995; Wolfe and English, 1995; Bradley and England, 1979; Keeler, 1964), and on glacial mass balance patterns throughout the Queen Elizabeth Islands (Alt, 1987; Koerner, 1979). A long-term project to study the polynya north of Baffin Island, called the "North Water Project", is an example of a study of a significant marine phenomenon (Muller et al. 1975). Other

studies have focussed on specific parameters, such as wind (e.g. Headley, 1990; Peters and Headley, 1992), temperature (e.g. Atkinson, 1994a), or a number of climatic parameters from a specific site (Alt et al. – submitted; Labine et al., 1994; Alt et al., 1992). Analyses of the interaction between climate and other physical parameters at specific sites have also been undertaken (e.g. Woo et al., 1990)

Other examples can be drawn from ecological and botanical studies. For example, Labine (1994) described and assessed the local-scale climate of a small coastal lowland area (Alexandra Fiord, Ellesmere Island) as an agent influencing the ecological richness of the area. Courtin and Labine (1979) presented a similar study of another lowland area (Truelove Inlet, Devon Island) in which the impact of the local climate was incorporated into the development of an understanding of ecological dynamics of that area. Tundra studies, including vegetation distribution pattern analysis (Edlund and Alt, 1989; Edlund et al, 1989), albedo studies (McFadden and Ragotzkie, 1967; Larsson, 1963; Larsson and Orvig, 1962) and energy budget investigations (Ohmura, 1981) have also been conducted.

Chapter 2

A NEW DATABASE OF HIGH ARCTIC CLIMATE DATA FROM THE POLAR CONTINENTAL SHELF PROJECT ARCHIVES

2.1 Introduction

The analysis of surface climate in the Canadian High Arctic is hampered by a low density of observational data. The Queen Elizabeth Islands alone occupy roughly 800,000 km² yet are served by only four weather stations, all of which are located on the coast (Figure 2.1). This represents one station for every 200,000 km²; a similar density in the conterminous United States would give a total of 40 weather stations. This is insufficient to characterize the surface condition in a spatially detailed fashion.

To augment the low spatial density of observing locations, a set of non-standard surface weather observations made at sites throughout the Canadian Arctic Archipelago has been made available. These data were gathered by the Polar Continental Shelf

Project (PCSP), an agency of the Canadian federal government. The PCSP was established in 1959 and continues to operate now, with a mandate to provide logistic support to research in the Canadian Arctic during the spring and summer months (March–September). Support includes aircraft and surface transportation, maintenance of radio communications with camps, lodging at the PCSP bases, provisioning of selected supplies, and loans of equipment, including meteorological equipment. The PCSP maintains two bases, the largest at Resolute Bay on Cornwallis Island and a smaller base at Tuktoyaktuk at the mouth of the Mackenzie River. Almost all scientific research in the Canadian High Arctic is supported by the PCSP.

In 1973 the PCSP initiated the “Aviation Weather Observation Program”. Under this program basic meteorological observations were collected at arctic research camps and relayed twice daily to the main PCSP base at Resolute Bay. This initiative was conceived to assist flight operations. However, the data were archived and can now be used to assist climatological research in this region. Recently these data were assembled from their disparate locations into a single database and reviewed for errors. This document details that process, summarizes the available data and provides some brief research results based on the data.

The data gathering initiative functioned until 1994, by which time the proliferation of global positioning equipment and access to satellite weather imagery had made contributions from surface observations less important for aviation. During the time data were gathered, from 1973 to 1994 inclusive, almost 60,000 observations were accumulated from almost 1200 sites. However, they are of short duration and very few

record data for more than one season. The most common elements are temperature, wind and cloud observations, but also gathered at a few sites were dew point temperature, maximum/minimum temperature, precipitation, and pressure data.

The information accumulated in the 'PCSP data set' was in danger of being lost because no one group was officially responsible for it. The main reason the initiative functioned as long and as effectively as it did can be attributed to the efforts of Dr. Bea Alt, a glacier meteorologist formerly with the PCSP and more recently with the Glaciology Section of the Geological Survey of Canada. Although never officially charged with the supervision of the project, she was largely responsible for its design and implementation, its year to year functioning and post-field season data recovery, and the subsequent archival of gathered data. However, the time commitments demanded by her other professional duties did not allow her to conduct a full compilation of this database.

This work was therefore undertaken to perform that compilation: organizing the data set with its associated station information, creating suitable database access programs, and making it available to the research community at large.

2.2 History of the data set

Each summer in the High Arctic the PCSP transports teams of researchers to field locations situated throughout the archipelago. This activity usually begins in April but sometimes as early as February and continues throughout the summer, often lasting into September. The transportation of groups to and from their field camps is a continuous

process during this period. The length of time a group spends in the field varies from a few days to almost four months and locations are similarly variable from year to year.

Before going into the field a research party received a briefing on how to take sky/cloud observations and use the meteorological equipment. Weather observations were made at the camps twice daily, at 0000 and 1200 UTC, recorded, and then relayed to Resolute Bay during radio check-ins, where they were copied by the base manager. Most camps provide a continuous twice-daily record, but gaps do exist. Some groups submitted more than two observations per day. At the end of each field season, observational records were retrieved where possible from both the field parties and from the base camp. At some camps the observation of weather was a necessary part of the research agenda, however, at many camps it was incidental. Here 'camps' refers to the same thing as does 'station', that is, an unbroken time series of weather observations from a given location. For example, a data set for the month of May and another for the month of August in the same year, both gathered from the same location, would be considered two camps in the context of this data file. Alt and Inkster (1987) described the process of field party data gathering and relaying of information.

The meteorological equipment used by the camps has varied over the years, being dependent upon the type and quantity of equipment PCSP had on hand at its warehouse and on the nature of the research being conducted at a given camp. Thus, the elements observed often differed between camps because they were not all provided with the same set of equipment. Table 2.1 details which climate elements were gathered with what type of equipment. Less equipment went to camps that were short term or which were mobile.

Table 2.1: Equipment used to gather climatic elements.

Element	Equipment
dry bulb temperature	AES** alcohol minimum thermometer electrically aspirated psychrometer sling psychrometer Davis "Digitar" electronic temperature sensing equipment Campbell Scientific RH101 temperature probe
Maximum temperature	AES mercury maximum thermometer Davis "Digitar" electronic temperature sensing equipment Campbell Scientific RH101 temperature probe
Minimum temperature	AES alcohol minimum thermometer Davis "Digitar" electronic temperature sensing equipment Campbell Scientific RH101 temperature probe
Radiation shielding	Stevenson screen PCSP "mini-screen" (small, portable version of Stevenson screen) 12-plate Campbell Scientific Gill screen
wind speed	Portable anemometer RM Young anemometer Beaufort scale groupings Davis "Digitar" anemometer
wind direction	Estimation from flags RM Young wind vane Davis "Digitar" wind vane
cloud fraction	Estimation, in tenths
cloud type	Estimation, written description with picture chart for guide
cloud height	Estimation
wet bulb temperature	Electrically aspirated psychrometer sling psychrometer
Radiation	Pyronometer (Kipp and Zonen, Epply)
Pressure	Barograph
Precipitation	graduated cylinder

**Atmospheric Environment Service Canada, the government agency responsible for all permanent weather stations in Canada.

Several documents reviewing the PCSP data set have been produced. Thomson et al. (1986) described all non-standard (i.e., non-AES) data sets available for the High Arctic, including those from Dome Petroleum, PanArctic Oil (both gathered from the early 1970s to the mid 1980s) and PCSP, and included brief statistical summaries and assessments of usefulness. Etkin (1988) focussed exclusively on the PCSP data set, summarizing record lengths at various stations, types and extents of local site influences and suitability of data for various types of research. Atkinson (1994b) provided a review of the data up to 1994, which included storage locations and an assessment for continued gathering and archiving efforts.

In the 20 years this project was underway there was never any single responsible agency maintaining archival continuity. Various contracts were issued to encode the data into digital format. This was problematic for two reasons: first, it meant that the data were stored in three different formats, reviewed individually below; and second, stations were identified in three different ways at different time periods. These problems have limited the use to which the data have been put over the years.

2.2.1 1974–1983

During the period 1974 through 1983, several contracts were issued to have the data quality checked and coded. Data were drawn preferentially from the field camp records, but where necessary, these were cross-referenced with base camp records for verification. In 1984, the Arctic Applications Division (now disbanded) of the Atmospheric Environment Service (AES) transferred the coded hardcopy records onto computer tape. These data were then transferred to an AES standard format data set for permanent archival. The particular format used, TDF-1129, was selected so that the data could be accessed by the AES Marine Statistics System (MAST) to assist in the production of a Marine Atlas (Agnew et al., 1987). Further encoding of data into this format occurred up to 1986. In 1988, in response to concerns about shortcomings of the TDF-1129 format for land data (it does not archive maximum and minimum temperatures), AES transferred the data archive. The data were written to an hourly observational format, HLY01, and the station data were transferred into the AES Station Information System (SIS) database. The HLY01 format was selected because PCSP data were hourly, with two or more observations per day. The format was organized by observational element. Each record possessed a header string containing station

identification, date and a climatic element label, followed by 24 7-character observation spots, one for each hour of the day. Missing data were coded with a standard flag, -99999M.

Thus, PCSP data from the years 1974 through 1986 were and are still stored by the AES data archive division. The AES data retrieval system, when working as designed, will format archived data in tabular format. However, the retrieval system did not work with the PCSP data because missing observations were coded with blank characters and not the proper flag. It was also discovered that the PCSP data held in the HLY01 format were missing maximum and minimum temperature observations, the result of an error in a data transferal operation.

The files as stored at AES were thus not readily usable. For this reason the open reel magnetic tapes submitted to AES by the original data coding contractor were sought. These were used as the source of data for this study because they were organized in a format that was much easier to use and because all original data were present. These data were organized in table format with one record per observation period (row or line) for each of the available weather elements (columns), sorted by date and grouped by year and station in an 80 column ASCII file. Unfortunately, only the first of the two reels was readable, that containing data from 1973–1984.

2.2.2 1984–1986

The years 1984–1986 had also been encoded by an external data entry contractor, turned over to AES and ultimately archived in HLY01 format. These years had to be obtained in HLY01 format because the magnetic tape on which the original table format

data had been stored had degraded and was unreadable. Maximum and minimum temperature data for this period are not available because of the programming error mentioned previously.

2.2.3 1987–1993

AES stopped archiving PCSP data after 1986. The years 1987 and 1988 were encoded under data-entry contracts issued by PCSP and the Geological Survey of Canada (GSC) and were available on floppy disk in spreadsheet format, one file per station. The data structure was tabular, one element per column and one observation per row, sorted by station and then by time.

Samples of data structures are presented in Appendix A.

2.2.4 Station information and original records

Information about camp topographical context was noted on information sheets that were returned with observation records by the field parties. These are currently stored in binders.

The original data records are stored in file folders and reside currently with the Glaciology Section of the Geological Survey of Canada who are not, however, technically responsible for the storage of these data. There are several instances of missing base station records, most notably the entire 1989 field season.

2.3 *Methods*

2.3.1 Reading and processing the original data files

The variety of formats in which the PCSP data had been stored necessitated the creation of computer routines in various software environments on various hardware

platforms that were tailored to the three time periods involved. Weather element data from all years were read by the computer routines (Appendix A) and stored in a single observation database; all data concerning station location and situation were stored in a station location database. The software system used for processing and storage of the final data sets was SAS™.

Reading the data for the years 1974–1983 into a SAS data set was relatively straightforward because the tabular structure of the original data for this time period was similar to the structure used by SAS. Each observation (one row) was treated as a string. Data were extracted from certain points along the string, assigned appropriate variable identifiers representing climatic elements, and written into the data set. Peculiarities of this data set included, for example, the use of the number 1 to represent the negative “-” symbol and the use of Fahrenheit temperature scale for 1974 data. These inconsistencies were handled as the data were read into the SAS data set.

Data for the years 1984–1986 were stored in AES HLY01 format. While a space-efficient way to store data (dates, times and station identifiers are not repeated for each observation, unlike a table format), this format was difficult to process into a useable form and required the most extensive SAS program (Appendix A). The coding algorithm consisted essentially of three parts: a loop to read observations in single element blocks (of 24 hourly observations) by day, sorting the data set by station and date, and then outputting in a tabular format. These data also required the conversion of wind speed from metric into Imperial units (knots).

Data for the years 1987–1993 were stored in PC spreadsheet formats (Macintosh Excel and Quattro for DOS), one spreadsheet per station. All spreadsheets were imported into Quattro. A program (Appendix A) was written to properly format each station and export it as ASCII text. From there a Unix 'korn shell' script concatenated the individual station files into a single file for each year which were then read by a SAS program. Inconsistencies within the data files required several stations to be manually input (i.e. problems with headings or number of variables that did not work with the Quattro formatting program).

Station data were stored only as hardcopy and had to be encoded into digital form. There existed, however, no one standard way of identifying the stations. Within the four time periods there were three different ways to identify the PCSP stations (Table 2.2).

Table 2.2: Station identifiers used in the different data storage formats.

Data period	Identifier	Identifier type	Identifier width
1974-1983, 1987-1988	PCSP station no.	Numeric	4
1984-1986	AES station id.	Character	7
1989-1993	PCSP station name (geographically based)	Character	25

The station database required at least one identifier for each station: a character name, the original PCSP identifier, and/or the identifier assigned by Atmospheric Environment Service, as well as a location references and start/end dates. These were input into a second database. Elevation was available on the original station files for approximately half of the stations. Dates and location information were obtained from the station data hardcopy.

For simplicity, all three ways of referring to the stations were retained in the final database, rather than trying to select one in particular or to come up with a new identifier. This required the construction of a reference table that contained for each station all possible identifiers it possessed, which was then merged with the station information database. This table was compiled from three original sources: one coming from the station information binders, one from Thompson (1986—providing the AES identifiers), and one created for the years 1987–1993.

Merging of the data files into a single database involved prescribing the final structure of the database, modifying each data file to match that structure, and then conducting the merge. A minor problem concerned the fact that each time period had variables that were unique to it. This problem was addressed during the final merge stage by incorporating all available variables into all observation periods. Those times that did not originally possess the variable had a missing value assigned.

Code and examples for these routines are provided in Appendix A.

2.3.2 Error checking

This consisted of examining the observation and station information databases for outliers or inconsistencies and making corrections or discards as necessary.

Observation database

Extreme temperature values were discarded during initial file import. After import, further error checking was performed on the following elements:

- dry bulb, dew point, wet bulb, maximum and minimum temperatures within the range -60 to +30 °C
- wind direction between 00 and 36; wind speeds of less than 100 knots

- total cloud cover between 0 and 10
- individual cloud layer amounts between 0 and 10; layer totals equal total cloud cover

A data visualization system (SAS/INSIGHT™) was used to determine the ranges used to assess datum veracity. These ranges formed the basis of assessment used by an error-reporting program. If an error was detected there were several approaches to handling it. In many instances obvious errors had occurred during data entry, such as reversed digits for wind direction or the addition of a second, identical digit (e.g. 99 instead of 9). Errors were corrected if the problem was obvious, otherwise the datum was deleted. Errors in which cloud layers did not equal the total cloud cover were not dealt with because use of the cloud layer data was not planned for this project.

Table 2.3 provides total number of occurrences by error type and indicates how errors were dealt with. Any error type with zero occurrences is not listed in the table.

Table 2.3: Number of errors by type.

Specific Problem	occurrences	corrected	Discarded
Maximum temperature too high	1	-	1
Minimum temperature too low	2	2	-
Dew point temperature too high	3	1	2
Total cloud cover > 100%	28	18	10
Cloud layers > 100%	11	10	1
Wind speed too high	1	1	-
Wind direction incorrect	41	18	23
Sum of cloud layer amounts do not equal total cloud cover*	849	-	-

* These errors were not dealt with.

A disproportionately large number of cloud error problems occur in the last four years of data. Table 2.4 contrasts, by year, the percentages of total errors and total number of observations. The last four years of data represent 13% of the total number of

observations yet possess 61% of all errors, almost all of which are cloud errors. This increase in cloud reporting problems is ascribed to a lack of expert advice available during the field season. Up until 1989 Dr. Alt had usually been on hand to assist field parties with cloud observation, a skill that often requires considerable experience. Advice was not available after this time and the quality of cloud observation suffered accordingly.

Table 2.4: Percentages of errors and total numbers of observations by year.

Year	Percent of total errors	Percent of total observations
1974	4	9
1975	1	8
1976	1	6
1977	1	6
1978	3	6
1979	1	6
1980	2	7
1981	5	5
1982	3	5
1983	2	5
1984	2	6
1985	3	6
1986	4	6
1987	3	2
1988	2	3
1989	3	1
1990	20	4
1991	19	4
1992	14	3
1993	8	2

Station information database

Quality control of the station information database included ensuring it matched the observational database and checking the coordinates. The station information database was compared with the observation database to determine if all stations represented in

the data possessed corresponding records in the station database, and *vice versa*. Station beginning and end dates were obtained from the weather observation database and compared with the same information in the station location database. In numerous instances discrepancies were found; they and their corrections, when possible, took several forms and are listed in Table 2.5. In some instances corrections were made from the original hardcopy.

A count was made of all station identifiers in the observation database that did not possess a corresponding record in the station database and *vice versa*. The total was 234 occurrences, but a number of these were due to slight inconsistencies in station name spellings. Spelling corrections reduced the number of mismatches to 211. Other types of mismatches are described in the column labeled "Situation" in Table 2.5. Corresponding confirmation codes for the observational database are listed in Table 2.6. Stations for which there was no location data were not corrected; these data are currently unavailable.

Verification of locations was also performed on the station information database, a process that consisted of two steps. The first was a check for gross location errors, that is, locations that fell outside the archipelago; the second was a check for smaller errors in the accuracy of location data. This was conducted by examining plots of the stations to see exactly where they were situated, to ensure, for example, that a station was not located 100km to the west of where it should be.

Several strategies were available to correct errors in location: finding references to the station in other years, looking up the geographical feature using the "GeoNames" internet database search service offered by the Geological Survey of Canada, referring back to the original hardcopy field observations, or referring back to the hardcopy base station records. In almost all cases the name given to a camp was simply that of the nearest named geographical feature on a 1:250000 National Topographic Survey map, a fact that greatly facilitated verification of location.

Table 2.5: Station database 'confirmation' field flag codes.

Flag	Situation	Action
Y	Station location, data and times confirmed in both databases	none required
A	Station database 'datein' and 'dateout' fields existed and differed slightly from the observational database	station database 'datein' and 'dateout' fields altered to match the times in the observational database
B	Station database 'datein' and 'dateout' fields existed and differed significantly (e.g. different years) from the observational database	station database 'datein' and 'dateout' fields altered to match the times in the observational database (note that no corresponding data existed in the observational database for that year)
C	Station database 'datein' and 'dateout' fields missing	'datein' and 'dateout' fields taken from the times in the observational database
D	Station database 'lat' and 'long' location fields missing	'lat' and 'long' data taken from another record possessing the same station identifier in a different year
E	Station database 'lat' and 'long' location fields missing	none possible: no 'lat' and 'long' reference for this identifier can be found in the station database. data are unusable
F	Station database record missing	station database record created: combination of actions C and D
G	Station database record missing	none possible: no 'lat' and 'long' reference for this identifier can be found in the station database. data are unusable
J	Station database record missing	record created from metadata records and dates from observational database
M	Situations A and B existed	actions for A and B taken
T	Observation database missing some dates but, with reference to the station database, appears to be part of an existing series	new station database record created for the assigned period. Original station database record modified to reflect time period for which the observation database had dates; original station database record assigned flag 'V' to reflect the modification.
V	See flag 'T'	see flag 'T'
W	Observational database possesses data for this identifier but no dates	none taken at this time: it is not possible to ascertain the veracity of dates assigned to the observational database based on those in the station database. data are unusable
X	Observational database has no records for this identifier	none possible: station database record retained for reference
U	Identifier mix-up in the station database	observational record with dates only corresponds to a station record possessing no dates or locational data, but with an identifier duplicate to an existing confirmed station record. Observational data assigned to station, but data are unusable
Z	Identifier mix-up in the station database	station database records sorted out with reference to metadata; observational record data assigned station that matched dates
H	Observational data had pcsp station identifier of 0	station database records with matching dates and no other data assigned to ob record; observational database records changed.

Table 2.6: Observational database 'confirmation' field flag codes.

Flag	Situation	Action/identifiers of affected stations
Y	Station location, data and times confirmed in both databases	none required
J	station database record missing	record created from metadata records and dates from observational database
T	observation database missing some dates but, with reference to the station database, appears to be part of an existing series	dates assigned to records in observational database
W	observational database possesses data for this identifier but no dates	none taken at this time: it is not possible to ascertain the veracity of dates assigned to the observational database based on those in the station database. data are unusable
X	Observational database has no records for this identifier	none possible: station database record retained for reference
U	Identifier mix-up in the station database	observational record with dates only corresponds to a station record possessing no dates or locational data, but with an identifier duplicate to an existing confirmed station record. Observational data assigned to station, but data are unusable station 172 changed to 8 station 7 changed to 25 station 15 changed to 17
H	Observational data had pcsp station identifier of 0	observational database records changed.

Overall, of the approximately 58000 records that exist in the observation database 60% possessed accompanying station information that did not require attention while 30% required some modification of the station information database. However, almost 10% of the observational data are unusable because the attendant location information is either missing or obviously in error. Thus, when assessing non-standard data, while concerns about the accuracy of observational data are important, in the case of the PCSP database a far more serious cause of data loss in fact came not from errors within the observational data but from errors in the recording or transcription of the associated location information.

Data reliability assessment

An important consideration regarding non-standard data is its reliability for use in research. Even though sources of gross error were addressed there might be more subtle errors within the data, such as a systematic misuse of a particular instrument, that can introduce inaccuracies into results based on analyses using these data.

The manner in which reliability was assessed depended on the climatic element. Objectively observed elements, such as any temperature measurement, pressure, and precipitation were expected to be more accurate than subjectively observed elements such as cloud amount, type or height, and weather element type and intensity. Wind velocity was represented by both instrumental and estimated observations, however, they were usually instrument based and all wind velocity observations will be treated as such in this paper. Wind direction observations were usually estimated using an improvised wind sock, but it will be treated as an objectively measured element because it has a large tolerance for error.

The main purpose for gathering these data was the facilitation of aviation safety and there was thus a vested interest in the accuracy of all elements, objectively or subjectively observed. In terms of objectively measured elements, those making the observations were researchers with science backgrounds who were presumably capable of handling meteorological instruments, which are simple to read. For these reasons the reliability of the objectively observed data may be considered high. There will be small errors of reading based on the finest graduation mark, however throughout the database the distribution of error will be random because there are so many individuals and

locations involved, and thus the mean of the observations should approach the actual value. Thus instrument-based observations in general may be used with confidence, taking into account the provisos noted below.

Observations that were purely subjective, however, must be treated with caution. The accurate observation of clouds is difficult. Within the PCSP database the most reliable cloud elements are total cloud cover and number of layers, because cloud identification was not required. The percent cover for individual layers can be fairly accurate or not, depending on the sky condition. The type of the lowest layer is often reasonably well identified. If the clouds are at mid- or high-level identification of type was more problematic, e.g. confusion between altostratus and cirrostratus. Accuracy of cloud height observations was poor. In general, cloud layer data from the PCSP database should be used with caution.

Weather element type was usually well identified, except when the distinction between precipitation forms was subtle, for example, snow grains, snow pellets and ice pellets. Intensity of precipitation forms was usually over-estimated.

Arguably the most important element of the PCSP database is temperature. As an example of the accuracy of this element, three sets of plots were prepared, one set for each of three years, showing temperature data from various PCSP stations and a map showing the station locations (Figures 2.2a,b; 2.3a,b and 2.4a,b). Data from AES stations were plotted for reference.

Air temperature at a given location is a function of processes acting at various spatial scales. One process is the seasonal cycle of net radiation balance. This cycle is manifested as a low-frequency pattern in temperature time-series with a signal that should be coherent over a large geographical area. To assess its prevalence a principal components analysis (PCA) of temperature was performed for the years 1977, 1979 and 1981 using data from the month of July. Stations possessing at least 28 days of observations in July were included. The components that result from a PCA represent physical processes that are responsible for the patterns in the temperature time series. Therefore, components possessing numerous large loadings represent phenomena that are spatially wide spread, as is the seasonal cycle. In each analysis the first component accounted for a large amount of variability, between 45 and 49 percent, and had many stations loading with large, positive values. All other components accounted for lower amounts of variability and possessed few large component loadings. These results are summarized in Tables 2.7, 2.8, 2.9, and 2.10. Elements in the temperature time series that are unique to one or a few stations are picked up by lower components (e.g. components 3, 4, or 5). These components may be considered indicative of local-scale phenomena at work and which are manifested at certain stations by their large loadings on certain factors. A more extensive PCA is presented in Chapter 3.

Table 2.7: Results from principal component analyses of July temperature data showing percent variance explained by first three components for stations used in figures 2.2, 2.3 and 2.4.

	1977				1979				1981			
	% variance explained by component #			# stns.	% variance explained by component #			# stns.	% variance explained by component #			# stns.
	1	2	3		1	2	3		1	2	3	
PCSP + AES*	45	21	8	12	46	14	10	11	49	12	10	12

*Includes Resolute Bay, Rea Point, Eureka and Alert

Table 2.8: PCA component loadings for 1977. Loading values range from +1 to -1. A large positive or negative value is indicative of a relationship to the factor.

Station	Factor 1	Factor 2	Factor 3
AES Resolute Bay	0.835	-0.005	-0.162
AES Alert	0.748	-0.18	0.359
AES Eureka	0.698	-0.545	-0.262
AES Rea Point	0.466	0.353	0.65
S34	0.825	0.332	0.05
S55	0.805	-0.31	-0.316
S75	0.299	0.796	-0.048
S102	0.158	0.722	-0.157
S120	0.822	0.061	-0.279
S125	0.559	0.67	-0.132
S126	0.814	-0.274	0.112
S130	0.594	-0.377	0.357

Table 2.9: PCA component loadings for 1979. Loading values range from +1 to -1. A large positive or negative value is indicative of a relationship to the factor.

Station	Factor 1	Factor 2	Factor 3
AES Resolute Bay	0.768	-0.374	-0.309
AES Alert	0.688	0.511	0.352
AES Eureka	0.737	0.137	0.458
AES Rea Point	0.651	0.321	-0.082
S34	0.859	-0.263	-0.297
S62	0.713	-0.41	0.081
S79	0.695	0.321	-0.031
S83	0.57	0.141	-0.355
S87	0.416	-0.343	0.643
S173	0.736	-0.431	-0.055
S190	0.489	0.616	-0.194

Table 2.10: PCA component loadings for 1981. Loading values range from +1 to -1. A large positive or negative value is indicative of a relationship to the factor.

Station	Factor 1	Factor 2	Factor 3
AES Resolute Bay	0.817	-0.009	0.351
AES Alert	0.707	-0.322	-0.26
AES Eureka	0.891	-0.09	-0.212
AES Rea Point	0.545	0.139	0.369
S43	-0.075	-0.68	0.318
S49	0.881	-0.046	0.248
S59	0.779	-0.105	-0.414
S64	0.629	-0.156	-0.534
S79	0.069	0.81	-0.26
S87	0.809	0.197	0.219
S199	0.643	0.369	0.315
S241	0.88	-0.049	-0.015

In a few instances station loadings on component one were low. These stations did not exhibit coherency most likely because they were situated in areas where local influence is extreme and which masked the station's response to synoptic-scale processes.

It must be noted that there was a possibility of non-random error occurring in the temperature data. In some instances a temperature reading may have been influenced by direct solar radiation. The error is non-random because it would not be manifested during cloudy conditions, but would appear as an over-estimation during sunny conditions. The chance of occurrence is small, for two reasons: observations are taken in the morning and evening when direct solar radiation is low, and the archipelago is more often cloudy than sunny (Maxwell, 1982). Despite these mitigating factors, the possibility still exists and is thus mentioned. The most likely situation under which this type of error could occur would be an evening observation after a clear day with little wind.

2.4 Data set summary

This section presents statistical information about the PCSP data set. To show from where most of the PCSP data were gathered, overall station densities for the years 1974–1993 were plotted (Figs. 2.5a, 2.5b). Density values were obtained by totaling the number of stations within a specified radius around each site and contouring those totals. Stations that possessed at least three days of observations were used, resulting in 990 stations for the 20 year period. Use of a relatively large search radius of 100 km (Figure 2.5a) highlighted the main areas of activity while a smaller 50 km search radius identified local concentrations of research activity (Figure 2.5b). These plots suggest

good locations for any future research that needs to incorporate PCSP data, and indicate where research has been most active over the years.

The contour pattern in Figure 2.5a is largely a byproduct of logistical, rather than research, considerations. The main areas of activity are centered around Resolute Bay and Eureka. Resolute Bay is the main base of operations for the PCSP; it also possesses an AES station and an airstrip and is an important high arctic center. Eureka, another AES station with a good airstrip, serves as the main hub for the central Queen Elizabeth Islands. Another area highlighted in Figure 2.5a, the northern part of Ellesmere, is largely encompassed by the Ellesmere Island National Park and has been the focus of many research projects over the years, especially centered around Lake Hazen. Other zones are identified on Figures 2.5a and 2.5b. These areas usually possess specific sites that have seen repeated annual occupancy for long-term studies, such as the research camps at Truelove Inlet or Expedition Fiord, or small areas which have been a focus for long-term activity, such as the on-going archaeological work in the Hazard Inlet area, and which have possessed multiple site locations within a few kilometers of each other.

The number of stations and their durations have varied considerably from year to year (Figure 2.6). Durations are broken into one-week intervals for short-term stations, into two-week intervals for longer stations lengths and into months for long-term stations. Note that in all years there are stations with record lengths exceeding one month. The sharp decline in 1987 of the total number of stations, and the trend in general towards fewer stations after 1986, is a reflection of a realignment of PCSP mandate away from data storage or analysis work (before 1986 PCSP possessed a small research group).

In general, of all stations available, 75% possess records that exceed one week in length and 30% possess records that exceed one month. Counts of various climatic elements, by year and by month, are presented at the beginning of Appendix A.

Seventy-five percent of PCSP data are within 10 kilometers of a coast and are at low elevation (Figs. 2.7, 2.8). This grouping is an artifact of logistics and research agendas. Bush aircraft need flat stretches on which to land; these are typically found on raised beaches and gravel terraces near coastlines. Often, however, phenomena under study by the individual camps involved are also located in such areas. The climate data from the near-coastal sites form an important source of information about the coastal regime, which is a highly variable environment in the arctic archipelago. The remainder of the data come from inland sources. This is perhaps one of the most important contributions the PCSP database can make. Data from island interiors augment AES data by providing information about environment types that the AES network does not sample.

The mean and median number of observations per station by year is presented in Figure 2.9. The median has remained relatively constant over the years, ranging between 20 and 40 observations per camp in all but four years, indicating a general consistency in station durations. The mean almost always exceeds the median, indicating that often there are one or two stations in each year that possess a large number of observations. There were proportionally more long-term stations in 1974, 1975, and 1991 than in other years.

2.5 Uses of the data

Since 1973 PCSP data have been used in several studies. Most of these studies have focussed on local-scale phenomena. Biological sciences have used the data as climatic input for studies of animal habitat or ecology/botany (Labine, 1994; Edlund and Alt, 1989; Courtin and Labine, 1979). Glaciological studies have made use of the data, again as climatic input for mass balance studies (Alt 1975, 1979, 1987; Wolfe, 1995; Bradley and England, 1979; Keeler, 1964). An archipelago-wide radiation model was produced by Young et al. (1995). Alt and Maxwell (1990) and Maxwell (1980) have used data from many stations to subjectively increase detail of climate plots.

As indicated in Alt and Maxwell (1990) and Maxwell (1980) the density of climate observation provided by the AES stations is inadequate to represent in detail the climate of an area the size and complexity of the arctic archipelago. Maps portraying climatic elements interpolated to form a continuous surface possess a resolution insufficient to capture the physiographical complexity of the region. The situation is such that subjective modification by an experienced analyst can form a useful source of additional information. For example, a frequently cited portrayal of temperature is Maxwell's map of July normals (Figure 2.10). This map was based on data from long-term AES stations, a few incidental stations, and Maxwell's experience. On it are portrayed contours which show cooler upland regions that are not backed up by observational data, yet which Maxwell believes to be correct and which contributes important information to the map.

The main contribution the PCSP data can make is to mitigate the problem of station paucity in the AES observing network by providing more detail about a given type

of local-scale environment currently sampled by AES stations, and by providing information about environments that are not sampled by AES stations. The PCSP database is used in a regional climate study, presented in Chapter 3, and in a verification role to support a high-resolution surface air temperature model for the Queen Elizabeth Islands, presented in Chapter 4.

2.6 Conclusion

This chapter has detailed the assembly of the Polar Continental Shelf Project database of climate data, collected under the Aviation Weather Observation Program, from a disparate collection of small data sets into a single, accessible unit. This process included the collation and organization of the small observation and station location data sets and dealing with errors in the data sets. The resultant database, from the period 1974 through 1993, consists of 53000 observations, each observation possessing one or more climatic elements, and 1000 locations, 60% of which are unique.

Given that the data are non-standard an assessment of data accuracy was conducted. The most often utilized element, temperature, was selected for examination. Assessment consisted of two analyses: a series of comparison plots (Figs. 2.2 – 2.4) and a principal components analysis. The plots, in which AES data were used as a reference standard, show the time series from the PCSP stations trending with the time series from the nearby AES stations. This shows the PCSP data responding in the same way to regional-scale influences as the AES station. In the PCA results both PCSP and AES stations are loaded heavily on the first component, indicating that most stations are responding to processes acting over large spatial scales, in this case the seasonal cycle,

in a similar manner and thus confirm that the PCSP data are seasonally coherent with the region as a whole. More detailed applications of the PCSP data set are conducted in Chapter 3 and the data set is used in a supporting role to verify the surface air temperature model presented in Chapter 4.

The utility and accuracy of the PCSP data set has thus been established. Of the climatic elements represented, cloud layer data and precipitation data should be used with caution. Data about weather type should be used with the previously stated provisos borne in mind. All other elements are fit for use.

Chapter 3

A PRINCIPAL COMPONENTS ANALYSIS OF THE JULY TEMPERATURE CLIMATE OF THE CANADIAN ARCTIC ARCHIPELAGO

3.1 Introduction

The level of detail at which an analysis of summer surface air temperature in the Canadian Arctic Archipelago (CAA) may be conducted is limited by several factors, including the coastal location of all meteorological stations, the large distances between the few weather stations, and the complex physiography of the archipelago (Figure 3.1).

Physiographic complexity can restrict representativeness, depending on the type and scale of the analysis. For example, the portrayal of general trends in time and of broad patterns over large areas can be captured adequately by relatively few stations. Location is not as significant a concern when the objective is the identification of patterns

caused by climatic controlling agents operating at the largest spatial scales. However, considerations of restrictions to representation become more important when detail within a smaller area is sought in short time-period analyses, especially in areas of complex physiography. An additional concern in the arctic is the fact that all of the meteorological stations are located within a few kilometers of the coast. In general, for stations around the world, location on a coast can result in a local-scale modification which consists typically, in summer, of a cooling caused by on-shore advective flow or by local-scale circulations called the 'sea-breeze' (King and Turner, 1997; Maxwell, 1980). Thus two factors, complex physiography and coastal proximity of stations, act to reduce the detail that weather stations of the Atmospheric Environment Service (AES) may resolve in analyses of surface climate elements in the Arctic Archipelago.

A further problem restricting the spatial resolution of arctic climate studies is the large distances between the observing sites. The mean distance between stations operated by the Atmospheric Environment Service of Canada (AES) is 600 kilometers and the Queen Elizabeth Islands, the group of islands north of the Parry Channel (north of 75° N latitude), are currently served by only four stations (Figure 3.1). Few AES stations in the archipelago possess a continuous record back to 1950 (Figure 3.2). There have been other stations in various locations in operation at different times since 1950, but most have closed. These additional stations were also located in close proximity to the coast, with the exception of Lake Hazen, on north Ellesmere Island (in operation 1957–1958 and summer, 1963), and Dewar Lakes, on central Baffin Island (in operation 1961–present).

The AES network of weather stations is dense enough only to reveal large-scale synoptic activity. Meso-scale patterns, caused by the topographic modification of the radiation and/or wind regime, or the modification of temperature due to elevation or as caused by major features such as ice fields, cannot be identified using these data.

In the CAA several research efforts have been conducted to increase the spatial detail of surface air-temperature maps; these efforts involve various ways of augmenting the existing AES network. Maxwell (1980) modified a plot of 1941–1970 July normal temperatures generated from long-term station data using his own experience and information from short-term, historical stations. Alt and Maxwell (1990) used non-standard, short term weather observations from several sources, including research camps supported by the Polar Continental Shelf Project and oil exploration camps, to increase the spatial detail of temperature maps in the Queen Elizabeth Islands. In work conducted on southern Baffin Island, Jacobs (1990) used automatic weather stations (AWS), automated mapping techniques, and transfer functions to more objectively deal with the problem of spatial gaps in data availability. These studies, combined with the surface heterogeneity of the arctic archipelago, suggest that patterns at the meso-scale should be regular features of the surface temperature climate in this region. It is also apparent from these studies that data from AES stations alone can not be used to explore this question.

The objective of this chapter is to utilize a database of non-standard temperature observations to identify air temperature patterns in greater spatial detail than those which can be identified using AES data alone.

This was accomplished using the method of principal components, a technique regularly used in climate research. Principal Components Analysis (PCA) identifies variance common in time and space and extracts this information as new variables called components which, when plotted, summarize the major spatio-temporal patterns that exist in the data. PCA was selected because it is able to capture the temporal aspect by indicating the strength of relationships through time, something direct contouring or interpolation onto a grid, two other common methods of spatial analysis in climatology, can not do.

A secondary objective of this chapter was to show how the spatial and temporal results produced by PCA were able to be linked directly to specific synoptic events. This study also demonstrated the utility of the non-standard Polar Continental Shelf Project (PCSP) data set, for climate studies in general and for assessing a model to predict surface air temperature in particular (Chapter 4). Computer code used in this analysis is found in Appendix B.

3.1.1 Principal components analysis: methodological review

Principal components analysis is a numerical method that replaces variables in a data set with an equivalent number of components. Variance in the components is redistributed such that a maximum amount of common variation is retained in each successive component, and each component is linearly independent of the previous one. The essential objective of a PCA is to account for, but not necessarily physically explain, the largest amount of variance using the fewest number of components (Legates, 1991). PCA is a member of a family of related methods known as

eigenmodels; other members include common factor analysis and empirical orthogonal functions.

PCA has been used in climatology and meteorology since the 1950s (Richman, 1981,1986). It has been used for various purposes, including the identification of recurring structure (i.e., pattern classification) in pressure or other meteorological parameters (e.g., Richman, 1981), the definition of coherent sub-regions in climatic elements (e.g., Mallants and Feyen, 1990), and the reduction of numerous, linearly dependent climatic elements into a few independent, synthetic variables. These synthetic variables may be used as input into further analyses (e.g, Diaz and Fulbright, 1981) or may be used to define climatic zones (e.g., Malmgren and Winter, 1999). Richman (1981) provides a lengthy review of applications of eigenmodel analysis in the meteorological and climatological literature between 1980 and 1986.

The first consideration when using these methods is the selection of an eigenmodel. Common factor analysis is rarely used in climate and meteorological analyses (Richman, 1986) because it discards variance that is unique to individual stations. It was of interest to retain this local-scale variance in this study; the choice was therefore restricted to empirical orthogonal functions (EOF) and PCA. Disagreement in the literature emerges at this point. Richman (1986) and Jolliffe (1986) frequently consider theoretical aspects of the application of PCA and disagree on the exact meaning of EOF: Jolliffe considers EOF another name for PCA, whereas Richman feels EOF is another form of analysis. Principal components analysis was the eigenmodel selected for this study. After selecting an eigenmodel, the form of input matrix, termed

the dispersion matrix, must be determined. The two most frequently used matrix types are covariance and correlation. In this study, the correlation matrix was used. An important difference between the correlation matrix and the covariance matrix is the sensitivity of the latter to differences in the magnitude of variance between variables (Jackson, 1991, pp 64–66). In this study temperature is the only variable being compared and thus all variances were of comparable magnitude. To verify this, correlation and covariance dispersion matrices were both tried, with identical results.

Results consist of components, equal in number to the original number of stations, that are composed of synthetic data time-series. The time-series for a component are values, called 'scores', which indicate the relative influence of the component through time. Another value, termed a 'loading', is generated for each station for each component which indicates the correlation between the original station time-series and the component time-series. Station loadings are mapped to produce a loading pattern. The total variance explained by each component is given by the eigenvalue. Eigenvalues can be used to determine how much of the overall variability a component has accounted for. They are also used to determine how many components to retain for subsequent analysis and interpretation. There have been several methods proposed for determining the number of components retained, including both subjective and objective techniques (e.g. Jolliffe, 1986). Only the first three components of any analysis were examined in this study, regardless of their explained variances.

Once a subset of components has been selected for retention a further option exists: the components can be 'rotated' to enhance their interpretability. The issue of

rotation was reviewed because it is often used in climatological research and was thus considered for this study.

In a simplistic sense, because a PCA maximizes explained variance, the position of the first component vector in a multi-dimensional space is that of a 'mean' vector which may not represent any particular cluster of variables. Rotation attempts to overcome this by shifting the axes of the component vectors so that they are better aligned with clusters of correlated variables. There is disagreement in the literature on the usefulness and necessity of rotation. Proponents of rotation, chief among them Richman (1980, 1981, 1986, 1987, 1993), advocate the application of rotation in all analyses (Richman, 1986). However, Jolliffe (1987, 1986) disagrees with Richman and another PCA theoretician, Legates (1991), further criticizes both authors. An important point about rotation is that the accuracy of rotated results depends upon the number of components retained. An error in determining how many components are retained is carried through into a subsequent rotation, possibly with detrimental effects.

The theoretical debate about rotation sparked by Richman, Jolliffe, and Legates has continued unresolved (Legates, 1993; Richman, 1993; Legates, 1991). An examination of several recent papers showed a continued lack of consensus. Leathers and Luff (1997) employed an unrotated PCA without justification. Vega et al. (1998) employed a rotated PCA using justification based on arguments found in White et al. (1991), which are essentially those forwarded by Richman (1986, 1981, 1980). Malmgren and Winter (1999) applied a rotated PCA, justifying their approach using two books on the subject (Jackson, 1991; Jolliffe, 1986). At the present time the consensus

in the meteorological and climatological community regarding the application of eigenmodels appears to be to use whichever method provides an interpretable result. A comparison was made between rotated and unrotated results using the AES and PCSP data. The rotated loading patterns were found to be more difficult to interpret and excessive variability was introduced into the time series plots, with no gain in amount of variance explained. For these reasons rotation was not applied.

3.2 Method

3.2.1 Analyses

Three sets of principal components analyses were performed. The first set was conducted to demonstrate how PCA can account for aspects of the spatial and temporal components of temperature patterns, by showing how specific synoptic events can be identified in the PCA results. The second set was performed on several multi-year time-series of AES data to establish recurring large-scale temperature patterns. The third set incorporated PCSP data to determine if additional spatial detail can be resolved in temperature patterns using an increased station density.

The first principal components analysis was conducted on mean daily temperature data from AES stations for June, July and August in each of two years, 1962 and 1964. These years were chosen because detailed synoptic analyses were available against which PCA results could be compared, to assess the ability of the PCA to identify synoptic-scale patterns and events (Alt, 1987).

The second PCA was conducted on mean July daily temperature data from AES stations. Four separate analyses were performed in order to maximize the number of stations available in each analysis, because the number of AES stations varies over time (Figure 3.2). The periods analyzed are listed in Table 3.1.

Table 3.1: Time periods covered in principal components analyses of AES data.

Analysis number	Years covered	# stations
1	1951 – 1974	8
2	1957 – 1977	11
3	1969 – 1985	11
4	1977 – 1988	12

A third analysis was conducted on mean July daily temperature data from both AES and PCSP stations. The PCSP data are reviewed in Chapter 2. Analyses were conducted on all years in the period 1974–1992 which represents the full extent of PCSP data availability. Five years (1975, 1983, 1984, 1991, 1992) were selected for direct comparison with single-year AES analyses; years with at least eight PCSP stations were selected to provide a clear indication of the additional detail resolved with the extra stations.

All stations were considered for inclusion, however because missing values were not estimated, station inclusion was restricted to those possessing a full set of data for the period being studied. In the case of the multi-year AES analyses, data missing from any station results in a missing observation from the analysis. Before performing the PCA in the second and third sets of analyses, data series had the mean time-series subtracted because it was found that, without this step, large explained variance values were obtained on the first component. For example, on a PCA conducted on a 5-month period (May–September) and a 12-month period (January–December) the first

component explained 87% and 93% of the total variance, respectively (Figure 3.3). In these cases the first component was picking up on the dominance of the seasonal cycle. This is a usual result for PCA performed on temperature time series. The subtraction of the mean time-series from all individual time series, i.e., on a site-by-site basis, was therefore deemed necessary to increase the sensitivity of the PCA to smaller-scale patterns. The first set of PCA did not have the mean subtracted because they were being compared directly to results obtained from observational data, that is, the synoptic analyses for 1962 and 1964, which had not had the series mean subtracted.

The results of a PCA consist of 'variance explained', 'component loadings' and 'component scores'. Variance explained by a component is expressed as a percentage and indicates how much of all the variability in all the time series the component was able to account for. The more spatially coherent a pattern is, the more variance a component that is accounting for that pattern will explain. A component loading measures the correlation between the original station data time series and the component score time series. A loading of 1.00 indicates identical curve patterns, -1.00, opposite curve patterns, and 0, no discernible linear relationship. Component loading values were mapped and referred to without the decimal point for ease of reading. A component score measures the relative dominance of that component through time.

PCA output consisted of maps of component loadings and time series plots of component scores, one set for each of the first three components. Issues of how many components to retain were not dealt with in this study – three were always retained, regardless of the amount of variance they explained. Contouring of loading values was

performed using a distance-weighting interpolation scheme. In general, analysis was restricted to the first component because meso-scale patterns were typically found there. Other components identified local-scale patterns beyond the scope of this discussion.

3.3 Results

3.3.1 Comparing PCA and traditional synoptic analyses

Synoptic analysis: summer 1962

Alt (1987) analyzed the synoptic situation for June, July and August of 1962 and identified the main synoptic events. All descriptions of synoptic events for this summer are taken from this study. In general, the summer was dominated by a large ridge of high pressure centered over the western archipelago from the beginning of June until the middle of August. The ridge brought clear skies and a southerly flow to the entire archipelago. During two periods the influence of this ridge was diminished by surface low pressure systems, both of about one week in duration. Alt (1987) indicates that this summer was markedly different from other years in the existing record. Typically, a semi-permanent upper-level polar vortex is in place at the 50 kPa level in the vicinity of the archipelago, bringing cool flows and frequent incursions by small, surface low-pressure systems. In 1962, however, the vortex was positioned on the Siberian side, allowing a series of ridges to develop over the archipelago.

All stations (except Sachs Harbour, Fig. 3.4j) showed rising temperatures from early June to the third week in July (Fig. 3.4a-l). After this there was a ten-day period during which many stations recorded lower temperatures when a surface low-pressure

system was established. The ridge and warm temperatures were re-established during the first two weeks in August, after which the upper-level vortex moved back over the archipelago and summer ended.

The influence of these synoptic events on the regional temperature patterns may be observed in the PCA. Note that the removal of the regional mean time-series was not performed for these PCA because the original synoptic analyses had been conducted using unmodified data. The first component accounted for 60 percent of the variance (Fig. 3.5h). This large value indicated the seasonal cycle affecting all stations over the entire region during the period of observation. All stations had positive loadings on the first component with values greater than 60, with the exception of Clyde, with a value of 43 (Fig. 3.5a). The component scores for component one (Fig. 3.5b) traced the establishment of the ridge during the first week of June and the associated increase in temperature at stations throughout the region (Fig. 3.4a-l).

The incursion of a surface low-pressure system into the Baffin Bay area from June 8 to 22 was evident in the time series plot of the first component as a halt in the temperature increase. In the plot of the second component a negative bulge occurred during this period, corresponding to the life of the low (Fig. 3.5d). The eastern source of the low was evident in the component loadings (Fig. 3.5c). A large, positive loading on a negative score indicates a decrease in temperature coincident with the period of activity associated with the low, for example, Clyde (Fig. 3.5f) and Hall Beach (Fig. 3.4d). A large negative loading on a negative score indicates an increase in temperature, such as

observed at Isachsen (Fig. 3.4g) and Mould Bay (Fig. 3.4e). These results were consistent with a northeasterly flow brought on by a low located to the east.

The time-series for component one shows that warming was re-established in early July. Component two identified regional differences in the response to the ridge; stations in the west record temperature increases before they affect the east, however Clyde and Hall Beach felt the effects of the Baffin Bay low until July 13.

During this period, from July 3–7, a strong but localized surface low-pressure system brought lower temperatures and precipitation to Sachs Harbour (Fig. 3.4j) and higher temperatures to Alert (Fig. 3.4b). This event was not large enough to be recorded in the first or second components but did appear in the third component. The component scores (Fig. 3.5f) showed a strong peak in the first week of July, with Alert and Clyde positively correlated with the peak, and Sachs Harbour negatively correlated (Fig. 3.5e).

A rapid decrease in the scores for component one then occurred in the third week in July. The cause of the decrease was a low-pressure system in the Beaufort Sea that moved towards the islands and became linked by a trough to another low-pressure system in Hudson Bay, accompanied by a well-developed frontal system that tracked around the trough (Alt, 1987). Scores for component two peaked during this period, which indicated the northwest-southeast trend in the system's effects, apparent in the loading pattern (Fig. 3.5c,d). As the Beaufort Sea low moved in from the southwest, the western stations showed a decrease in temperature while the eastern stations showed no change or an increase. Component two rapidly reversed sign at the end of July (Fig

3.5d), indicating the effects of the warm sector on the western stations (Fig 3.4 e,g,j). The event of late July was also evident in the third component score, which showed a strong peak in the component score time series (Fig. 3.5f). This was driven largely by the strong decrease in temperature at Sachs Harbour (Fig 3.4j) and a large increase in temperature at Alert (Fig. 3.4b).

The strengthening of the ridge in early August caused a last period of warming across the region. The polar vortex returned by the second week in August, bringing northwesterly flow. These events were reflected in the scores for components one and two. Component one scores reflected the regional decrease in temperature (Fig. 3.5b). Component two scores (Fig. 3.5d) increased into the middle part of August, indicating a cool northwesterly flow that caused temperatures to decrease at stations in the northwest (Alert, Eureka, Mould Bay, Isachsen, Sachs Harbour – Fig. 3.4 b,c,e,g,j). These decreases were reflected as negative loadings on these stations (Fig. 3.5c).

Component one scores (Fig. 3.5b) increased as region-wide control of temperature strengthened, such as when a large ridge dominated the entire archipelago, and thus recorded events that affected the entire archipelago. They decreased when spatial heterogeneity increases, which occurs when smaller-scale climate processes dominate the temperature signal, such as traverses of low-pressure systems. Component two reflected events that affect smaller regions for shorter periods of time, e.g., northwesterly flow or a large low-pressure system. This component tended to become more significant whenever a low passed through.

Climatologically, different parts of the archipelago respond differently to a low-pressure or frontal system for two reasons. First, a system does not cover the entire region at once, but proceeds across the region over some period of time; because of this the timing of its effects differ. Second, the distance and position of a station with respect to the location of the system will determine how the station is affected. A station exposed to onshore flow will be moist and cool, whereas a station exposed to air descending from a mountainous area or moving over a large land area will observe a temperature increase. Component three recorded localized events or local accentuations of larger scale events. In general, the third component seemed sensitive to local responses of stations situated far from the central region and which were bordering other synoptic zones: the Beaufort Sea and Mackenzie region to the west/southwest, affecting Sachs Harbour; Baffin Bay to the east, affecting Clyde; and the polar marine environment of northeast Ellesmere Island, affecting Alert. It is possible that the relatively small amount of variance explained by the third component (Fig. 3.5h) indicated noise, however reasons have been presented here to support notion that it accounted for unique aspects of the temperature time-series of a few stations.

Synoptic analysis: summer 1964

Alt (1987) analysed the synoptic situation for the summer of 1964, identifying the main synoptic events. Unlike 1962, summer temperatures were 1° to 3° C below normal and all stations in the QEI received above normal snowfall. Beginning in mid-June a succession of low pressure systems at the 50 kPa level moved into the region resulting in a persistent northwest flow as well as numerous incursions by surface-based 'cold-

core' low pressure systems. Passage of these systems through the region continued until a strong ridge moved into the area in mid-August.

The PCA recorded specific features of the description found in Alt (1987). An early increase in temperature was halted by June 12 (Fig. 3.7b) as a 50 kPa low, with an associated surface low, positioned itself over the islands. The 50 kPa low moved to the southwest by the end of June. This gave a southerly flow to northern stations (Alert, Eureka) between June 28 and July 4, which responded with temperature increases (Fig. 3.6b,c), whereas stations in the south received northwesterly flow and did not observe an increase (Sachs Harbour, Cambridge Bay). This was recorded as a peak in the second component score (Figs. 3.7c,d).

Between July 13 and 18, two 50 kPa lows moved through the region, resulting in cooler temperatures at most stations (Fig 3.6 a,e-l). This was recorded as a decrease in the first component score (Fig 3.7b). After that was a brief period of warming around July 20 as a ridge briefly developed (Fig 3.6 a,d,f,h-l); this was seen as the peak in component one scores (Fig. 3.7b). After this the region was dominated by a succession of transient low-pressure systems until mid-August, when a strong ridge established itself in the area, bringing warming to all stations except Alert (Fig 3.6 a, c-l; Fig. 3.7b).

The preceding discussion shows that PCA results can be explained in terms of the synoptic processes which control daily temperature and precipitation. Even relatively small-scale and short-term events were recorded in the first three PCA scores. This suggests that PCA can be used to summarize the pattern of summer temperature.

3.3.2 July PCA analyses – AES data only

PCA were conducted on July mean daily data from AES stations for the month of July over the period 1954 – 1993 (Table 3.1 – Section 3.2.1). Four separate overlapping analyses were performed because the stations had records of varying length. Only the results common to all analyses will be discussed here. It must be borne in mind throughout the discussion that the interpretation of PCA results is fundamentally a subjective exercise, and that the components do not come with any sort intrinsic indication of their physical basis. At each station the period mean was subtracted from each datum before the PCA was performed; this removed the seasonal trend.

The most prevalent pattern on the component one loadings map was one described as 'northwest lobe'. This pattern appeared in each analysis (Figs. 3.8 – 3.11). Its defining aspect was a zero-line (the line separating the zone of positive loading from the zone of negative loading) that possessed a bulge which penetrated down into the islands from the northwest. Specifically, it runs from north Ellesmere Island towards the southeast, curving beyond and to the east of the archipelago then cutting across northwest Baffin Island as it moves across the lower-central region of the archipelago to the west/southwest, exiting the region to the west/southwest.

On the first component the stations in the central, west and northwest region formed a persistent group that included, when present, Eureka, Isachsen, Mould Bay, Resolute Bay, Rea Point, and Sachs Harbour. These stations possessed loadings ranging from 46 to 80. A second grouping of stations existed in the south and southeast that included, when present, Cambridge Bay, Shepherd Bay, Pelly Bay and Hall Beach.

These stations possessed loadings that were always opposite in sign to the northwest group, and which ranged from 37 to 78. In each analysis Alert, in the north, and Pond Inlet and Nanisisvik, in the east (present in one analysis - Fig. 3.11), were not associated with the first component. In physical terms this pattern was recording the persistent northwest flow off the Arctic Ocean. It penetrates southeast into the archipelago over the lowland area of the central islands, often reaching Resolute Bay, but in the north is blocked by topography. Other regional temperature analyses available in the literature have identified this feature (Alt and Maxwell 1990, Maxwell 1980, Bryson and Hare 1974). This component was important in several years (e.g. 1958, Fig. 3.8; 1968, Fig. 3.9).

Component two loadings maps (Figs. 3.8, 3.9, 3.10) were similar and exhibited a southwest/northeast gradient. In Figure 3.11 this pattern appeared in the third component. This pattern represented the fact that the temperature regimes that exist on the maritime north and east coastal zones of the CAA and in central Ellesmere Island, where Alert, Clyde, and Eureka were highly loaded, were often different from those which were noted on the western coastal zone and southern part of the CAA, where Sachs Harbour and Cambridge Bay loaded with moderate to high values and opposite sign. This component frequently showed a decreasing trend through the month of July (e.g. 1960 and 1967, Fig. 3.8; 1970, Fig. 3.9).

The third component in Figures 3.8, 3.9 and 3.10, and the second component in Figure 3.11, exhibited a bullseye pattern with large loading values in the central and southern part of the arctic islands. This represented the continental influence of the mainland land mass to the south of the CAA. Stations that loaded most consistently on this score were in the south and central regions of the CAA, and included Shepherd Bay,

Pelly Bay, and Resolute Bay. This component was not very important in the 1960's (Fig.3.8).

3.3.3 July PCA – AES and PCSP data

Due to the fact most PCSP stations were only at a given site for one season, analyses were performed on a year-by-year basis, i.e., as opposed to incorporating all years in one analysis, which is the usual method of application. The number and location of stations has varied over the years; this determined, in part, the spatial variability observed in the component loadings. Two sets of analyses were performed for each of five selected years. These years were selected because as a set they displayed a range of possible PCA results and they had some of the largest number of stations available. The first figure in each pair shows results of a PCA performed using only AES data (referred to as 'AES analyses'), while the second figure shows the results for a PCA performed on both AES and PCSP data (referred to as 'combined analyses'). For this section the assessment of PCA results was not be taken beyond a 'descriptive' analysis, i.e., specific physical mechanisms for each component were not deduced. Rather, the assessment was confined to comparing the results to those obtained from the PCA on the AES data alone.

When PCSP stations were added to AES stations for the analysis of 1975, the first and second components were switched. The results for the AES analysis show that the variance explained by components one and two was almost equivalent (component 1 = 26%, component 2 = 23%). The addition of PCSP data provided enough extra information to better identify the dominant pattern of July temperature for that year. The north bulge of the zero line in the loadings of component two (AES analysis – Fig. 3.12),

just north of Melville Island, was better specified in component one of the combined analysis (Fig. 3.13). In component two of the combined analysis Bathurst Island was seen to be more closely related to the eastern islands than to the western ones or to neighbouring Melville Island. The NW/SW pattern of component one of the AES analysis was not as apparent in component two of the combined analysis, due in part to the variance removed in the first component. The third component of the analysis of AES analysis showed three days when Resolute Bay and Arctic Bay were correlated, and showed that nearby stations were not necessarily related to these two.

In the 1983 analysis the PCSP stations served to emphasize the pattern established by the AES analysis (Figs. 3.14, 3.15), more so than in the 1975 comparison. Note that the signs of the component two scores and loadings switched between Fig. 3.14 and 3.15, again indicative of the extra information brought in by the PCSP data. Here AES stations correlated with nearby PCSP stations and the additional information in the combined analysis served to better locate the isolines. Note the deep penetration of the northwest flow during several days in early July (5th – 8th) and again in mid July (14th – 26th) with both periods followed by warming in the central islands.

In the 1984 analysis the addition of PCSP stations again emphasize the pattern exhibited in the AES analysis (Figs. 3.16, 3.17). For example, the high positive loadings on the second component of both analyses were centered on Resolute Bay, however the combined analysis showed that this extended to Bathurst, Devon and Somerset Islands.

In the 1990 analysis the PCSP stations formed a group of such coherence on Ellesmere Island that the variance explained by the first component exceeded that for the AES analysis, although the total for all three components was still lower (Figs. 3.18, 3.19). The pattern in the first component of both analyses was similar; the second components are only generally similar but both emphasize the similarities of the eastern arctic stations. The third component was dominated by an incursion of warm air on the 18th of July that was centered over northern Baffin Island.

The analysis of July 1991 temperatures again revealed the northwest/southeast difference in the arctic (Figs. 3.20, 3.21). Component two in both analyses illustrated a cooling period in the northwest in the early part of July, followed by a warming. The third component of the combined analysis illustrated that the central ridge, seen in the AES stations, was not exhibited at all stations.

3.4 Discussion

The first component identified the largest features of the regional temperature trend. There need not be a single physical agent responsible for each component. For example, although the first component usually represents regional-scale flow patterns, it reflects whatever climate processes that possessed a magnitude sufficient to affect a large area, such as large and/or persistent synoptic events or wide-spread heating.

Two main results were observed when PCSP data were incorporated. First, existing patterns observed initially in the AES analyses were emphasized. The level of detail was increased but the essential nature of the patterns remained discernible.

Second, areas of spatial coherence smaller than and in addition to those observed in the AES analyses were identified. These zones suggested the presence of meso-scale processes.

This analysis has demonstrated that PCSP data may be utilized to identify climatic activity at scales smaller than those identifiable using AES data alone. It has also shown that the PCSP data set may be used with confidence. Problems with the data would have appeared, if systematic, as a disruption to existing trends identified by the AES data. These problems were not evident. Instead, the use of PCSP data has provided support to the notion that persistent regional trends can be identified, as shown by the fact that the PCSP data were largely consistent with the patterns established by the AES analyses. Use of PCSP data also indicated the presence of smaller areas of spatial coherence in which meso-scale processes influence the climatic signal. An increase in spatial detail is expected if station density is increased, but only if there are smaller-scale climatic features present to be resolved.

The large-scale patterns observed in the AES analyses were similar to those observed in plots of isotherm normals produced using the same data (Figure 2.10, Chapter 2). This demonstrated the fact that PCA can be used to synthesize and summarize data sets to objectively arrive at conclusions similar to those obtained using other analytical techniques. Another point demonstrated by this analysis is that the spatial density of the input data set determines the scale of climate patterns, and thus the scale of the climate processes, that are discernible.

Chapter 4

A MODEL TO ESTIMATE SURFACE AIR TEMPERATURES IN THE CANADIAN ARCTIC ARCHIPELAGO

4.1 Introduction

A common way to map surface air temperature consists of interpolating a continuous surface between weather observing stations possessing long-term data records ('primary' stations). In the Canadian Arctic Archipelago (CAA) this portrayal of surface temperature is problematic for several reasons: topographic complexity; all observing stations of the Atmospheric Environment Service of Canada (AES) are located on the coast; and the distances between these stations are large. The mean distance between stations is 600 kilometers and the Queen Elizabeth Islands, the group of islands north of the Parry Channel, is served by only four stations (Figure 4.1).

Various authors have commented on the problems posed by using a sparse, location-biased network for the detailed portrayal of surface temperature (Maxwell, 1980; Jacobs, 1990; Barry, 1995). Efforts to increase the accuracy and spatial resolution of surface temperature plots in the archipelago have been undertaken and have included various approaches. Maxwell (1980) utilized information from historical short-term stations and his own experience to subjectively modify isotherms to depict the fact that the ice field/upland regions are usually cooler than the coastal margins. Alt and Maxwell (1990) employed non-standard, short-term weather observation data from several, more recent sources, including research camps supported by the Polar Continental Shelf Project and oil exploration camps, to increase spatial detail of the 30-year July normal temperature plot for the Queen Elizabeth Islands. Jacobs (1990) utilized a more objective approach to address the problem of lack of data in southern Baffin Island. He installed two automatic weather stations (AWS) in the interior and then mapped the result objectively. He further demonstrated how an AWS could be linked to AES weather stations using transfer functions which, after the AWS is removed, allow for the generation of data at a 'virtual' station. In this manner data could be generated for the virtual site as long as the AES site(s) it is linked to remain in operation. This assumes the transfer function remains constant through time.

In the cases presented above the goal was to increase detail by using short-term stations, including automatic weather stations, to provide information from locations within the large spatial gaps between the primary stations. By incorporating short-term observing stations station density is effectively increased, however some interpretation is required to incorporate the short-term data. This is a labour-intensive process requiring

experienced analysts. Jacob's approach of transfer functions works only where stations are available for comparison with AES stations and its accuracy decreases with distance from the predictor (i.e. AES) station(s) and with physiographic complexity of the terrain. It is also limited to the time period for which it was constructed, for example, a transfer function derived using summer conditions would be poorly suited for use in the winter.

Another approach that may be used to improve detail in temperature plots is to generate temperature estimates by modeling processes that govern temperature. There are two main approaches to climate modeling: theoretical and process-based. Theoretical models reproduce the atmospheric system using mathematical representations of physical laws that govern it (e.g. general circulation models). A process-based model utilizes a semi-empirical approach that combines existing data from a few primary stations with the known effects of physical laws, that is, the processes at work in the atmosphere, to render for remote locations temperature estimates that are related to a primary station. Estimates made using existing data are guided by the application of the processes for selected time periods. For example, if an on-shore flow is known to be affecting a given location, then an advective process is applied to refine the temperature estimates.

The development of a process-based model to estimate temperature requires identifying those processes which control temperature at a scale appropriate to the model. This involves consideration of what climatic elements are available at the primary stations and how the required processes may best be captured using the available data.

The issue at hand is thus which processes should be modeled. Given that the general objective was to increase detail in temperature plots of the CAA, processes which control temperature at the meso-scale, and their manifestation in terms of model operation, were considered. Site elevation and proximity to a coast are two examples. Although these are physical aspects of a location, and not 'processes', knowing the elevation and distance from a coast for a site provides a way to parameterize the effects of processes that are of interest, and to thus include them in the model.

The link between the physical aspects of a location, used as model parameterizations, and the processes they represent is outlined below. A broad feature of the troposphere is a general decrease of temperature with height. This is the net result of two processes: 1) the atmosphere is heated from below, via radiation from the surface, and 2) temperature and pressure are directly proportional (gas law) which means that, when the mass distribution of the atmosphere is considered (highest pressure at the surface decreasing continually upwards), temperature must decrease with elevation. For these reasons there is a link between elevation and temperature. Another important process that can modify the vertical temperature structure of the atmosphere is advection. This is especially true when two different surface types, such as ocean and land, are in close proximity. This juxtaposition, combined with favorable wind conditions (e.g., on-shore), causes the characteristics of one surface to be imposed on another with an intensity that decreases steadily from the point of contact. For this reason there is a link between site proximity to the coast and temperature.

The primary goal of this chapter, therefore, was to examine the following assumption: The elevation and proximity to the ocean of a given land location can be used to represent meso-scale processes acting to modify summer air temperature determined by synoptic-scale factors.

A model was constructed to explore this assumption. Estimates of temperature were obtained from vertical temperature profiles, as determined by the environmental lapse rate, and by a digital terrain model. The latter provided the spatial context for the model, including point topology, for use in resolving wind/advective processes. Model estimates were compared at specific locations with observations taken at short-term, non-standard weather stations and residuals calculated. Accuracy was assessed using plots of the residuals to identify areas showing consistent deviation.

The following sections describe the model operation, the upper air data used to drive the model, model results, and assessment. The closing section considers future work required to improve the accuracy of the model. Model code is found in Appendix C.

4.2 Model Description and Operation

The surface air temperature model is a process-based model implemented at a resolution of one kilometer by one kilometer. The mechanisms by which physical processes were accommodated in this model include:

- Use of an environmental lapse rate from the time period being modeled to define the rate of temperature change with elevation. Specific lapse rates were measured by rawinsonde ascents at weather stations situated in the study region.

- Use of an average, low-level wind direction and velocity, supplied by rawinsonde ascents, to determine the extent to which coastal zones were modified by onshore breezes.
- Stipulation of surface temperatures for major icefields using a linear modification of the base temperature estimate.

The basis of the model was a digital elevation model (DEM) of the arctic archipelago, organized as a matrix of 1996 columns by 1833 rows, subset from the United States Geological Survey GTOPO 30 arc-second DEM of the world. Hereafter any matrix of similar dimension will be referred to as an *image*, irrespective of what type of values its grid cells contain; all images used in the model were matrices that possessed these dimensions, with the exception of the wind filters. Each point (pixel) in an image was about 1 square kilometer; each pixel possessed a single datum that varied according to the image. For example, in the case of the DEM, each pixel possessed a single elevation datum. At each pixel an initial estimate of surface temperature was generated directly from the vertical temperature profile. For pixels containing an icefield a modified temperature value was applied. The resultant winds were then determined from the rawinsonde ascents and a series of filters were applied to the initial temperature estimate to implement the effects of wind. These steps are described in detail below. Within the discussion references such as "images were added together" refer to processes carried out on corresponding pixels in each of the image matrices involved in the operation.

Limited use of a digital elevation model as a basis for modeling climatic parameters has been made. The most significant contribution in this field to date has

been the extensive work led by Daly (Daly et al. 1994, Daly et al. 1997, Daly and Johnston 1998, Johnson et al., 2000). He has constructed a complex topoclimatic model, called "Parameter-elevation Regression on Independent Slopes Model" (PRISM), that parameterizes processes to generate estimates of several climatic variables at 4 km resolution. His PRISM model was recently selected to produce all the maps for the new *Climate Atlas of the United States*. The PRISM model differs from that described here in two respects: PRISM does not use upper-air data to aid the interpolation process, and PRISM is dependent on observational data from surface stations to help generate and refine base estimates. Wilmott (1995) used a DEM driven by a standard lapse rate in conjunction with a high-resolution climate network to generate high spatial resolution estimates of mean annual air temperatures for the United States. Santibanez et al. (1997) report on a study in which surface observations at a reference station were combined with a DEM and satellite radiative data to forecast minimum temperatures at high spatial resolution. Goodale et al. (1998) mapped, onto a DEM, polynomial surfaces generated using the normals of various climatic parameters. These high resolution normals surfaces were then used as climatic input into ecosystem carbon and water cycling models. Dodson and Marks (1997) modeled the lapse rate and apply it to a DEM in order to accommodate the effect of elevation on temperature. This was applied in a study to produce a high resolution plot of temperature in a mountainous region.

4.2.1 Initial estimate of surface air temperature

Initial surface air temperature values were estimated using an environmental lapse rate averaged over time and space for the period under consideration. Elevation data from the DEM were used to generate estimates on a pixel by pixel basis. The procedure is outlined below.

Vertical profiles of dry-bulb temperature were obtained from twice-daily radiosonde ascents for the period under consideration. Each individual ascent was fitted with a 5th order polynomial. The six coefficients that describe each polynomial curve were averaged to produce a mean curve of temperature with elevation for the period for the station. From each ascent, data up to 5000m were extracted to ensure the resulting curve extend well beyond the maximum elevation in the region (2620m – Barbeau Peak, Ellesmere Island). This procedure was repeated for each upper air station used in the study (Table 4.1).

Table 4.1: Upper air stations used to generate regional estimates of environmental lapse rate. All stations operated by the Atmospheric Environment Service Canada unless located in Alaska.

Station	Lat (N)	Long (W)
Alert	82° 20'	62° 30'
Eureka	80° 00'	85° 56'
Mould Bay	76° 14'	119° 20'
Resolute Bay	74° 43'	94° 59'
Iqaluit (Frobisher Bay)	63° 45'	68° 33'
Hall Beach	68° 47'	81° 15'
Cambridge Bay	69° 07'	105° 01'
Barter Island (Alaska)	71° 18'	156° 47'
Barrow Point (Alaska)	70° 05'	143° 36'

A curve was fitted to the data to simplify calculation of a mean ascent profile, especially given the fact that the sampling interval differed between ascents, which would have made the calculation of a mean ascent curve difficult. Use of an equation to represent the mean ascent also simplified the subsequent estimation of temperature values using elevation data. A fifth-order polynomial was used because many of the ascent profiles showed a surface inversion to be present, and a high order polynomial can capture this feature while also following closely the rest of the ascent profile. In addition, because these ascent curves were used as the basis for estimates derived

across the study region, it was important that erroneous upper air temperature observations be filtered out. The process of fitting a curve to the ascents accomplished this. Examples of curve fits are presented in Figure 4.2a,b.

Many of the stations listed in Table 4.1 often exhibited surface-based inversions. Although they were found to be smaller in magnitude than those found in the winter (Bradley et al., 1992; Maxwell, 1980), it was deemed necessary to remove the inversion signal for the theoretical reasons outlined below. A breakdown of inversion frequency by station and height is presented in Table 4.2; mean ascent curves for the period of the model runs (Table 4.4) were considered for this table.

A surface inversion can be caused either by negative surface radiation balance or by the advection of warmer air aloft/cooler air below. In winter a strong surface inversion is present at most locations, caused by the large negative surface radiation balances experienced during the arctic night (Bradley et al., 1992; Jackson, 1959). In summer, however, the net surface radiation balance is positive, meaning that the inversion is unlikely to be radiation-based. Since the cause of the summer inversion cannot be radiation based it is likely due to advection. The source of cool air at low levels is probably the ocean and sea-ice filled channels, because in summer land surfaces are more likely to possess positive balances and are therefore more likely to be warmer than the atmosphere above. Therefore, it is assumed that a summer surface inversion at a coastal location is a local-scale effect of limited extent caused by proximity to a large body of water often covered with sea-ice. This effect must be removed if the

temperature ascent curve is to be representative of inland regions that are presumably free of this influence.

Table 4.2: Frequency of inversions observed in mean ascent curves for the periods of model runs. Value class is height of the inversion maximum in meters above the ground. GT refers to "greater than 700 meters" (observed only at the Alaska stations).

Upper air station	No inversion	<100	<200	<300	<400	<500	<600	<700	GT	
Alert	10	2	-	2	2	-	-	1	-	
Barrow Point (Alaska)	4	-	-	-	1	5	1	2	4	
Barter Island (Alaska)	2	1	-	-	2	2	2	3	5	
Cambridge Bay	14	-	3	-	-	-	-	-	-	
Eureka	4	3	5	5	-	-	-	-	-	
Iqaluit (Frobisher Bay)	9	1	4	2	-	1	-	-	-	
Hall Beach	6	3	2	2	2	2	-	-	-	
Mould Bay	14	1	1	-	-	-	-	1	-	
Resolute Bay	8	2	1	3	1	1	-	1	-	
Totals	71	13	16	14	8	11	3	8	9	153

Removal of the inversion involved first detecting the inflection point on the curve where it begins sloping into the inversion component. This was done using a global-maximum detection algorithm driven by a Newton-Raphson root solving methodology (McCracken and Dorn, 1964). Next, data were generated from this point down to the surface using the rate of change that existed in the curve above the inversion. The new data series then had an equation fit to it to re-establish the set of polynomial coefficients (Figure 4.2a).

The manner in which the inversion was removed was determined by observing the difference between mean surface temperatures from a coastal location and an inland location. Several such comparisons were performed using PCSP camps located near, but inland of, three of the AES upper air stations used in the model (Alert, Eureka and Resolute Bay). The PCSP stations were situated far enough inland that their

temperature records were assumed to be unaffected by coastal proximity (>20 km). The temperature value at the inland site was thus considered to be the surface temperature a lapse rate would possess if not affected by the inversion. To check this assumption the inland station temperature was plotted on the mean ascent curve. In all cases examined the inland station temperature was found to lie at a point that represented a direct extension of the upper air profile from a point located above the inversion. Based on this, use of a direct extension of the ascent curve to the ground level was adopted as the coastal effect correction. The nature of the correction is detailed in Figure 4.2a.

Once a final ascent curve was available for each station, the next step was to interpolate the polynomial coefficients from each station such that a set of polynomial coefficients was available at every pixel. Each coefficient was interpolated individually, one image per coefficient. Interpolation onto the grid was performed using an inverse distance weight procedure with distance decay set to a factor of 2; this was selected to provide a balance between local weighting and range of influence. The paucity of observing sites and a lack of spatial structure (e.g., no clustering) did not warrant use of more specialized interpolation techniques (e.g. McCullagh, 1981; Shepard, 1965).

Temperature values were obtained by solving the equation on a pixel-by-pixel basis using elevation data as the independent (x) value. This produced six images of intermediate results, one for each coefficient. These images were added together to give a region-wide estimate of surface temperature that reflected the environmental lapse rate without any advective modification. A schematic representation of this process is presented in Figure 4.3.

At this point temperatures for ice fields were estimated. Havens et al. (1965) demonstrated an average 'ice field cooling' factor of about 3° C using data from two meteorology stations, one on top of an ice field and the other near the ice field on a non-ice surface. This cooling factor was adopted for use and applied in the model in the following way. A binary image of ice field locations, in which pixels possessing an ice field were assigned 1 and all other pixels were assigned 0, determined where the cooling factor was applied. At these locations the pixel was assigned a new value that consisted of the initial temperature estimate minus the cooling factor.

4.2.2 Coastal advection

In the presence of a steady on-shore flow, coastal regions can experience an advective modification of temperature (Malone, 1960). The magnitude of a resultant change in temperature is dependent on the temperature difference between the surface of the ocean and the land surface. Inland penetration is dependent on the strength of the prevailing on-shore flow and the distribution of high ground.

In the model, the influence of winds was accommodated by first preparing a resultant wind value for each pixel. In summary, this involved extracting wind data from a specified level (90 kPa) in the upper air ascents, interpolating these data across the region, grouping the interpolated data by direction and velocity, and determining the magnitude and location of maximum possible coastal proximity modification. This was expressed as a value, in the form of a percentage, that indicated the degree to which the temperature of the ocean and the land temperature at the pixel under consideration should be mixed. Note that topography was not explicitly considered, although the

potential impact of wind was reduced in areas of low ocean fetch, such as a fiord. The 'wind-effect' image was then combined with the previously created temperature surface. The resultant image was a temperature surface on which the effects of advection of ocean-influenced temperature has been incorporated. The procedure is outlined in detail below.

First, wind direction and velocity data from the 90kPa level was extracted from upper air ascents. For each ascent at each station the resultant wind vector was obtained and the u and v (N/S; E/W) components resolved. A mean for the time period was obtained by averaging the u and v components individually; these results were then interpolated separately. The interpolation method used was identical to that used to interpolate the polynomial coefficients. The 90kPa level was selected to represent a balance between steady upper level flow and surface perturbations. This level was high enough (~900m) to avoid most topography and to possess the steady characteristics of winds at higher levels, yet low enough to represent reasonably the direction and velocity of winds felt at the surface.

After interpolation a resultant wind vector was reconstructed at each pixel. Based on these values the image was classified into the categories listed in Table 4.3.

Table 4.3: Wind direction and velocity classification categories.

Direction (° true N)	Class	Velocity (km/hr)	Class
336 – 45	North	0	Calm
46 – 135	East	1 – 13	Low
136 – 225	South	14 – 26	Medium
226 – 335	West	27 +	High

Four direction classes were used because it was felt that more classes would represent an unrealistic level of detail given the extent to which the input wind data are generalized in time and space. The velocity classes were established based on examination of the wind regimes at several PCSP stations; the selected classes roughly represent the breakdown of wind velocities that were observed. Note that when the wind was zero there was no modification of the coastal environment. Thus 13 categories exist: calm, low speed from the north, low speed from the west, and so on.

An observation made while examining the wind regimes at the PCSP stations was that, when wind velocities exceeded 40 km/hr, cool temperatures were observed irrespective of wind direction, even if the wind was blowing from a warm inland region. Although this may be expected during the passage of surface low-pressure systems and associated cold front or cold sector, these conditions were also observed on occasions when total cloud cover was low. This leads to the speculation that strong turbulent mixing events, during which air at the surface is rapidly exchanged with air hundreds of meters above the surface, were being observed (Stull, 1988; Malone, 1960). The net result of such events was cool temperatures at the surface. Although of general interest, such events were not specifically modeled because they were confined to periods of high wind velocities and usually did not persist more than a few days.

The 13 wind categories formed the basis for the selection of a matrix 'filter' that was applied to the DEM to determine the representation of the wind in the model. The filter is a square matrix of odd-numbered rows that is symmetric and opposite on either side of an axis, determined by the wind direction, that extends through the center cell.

The physical size of the matrix was determined by the velocity class of the wind, as were the specific values it took. The structure of the filter was modified from what is known as a 'directional' filter in geographic information systems (GIS) literature (Bonham-Carter, 1994). A similar use of a filter to determine the 'exposure' of underwater reefs to the prevailing wind was reported in Bonham-Carter (1994).

The filter was then applied to a binary representation of the original DEM, in which land pixels were assigned the value 1 and ocean pixels 0. At each pixel in the binary image the values of the filter matrix were multiplied with the values of binary image pixels in the region coinciding with the filter. The resulting products were totaled and the value, expressed as a percentage, was assigned to the pixel being considered (Fig. 4.4). Thus when the filter was located completely in the ocean the result was zero; when completely on land the result was also 0. However when the filter straddled a coastal margin the filter values were multiplied by 1 over land and are not countered by an identical value of opposite sign because the rest of the filter was multiplied by zero over the ocean. The result assigned to the pixel under consideration was then a positive or negative value of varying magnitude, depending on the orientation of the coast, the position of the filter with respect to the coast, and the distance of the pixel away from the coast. The values in the filter were arranged to emulate a gradual transition between coastal and inland temperature regimes, based on observations made at several PCSP stations.

The maximum possible value for wind effect that can be returned by the filter matrix was set to 90, or 90%. It was felt that only the few hundred meters nearest the

ocean would completely take an ocean temperature. Given that the resolution of the DEM is one kilometer, it was felt that mitigation of the maximum possible (100%) was warranted.

In many coastal areas of the world when a strong prevailing flow is absent and when sky conditions are largely cloud-free it is common for a diurnal circulation driven by temperature differences between the land and water surface to commence. The strength of this thermally-driven circulation, termed a 'sea-breeze', is proportional to the difference in temperature between the juxtaposed surfaces and to the instability in the low-levels of the atmosphere (Atkinson, 1981). Two reports did indicate the possibility of high-latitude sea breezes. Maxwell (after Jackson, 1969) noted their presence in the arctic and King and Turner (1997) described them in the Antarctic. It was thus decided to conduct a search for the occurrence of sea breeze events with a view to explicitly including them in the model if they were found to occur with sufficient frequency.

The location was selected for a sea breeze search was AES Eureka, one of the warmest stations in the region, and was conducted by looking for a closed ellipse pattern on hodograph plots (plots of wind speed and direction employed to search for these events) (Atkinson, 1981). None were observed for the periods examined. In the High Arctic, temperature differences between the land and water surfaces are often less than for coastal regions to the south, and the occurrence of atmospheric instability in the arctic is less frequent and not as deep when it is present. Therefore, sea-breeze circulations should occur infrequently and, when they do occur, they will be weak and of limited spatial extent. The observed lack of occurrence of these events at AES Eureka

seemed to corroborate the general theoretical concepts. For this reason it was decided not to explicitly include sea-breeze in the model.

The air temperature over the ocean was set at 2°C for all water pixels in the image (Maxwell, 1980). The values in the wind effects image were percentages. They and the values from the temperature estimates image were combined in the following equation:

$$T_r = W \times T_s + (1 - W) \times T_L,$$

where T_r = resultant temperature value for a pixel,

W = wind modification value (as a percentage),

T_s = air temperature over the ocean surface, and

T_L = air temperature over the land surface obtained from the polynomial-based estimate.

The final output of a model run was an image of estimates of the mean surface air temperature for the period of the model run. Values were estimated for all land surfaces at a resolution of one square kilometer over a region encompassing all the islands in the Canadian arctic archipelago, Boothia Peninsula, and some of the mainland north coast.

4.2.2 Sensitivity analysis

Sensitivity of model temperature estimates to variations in the application of factor parameterization was explored by successively altering the magnitude of application of

the various parameters. A series of seven additional combinations of factor parameterizations, different from the original combination, were assembled to explore the question of model sensitivity. Each combination was run on five separate years. The same five years were used in each case so that inter-comparisons could be performed. Three main model factor parameterizations were targeted: the inversion removal algorithm, the sea-surface temperature (SST), and the wind effect. Specific alterations to the inversion algorithm consisted of three approaches to removal in addition to that already described: a "low-slope" removal, in which the slope removal line was drawn from the point at which the lapse rate begins a stable rate of cooling, a "peak-point" removal, in which the slope removal line was drawn vertically down from the point of maximum warming to the surface, and "none", in which no alteration to the observed lapse rate was performed (Fig. 4.2b). The inversion removal algorithms represent a gradation in the magnitude of inversion removal, from a maximum in the original model ("high-slope" removal) to no alteration ("none"). For SST, the region-wide constant value was replaced by a map of mean observed SST (Fig. 4.5 - Maxwell, 1982). The observed SST was used in two sets of runs, one set using the "low-slope" removal inversion algorithm and another using the "peak-point" removal algorithm. The existing wind effect was increased in strength, such that its influence could be felt twice as far inland as in the original model. Two sets of runs were conducted with the strengthened wind, one using the "low-slope" removal inversion algorithm and a constant SST and another using the "low-slope" removal inversion algorithm and the observed SST. In the case of the wind sensitivity runs SST was varied, rather than the inversion removal algorithm, because the effect of wind is influenced as much by the SST as by the inversion, and it was deemed important to explore how the two alterations function together. Table 4.4

summarizes model runs by parameter set. In all, 35 additional runs were conducted to explore model sensitivity. Table 4.5 details the five time periods for which each sensitivity run was conducted, along with reasons for their selection.

Table 4.4: Nature of alterations to runs for sensitivity analyses.

Parameter	Nature of alteration	Total runs
Inversion removal	High-slope removal (original)	18
	Low-slope removal	5
	Peak-point removal	5
	No modification to temperature profile	5
Sea surface temperature	Constant over entire region (original)	-
	Variable over region with Low-slope removal	5
	Variable over region with Peak-point removal	5
Coastal wind effect	Moderate effect application (original)	-
	Maximum effect application with Low slope removal and constant SST	5
	Maximum effect application with Low slope removal and variable SST	5

Table 4.5: Periods for which sensitivity analyses were run.

Period	Reason for selecting
9 – 22 July 1974	Large zone of negative residual in original model
21 July – 03 August 1978	Lack of inversions for the time period
28 June – 11 July 1983	Large zone of positive residual in original model
16 – 29 July 1984	Large number of stations available for verification
16 – 29 July 1988	Climatologically warm summer

4.3 Results

Model runs were conducted in all years for which adequate PCSP data were available for verification, 1974 – 1990, less 1989. In addition to the model results, mean temperature values were calculated for all stations present during the period of the run. Model estimates for pixels coinciding with the station locations were extracted and residual values, which consisted of mean temperature values calculated by the model subtracted from mean temperature values determined from observed data, were

calculated. The residual values were contoured and plotted to gauge the performance of the model. Eighteen runs using what will be referred to as the "original" model parameters were performed. Another 35 runs were performed to gauge model response to variations in factor parameterization (the sensitivity runs) and will be considered apart from results generated by the original runs.

4.3.1 Run periods

The majority of original model runs covered a 14-day period; one run was performed in each year. The starting date was selected to maximize the number of stations available for verification. A length of 14 days was selected because it was long enough to ensure that mean values calculated from observations were reasonably free of deviations caused by random variations, yet short enough to maximize availability of PCSP stations for testing purposes. The minimum number of days a time-series should possess was determined using a Monte-Carlo simulation in which the variability associated with using successively larger numbers of observations to estimate a mean value was recorded. The variability was stable by the 14 day period, indicating the station is likely a good estimator of temperature at that point. In addition to the 14-day runs, several one-month runs (July) were conducted to explore model response when run for longer time periods.

Model start dates vary from year to year and were timed to maximize availability of observing stations against which to test model output. However, all runs fell mostly within the month of July and all were within the high arctic summer climatological period, considered to begin sometime in June and continue at least until mid-August. Specific 14-day run periods are listed in Table 4.6; dates for the full-month (July) runs are listed in Table 4.7.

Table 4.6: Model run dates for 14-day periods.

Year	Dates
1974	09 – 22 July
1975	08 – 21 July
1976	04 – 17 July
1977	19 July – 01 August
1978	21 July – 03 August
1979	09 – 22 July
1980	23 July – 05 August
1981	29 June – 12 July
1982	08 – 21 July
1983	28 June – 11 July
1984	16 – 29 July
1985	06 – 19 July
1986	09 – 22 July
1987	05 – 18 July
1988	16 – 29 July
1989	13 – 26 July
1990	09 – 22 July

Table 4.7: Additional run periods

Year	Dates	Feature of run
1974	01 – 31 July	Full month run
1977	01 – 31 July	Full month run
1979	01 – 31 July	Full month run

4.3.1 Surface air temperature estimates

Output from selected model runs, including 1974, 1976, and 1988 is presented in Figs. 4.6 – 4.8 and 4.18 – 4.19. Values have been rounded to the nearest whole degree Celsius. As expected, cooler temperatures were found at higher elevations in the eastern arctic. The general north-south gradient was modeled by warmer temperatures in the central islands (south Victoria Island).

4.3.2 Residual analysis, original model runs

Residuals were contoured and plotted for all original model runs and are presented in Figures 4.9a–p. Negative residuals indicated an estimated temperature that exceeded

the observed mean; positive residuals indicated that an observed temperature mean exceeded the model estimate. Any model estimate that possessed a residual in the range -1.4 to $+1.4$ was considered to be a good estimate. The criterion used to establish this was based on the standard deviations of the observed data against which the model estimates were compared, using the following logic. Observed mean temperatures are by-products of climatic processes operating at several scales: synoptic, meso, local, and random. The model is designed to capture synoptic- and meso-scale signals. This means that spatial variability caused by local-scale and random fluctuations are noise for the model. The level of background noise is the level below which the model cannot be expected to function. This variability is contained in the standard deviation associated with the observed mean. Standard deviations for the observed time-series, which covered the two-week periods for which the model was run, ranged between $1.5 - 2.8^{\circ}\text{C}$. Therefore, it was decided that if the residual was less than the standard deviation of the observed values, then the estimate was likely free of systematic deviation and possessed only fluctuations caused by local or random processes. The final cut-off value used was, however, at the lowest end of the range of standard deviations for the observed time series because it is recognized that some variation is attributable to large-scale processes, such as weather systems, the signals of which can be captured by the model. As well, establishing this range of acceptability allowed a workable assessment of the model to be conducted – the range was large enough to avoid highlighting every little residual, which consisted of random and local-scale fluctuations that the model cannot capture, yet small enough to portray zones of persistent residual that were caused by model shortcomings, facilitating meaningful model diagnostic work.

The total number of available residual values for original runs was 386; the distribution is listed in Table 4.8. Most values fell within the range ± 1.4 and there were more than twice as many under-estimations as over-estimations.

Table 4.8: Residual totals by residual classification.

Residual	Number	%
Less than - 1.5	99	25.6
-1.4 to +1.4	245	63.5
Greater than +1.5	42	10.9

Table 4.9 gives a breakdown of the mean and standard deviations, by year and overall, of the mean observed temperature, mean estimated temperature and residual. Residual results ranged from very close to the mean with little variation (e.g. 1976) to large deviations from the mean with large variation (e.g. 1974). The general trend is towards negative mean residual values.

Table 4.9: Mean, standard deviation, and sample size of observed values, model estimates, and residuals for each year of model run and for all years.

	Observed		Estimate		Residual		N
	mean	Std	Mean	std	Mean	std	
1974	5.5	2.6	7.8	1.8	-2.3	2.5	26
1975	3.4	1.9	4.1	1.3	-0.6	1.4	27
1976	2.9	1.8	2.9	1.3	0	1.1	23
1977	5.9	2	6	1.6	0	2	27
1978	4.9	1.7	5.4	1.8	-0.5	2.3	23
1979	4.2	1.8	5	1.5	-0.8	1.5	26
1980	4.7	1.3	5.3	1.2	-0.6	1	24
1981	6.4	2.2	7.4	1.6	-1	2.3	24
1982	6.8	1.6	7.3	1.4	-0.4	1.8	21
1983	3.5	1.9	2.9	1.8	0.6	1.3	23
1984	5	2.4	4.5	2.1	0.5	1.5	33
1985	5.9	2	7.4	2.2	-1.5	1.8	24
1986	4.3	2.4	4.8	2.2	-0.4	1.2	26
1987	6.2	2.6	5.8	2.4	0.4	1.5	14
1988	8.3	2.2	9.6	2.4	-1.3	2.8	23
1990	6.3	2.1	5.7	2.6	0.6	1.7	22
OVERALL	5.21	2.4	5.69	2.5	-0.5	2	386

In addition to the aggregate comparisons, residuals were also summarized according to size of spatially continuous zones. The idea behind this approach was that the larger a residual zone the more likely it was a result of a systematic shortcoming in the model, and should thus be a focus for attention. In the maps, the 141 residual values that fell outside the acceptable range formed a total of 59 spatial zones, of which 33 possessed a single station and 26 possessed multiple stations. Table 4.10 presents information about the frequency of occurrence of zones of different sizes (in terms of numbers of stations), and Table 4.11 presents similar information broken down by residual type, that is, positive or negative. Residual zones that encompassed more than 2-3 stations represented 18% of all observed zones.

Table 4.10: Frequency of occurrence of residual zones possessing certain number of stations.

Number of stations in zone	Number zones of this size in plots	Percent
1	33	56
2	12	21
3	3	5
4	3	5
5	2	3
6	2	3
7	2	3
8	1	2
14	1	2

Table 4.11: Frequency of occurrence of residual zones possessing certain number of stations, by residual type.

Zones of negative residual			Zones of positive residual	
Number of stations in zone	Number zones of this size in plots	Percent	Number zones of this size in plots	Percent
1	18	55	15	58
2	4	12	8	30
3	3	9	-	-
4	2	6	1	4
5	1	3	1	4
6	1	3	1	4
7	2	6	-	-
8	1	3	-	-
14	1	3	-	-
Total	33		26	

Six of 16 years possessed residual zones that exceeded 3 stations in size: 1974, 1979, 1981, 1985, 1988 and 1990. In each case except 1990 the residuals were negative and were situated largely in the northern part of the archipelago. Small zones of one and two stations constituted the majority, together accounting for 77 percent of all zones (Table 4.10). Separation into residual type indicated that zones possessing one or two stations accounted for 67 percent of negative residuals but 88 percent of positive residuals. The three largest zone classes were represented only by negative residual groups (Table 4.11).

Some persistent features were noted in the residuals plots. A negative residual was frequent in the northwest, present in 6 of 11 possible years. Large and small zones of positive residual were frequent in the north, on Ellesmere and Axel Heiberg Islands (1977, 1978, 1982, 1983, 1984, 1988, 1990), and often in the eastern parts of Ellesmere (1977, 1988, 1990). Hall Beach exhibited residuals on a regular basis (ten of 16 possible

times), as did Sachs Harbour to a lesser extent (four of fourteen possible times). Residuals at these two stations did not favour either sign.

On a larger scale, residuals of both signs tended to be found in the Queen Elizabeth Islands and much less so south of the Parry Channel. All the large residual groups were located in the QEI; in fact, on only two occasions did a large group possess any connections with the southern islands. If Hall Beach and Sachs Harbour were not included, the southern islands would possess very few residuals.

Three years were selected for a one month run: 1974, 1977 and 1979. Years were selected from those that possessed numerous residuals that exceeded the acceptable range. Comparative plots are presented in Figures 4.10a–f. In each case the total number of residuals in the full month plot exceeded that of the 14-day plot. In 1974 the intensity of the negative zone decreased but its extent increased slightly to encompass Hall Beach, which lost its positive residual. In 1977 some of the zones decreased but two more single-station residuals were introduced. In 1979 several more stations in the west were added to the negative residual zone, while in the east it intensified, although two small zones in the south disappeared.

It must be noted, when looking at the residual plots, that a residual zone generated by one station can visually dominate an entire region where there are no other stations to counter it. For example, in 1988, Resolute Bay is linked to a negative residual in the north and the resulting pattern appears to dominate the entire northwest, even though there are no stations in that region.

4.3.2 Residual analysis, sensitivity runs

Residuals were contoured and plotted for all sensitivity model runs (Figs. 4.11 – 4.15). The total number of available residual values for each sensitivity run was 137; the distribution is listed in Table 4.12. In general, none of the sensitivity combinations investigated yielded a clearly superior result. The most significant result was that the original inversion removal algorithm ("high-slope") was inferior to any of the sensitivity combinations. Applying different inversion removal algorithms while maintaining the constant sea-surface temperature resulted in skewed residual groupings: skewed negative using the high-slope and low-slope removals, and skewed positive using the peak-point and "none" residuals, although the total number of residuals in the acceptable category did not change by much. Similar skewed results were observed when using the stronger wind field with the low-slope removal. The most even distribution of residual values was obtained using the low-slope inversion removal with a variable sea-surface temperature, however, it must be noted that this method also yielded one of the lowest numbers of residuals in the acceptable range. The largest number of residual values in the acceptable range was obtained using no inversion removal, however it also generated the most highly skewed residuals set.

Table 4.12: Residual totals by model factor parameterization set.

Residual	Factor parameterization set							
	Orig.	Low slope	Peak Point	None	Var SST, low slope	Var SST, peak slope	Max wind, const SST, Low slope	Max wind, Var SST Low slope
Less than - 1.5	33	31	18	15	31	16	33	33
-1.4 to +1.4	76	87	89	92	80	86	83	85
Greater than +1.5	19	19	30	30	26	35	21	19
Total obs	128	137	137	137	137	137	137	137

Table 4.13 gives a breakdown of the mean and standard deviations, by year and overall, of the mean observed temperature, mean estimated temperature and residual. More detail regarding how the different alterations affected specific years is provided in the table. It is apparent from Table 4.13 that much of the observed change occurred in two years: 1974 and 1988.

Table 4.13: Mean, standard deviation, and sample size of observed values, model estimates, and residuals for each year of model run and for all years.

		Observed		Estimate		Residual		N
		Mean	std	Mean	std	Mean	std	
Low slope	1974	4.9	2.5	6.8	2.1	-1.9	2.4	23
	1978	4.9	1.6	4.9	1.1	0	1.1	25
	1983	3.4	1.9	3.1	1.8	0.3	1.2	24
	1984	5	2.3	4.2	2	0.9	1.6	34
	1988	8.8	2.7	9.6	3	-0.8	3.1	25
	Overall	5.4	2.2	5.7	2	-0.3	1.9	
Peak slope	1974	4.9	2.5	5.4	2.1	-0.5	1.7	23
	1978	4.9	1.6	4.7	1	0.2	1.2	25
	1983	3.4	1.9	3	1.7	0.4	1.4	24
	1984	5	2.3	4	1.7	1	1.5	34
	1988	8.8	2.7	8	2.3	0.8	2.6	25
	Overall	5.4	2.2	5	1.8	0.4	1.7	
No removal	1974	4.9	2.5	5.3	2.2	-0.3	1.5	23
	1978	4.9	1.6	4.7	1.1	0.2	1.2	25
	1983	3.4	1.9	2.9	1.6	0.5	1.5	24
	1984	5	2.3	4.1	1.8	0.9	1.4	34
	1988	8.8	2.7	7.5	2.4	1.3	2.5	25
	Overall	5.4	2.2	4.9	1.8	0.5	1.6	
Variable SST with Low slope	1974	4.9	2.5	6.6	2.2	-1.7	2.2	23
	1978	4.9	1.6	4.8	1.2	0.1	1.2	25
	1983	3.4	1.9	3	1.8	0.4	1.2	24
	1984	5	2.3	4.2	2.1	0.9	1.6	34
	1988	8.8	2.7	9.7	3	-0.8	3.1	25
	Overall	5.4	2.2	5.6	2.1	-0.2	1.9	
Variable SST with Peak point	1974	4.9	2.5	5.3	2.2	-0.4	1.8	23
	1978	4.9	1.6	4.5	1.1	0.4	1.2	25
	1983	3.4	1.9	3.4	2.9	0.5	1.3	24
	1984	5	2.3	3.9	1.8	1.1	1.5	34
	1988	8.8	2.7	7.8	2.3	1	2.5	25
	Overall	5.4	2.2	4.9	1.8	0.5	1.7	
Max wind with Low slope and constant SST	1974	4.9	2.5	6.8	2.1	-1.9	2.1	23
	1978	4.9	1.6	4.8	1.2	0.1	1.1	25
	1983	3.4	1.9	3.2	1.8	0.2	1.1	24
	1984	5	2.3	4.2	2	0.8	1.6	34
	1988	8.8	2.7	9.7	3.1	-0.9	3.1	25
	Overall	5.4	2.2	5.7	2.0	-0.3	1.8	
Max wind with Low slope and variable SST	1974	4.9	2.5	6.8	2.2	-1.8	2.1	23
	1978	4.9	1.6	4.8	1.2	0.1	1.1	25
	1983	3.4	1.9	3.1	1.9	0.2	1.2	24
	1984	5	2.3	4.2	2	0.7	1.6	34
	1988	8.8	2.7	9.7	3.1	-0.9	3.1	25
	Overall	5.4	2.2	5.7	2.1	-0.3	1.8	

The general response to progressive decrease of the magnitude of inversion removal was for the large negative residual zones to be reduced in size and/or broken up into smaller zones, e.g. Fig. 4.11a. In the 1974 series (Figs. 4.11a-d) there were a few, localized residual zones at the end of the progression (Fig. 4.11d), however in the 1988 series the negative residuals were replaced by large positive residual zones at the end of the progression (Fig. 4.15d). Very little change was noted in the years 1978 (Figs. 4.12a-d), 1983 (Figs. 4.13a-d) and 1984 (Figs. 4.14a-d).

Changes introduced by alteration of inversion removal were more significant than those resulting from altering the SST or wind fields. In general, model response to alterations in these factors consisted of sporadic, low-magnitude changes when contrasted with changes to inversion removal. An example of the changes to model output brought about by introducing a variable SST field is presented in Fig. 4.16.

4.4 Discussion

Estimates of surface air temperature generated by the model were largely reflective of elevation and the shape of the vertical temperature profiles. The central region of Ellesmere Island, guided by the profile from Eureka, was often relatively warm, as were areas in the south, especially Cambridge Bay. Alert, Mould Bay, and Resolute Bay were often relatively cool, although the northwest archipelago through to the Resolute Bay area was often cooler than Alert.

4.4.1 Original model

Original model estimates were compared to observed temperatures from PCSP and AES stations to generate residuals. 63.5% of residuals fell within a range of +/-1.4.

When the original model was in error it tended to overestimate temperatures. This fact emerged both in numerical assessments of residual frequency distribution and in visual assessments of the residuals plots. There were several instances of large areas in which multiple stations possessed negative residuals. This was most likely a systematic problem, although it is possible that each of the stations had been overestimated for local reasons (see below) and that the contouring procedure has connected the residuals together to form what looks like a large residual zone.

The large negative residual zones were observed to form two groups: a northwest group and another group along the eastern edge of the archipelago. A northwest negative residual group was observed in Figures 4.9a – d, h, and o. An eastern negative residual group was observed in Figures 4.9a, f – h, and l.

There were most likely two separate reasons for a northwest negative residual which were dominant in different years but which yielded a similar net effect. The first was a problem with deep inversions. In 1974, Mould Bay and Resolute Bay possessed unusually deep inversions, as did Eureka and Alert (Figure 4.18a). In these cases, the inversion removal algorithm would have generated a large correction factor. This would be especially true if the inversion gradient was large. The correction factor likely overestimated in these situations, and thus there was a corresponding overestimation by

the model. Another example of this occurred in 1988 (Fig. 4.9o). When compared to the inversion plot for the same year (Fig. 4.17o) many stations showed a negative residual in the areas of deep inversion. This appeared to be the source of negative residuals at Hall Beach, for example, 1978 (Figs. 4.9e, 4.17e), 1982 (Figs. 4.9i, 4.17i), 1984 (Figs. 4.9k, 4.17k), 1985 (Figs. 4.9l, 4.17l), and 1988 (Figs. 4.9o, 4.17o). The second cause for a northwest negative residual zone was likely a lack of inversion at Mould Bay. In only two years did Mould Bay exhibit an inversion: 1974 and 1988 (Table 4.2). This means the residual occurred in several years for which Mould Bay did not possess an inversion. In several of these years the vertical temperature profile from Mould Bay exhibits a warming in the lowest 500m. In appearance it has the form of an air mass whose temperature profile contains an inversion but which has undergone a short period of heating from below. This scenario of short-term modification of an existing air mass likely occurred at Mould Bay in the following manner. Maritime-arctic air masses, entrained in the prevailing northwest flow, move off the Arctic Ocean and over Prince Patrick Island. While traversing the island the air mass, which often possesses a deep inversion, is heated from below by the relatively warm land surface such that, when it has reached Mould Bay on the east side of the island, the temperature structure of the lowest level has been modified. The surface inversion has been removed and the model does not modify the curve. However within the model that several degrees of warming still acts like a corrected inversion. The reason this tends to be a problem more often in the northwest is detailed below.

In several cases the model over-estimated when the resultant wind was zero. An examination of specific situations revealed a number of shortcomings that could have

affected estimates made for the near-shore environment. A related problem is that the model over-estimated temperatures on small islands, such as Prince Leopold Island or Seymour Island. Coastal over-estimation probably occurred because, without an onshore wind component, the model did not apply any cooling to coastal pixels. For some cases this was probably a result of the model over-simplifying the resultant wind vector, such that it did not apply a small net on shore wind component that had been present and that had effected a cooling. This can be improved in the model by parameterizing some correction factor for times when net wind equals zero. The situation at small islands is simply that the entire island exists within the maritime arctic air mass. The radiative heating available from the small land area of the island is insufficient to modify the cool lowest levels of the atmosphere. Application of a corrected vertical temperature profile, which has been designed for the interiors of large islands, thus overestimated at these locations. Furthermore, with respect to the modified inversion condition at Mould Bay, a temperature profile that possesses an inversion but which has undergone a brief period of heating from below, such that the inversion is removed at the lowest levels, will also cause the model to overestimate. Finally, an overly literal application of wind effect was another wind-related cause of model over-estimation at coastal locations. Wind effects are applied in the model strictly along cardinal directions, when in fact a range of directions, centered on the calculated resultant, would probably be more realistic.

There are several topographic features of the northwest that can also increase the likelihood of overestimation: small islands are more numerous here than elsewhere in the archipelago, there is a lack of blocking topography, and there is a lack of

sheltering relief, in the form of fiords. The problem with small islands has been discussed. A lack of blocking topography means that there is no highland area to deflect the northwest flow, with the result that all areas in the northwest are subjected to the full cooling effects of the maritime arctic air mass. This is in contrast to locations in the vicinity of AES Eureka, situated behind the blocking influence of the Axel Heiberg highlands. A lack of sheltering relief refers to the fact that coastal locations in fiords are not exposed to the same magnitude of cooling as locations on an exposed coast. Several instances in which the model under-estimated in fiord areas were noted. Another problem that may contribute to a model over-estimation in the northwest is the value used for the air temperature over the ocean, that is, 2°Celsius, is probably too high for an ocean that is usually ice-covered. Another potential source of error, not confined to the northwest, was cooling due to proximity to an ice field. This was more difficult to confirm, however a model over-estimation at a station in southeast Devon Island could be the result of an ice field effect.

Residuals along the eastern edge of the archipelago most likely occurred because none of the upper air profiles is characteristic of this region. Alert, while on the coast near the eastern coastal region, is located at the extreme northern limit of this area, which limits the representativeness of its profile. Furthermore, the nature of the interpolation procedure is such that south of the central-east coast of Ellesmere Island the influence exerted by Eureka's vertical profile will exceed that of Alert. This can be tested by incorporating upper air ascents from Thule AFB (Greenland) and seeing if this improves the estimates. However, the large area of open water in the Davis Strait, and

the possible influence of the Greenland ice cap, may render data from Thule as unrepresentative as that from Eureka or Alert.

Positive residuals occurred less frequently than negative residuals. The majority of positive residuals were found in zones with one or two stations (Table 4.11). That fact, coupled with the spatial distribution of residuals, did not indicate systematic model underestimation. The most likely situation in which positive residuals would occur is during periods of low cloud cover and wind speed when surface heating is greatest. Given that a free-air temperature estimate of a land location at the same elevation is not always the best estimator of surface temperature (Maxwell, 1980), model underestimation should be more common during periods of low cloud cover and wind speed. However, examination of specific situations revealed, for single station residuals, that this was not necessarily the case and that there were several instances when model underestimation occurred when reported cloud cover was high. Stations within the large positive residual zones in the central islands in 1977 (Fig. 4.9d) and in the north in 1983 (Fig. 4.9j) did possess occurrences of low cloud cover amounts. A more thorough investigation of the relationship between clouds, wind and temperature should be undertaken to ascertain whether the inclusion of cloud cover, perhaps based on wet bulb depression from the ascents, should be considered.

There were few situations in which the modeled wind effect could be assessed. In 1981 (Fig. 4.9h) three stations – Hall Beach, a station on southwest Ellesmere Island and a station on northwest Devon Island – were located within an onshore wind zone that correctly modified their temperatures. In 1988 the wind effect captured most of the

coastal signal, despite an inversion (Fig. 4.17o), to reapply the coastal cooling. The estimate was slightly into the positive residual range (Fig. 4.9o), indicating that the amount of cooling was too large. The cause of this was probably a value for air temperature above the ocean that was not set high enough. The nature of the wind filter design also contributed to model over-estimation of small islands. To activate its full cooling potential, many of the pixels on one side of the filter must be underlain by land. Small islands possess few pixels, are able to activate little wind potential, and thus are subjected to less cooling by wind.

In several years there were large areas in which the model apparently overestimated temperature (e.g. 1975, 1976, 1977, 1979; Figs. 4.9b–d, f), but these large areas consisted of just a few stations, all situated on the coast, that were joined to form a contiguous zone by contouring. The likely causes of the over-estimation have been detailed, however it is important to note that these sources of error are confined to coastal regions, and that the much larger majority of land areas will not be susceptible to these problems. This suggests that the model is capable of generating accurate temperature estimates for large areas of the arctic.

Areas well represented in the model included the central, south-central and west/southwest regions. When considering model output and its evaluation, it is useful to remember that this model is producing areally averaged temperature estimates on a grid with a resolution of 1 kilometer by 1 kilometer over an area that is almost 2 million square kilometers using only two dynamically scaled parameters, and that the error is

being gauged, using an accuracy of one tenth of one degree Celsius, not by comparison against areally averaged observed data, but against point observations.

In general, it is important to realize that the majority of test sites were situated near the coast. This means identifying possible occurrences of model under-estimation is likely poorly represented. This fact should be borne in mind before considering a larger number of negative residuals to be an indication of a fundamental flaw in model functioning.

4.4.2 Sensitivity analysis

Sensitivity analysis involving modification to the inversion removal algorithm had the largest impacts in those years for which inversions were observed (1974 and 1988). A successive decrease in the magnitude of inversion removal resulted in corresponding decreases in the magnitude and occurrence of negative residuals. This suggests that the original model algorithm over-compensated when a deep inversion was present, resulting in temperature estimates that were too large.

Although the occurrence of over-compensation was reduced using inversion removals of lower magnitude, a problem of model under-estimation began to appear. In the "peak-point" and "none" removal results for 1974, under-estimation was almost as common as over-estimation, but these residuals did not form a continuous zone similar to the large negative residual zone formed using the original algorithm. In 1988 larger zones of positive residual appeared, notably in the central north and the west. 1988 was a warm year in the region surrounding AES Eureka (Edlund and Alt, 1989). In general, under-estimated sites were situated inland or in sheltered areas, such as at the head of a fiord.

In 1988, however, sites that were exposed to a stronger maritime influence, such as that to the east of Ellesmere Island or to the northwest, were still over-estimated by the model, even in the absence of an inversion correction (Fig. 4.15d), which means that an erroneous inversion removal algorithm can not take full blame for model over-estimation.

A large under-estimation that occurred in north Ellesmere Island in 1988 (Biederbick Lake) provides a good example of the potential problem that can exist with inland sites. The main estimator for this site is the temperature profile from AES Alert. A location like Alert, unlike a sheltered, inland site, is cooled by two mechanisms: cool air advection, and blocking of insolation by low-level cloud. Biederbick Lake, however, experienced many days of low wind and cloud, which most likely allowed the site to realize its maximum potential warming, and which made it different enough from Alert that even a corrected temperature profile was unable to reproduce its observed temperature.

It should be noted that in 1974 the stronger wind field increased the negative residual zone in the northwest, however in this case the inversion removal was probably the largest contribution to model over-estimation. In general, modifications to SST and the wind field, although producing little discernible effect in the residuals plots, had large impacts on the model estimations (Figs. 4.16 and 4.17). As alluded to before, a lack of testable effect does not mean these modifications are or are not improvements, just that there are few stations in place to adequately test the output.

Overall, the model was most sensitive to the manner in which inversions were handled. In terms of area of effect, modifications to this parameter also had the largest effect. An overall assessment of this effort as a model-building exercise must be guided by the residuals. They demonstrated consistency, in that for all combinations of parameterization, the majority of residuals were within the acceptable range, and where they exceeded the acceptable range, examination of their patterns suggested logical physical causes. The general accuracy and physical interpretability of the model results suggest that this is a promising avenue for arctic climate research, both to generate high spatial resolution temperature data and to explore the physical processes that control the climate in this region.

4.5 Future work

The assumptions governing the operation of this model include:

1. The surface air temperature at a location is largely a function of its elevation and proximity to a coast,
2. Estimates of surface temperature may be made using environmental lapse rates obtained from upper air ascents,
3. The occurrence of inversions in the upper air ascents is a by-product of coastal proximity and must be removed to accurately estimate temperatures at interior locations,
4. The reason coastal environments are relatively cool is the advection of marine air over the land surface, and
5. Coastal advection events may be captured using resultant winds from upper air ascents.

The course of this work has demonstrated that some of these assumptions are reasonable while others require varying degrees of modification. The primary assumptions, number 1 and 2 above, have proven a sound basis on which a model of this type may operate. The general validity of assumption number 3 was demonstrated because systematic model under-estimation occurred when the two weakest inversion removal algorithms were applied ("peak-point" and "none"). The "high-slope" and "low-slope" removals worked well and possessed problems that were limited to certain upper-air temperature profile patterns; specifically, if either steep inversions or a relatively steep low level increase in temperature occur, the model will systematically overestimate. Assumption number 4 was demonstrated to be only partially correct: even when there was no mean onshore flow coastal regions are cooled by proximity to the ocean. This meant that small islands which did not possess the land area to modify the temperature structure of the lower atmosphere were not accurately modeled with an ascent curve 'corrected' for coastal proximity. Assumption number 5 seemed reasonable the few times it could be directly evaluated. Sensitivity analyses added weight to the importance of dealing appropriately with Assumption 3. This was supported when considering the sensitivity testing using realistic SST.

In general the model has performed satisfactorily. Physical mechanisms have been suggested for most residuals. There are additional areas in which work would stand to benefit the model, including:

1. Improve the near-shore low-wind cooling effect, such that unless there is a strong wind from the land side, the model should have some sort of coastal cooling at all land pixels beside an ocean pixel;
2. Improve the spatial distribution of air temperature estimates above the ocean by incorporating a more detailed sea surface temperature or mean sea ice conditions map;
3. Use the vertical profile of dew point temperature in conjunction with the dry bulb temperature profile to get an estimate of clouds; and
4. Incorporate a more detailed ice field/glacier map and model downslope katabatic effects in large glacial valleys.

This model has shown that a significant amount of the meso-scale climatic variability of the arctic may be explained by a few factors operating on the complex topography of the arctic. Future work will improve the model by better parameterization of the climate of the lower atmosphere and condition of the surface. Another consideration for future work is to expand the time periods over which the model can currently operate by using vertical temperature profiles and winds generated by general circulation models (GCM) or the NCEP/NCAR reanalysis upper-air gridded data set. A model of this nature also provides a means of rendering GCM output in high spatial resolution because the low-resolution, gridded vertical temperature fields generated by the GCM can be used as upper-air stations in the model.

Chapter 5

SUMMARY

This thesis has presented an analysis of the summer meso-scale climate of the Canadian High Arctic. The goal was to improve existing knowledge of the climate using a model of surface air temperature and a new data set from a 20-year period. The underlying assumption, that modification of the synoptic-scale temperature climate by elevation and distance from a coast will largely determine the temperature at a location, was evaluated. The model, which was semi-empirical process-based in operation, was built to evaluate the assumption. Model results were assessed using a data set of non-standard weather observations from the Polar Continental Shelf Project. The process of assembling the data set and reviewing it for errors was detailed, as was a principal components analysis performed using the data to ascertain its veracity.

The following results have been obtained in this study:

- 1) A new data set of summer temperatures was assembled from disparate sources and made available to the research community. These data were based on

measurements collected and archived by the Polar Continental Shelf Project over the years 1974 through 1992. Data are available from across the Arctic Archipelago and many are available from the interiors of the islands. Although a few records span the entire season or several seasons, many are short-term. Some preliminary results indicated that these records constitute a useful supplement to existing AES standard weather observations (Chapter 2, Appendix A). The largest single problem was missing station information, which occurred in almost 10% of the data. Comparative plots were done which showed coherency with nearby stations.

- 2) A series of principal components analyses of surface temperature in the Arctic Archipelago (Chapter 3), based in part on the new database, demonstrated the following:
 - a) The timing and spatial extent of synoptic events, including tracing influence of large features, such as ridges of high pressure over the entire archipelago, were identified in the PCA results. Smaller features in space and time, such as the traverse of low-pressure systems and local deviations from the general temperature pattern of the archipelago, caused either by local forcing agents or by selective exposure to neighbouring weather regimes, were also identified by this analysis. This indicated that PCA of daily data can be used to trace the effects of synoptic climate processes.
 - b) Using a multi-year analysis of primary (AES) station data large-scale features of the temperature climate were identified. The most dominant was a persistent flow from the northwest. The next most prevalent feature was a difference in temperature regimes between the eastern/northern areas and the

western/southern areas. Finally, a pattern indicating the influence of the continental land mass on temperatures in the south central part of the archipelago was identified. These represent the major large-scale features of the temperature climate of the Canadian Arctic Archipelago.

- c) Comparison of analyses incorporating PCSP data with analyses of AES data alone further demonstrated the usefulness of the PCSP non-standard data set for use in regional climate analysis. Information provided by the additional data verified spatial patterns of summer temperature determined using the AES data because the PCSP stations showed general regional coherence. The PCSP data also exhibited coherence at smaller spatial scales which suggested that the PCSP data were able to provide some measure of meso-scale climate activity, although specific physical explanations for the components were not elaborated on in this study. The indication of meso-scale structure warrants further analyses of this kind to permit a better understanding of the arctic climate and its possible relation to physiographic factors. Potential approaches include applying methods of classification of other multivariate analyses to these data, use of transfer functions to extrapolate these records in time, thus permitting longer analyses, or many other methods being developed that depend on the sampling to give indication of errors as well as mean values of the signals.
-
- 3) The underlying assumption of this study was explored by developing a model that was based on the summer, synoptic-scale temperature signal of the AES upper-air stations in the High Arctic (Chapter 4). This signal was modified to include effects of distance from the coast, altitude and presence of ice caps. The resulting climatic

patterns captured much of the signal of July temperatures in the arctic. This was indicated by the fact that the majority (64%) of residuals between the predicted temperatures and point estimates from AES and PCSP stations, for all combinations of parameterization, fell within a reasonably narrow range. Some discrepancies were evident that seemed to be due to two factors: 1) model response was sensitive to the parameterization of the temperature profile, and specifically the manner in which an inversion was removed, and 2) the presence of persistent cloud cover was not accounted for. Future work with this model can address these problems. For example, wet bulb depression of the soundings input to the model can be used to introduce a cloud cover which will serve to reduce the coastal effect. The method by which the inversion is removed can be improved, possibly guided by the presence of cloud. A field study could be undertaken in which temperature is monitored at a number of stations along a transect perpendicular to the coast for a long period, to better understand the relation of coastal modification of temperature to wind speed, wind direction, and cloud cover.

An important application of the model described here will be to translate various scenarios of global warming, produced using GCMs, into a high spatial resolution temperature signal (i.e. what is termed 'downscaling') for detailed local- and meso-scale impact work in disciplines such as hydrology, urban climatology, or ecology. This can be accomplished using the gridded vertical temperature output from the GCM as surrogate rawinsonde data. This type of approach, that is, replacing upper-air stations with gridded upper-air air data, can also serve to extend the geographical application of the model to areas of the world that are not well-served by upper-air stations. In these cases gridded

data sets based on modern data, such as the NCEP (National Centers for Environmental Prediction) upper-air reanalysis data, available for anywhere in the world, could also be used.

Although it is frequently felt that the arctic climate is variable, this study has also indicated that processes operating at the synoptic- and regional-scales represent a large component of arctic temperature time series signals, and that local variability is not so dominant as to overwhelm the large-scale coherency. Meso-scale climate patterns can be explained to a large degree as modification of this signal by physiographic factors. This study has explained some aspects of arctic climate. It has indicated some paths for future work and presents an input data set for use in factor analysis.

References

- Agnew, T., L. Spicer and B. Maxwell. 1987. *Marine Climatological Atlas - Canadian Beaufort Sea*. Atmospheric Environment Service, Canadian Climate Centre Report No. 87-14. 196p. Downsview.
- Alt, B.T. 1987. Developing Synoptic Analogues for Extreme Mass Balance Conditions on the Queen Elizabeth Island Icecaps. *Journal of Climate and Applied Meteorology*, 26(12), 1605-1623.
- Alt, B.T. 1979. Investigation of summer synoptic climate controls on the mass balance of Meighen Ice Cap. *Atmosphere-Ocean*, 17(3), 182-199.
- Alt, B.T. 1975. *The Energy Balance of Meighen Ice Cap, Volumes 1 and 2*. Energy Mines and Resources Canada, Polar Continental Shelf Project:Ottawa, 60p. and 101p.
- Alt, B.T. and B. Inkster. 1987. Networking, Arctic Weather Style, *GEOS*, 16(3), pp6-10.
- Alt, B.T. and J.B. Maxwell. (submitted). Modern Climate: Arctic Overview, chapter.1.1.1 in M. Garneau (ed.) *Environmental Response to Climate Change in the Canadian High Arctic*. Geological Survey of Canada Bulletin 529.
- Alt, B.T. and J.B. Maxwell. 1990. The Queen Elizabeth Islands: A case study for arctic climate data availability and regional climate analysis, pp294-326 in *The Canadian Arctic Islands, Canada's Missing Dimension*. Ottawa, Nov.21-24, 1987. Proceedings, Volume 1.
- Alt, B.T., C.L. Labine, D.E. Atkinson, A.N. Headley and P.M. Wolfe. (submitted). Modern Climate: Fosheim Peninsula: Automatic Weather Station Results, chapter 1.1.2 in M. Garneau (ed.) *Environmental Response to Climate Change in the Canadian High Arctic*. Geological Survey of Canada Bulletin 529.
- Alt, B.T., C.L. Labine, A. Headley, R.M. Koerner, S.A. Edlund. 1992. *High Arctic IRMA [Integrated Research and Monitoring Area] Automatic Weather Station Field Data 1990-91*. Geological Survey of Canada, Open File 2562, 274p.
- Andrews, R.H. 1964. Meteorology and Heat Balance of the Ablation Area, White Glacier, Canadian Arctic Archipelago, summer 1960 (Lower Ice Station; 78°26'N 90°39'W, 208m). *Axel Heiberg Research Reports: Meteorology #1*. McGill University: Montréal.
- Atkinson, B.W. 1981. *Meso-scale Atmospheric Circulations*. Academic Press: London, 495p.
- Atkinson, D.E. 1994a. Aspects of Local- and Regional-Scale climatologies in the Canadian Arctic Islands: Coastal Effect at AES Eureka. Program and Abstracts, 28th Canadian Meteorological and Oceanographic Society (CMOS) Congress, Ottawa, Ontario, May 30 – June 3, 1994, p. 145.

- Atkinson, D.E.** 1994b. Polar Continental Shelf Project Aviation Meteorology Program: Current Status and Future Directions. Unpublished report submitted to the Ministry of Natural Resources, Canada. 100pp.
- Bonham-Carter, G.F.** 1994. *Geographic Information Systems for Geoscientists: Modelling with GIS*. Pergamon Press: New York, 398p.
- Bradley, R.S. and F. Keimig.** 1992. Climatology of Surface-Based Inversions in the North American Arctic. *Journal of Geophysical Research*. 97(D14), 15,699-15,712.
- Bradley, R.S. and J. England.** 1979. Synoptic Activity of the Canadian high arctic. *Geographiska Annaler*. 61A, 187-201.
- Bradley, R.S., F.T. Keimig and H.F. Diaz.** 1992. Climatology of Surface-Based Inversions in the North American Arctic. *Journal of Geophysical Research*, 97(D14), 15,699-15,712.
- Bryson, R.A. and F.K. Hare.** 1974. *Climates of North America: vol. 11, World Survey of Climatology*. Elsevier Publishing Company: New York, 370p.
- Cogley, J.G. and S.B. McCann.** 1976. An Exceptional Storm and its Effects in the Canadian High Arctic. *Arctic and Alpine Research*, 8(1), 105-110.
- Courtin, G.M. and C.L. Labine.** 1979. Microclimate Studies on Truelove Lowland, pp73-104 in L.C. Bliss (ed.). *Truelove Lowland, Devon Island, Canada: A High Arctic Ecosystem*. University of Alberta Press: Edmonton, 714p.
- Daly, C. and G.L. Johnson.** 1998. PRISM spatial climate layers: their development and use. <http://www.ocs.edu/prism/prisguid.pdf>.
- Daly, C., G.H. Taylor, and W.P. Gibson.** 1997. The PRISM approach to mapping precipitation and temperature. In: *Proc., 10th AMS Conf. on Applied Climatology*, American Meteorological Society, Reno, NV, Oct. 20-23. 10-12.
- Daly, C., Neilson, R.P., and Phillips, D.L.** 1994. A statistical-topographic model for mapping climatological precipitation over mountainous terrain. *Journal of Applied Meteorology* 33: 140-158.
- Dodson, R. and D. Marks.** 1997. Daily air temperature interpolated at high spatial resolution over a large mountainous region. *Climate Research*, 8, 1-20.
- Diaz, H.F. and D.C. Fulbright.** 1981. Eigenvector analysis of seasonal temperature, precipitation and synoptic-scale system frequency over the contiguous United States. Part I: Winter, *Monthly Weather Review*, 109, 1285-1304.
- Edlund, S.A. and B.T. Alt.** 1989. Regional Congruence of Vegetation and Summer Climate Patterns in the Queen Elizabeth Islands, Northwest Territories, Canada. *Arctic*, 42(1), 3-23.
- Edlund, S.A., B.T. Alt and K.L. Young.** 1989. Interaction of climate, vegetation and soil hydrology at Hot Weather Creek, Fosheim Peninsula, Northwest Territories. *Current Research Part D, Geological Survey of Canada*, 89(1D), 125-133.

- Etkin, D. 1988. *Assessment of the Polar Continental Shelf Project weather database*. Unpublished Canadian Climate Centre (CCC) report, 45p.
- Goodale, C.L., J.D. Aber and S.J. Ollinger. 1998. Mapping monthly precipitation, temperature, and solar radiation for Ireland with polynomial regression and a digital elevation model. *Climate Research*, 10, 35-49.
- Havens, J.M. 1964. *Meteorology and Heat Balance of the Accumulation Area, McGill Ice Cap, Summer 1960*. McGill University: Montréal.
- Havens, J.M., F. Müller and G.C. Wilmot. 1965. *Comparative Meteorological Survey and a Short Term Heat Balance Study of the White Glacier, Summer 1962*. McGill University: Montréal, 68p.
- Headley, A. 1990. A comparison of wind speeds recorded simultaneously at three metres and ten metres above the ground. *Report No. 90-1*, Canadian Climate Centre, Atmospheric Environment Service. 36p.
- Higuchi, K. 1988. Trends in Year-to-Year Fluctuations in Surface Air Temperature, Canadian Arctic. *Physical Geography*, 9(2), 139-150.
- Jackson, C.I. 1969. The Summer Climate of Tanquary Fiord, N.W.T.: A five year review. *Publication in Meteorology No. 95 Geophysics: Hazen 35*, Defence Research Board / Arctic Meteorology Research Group, McGill University: Montréal.
- Jackson, C.I. 1959. Coastal and Inland Weather Contrasts in the Canadian Arctic. *Journal of Geophysical Research*, 64(10), 1451-1455.
- Jackson, J.E. 1991. *A User's Guide to Principal Components*. John Wiley and Sons: Toronto. 569p.
- Jacobs, J.D. 1990. Integration of Automated Station Data Into Objective Mapping of Temperatures for an Arctic Region. *Climatological Bulletin*, 24(2), 84-96.
- Jacobs, J.D. and L.D. Grondin. 1988. The Influence of an Arctic Large-Lakes System on Mesoclimate in South-Central Baffin Island, N.W.T., Canada. *Arctic and Alpine Research*, 20(2), 212-219.
- Jolliffe, I.T. 1987. Rotation of Principal Components: Some Comments, *International Journal of Climatology*, 7, 507-510.
- Jolliffe, I.T. 1986. *Principal Components Analysis*. Springer-Verlag: New York. 271p.
- Jones, P.D., T.M.L. Wigley and P.M. Kelly. 1982. Variations in Surface Air Temperatures: Part 1. Northern Hemisphere, 1881-1980. *Monthly Weather Review*, 110(2), 59-70.
- Johnson, G.L., C. Daly, G.H. Taylor and C.L. Hanson. 2000. Spatial Variability and Interpolation of Stochastic Weather Simulation Model Parameters. *Journal of Applied Meteorology*, 39, 778-796.
- Kahl, J.D., M.C. Serreze and R.C. Schnell. 1992. Tropospheric Low-Level Temperature Inversions in the Canadian Arctic. *Atmosphere-Ocean*, 30(4), 511-529.

- Keeler, C.M.** 1964. Relationship Between Climate, Ablation and Runoff on the Sverdrup Glacier, 1963, Devon Island, N.W.T. *Research Paper 27, Arctic Institute of North America*. McGill University: Montréal, 80p.
- Kelly, P.M., P.D. Jones, C.B. Sear, S.G. Cherry and R.K. Tavakol.** 1982. Variations in Surface Air Temperatures: Part 2. Arctic Regions, 1881-1980. *Monthly Weather Review*, **110**(2), 71-83.
- King, J.C. and J. Turner.** 1997. Mesoscale systems and processes, Chapter 6 in *Antarctic Meteorology and Climatology*. Cambridge University Press: New York, 409p.
- Koerner, R.M.** 1979. Accumulation, ablation, and oxygen isotope variations on the Queen Elizabeth Islands icecaps, Canada. *Journal of Glaciology*, **22**(86), 22-41.
- Labine, C.L.** 1994. Meteorology and Climatology of the Alexandra Fiord Lowland, in **Svoboda, J. and B. Freedman (eds.)**. *Ecology of a Polar Oasis: Alexandra Fiord, Ellesmere Island, Canada*. Captus University Publications: Toronto, 268p.
- Labine, C. L., B.T. Alt, D. Atkinson, A. Headley, R.M. Koerner, S. A. Edlund, M. Waszkiewicz.** 1994. *High Arctic IRMA automatic weather station field data 1991-92: part 1, documentation, part 2, plots*. Geological Survey of Canada, Open File 2898, 190p.
- Larsson, P.** 1963. The Distribution of Albedo over Arctic Surfaces. *Geographical Review*, **53**(4), 572-579.
- Larsson, P. and S. Orvig.** 1962. Albedo of Arctic Surfaces. *Publications in Meteorology #54*. Arctic Meteorological Research Group, McGill University: Montréal.
- Leathers, D.J. and B.L. Luff.** 1997. Characteristics of Snow Cover Duration Across the Northeast United States of America, *International Journal of Climatology*, **17**, 1535-1547.
- LeDrew, E.F.** 1988. Developmental Processes for Five Depression Systems Within the Polar Basin. *Journal of Climatology*, **8**, 125-153.
- Legates, D.** 1993. The Effect of Domain Shape on Principal Components Analysis: A reply, *International Journal of Climatology*, **13**, 219-228.
- Legates, D.** 1991. The Effect of Domain Shape on Principal Components Analysis, *International Journal of Climatology*, **11**, 135-146.
- Mallants, D and J. Feyen.** 1990. Defining Homogenous Precipitation Regions by Means of Principal Components Analysis, *Journal of Applied Meteorology*, **29**, 892-901.
- Malmgren, B.A. and A. Winter.** 1999. Climate zonation in Puerto Rico based on principal components analysis and an artificial neural network, *Journal of Climate*, **12**, 977-985. (abst.)
- Malone, T.F (ed.)**. 1960. *Compendium of Meteorology*. American Meteorological Society: Boston, 1334p.

- Maxwell, B.** 1982. The Climate of the Canadian Arctic Islands and Adjacent Waters, vol. 2. *Climatological Studies No. 30*. Atmospheric Environment Service, Environment Canada: Downsview. 589p.
- Maxwell, B.** 1980. The Climate of the Canadian Arctic Islands and Adjacent Waters, vol. 1. *Climatological Studies No. 30*. Atmospheric Environment Service, Environment Canada: Downsview. 532p.
- McCracken, D and W. Dorn.** 1964, *Numerical Methods and Fortran Programming*, John Wiley and Sons: New York. 457 pp. (reference on pgs 124-159, chapter 5 "Roots of Equations". QA76.73.F25M315 1964.)
- McCullagh, M.J.** 1981. Creation of Smooth Contours over Irregularly Distributed Data Using Local Surface Patches. *Geographical Analysis*, 13(1), 51-63.
- McFadden, J.D. and R.A. Ragotzkie.** 1967. Climatological Significance of Albedo in Central Canada. *Journal of Geophysical Research*, 72(4), 1135-1143.
- Müller, F. and N. Roskin-Sharlin.** 1967. *A High Arctic Climate Study on Axel Heiberg Island, Canadian Arctic Archipelago, Summer 1961. Part I - General Meteorology*. McGill University: Montréal, 82p.
- Müller, F., H. Blatter, R. Brathwaite, H. Ito, G. Kappenberger, A. Ohmura, K. Schroff and A. Züst.** 1975. North Water Project: Progress Report 1 April to 29 September 1974. *Glaciological and Climatological Investigation of the North Water Polynya in Northern Baffin Bay*. Eth Zurich and McGill University: Montréal.
- Nakamura, N. and A.H. Oort.** 1988. Atmospheric Heat Budgets of the Polar Regions. *Journal of Geophysical Research*, 93(D8), 9510-9524.
- Ohmura, A.** 1981. *Climate and Energy Balance on Arctic Tundra: Axel Heiberg Island, Canadian Arctic Archipelago, Spring and Summer 1969, 1970 and 1972*. Zürcher Geographische Schriften ZGS #3: Zürich, 448p.
- Orvig, S.** 1970. Chapter 1 – Introduction, *Climates of the Polar Regions: vol. 14, World Survey of Climatology*. Elsevier Publishing Company: New York, 370p.
- Parks Canada.** 1994. *Ellesmere Island National Park Reserve Resource Analysis and Description: Chapter 5 - Climate*. Canadian Heritage Parks Canada: Ottawa.
- Peters, B. and A. Headley.** 1992. A comparison of winds at Hot Weather Creek and Eureka, N.W.T. *Report No. 92-2*, Canadian Climate Centre, Atmospheric Environment Service. 22p.
- Power, S.B. and L.A. Mysak.** 1992. On the Interannual Variability of Arctic Sea-Level Pressure and Sea Ice. *Atmosphere-Ocean*, 30(4), 551-577.
- Richman, M.B.** 1993. Comments on "The Effect of Domain Shape on Principal Components Analysis", *International Journal of Climatology*, 13, 203-218.
- Richman, M.B.** 1987. Rotation of Principal Components: A Reply, *International Journal of Climatology*, 7, 511-520.

- Richman, M.B. 1986. Rotation of Principal Components, *Journal of Climatology*, 6, 293-335.
- Richman, M.B. 1981. Obliquely Rotated Principal Components: An Improved Meteorological Map Typing Technique?, *Journal of Applied Meteorology*, 20, 1145-1159.
- Richman, M.B. 1980. Map Typing Patterns Associated with Potential Urban Enhanced Precipitation. Unpublished Master of Science thesis: University of Illinois at Urbana-Champaign. 122p.
- Santibanes, F., L. Morales, J. Delafuente, P. Cellier and A. Huete. 1997. Topoclimatic modeling for minimum temperature prediction at a regional scale in the central valley of Chile. *Agronomie*, 17(6-7), 307-314. (abst.)
- Serreze, M.C. and R.G. Barry. 1988. Synoptic Activity in the Arctic Basin. *Journal of Climate*, 1, 1276-1295.
- Shea, D.J., S.J. Worley, I.R. Stern and T.J. Hoar. 1994. (unpublished report) *An Introduction to Atmospheric and Oceanographic Data*, NCAR Technical Note NCAR/TN-404+1A, Climate and Global Dynamics Division, National Center for Atmospheric Research: Boulder, Colorado. 132p.
- Shepard, D. 1968. A two-dimensional interpolation function for irregularly-spaced data. *Proceedings - 1968 Association for Computing Machinery (ACM) National Conference*, pp517-524.
- Stull, R. 1988. *An Introduction to Boundary Layer Meteorology*, Kluwer Academic Publishers: Dordrecht. 666 pp.
- Thomson, R.B., M.P. Brennan and J.P. Ross. 1986. *Arctic Non-Standard Data Catalogue*; Prepared for Arctic Meteorology Section, Canadian Climate Centre by Scientific Services AES Western Region, Unpublished CCC Report 86-5.
- Vega, A.J., C.H. Sui and K.M. Lau. 1998. Interannual to interdecadal variations of the regionalized surface climate of the United States and relationships to generalized flow parameters, *Physical Geography*, 19, 271-291. (abst.)
- Walsh, J.E. and W.L. Chapman. 1990. Short-term Climatic variability of the Arctic. *Journal of Climate*, 3, 237-250.
- White, M.B. Richman and B. Yarnal. 1991. Climate Regionalization and rotation of Principal Components Analysis, *International Journal of Climatology*, 11, 1-25.
- Wilmott, C.J. and K. Matsuura. 1995. Smart Interpolation of Annually Averaged Air Temperature in the United States, *Journal of Applied Meteorology*, 34, 2577-2586.
- Wolfe, P.M. 1995. Hydrometeorological Investigations on a Small Valley Glacier in the Sawtooth Range, Ellesmere Island, Northwest Territories. *M.Sc. Thesis, Wilfred Laurier University*, 205p.
- Wolfe, P.M. and M.C. English. 1995. Hydrometeorological Relationships in a Glacierized Catchment in the Canadian High Arctic. *Hydrological Processes*, 9, 911-921.

- Woo, M.K., Young, K.L. and Edlund, S.A.** 1990. 1989 observations of soil, vegetation and microclimate, and effects on slope hydrology, Hot Weather Creek basin, Ellesmere Island, Northwest Territories. *Current Research, Part D, Geological Survey of Canada*, Paper 90-1D, 85-93.
- Young, K.L., M.K. Woo and D.S. Munro.** 1995. Simple Approaches to Modelling Solar Radiation in the Arctic. *Solar Energy*, **54**, pp33-40.

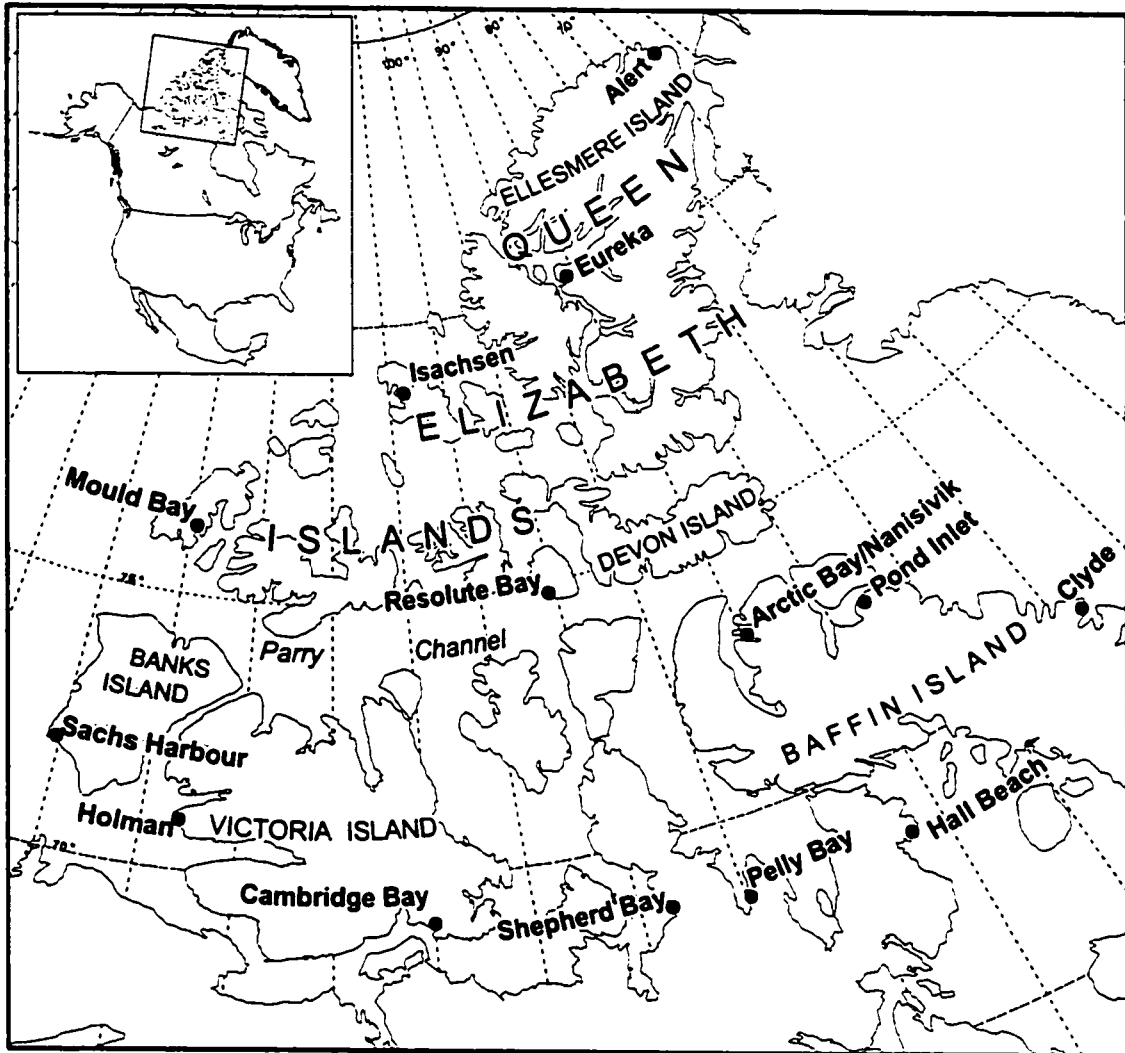


Figure 2.1: The Canadian Arctic Archipelago. Stations of the Atmospheric Environment Service of Canada (AES) are indicated. AES stations with upper air facilities are underlined.

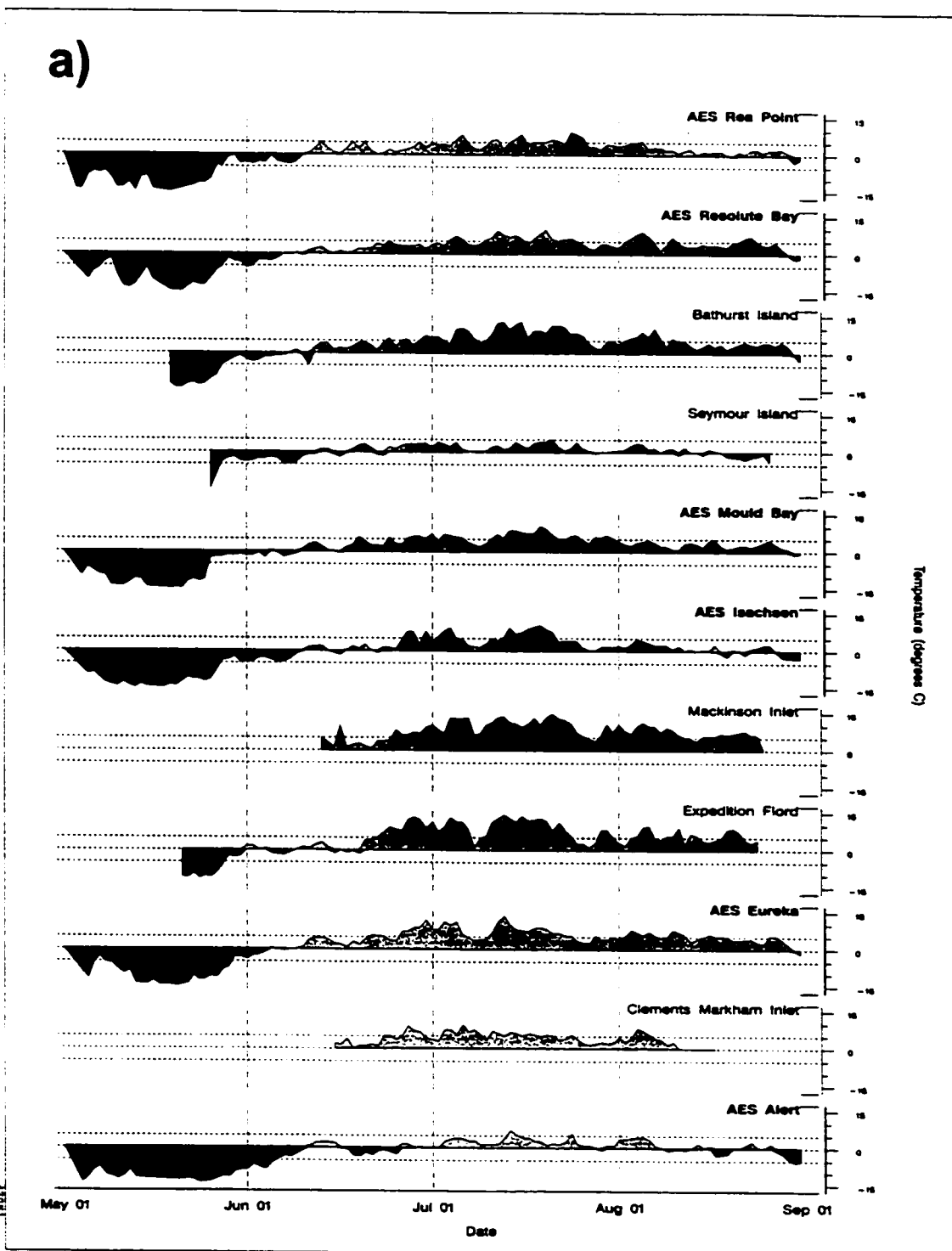


Figure 2.2a: Comparisons of AES and PCSP mean daily temperature, summer 1977: Plots of selected stations. Vertical dashed lines indicate the beginning of the month. Horizontal dashed lines are placed for reference at +5°C, 0°C, and -5°C.

b)

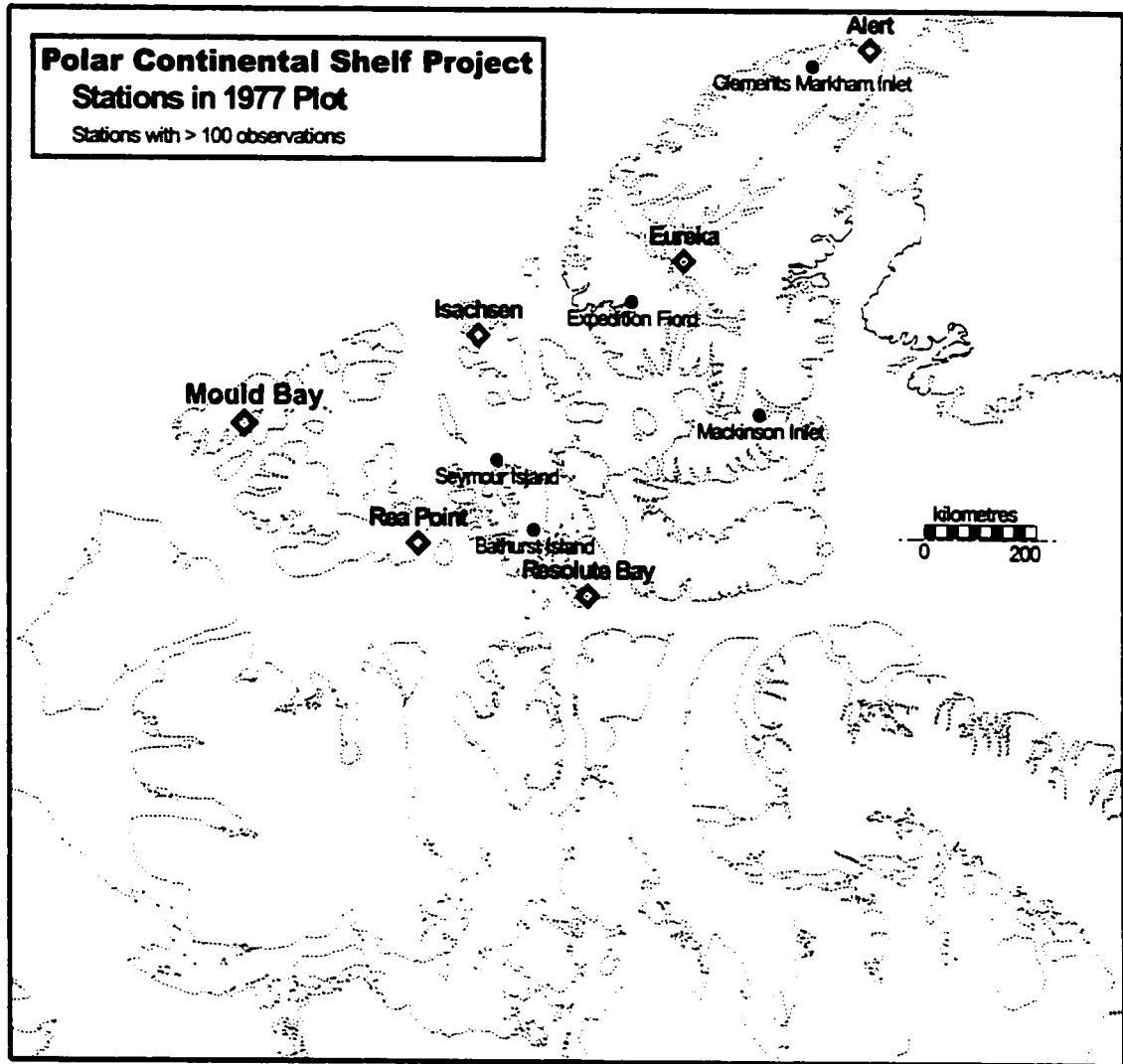


Figure 2.2b: Comparisons of AES and PCSP mean daily temperature, summer 1977: locations of stations presented in the comparison. AES stations are indicated with a diamond, PCSP stations with a dot.

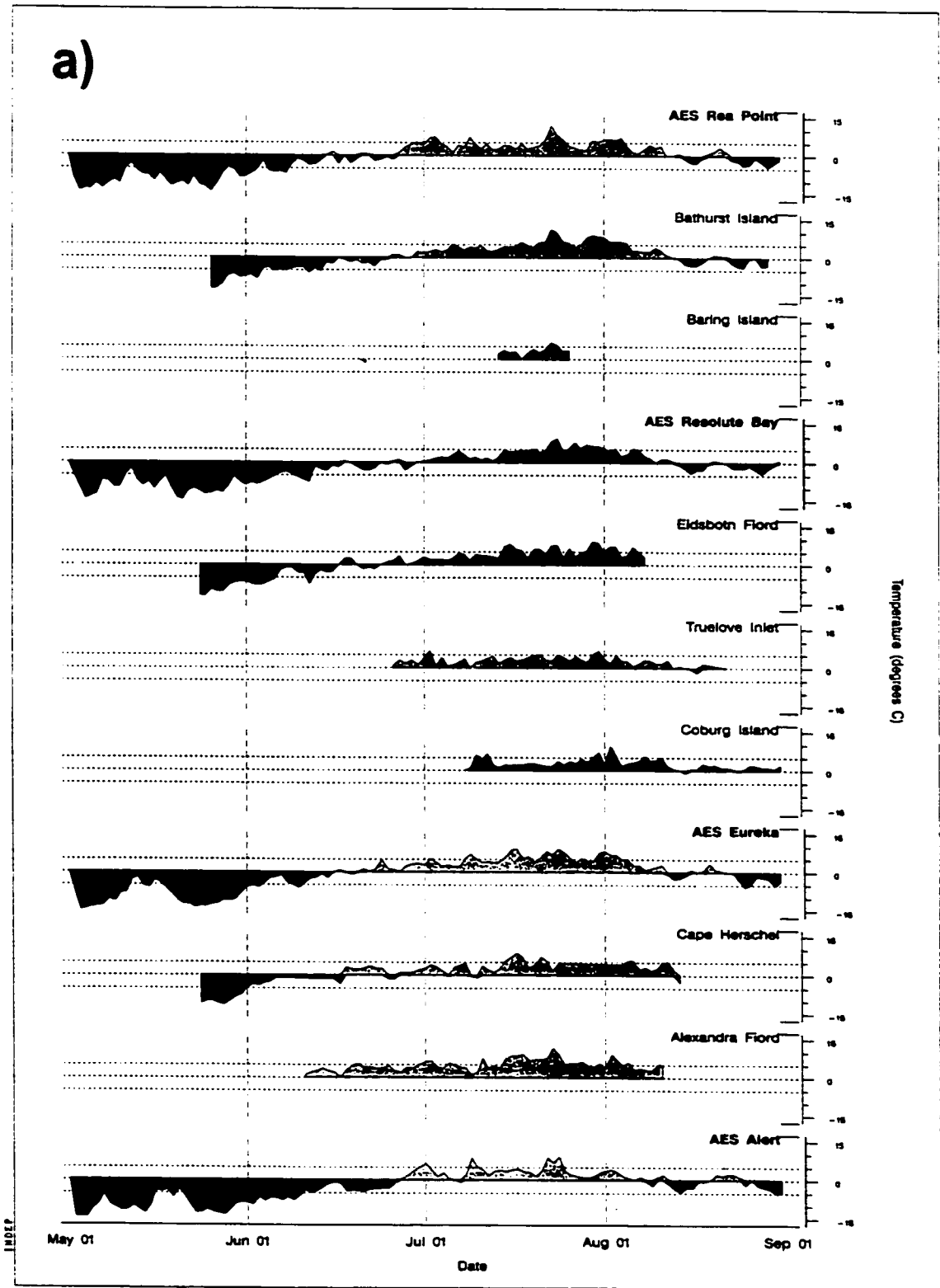


Figure 2.3a: Comparisons of AES and PCSP mean daily temperature, summer 1979: Plots of selected stations. Vertical dashed lines indicate the beginning of the month. Horizontal dashed lines are placed for reference at +5°C, 0°C, and -5°C.

b)

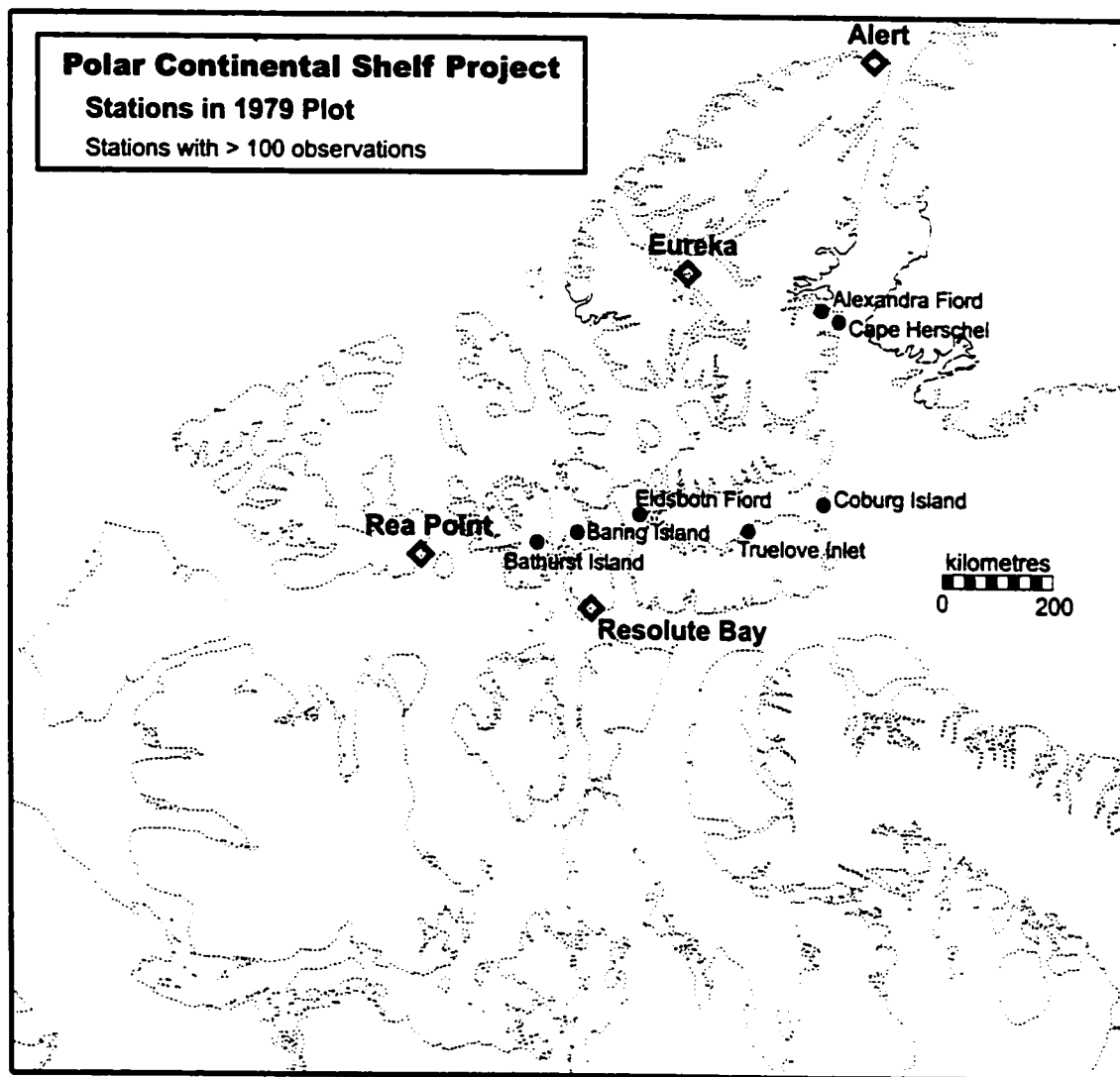


Figure 2.3b: Comparisons of AES and PCSP mean daily temperature, summer 1979: locations of stations presented in the comparison. AES stations are indicated with a diamond, PCSP stations with a dot.

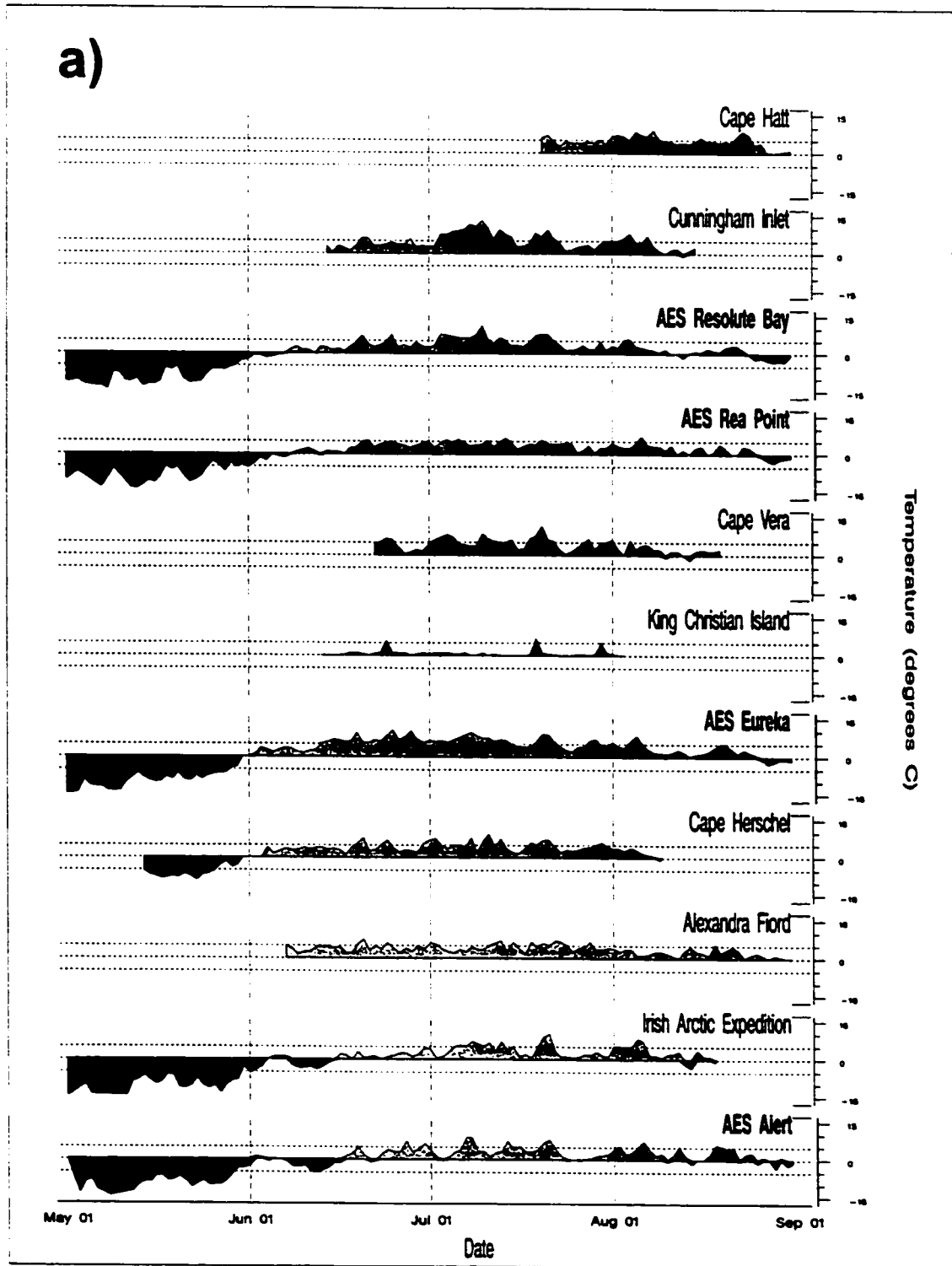


Figure 2.4a: Comparisons of AES and PCSP mean daily temperature, summer 1981: Plots of selected stations. Vertical dashed lines indicate the beginning of the month. Horizontal dashed lines are placed for reference at +5°C, 0°C, and -5°C.

b)

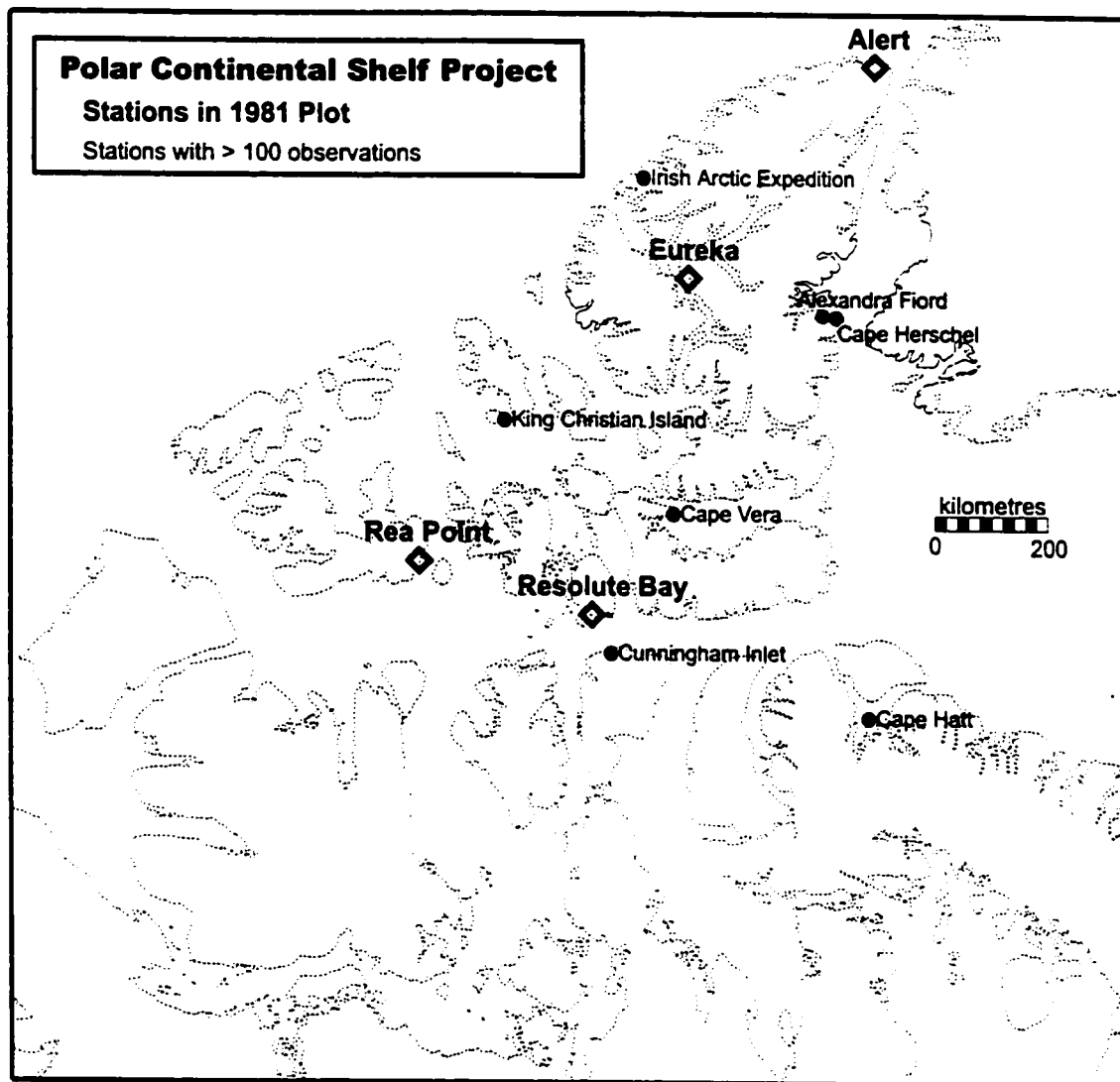


Figure 2.4b: Comparisons of AES and PCSP mean daily temperature, summer 1981: locations of stations presented in the comparison. AES stations are indicated with a diamond, PCSP stations with a dot.

a)

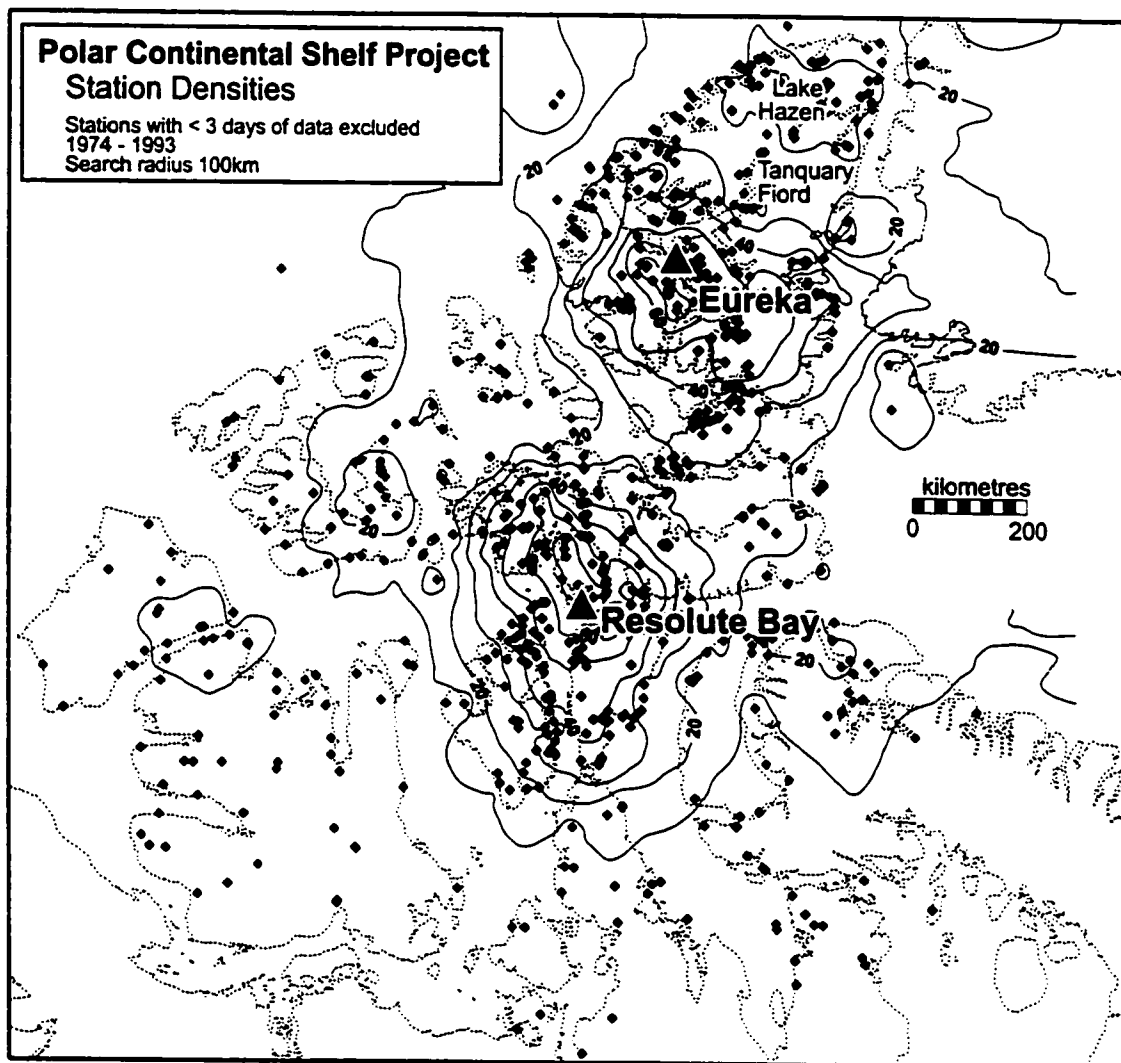


Figure 2.5a: PCSP station densities using 100 km search radius. This pattern is indicative of logistical constraints, such as aircraft operating distances from the Resolute Bay - Eureka corridor.

b)

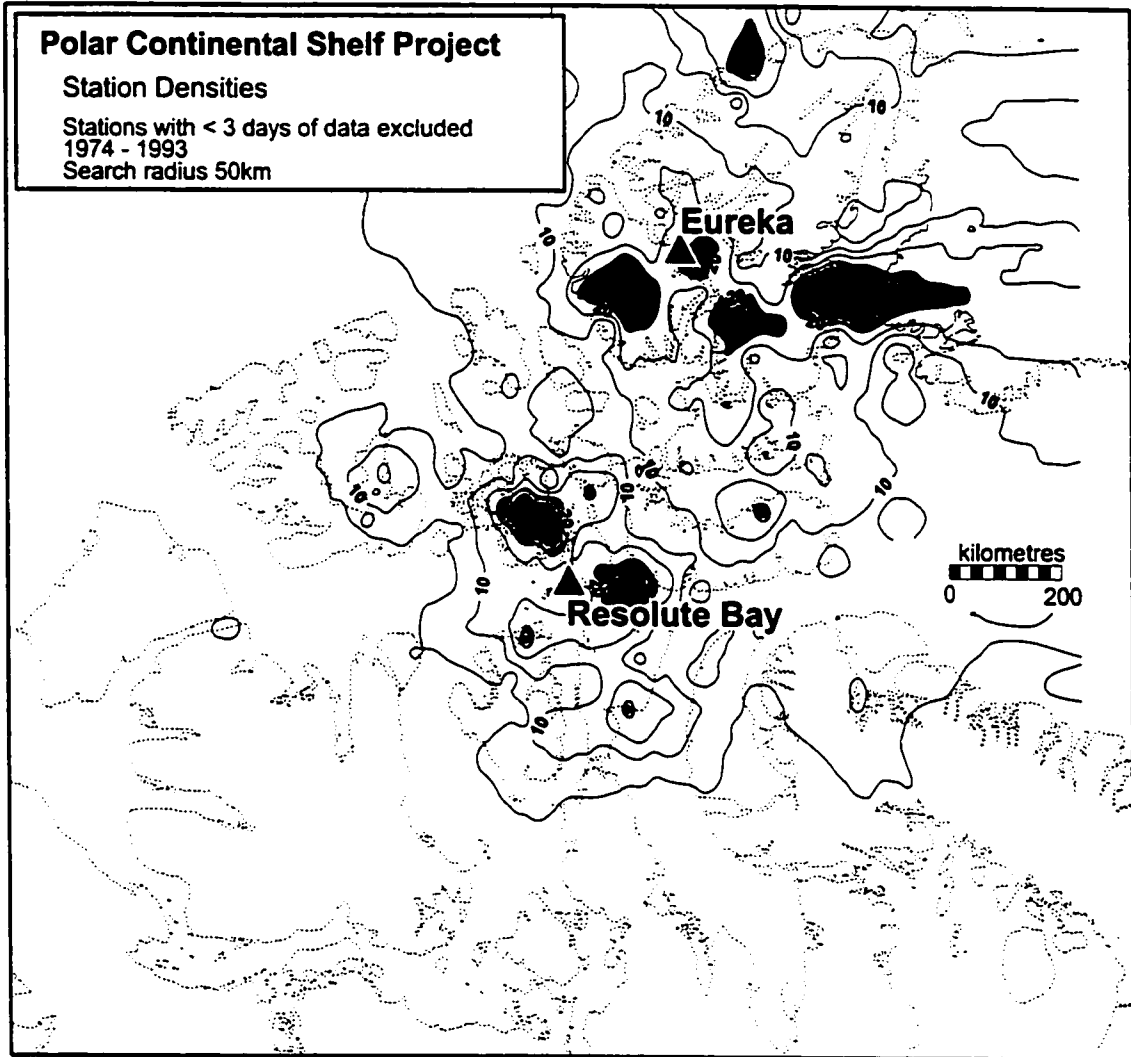


Figure 2.5b: PCSP station densities using 50 km search radius. This pattern reflects some of the most active research areas over the 1974 – 1993 period.

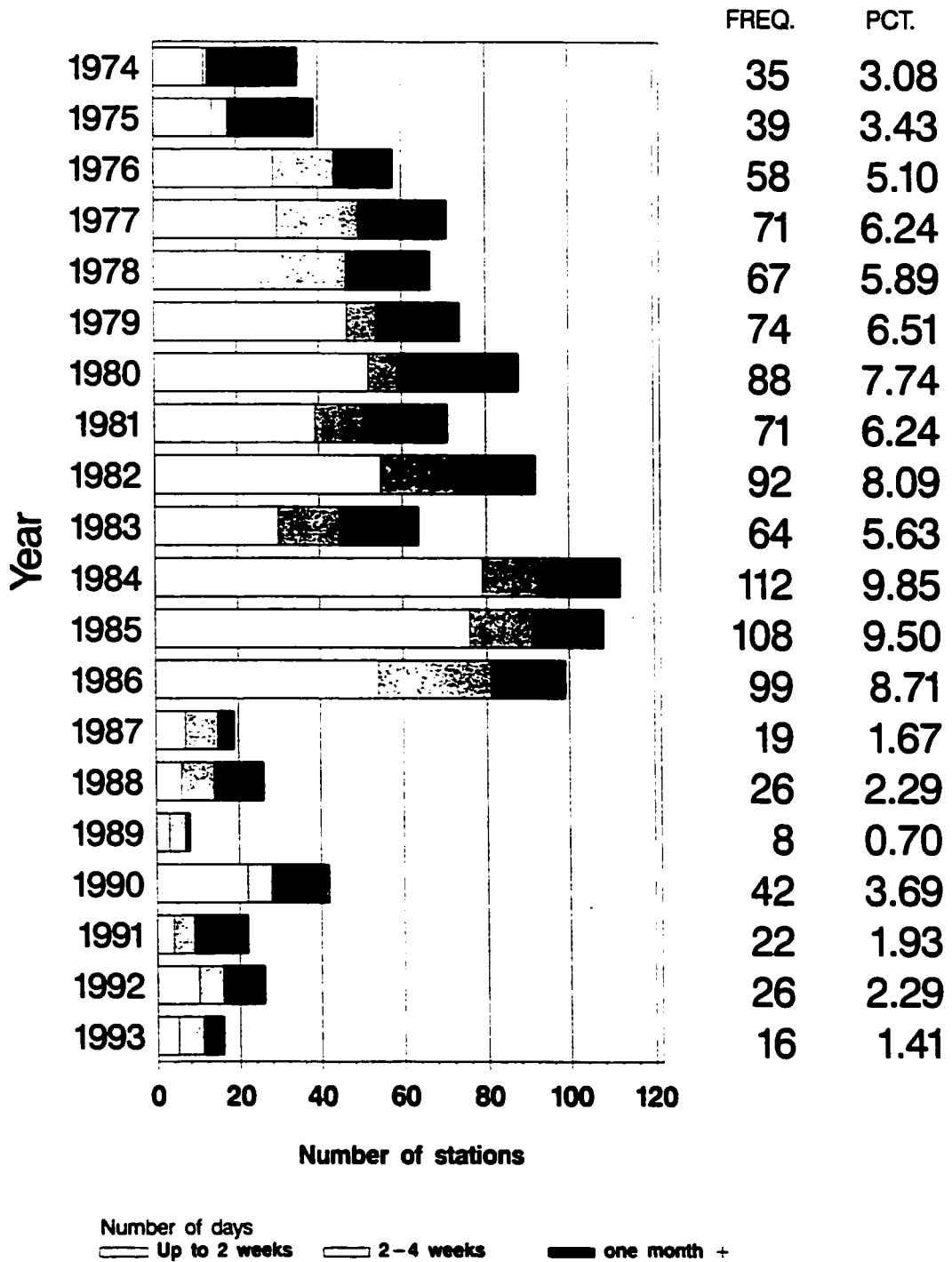


Figure 2.6: Number of stations in the PCSP database per year, broken down by duration.

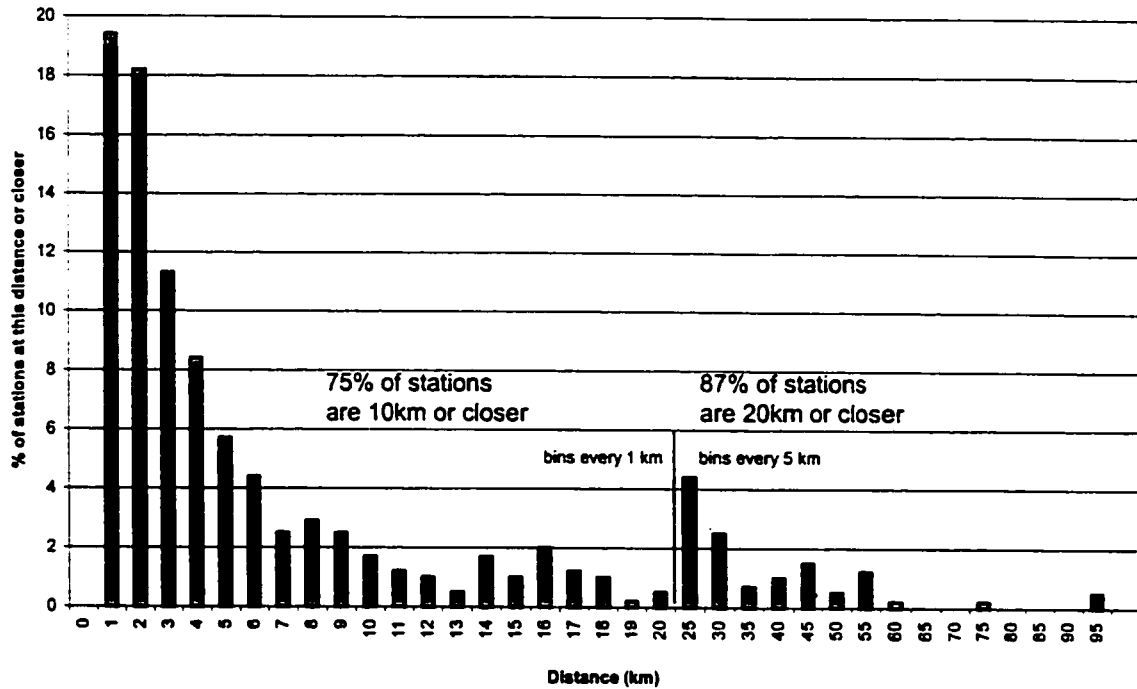


Figure 2.7: Breakdown of PCSP stations by distance from coast (km).

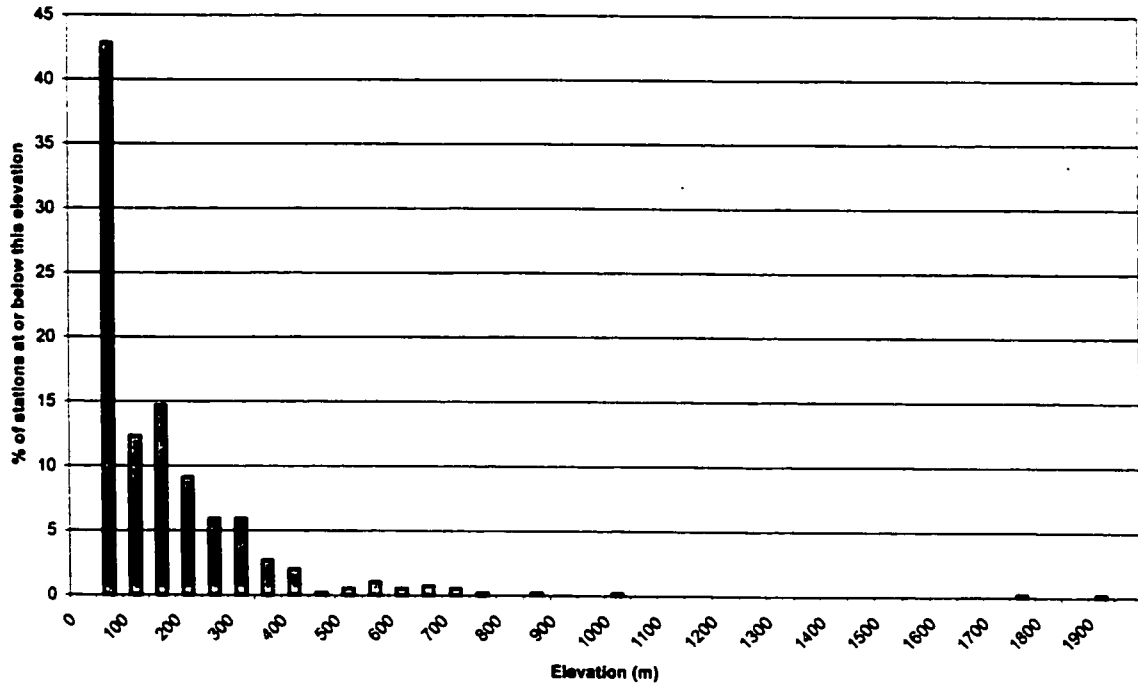


Figure 2.8: Breakdown of PCSP stations by elevation (m).

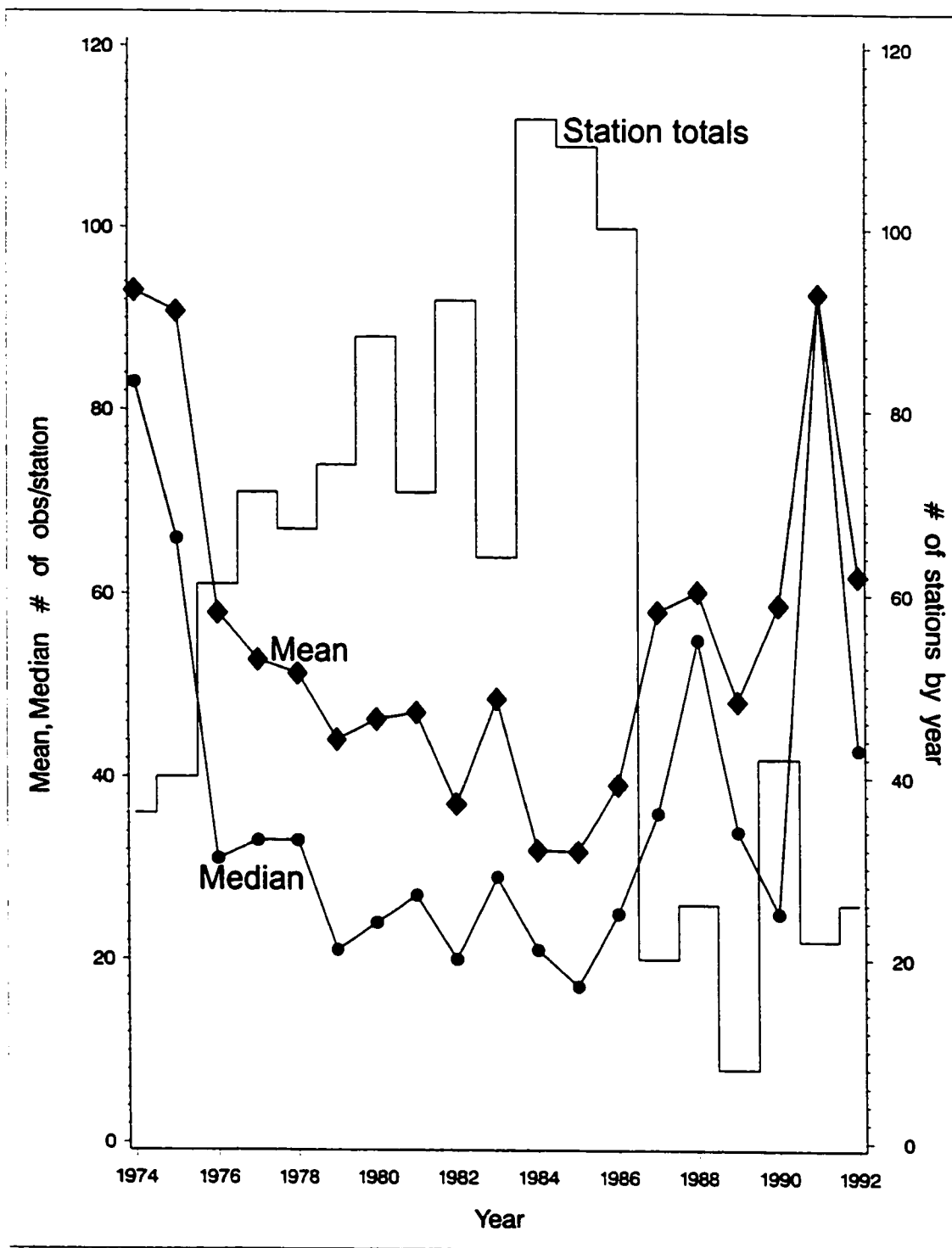


Figure 2.9: Mean and median number of observations per station, PCSP data set.

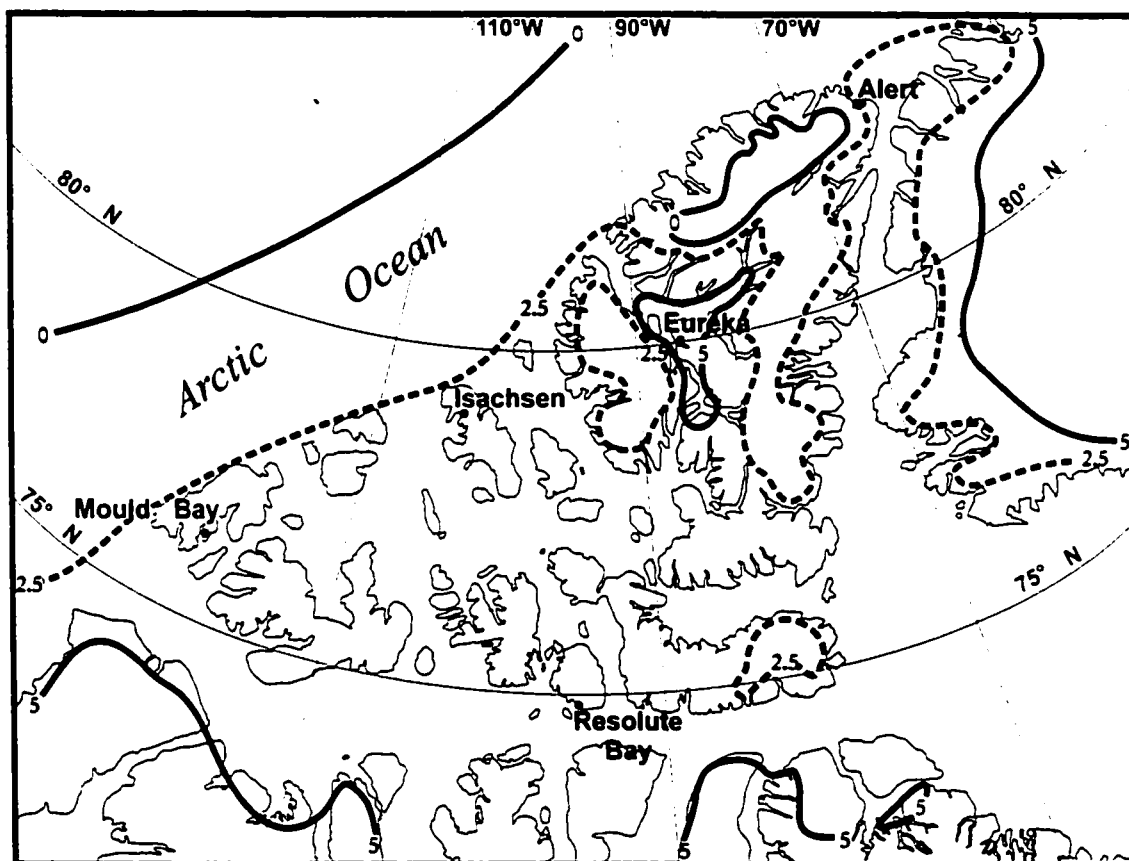


Figure 2.10: July mean temperature normals, 1941-1970 (after Maxwell, 1980). Solid lines are 0 and 5°C isotherms, dashed lines are 2.5°C isotherms.

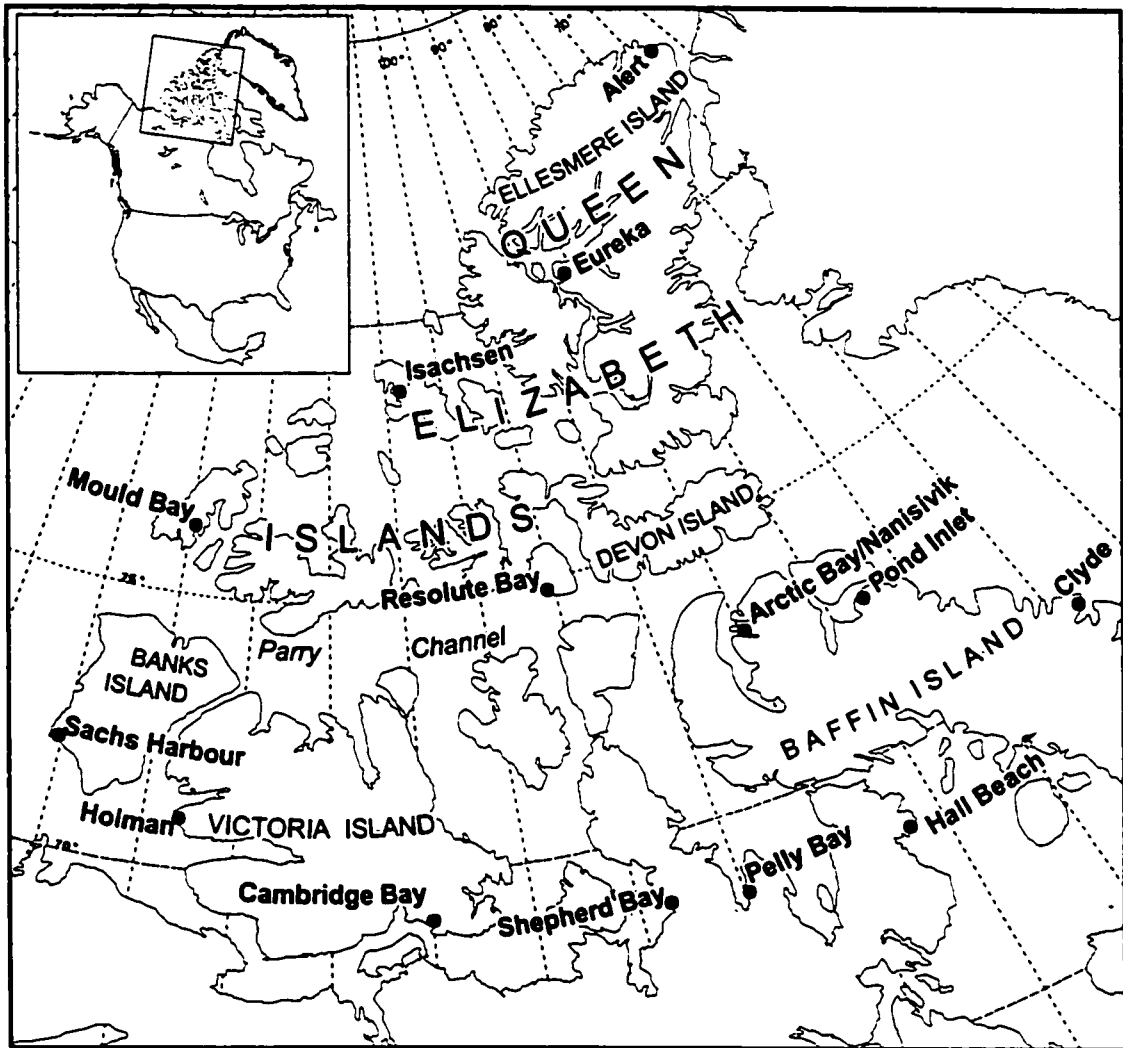


Figure 3.1: The Canadian Arctic Archipelago. Stations of the Atmospheric Environment Service of Canada (AES) are indicated. AES stations with upper air facilities are underlined.

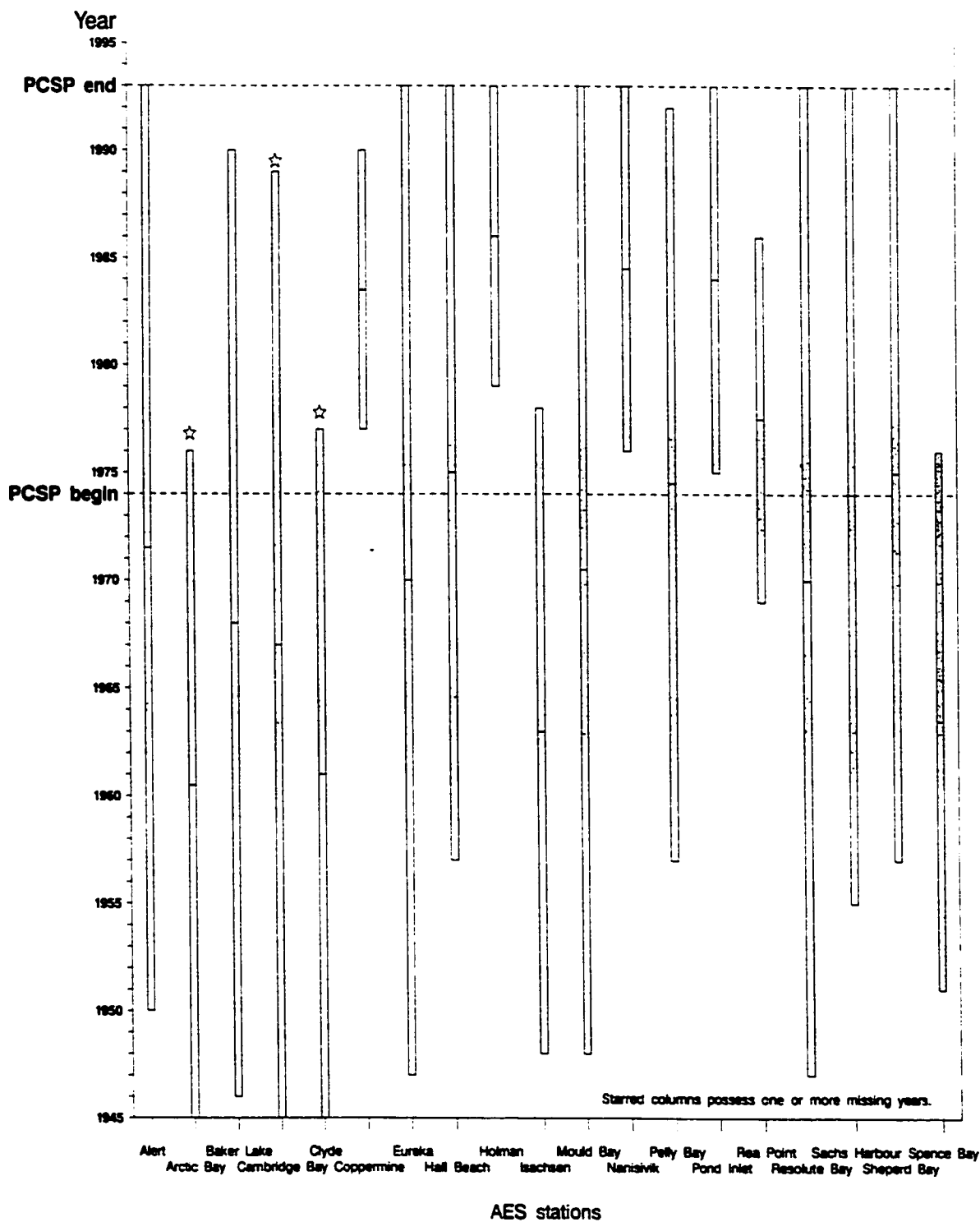
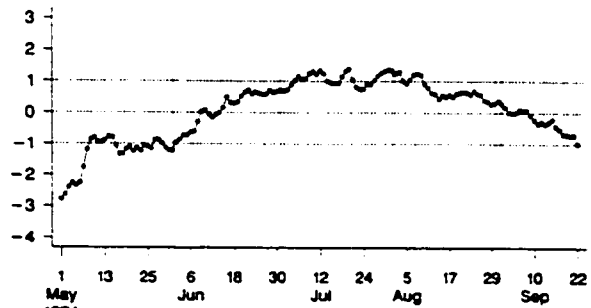
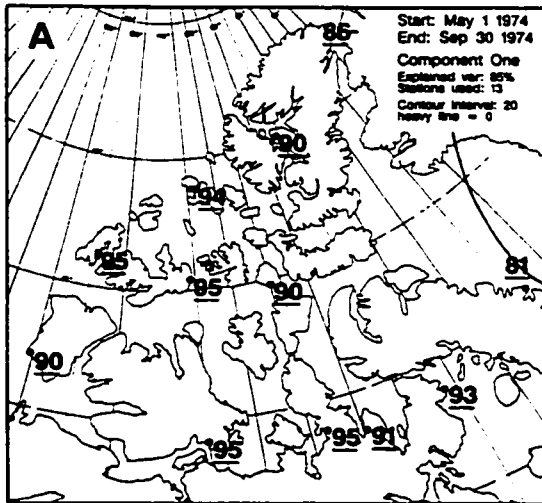
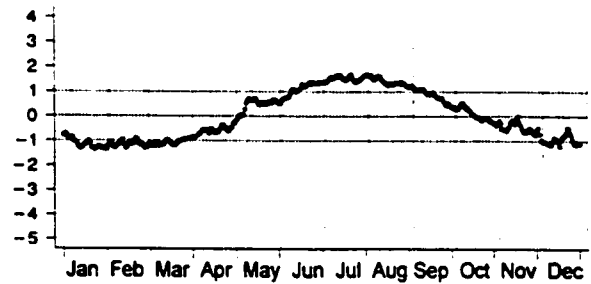
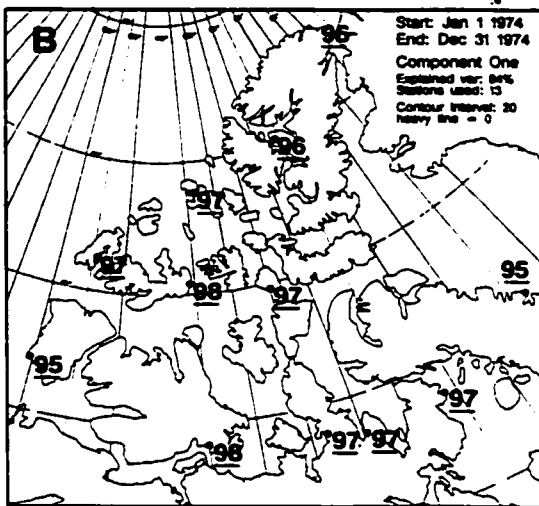


Figure 3.2: Record lengths (daily mean temperature) from AES stations in the Canadian Arctic Archipelago. Starred columns indicate stations with one or more years of missing data.



Date
Component scores for the dates indicated.
Analysis performed on daily temperature data.



Date
Component scores for the dates indicated.
Analysis performed on daily temperature data.

Figure 3.3: Principal components analysis results for a five-month period (a) and a 12-month period (b). Note the amount of variance the explained by the first components. Performed on mean daily temperature data.

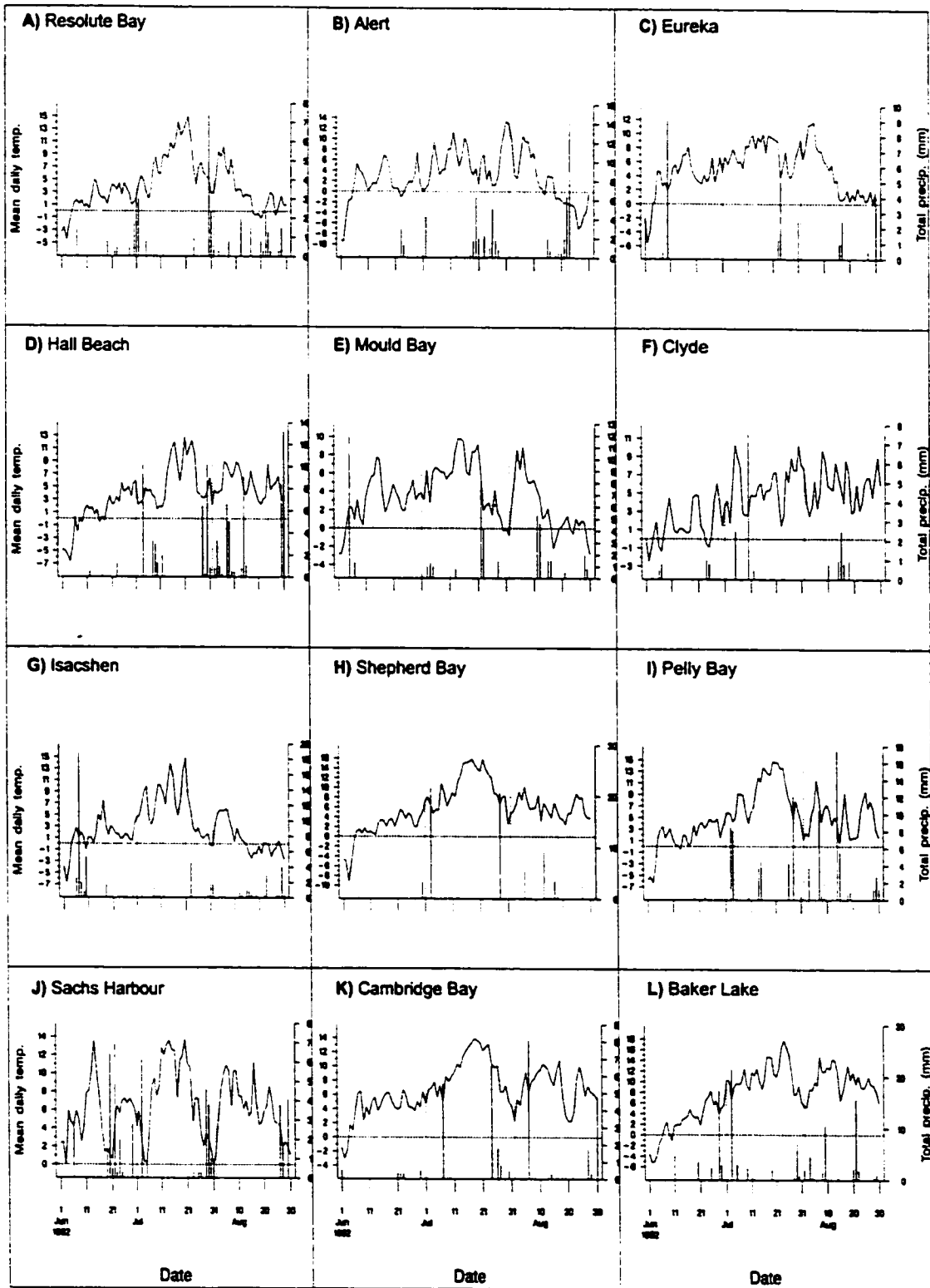


Figure 3.4: Daily precipitation totals and mean daily temperature data for the period June 1 – August 31, 1962. Stations are indicated on the plots. Vertical dashed lines indicate the beginning of a month; horizontal dashed line is a reference line placed at 0°C.

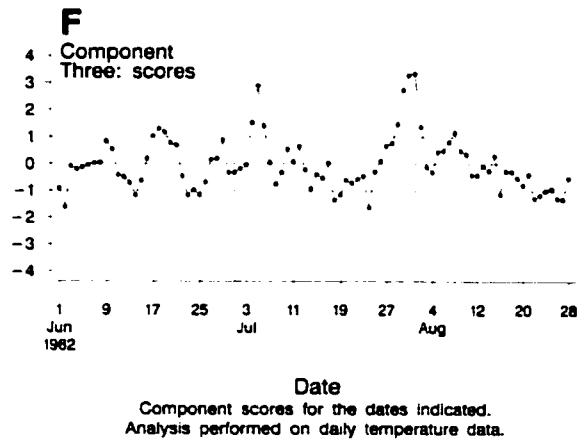
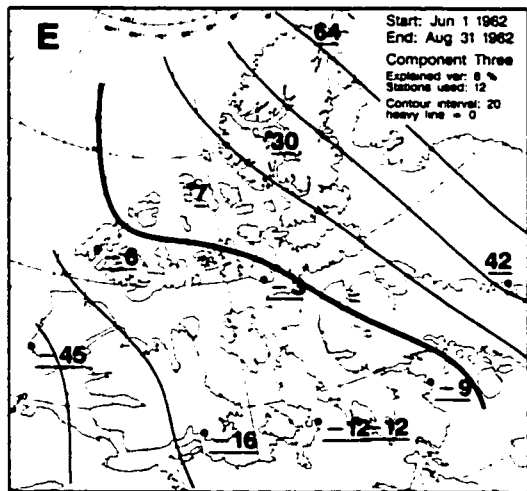
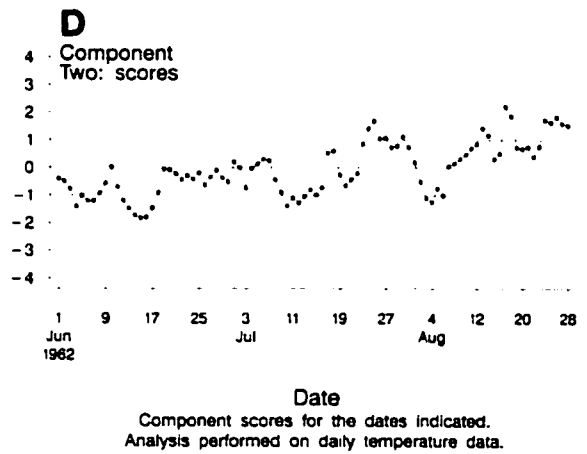
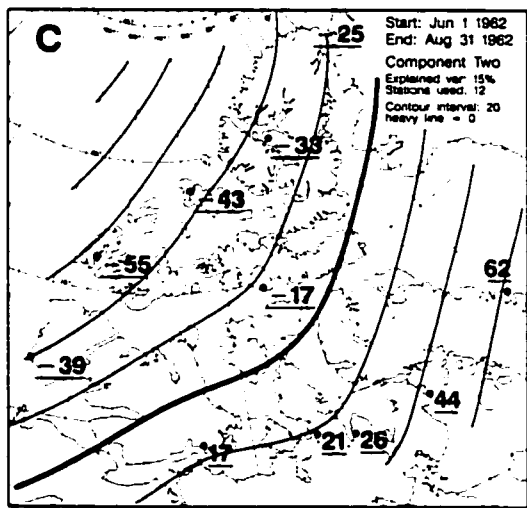
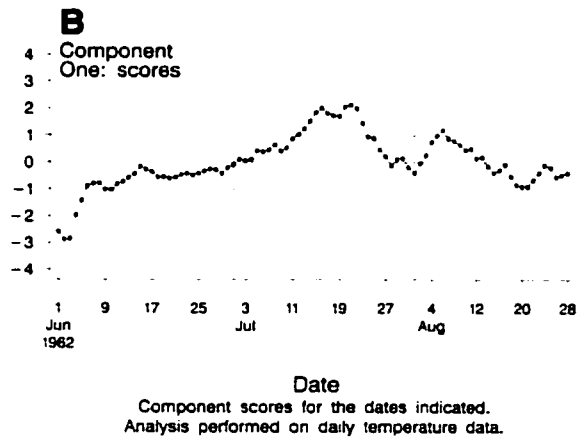
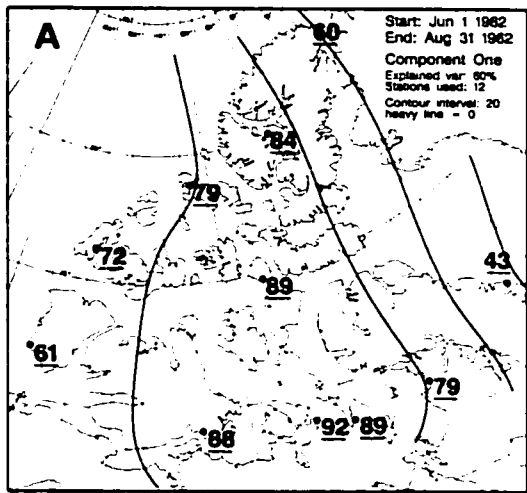


Figure 3.5: Principal components analysis of AES July mean daily data for summer, 1962. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. k): Plots of mean temperature for the period. l): Plot showing percent explained variance by each component. Vertical lines in time series plot indicate the beginning of a month; horizontal lines are reference lines placed at component score values +1, 0, -1.

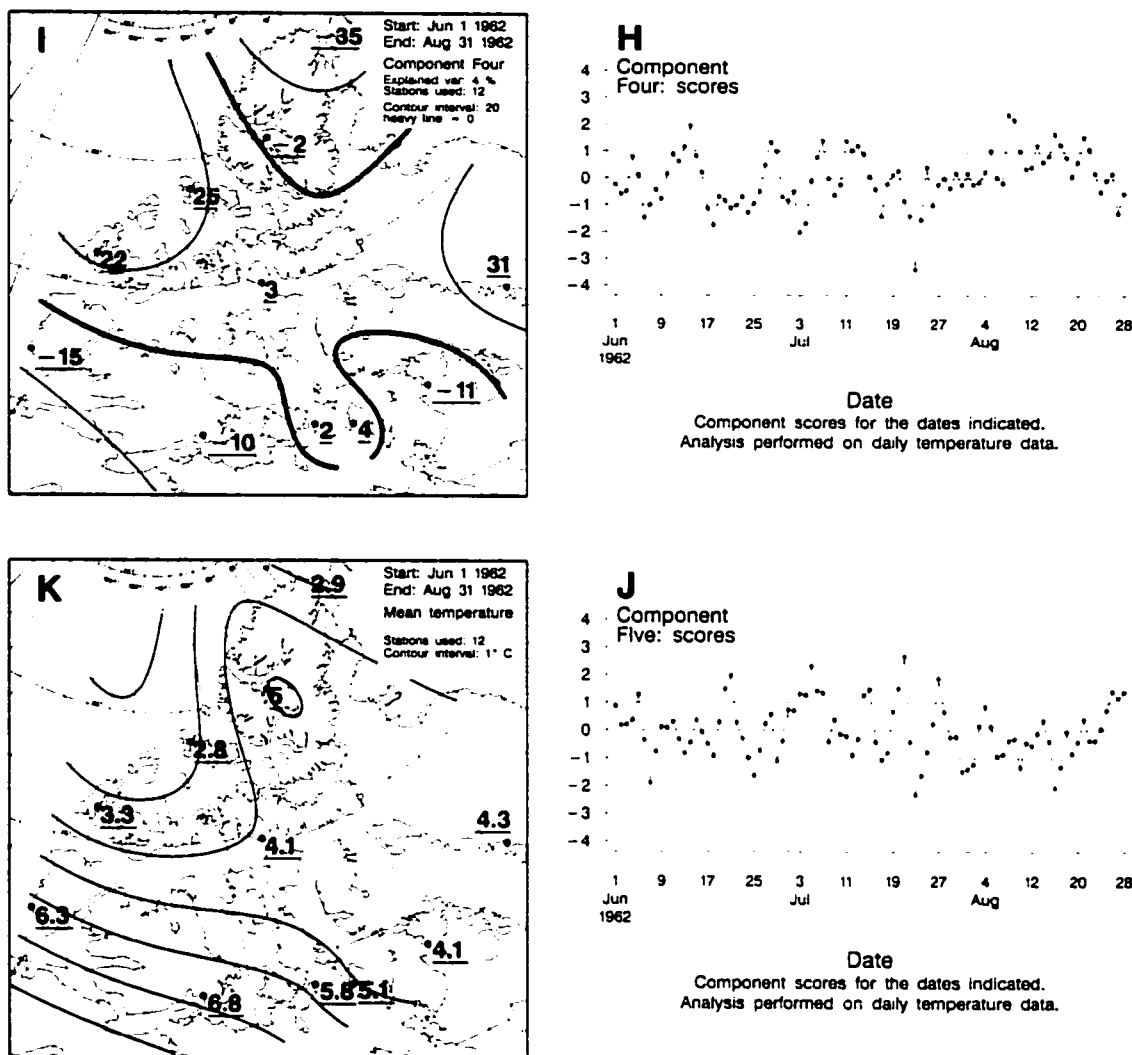


Figure 3.5: Principal components analysis of AES July mean daily data for summer, 1962. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. k): Plots of mean temperature for the period. l): Plot showing percent explained variance by each component. Vertical lines in time series plot indicate the beginning of a month; horizontal lines are reference lines placed at component score values +1, 0, -1.

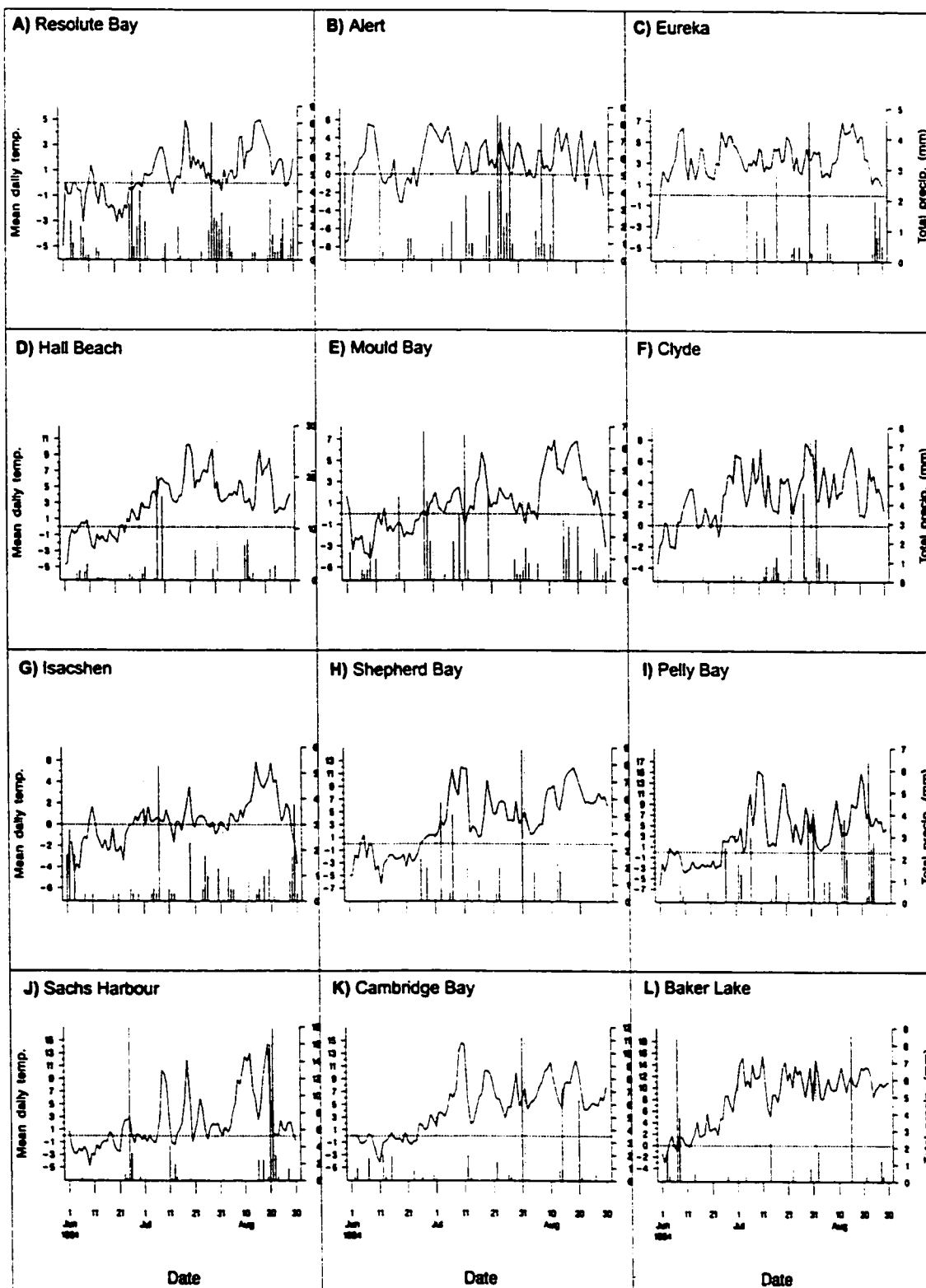


Figure 3.6: Daily precipitation totals and mean daily temperature data for the period June 1 – August 31, 1964. Stations are indicated on the plots. Vertical dashed lines indicate the beginning of a month; horizontal dashed line is a reference line placed at 0°C.

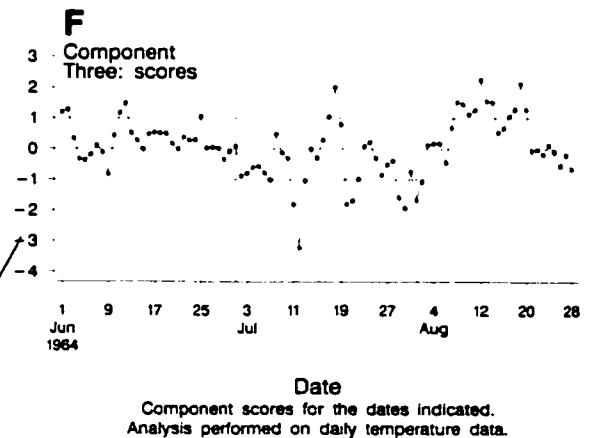
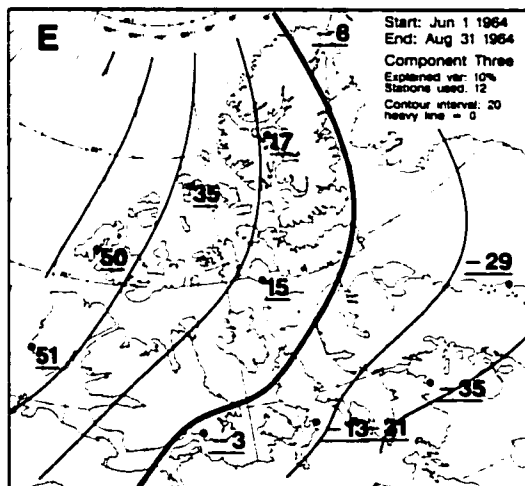
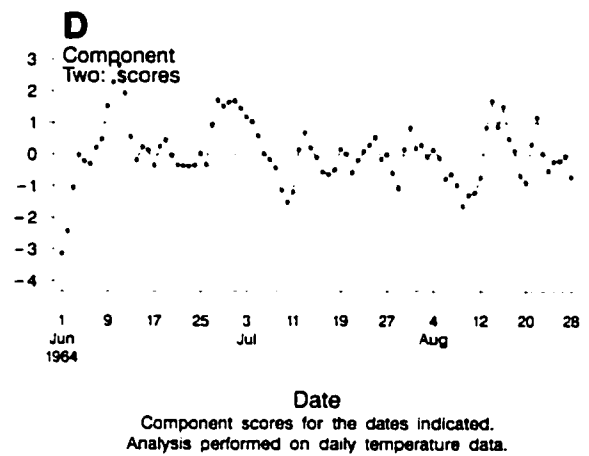
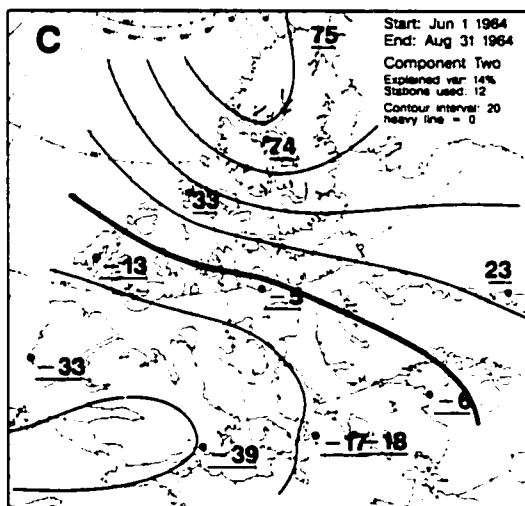
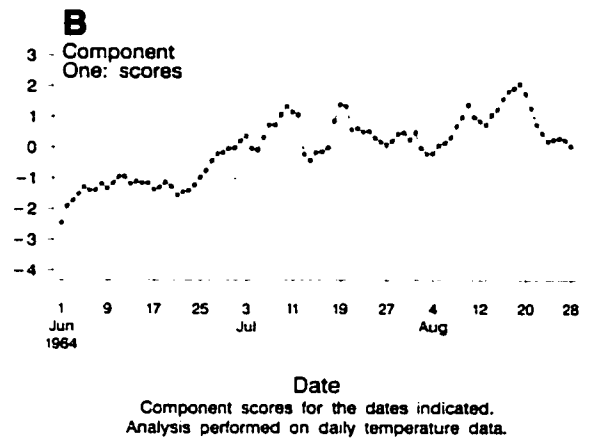
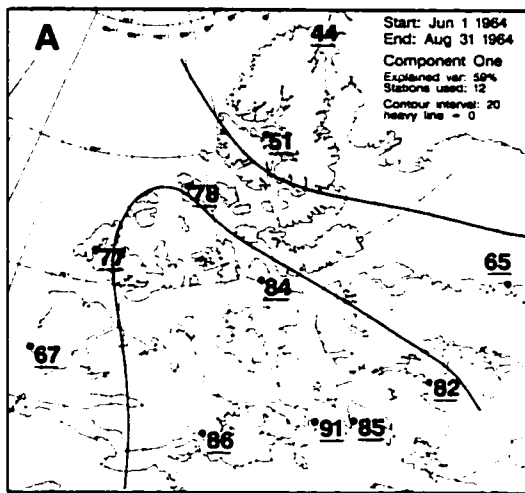


Figure 3.7: Principal components analysis of AES July mean daily data for summer, 1964. a), c), e) and i) are maps of loadings for components one, two, three and four; b), d), f) and h) are time-series plots of component scores. k): Plots of mean temperature for the period. l): Plot showing percent explained variance by each component. Vertical lines in time series plot indicate the beginning of a month; horizontal lines are reference lines placed at component score values +1, 0, -1.

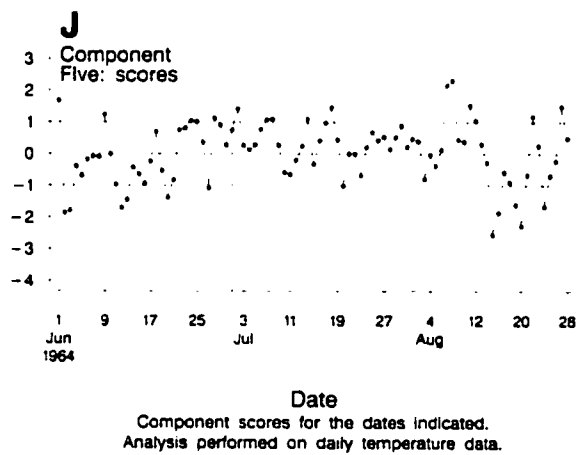
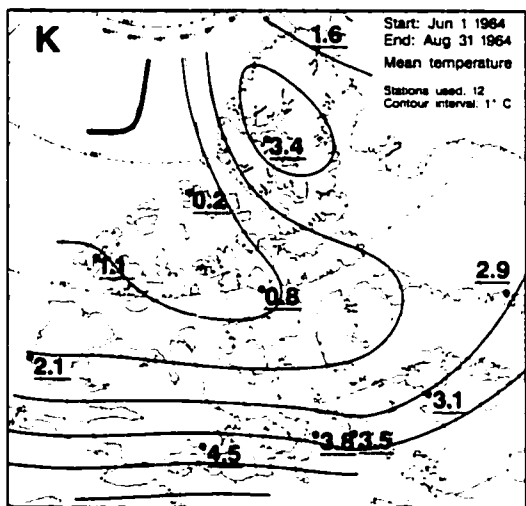
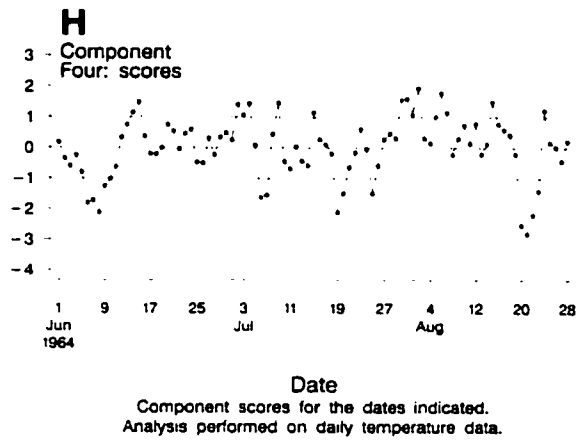
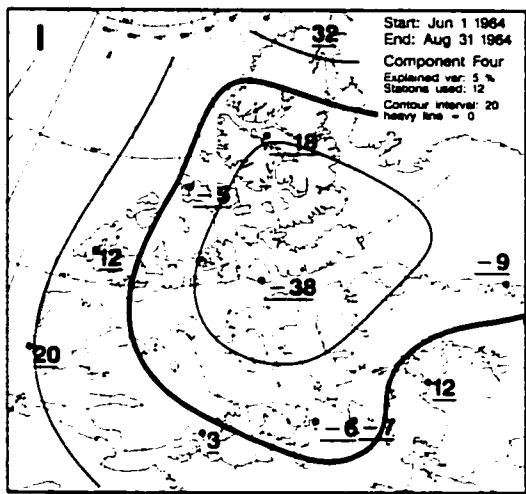


Figure 3.7: Principal components analysis of AES July mean daily data for summer, 1964. a), c), e) and i) are maps of loadings for components one, two, three and four; b), d), f) and h) are time-series plots of component scores. k): Plots of mean temperature for the period. l): Plot showing percent explained variance by each component. Vertical lines in time series plot indicate the beginning of a month; horizontal lines are reference lines placed at component score values +1, 0, -1.

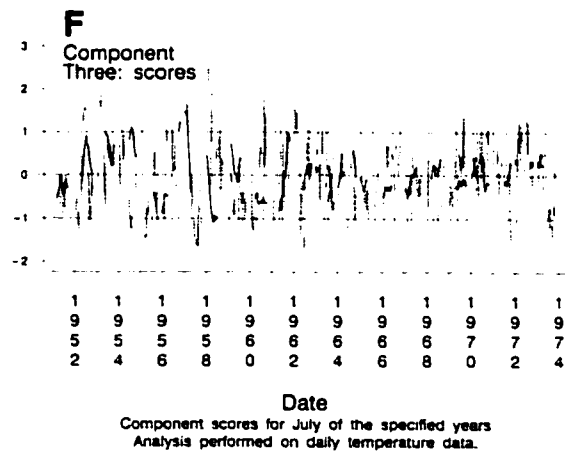
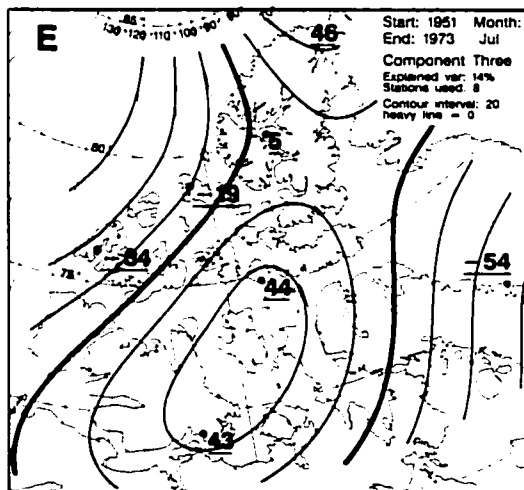
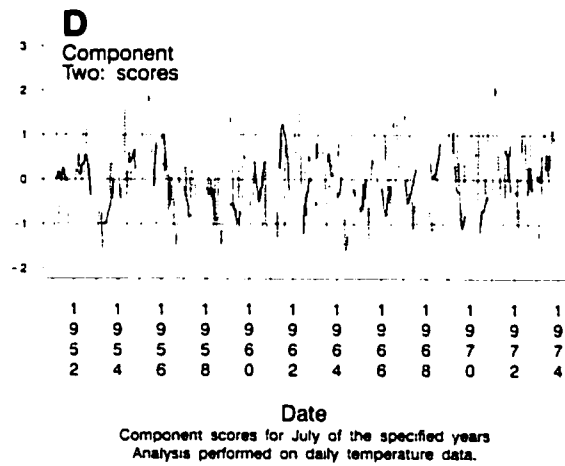
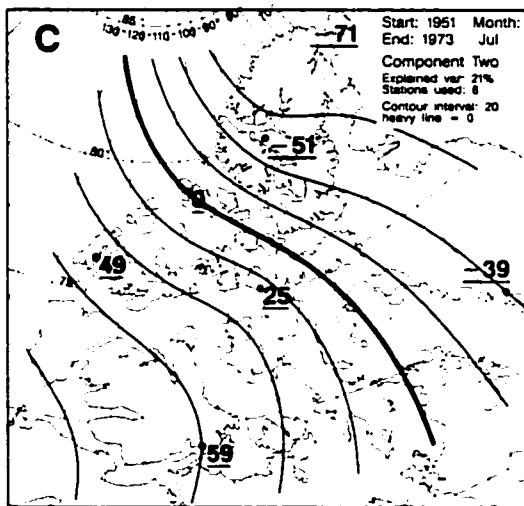
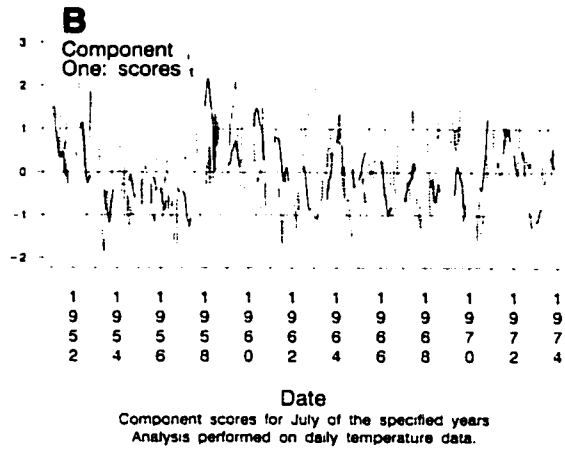
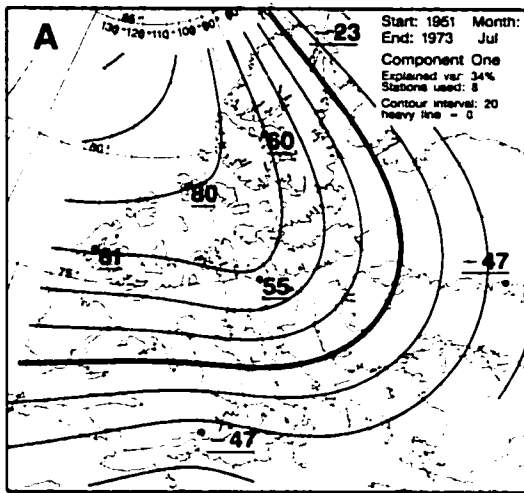


Figure 3.8: Principal components analysis on AES July mean daily data, 1951 - 1974. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.

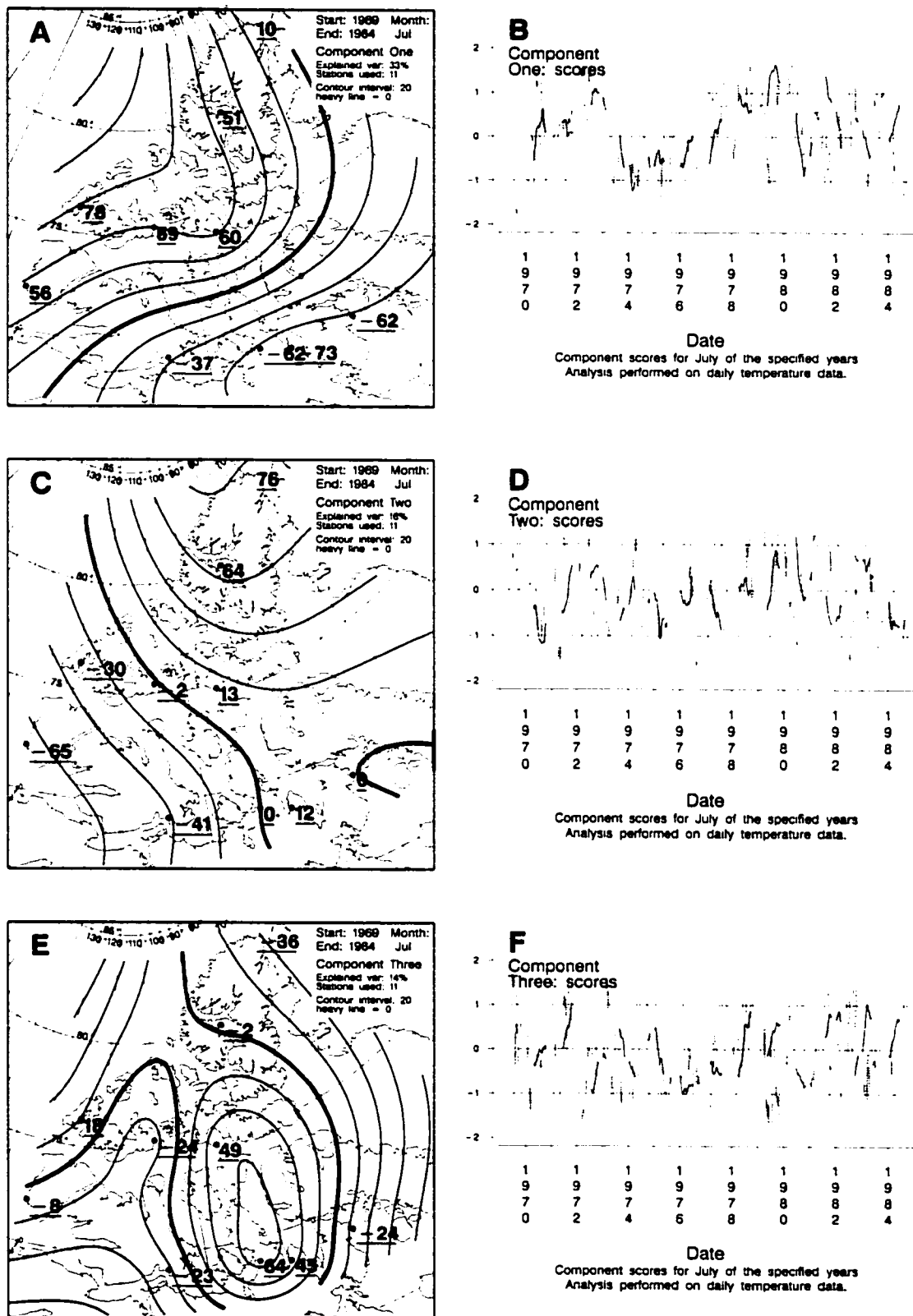


Figure 3.10: Principal components analysis on AES July mean daily data, 1969 - 1985. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.

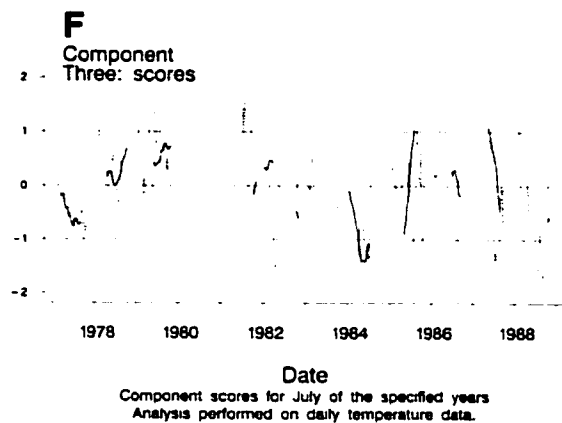
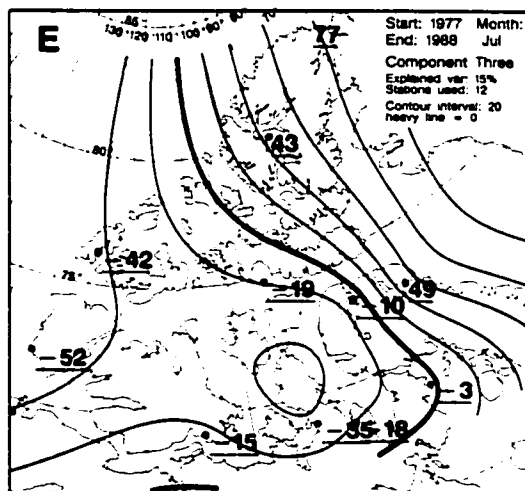
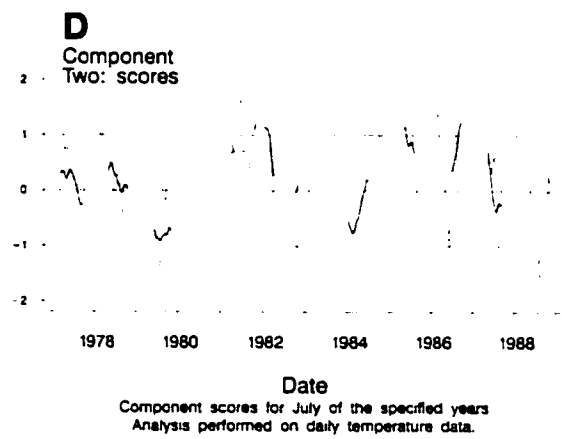
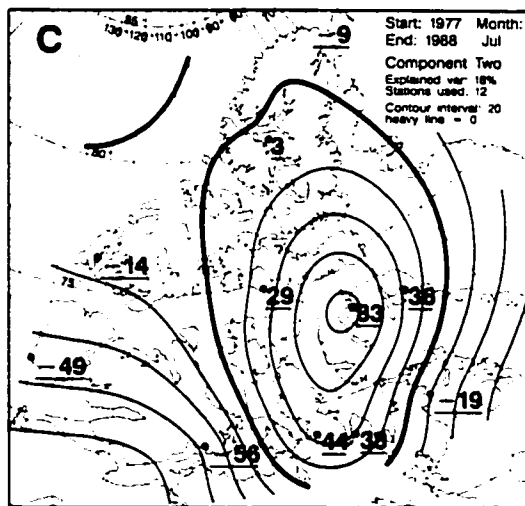
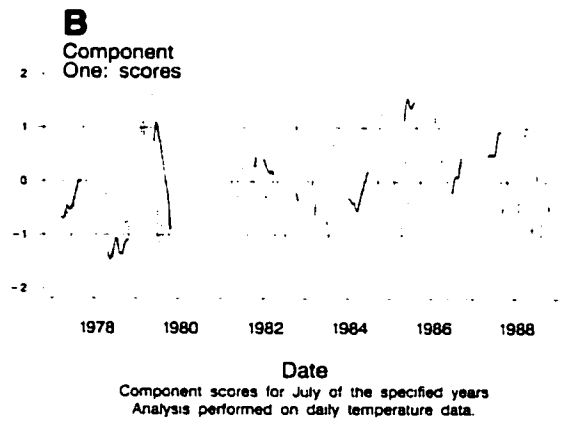
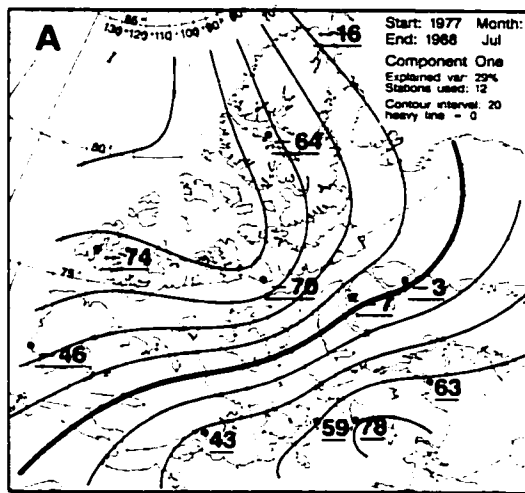


Figure 3.11: Principal components analysis on AES July mean daily data, 1969 - 1989. Raw component scores are plotted with a dotted line; smoothed scores are plotted with a heavy solid line. There is a break to separate each year.

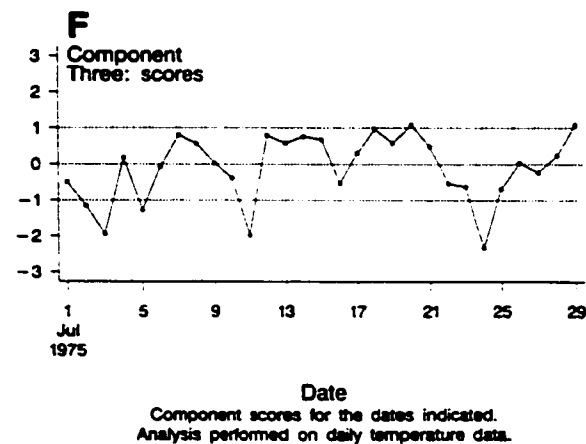
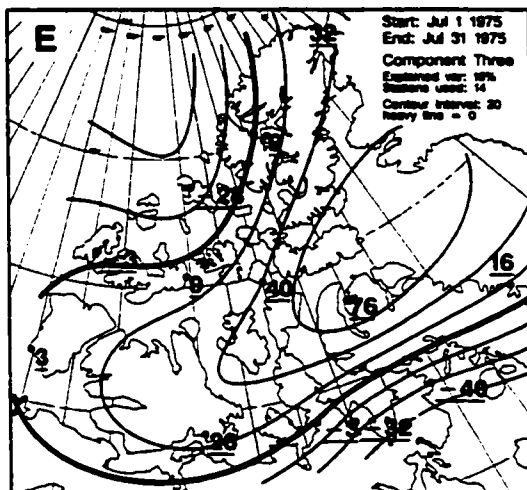
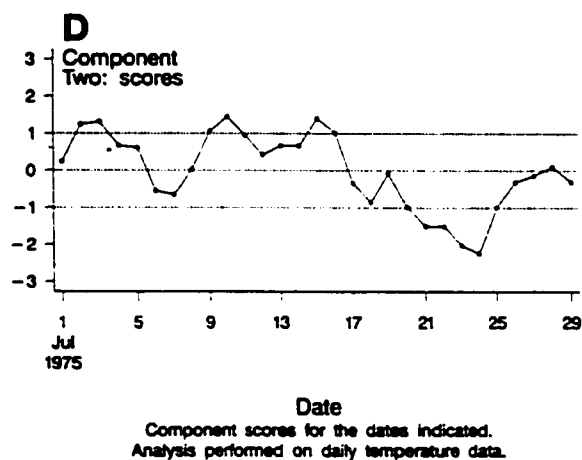
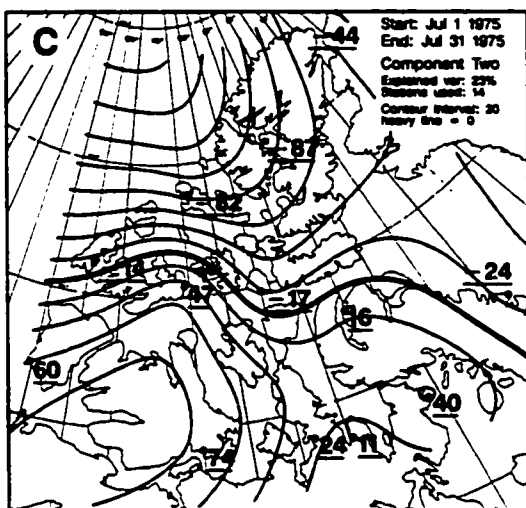
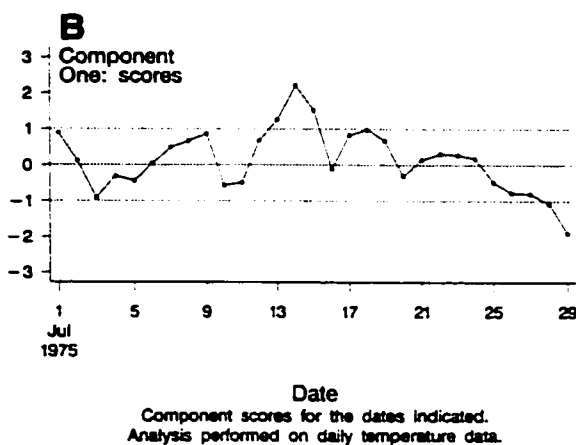
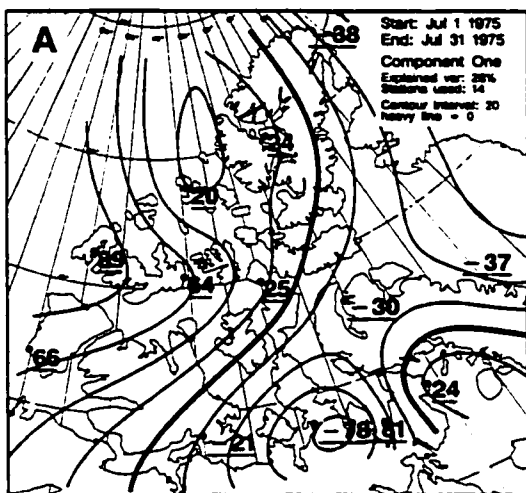


Figure 3.12: Principal components analysis of AES July mean daily data for summer, 1975. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.

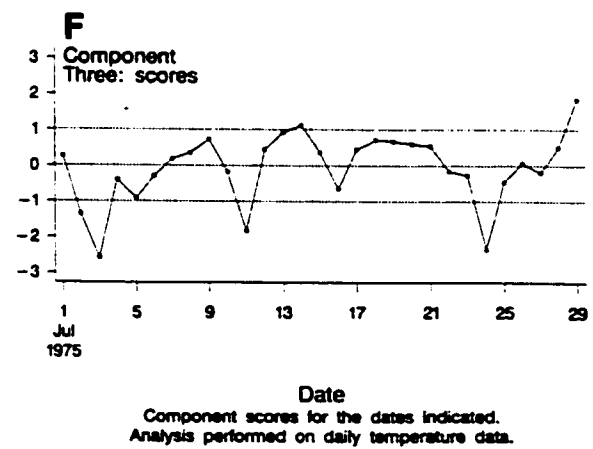
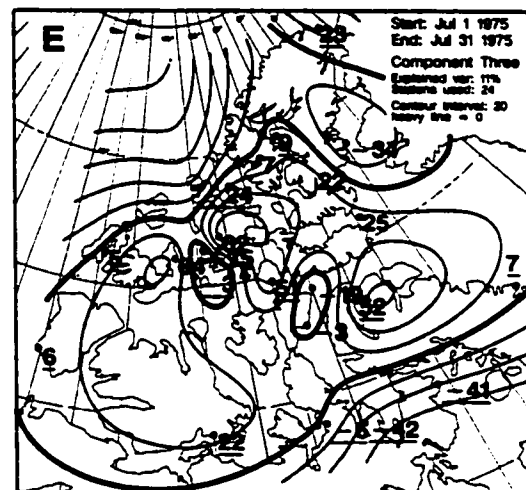
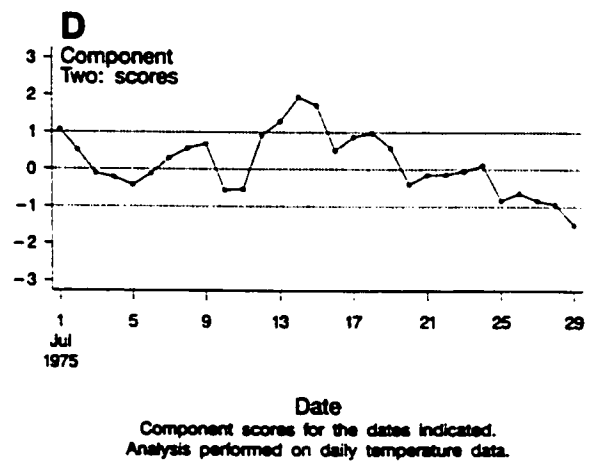
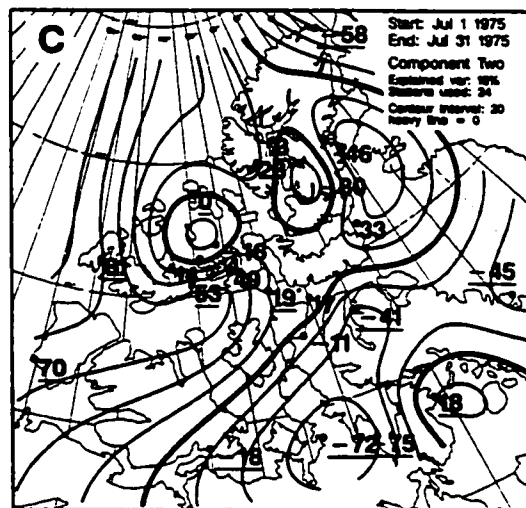
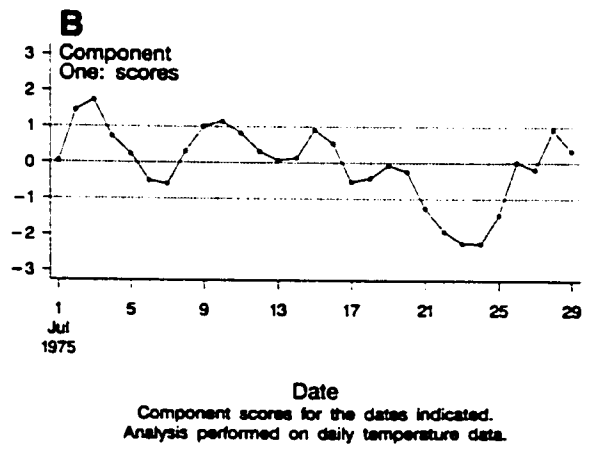
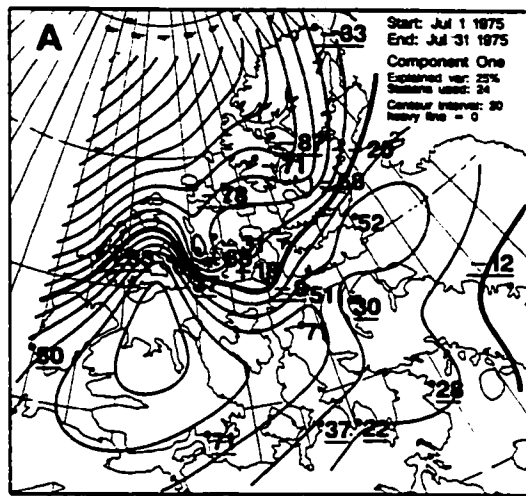


Figure 3.13: Principal components analysis of AES and PCSP July mean daily data for summer, 1975. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined

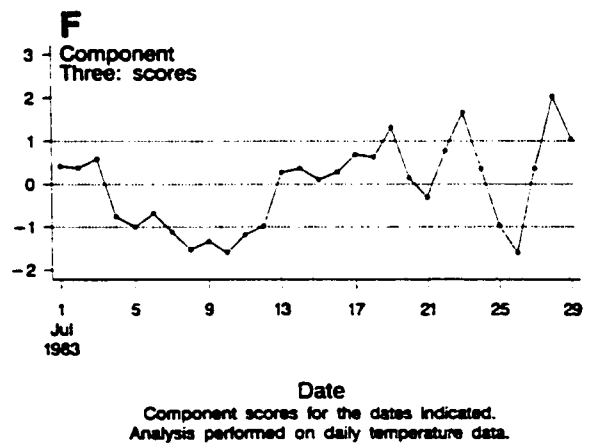
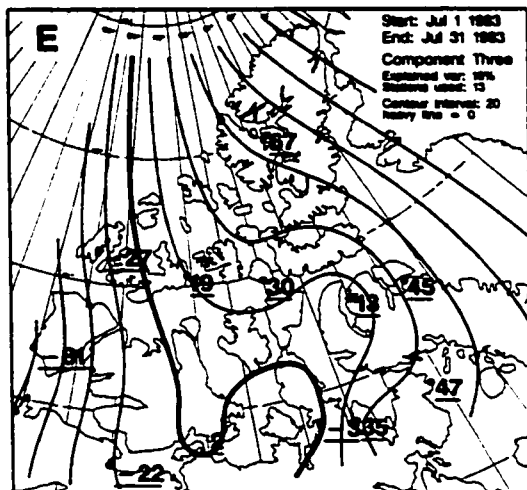
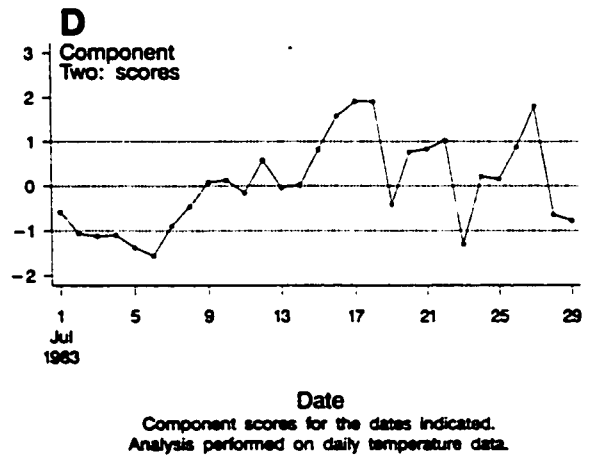
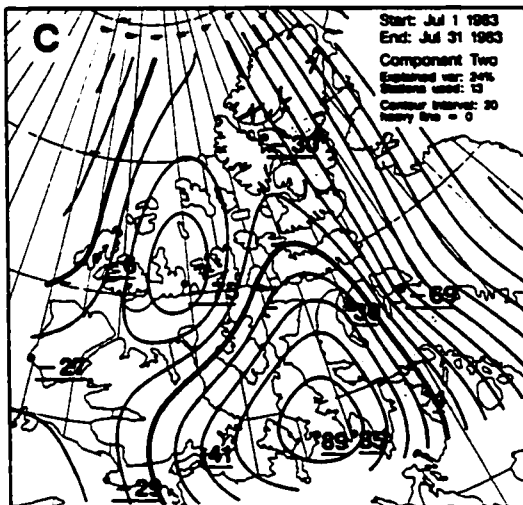
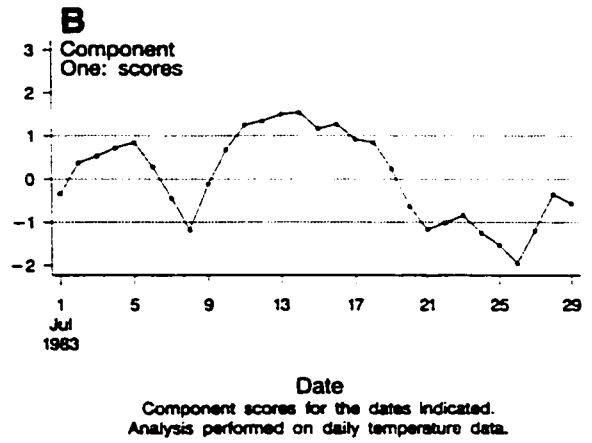
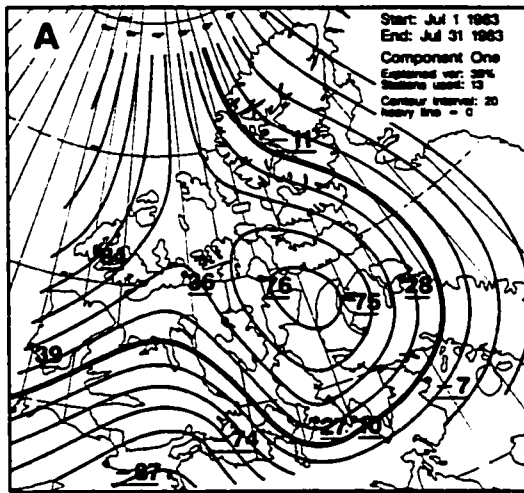


Figure 3.14: Principal components analysis of AES July mean daily data for summer, 1983. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.

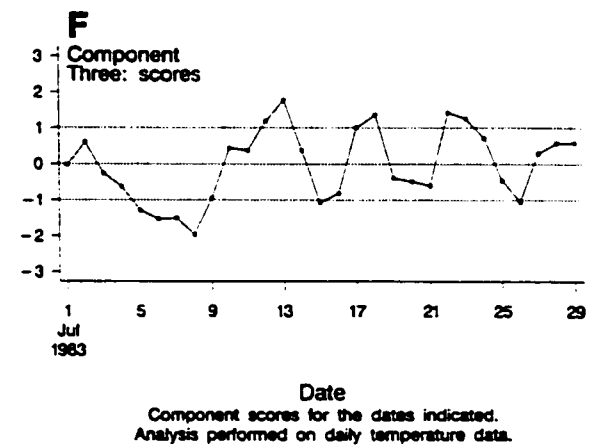
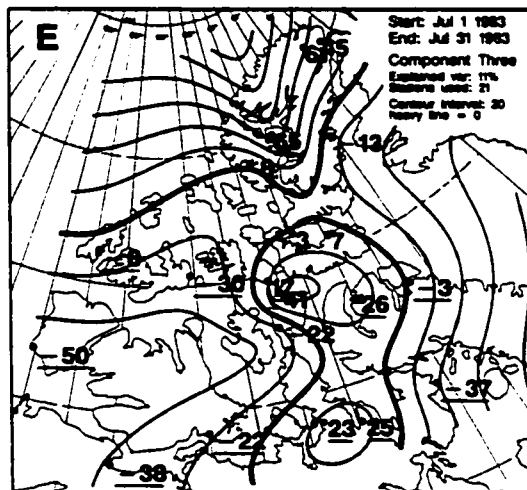
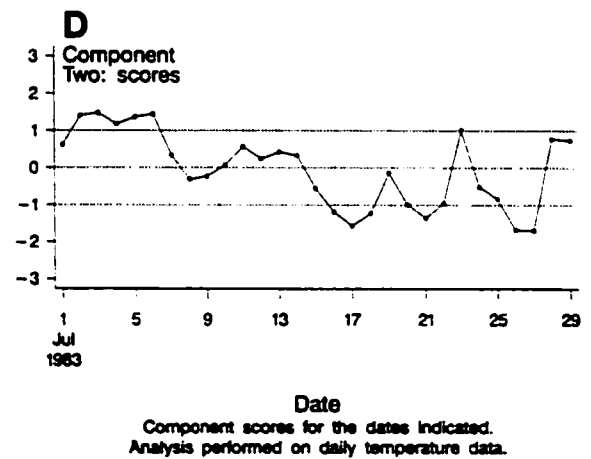
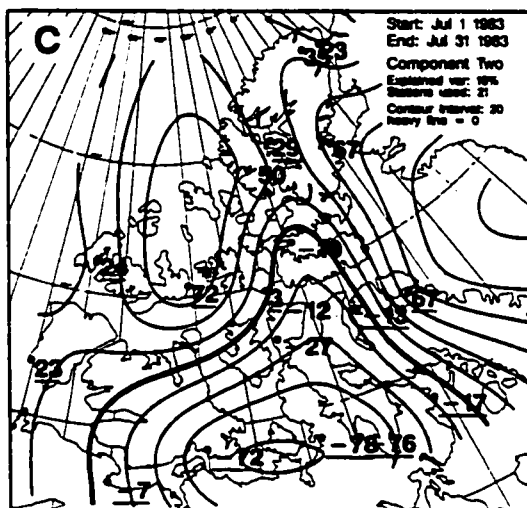
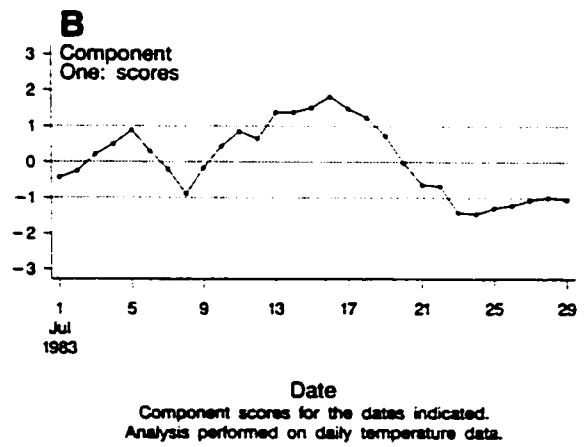
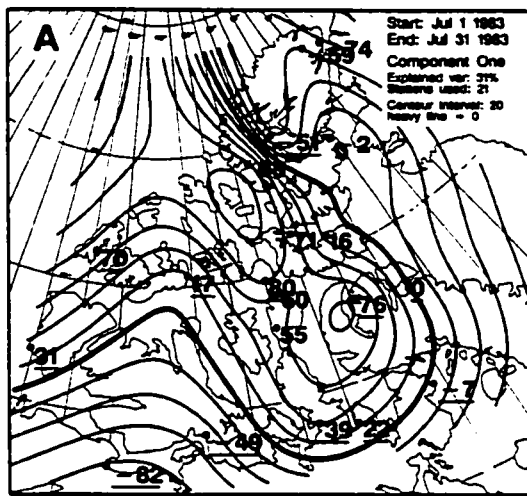


Figure 3.15: Principal components analysis of AES and PCSP July mean daily data for summer, 1983. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined

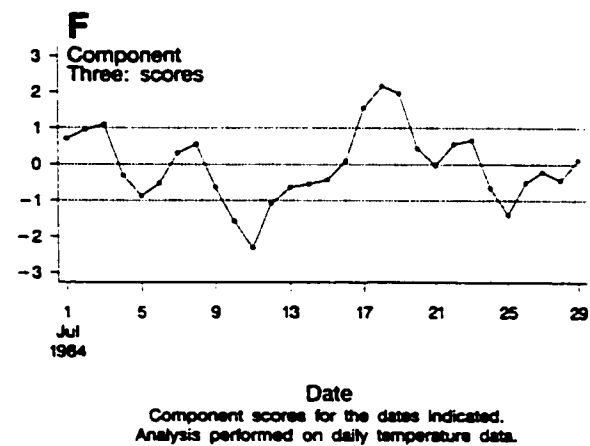
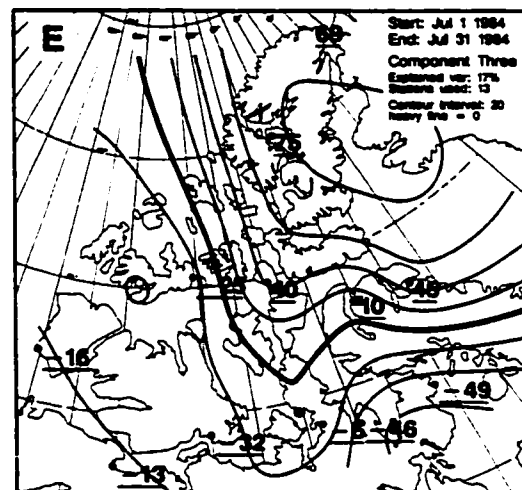
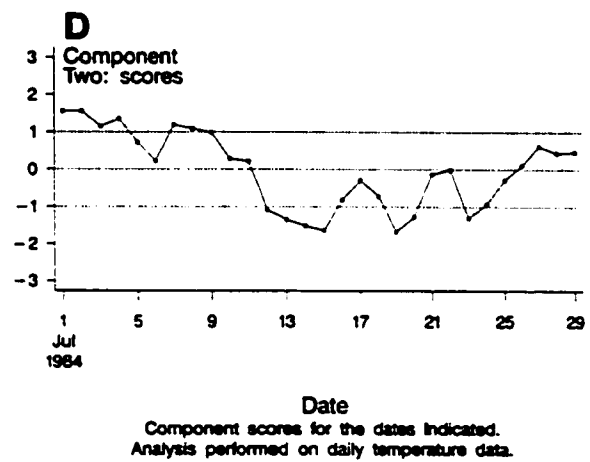
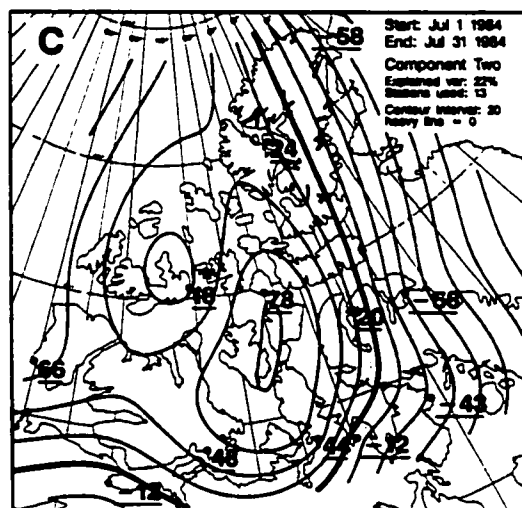
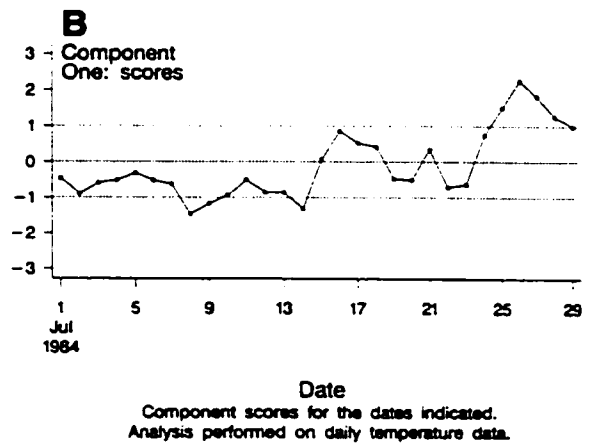
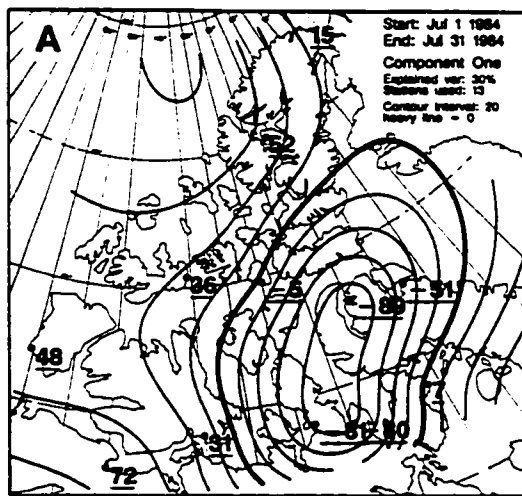
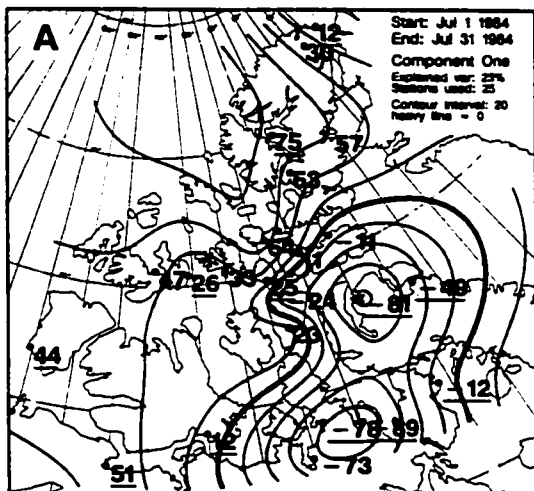


Figure 3.16: Principal components analysis of AES July mean daily data for summer, 1984. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.



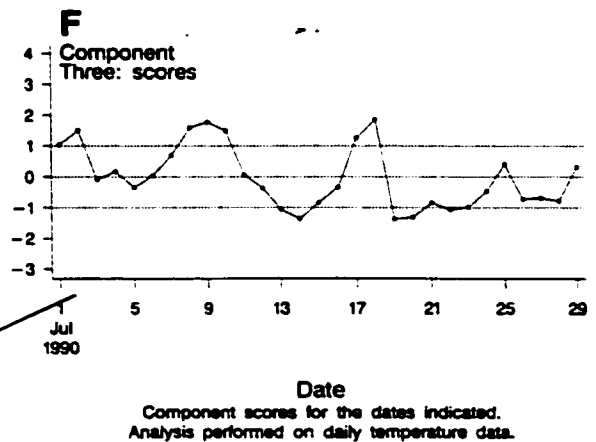
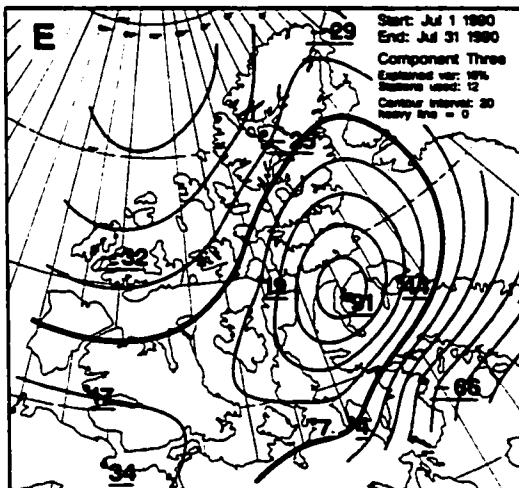
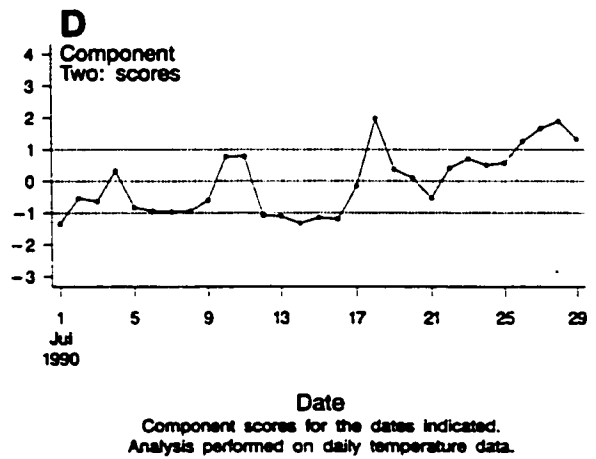
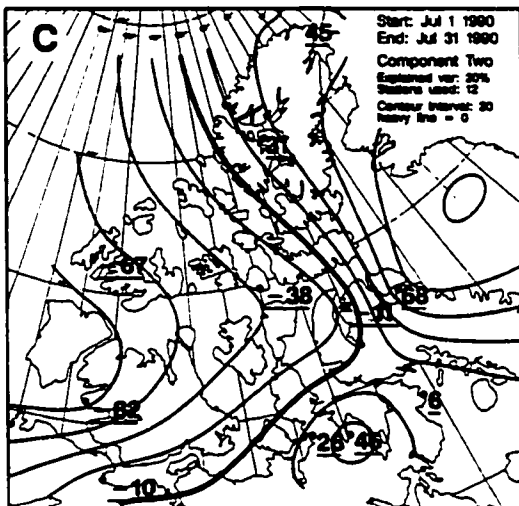
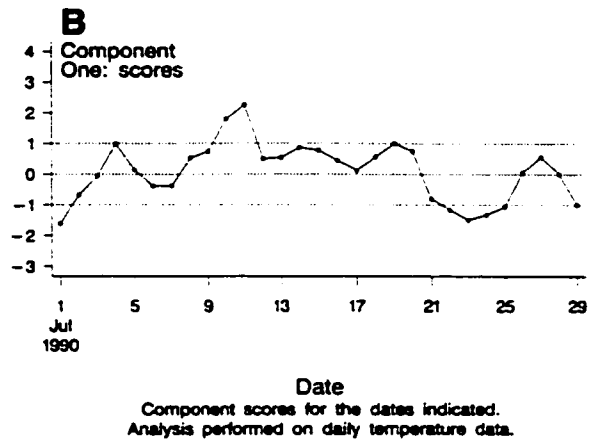
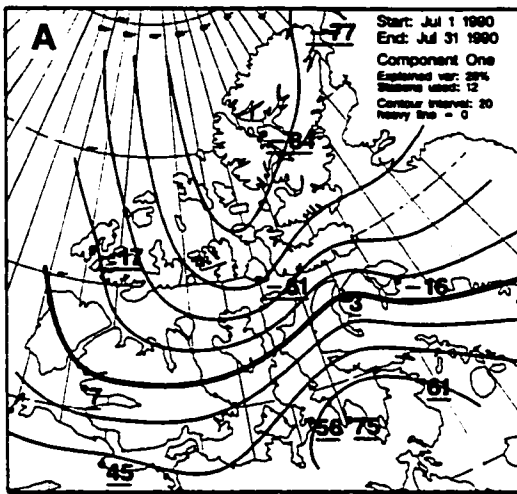


Figure 3.18: Principal components analysis of AES July mean daily data for summer, 1990. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores.

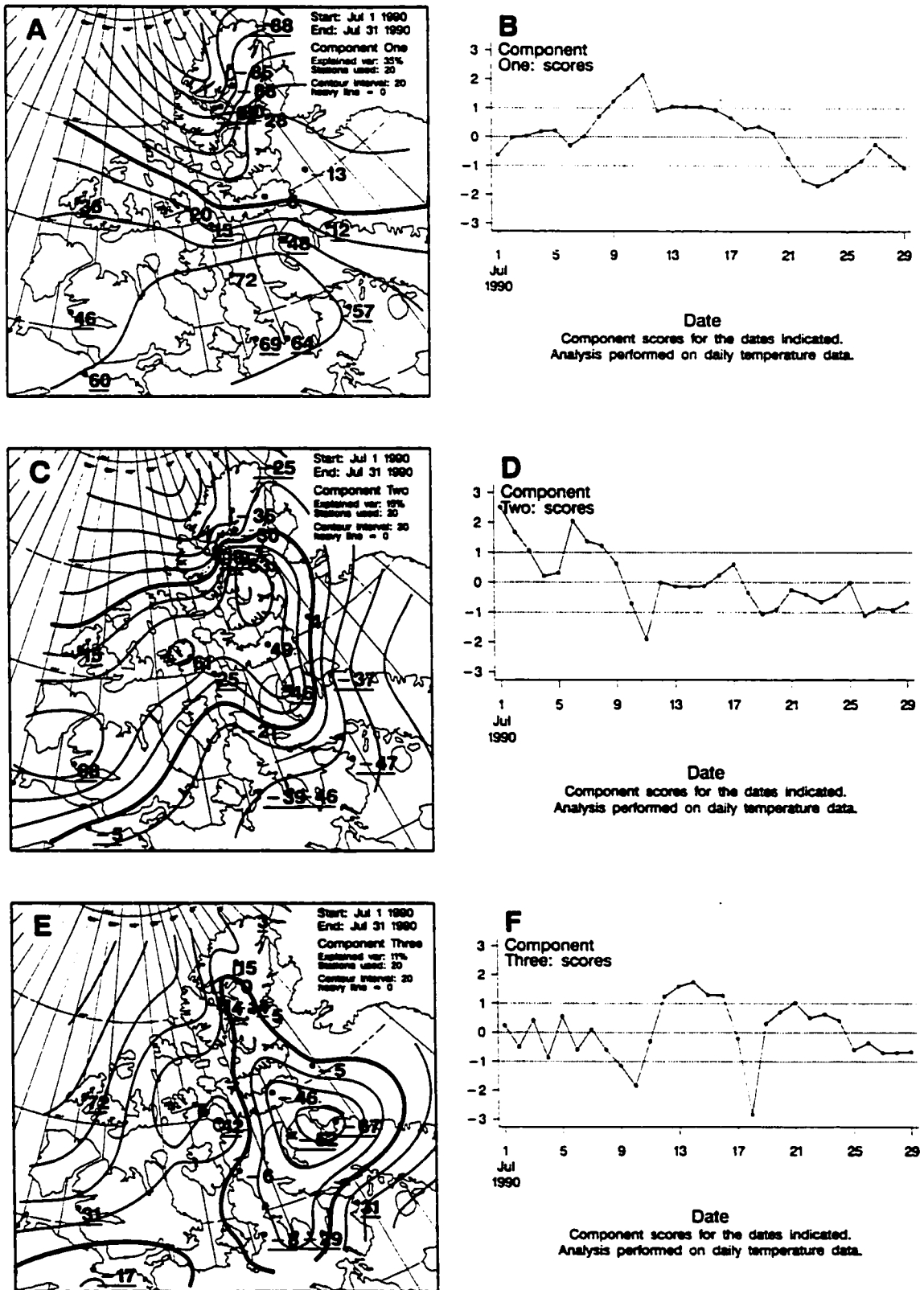


Figure 3.19: Principal components analysis of AES and PCSP July mean daily data for summer, 1990. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined

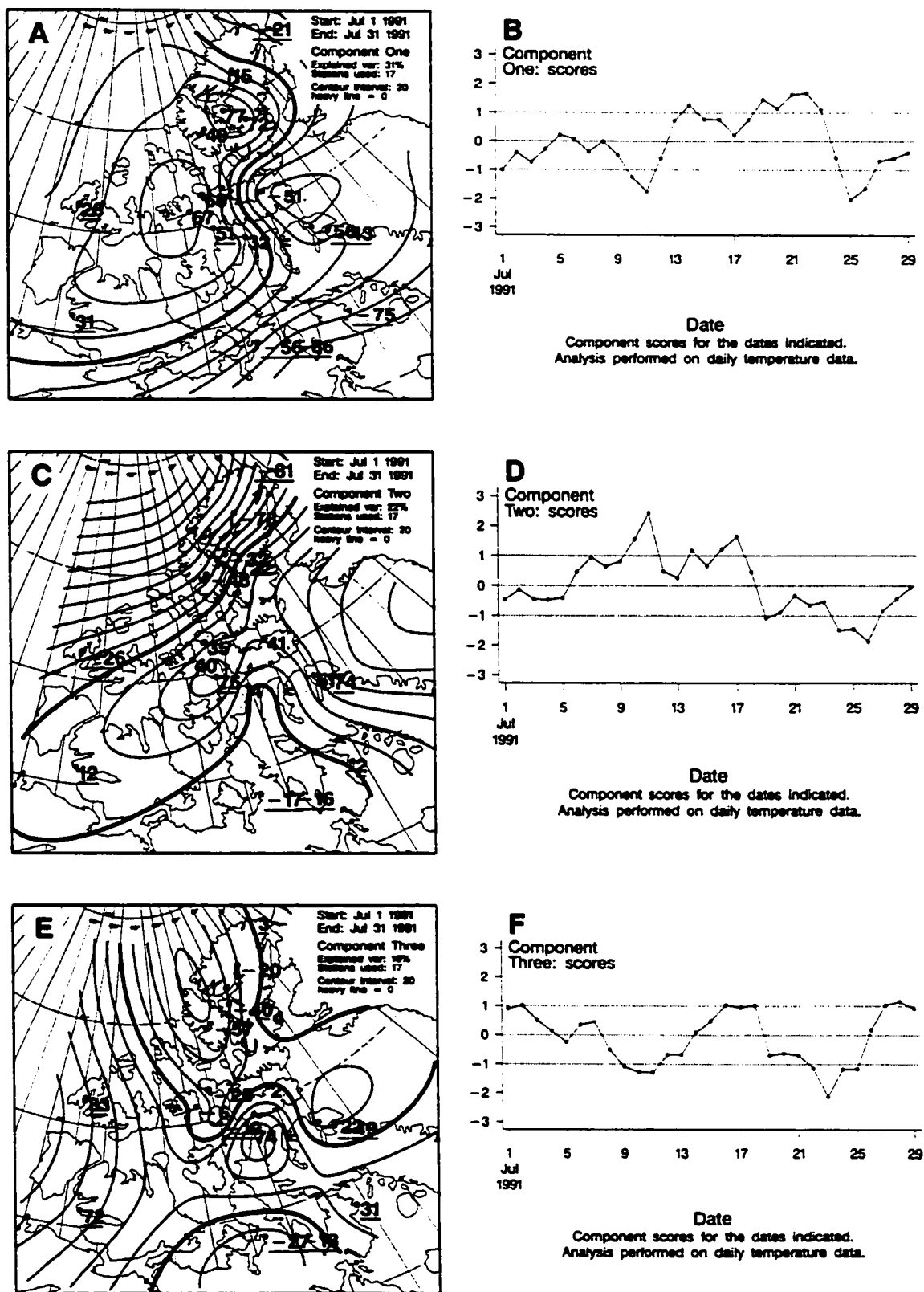


Figure 3.21: Principal components analysis of AES and PCSP July mean daily data for summer, 1991. a), c) and e) are maps of loadings for components one, two and three; b), d) and f) are time-series plots of component scores. AES stations are underlined

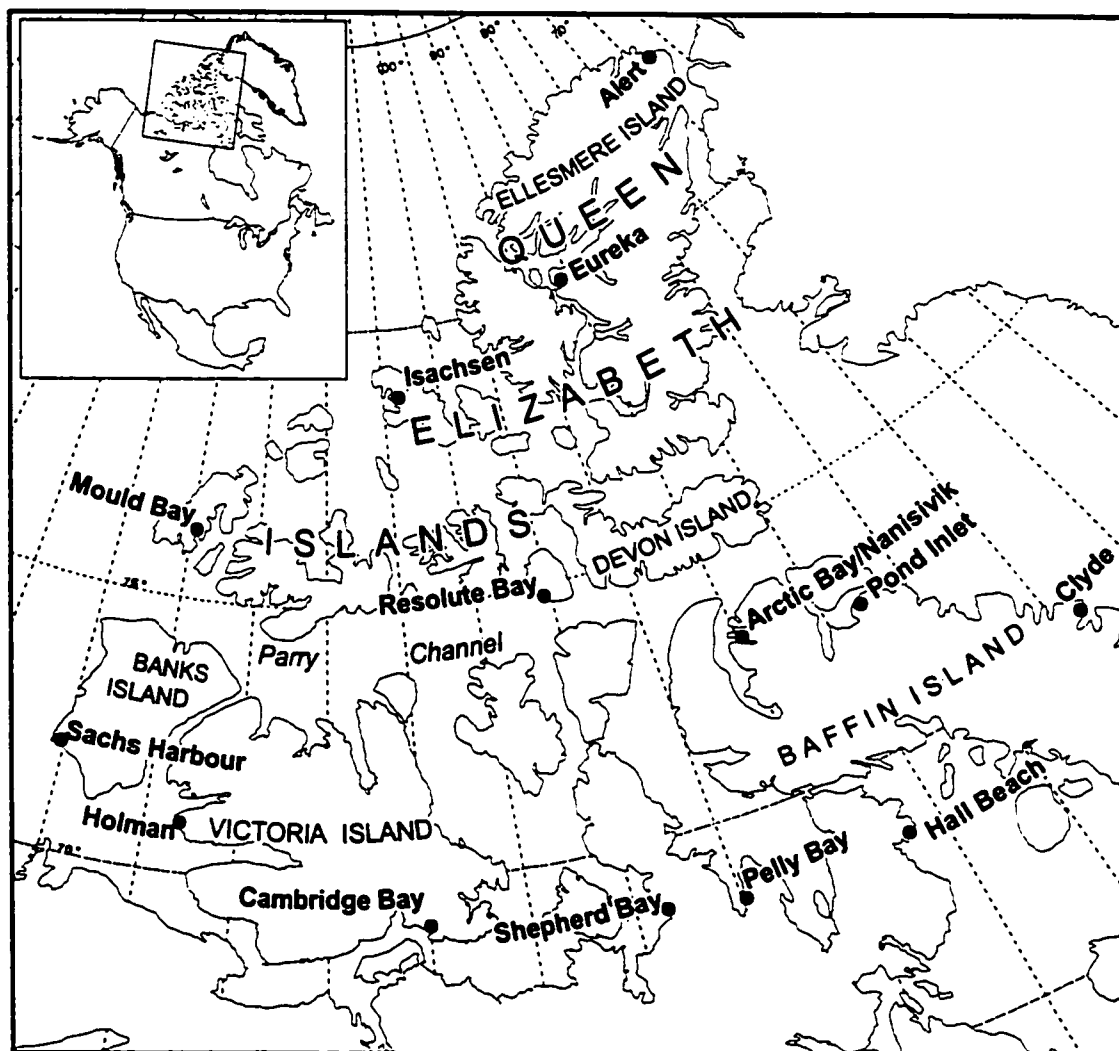


Figure 4.1: The Canadian Arctic Archipelago. Stations of the Atmospheric Environment Service of Canada (AES) are indicated. AES stations with upper air facilities are underlined.

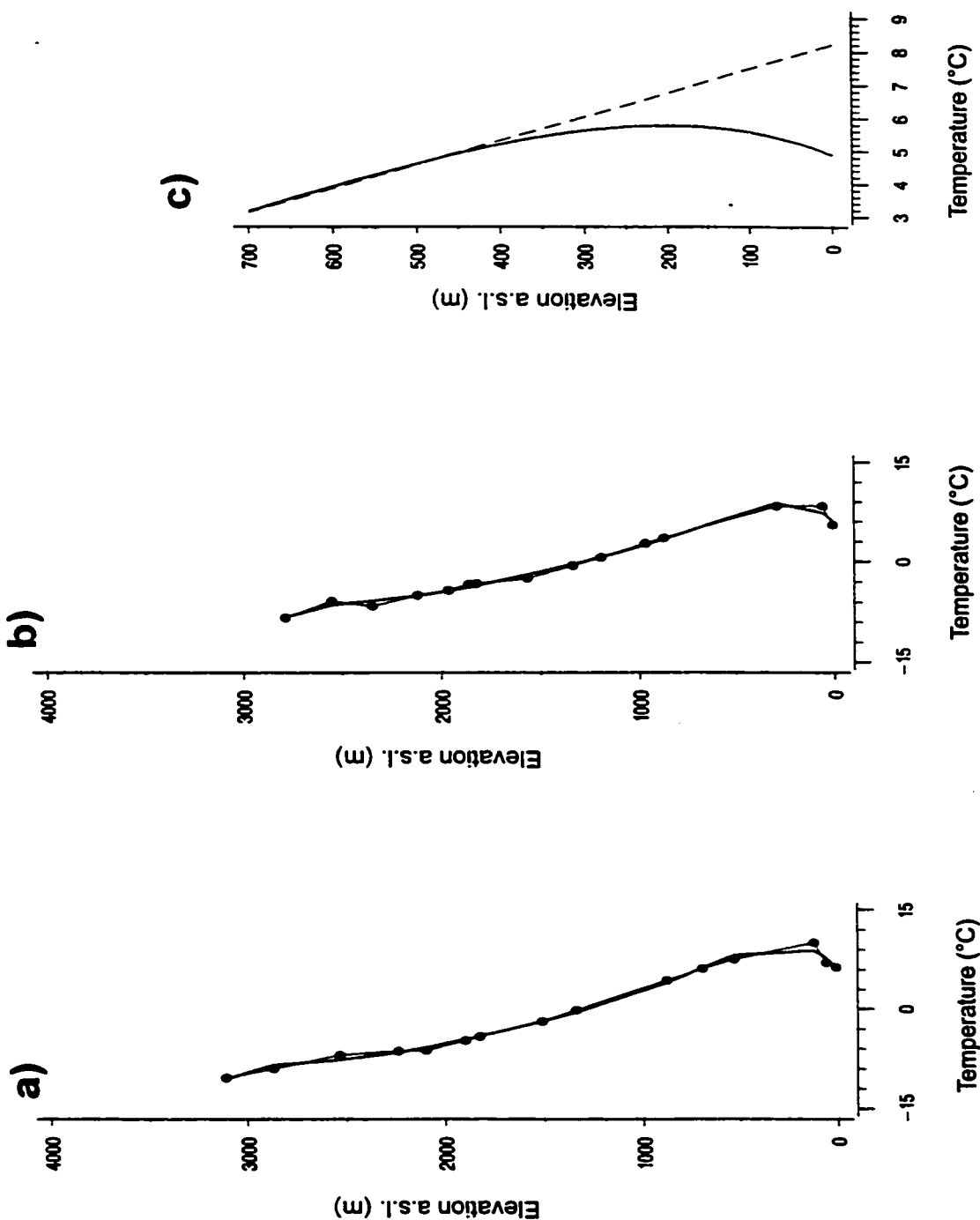


Figure 4.2a: Examples of upper air temperature profiles derived from individual rawinsonde ascents at AES Eureka. a): July 02 1987, 0000 UCT; b): July 23 1987, 1200 UCT. Thin line with black dots represents original data points; thicker line is the polynomial fit. c) Mean rawinsonde ascent profiles for July generated by averaging all polynomial estimates for each ascent over the month of July for a given year. Coastal effect is estimated by taking the difference between the surface temperature (solid line) and the extrapolation of the straight portion of the curve (dashed line) to the surface.

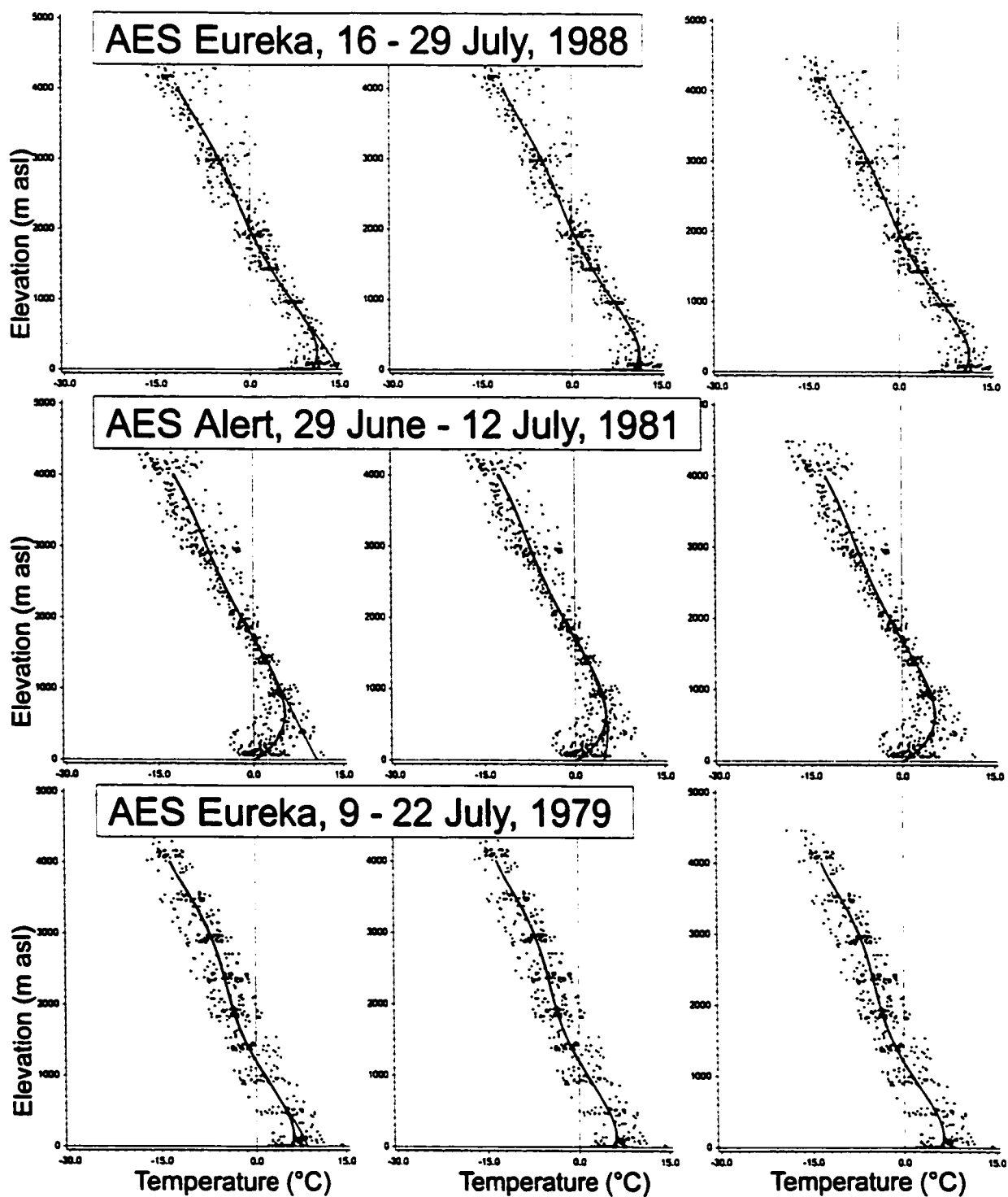


Figure 4.2b: Comparison showing difference between the three new algorithms for inversion removal introduced for sensitivity testing. The first column shows the "low-slope" removal, the second column the "peak-point" removal, and the third column shows no removal. The first row are upper air data from AES Eureka, 16-29 July, 1988, the second row, AES Alert, 29 June - 12 July, 1981, and the third row, AES Eureka, 9 - 22 July, 1979. Heavy line represents mean temperature profile curve, light line represents result of removal algorithm.

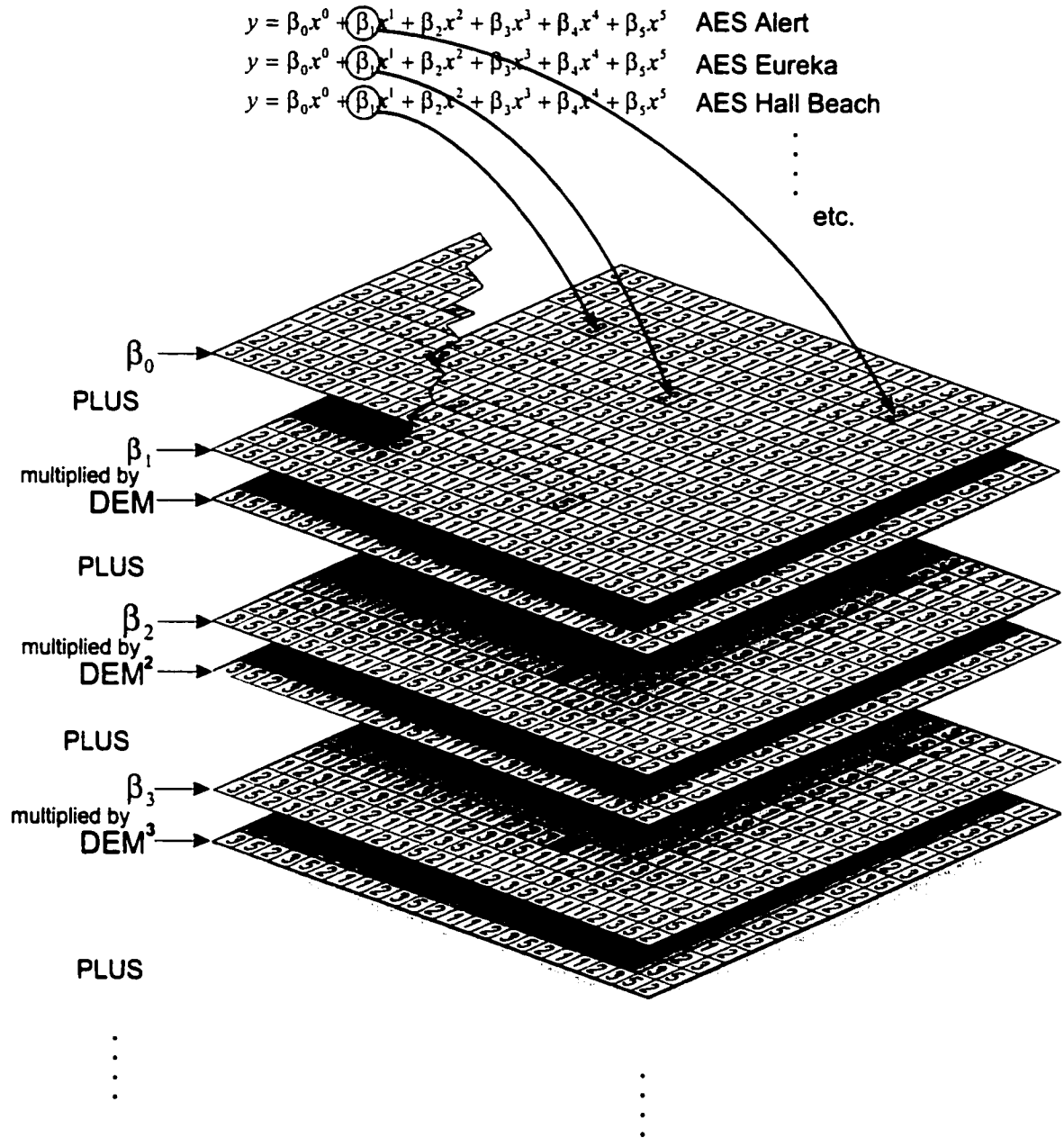


Figure 4.3: Schematic representation depicting how the base temperature estimate is obtained from the polynomial coefficient images and the DEM. Dark pixels indicate the locations of upper air stations on the image. The same coefficient from each station is interpolated to obtain values to form a new image in which each pixel contains an interpolated value for that coefficient. This is repeated for each coefficient. Resultant images are then multiplied with the DEM and these results are totaled, as per the equation.

An example of a 'wind effect filter'. This is not one that is used in the model; they are larger. The values are symmetric and opposite about the center:

0	1	2	1	0
0	0	5	0	0
0	0	0	0	0
0	0	-5	0	0
0	-1	-2	-1	0

The filter is applied to each pixel in a binary representation of the DEM (land=1, ocean=0). At each pixel the filter is multiplied with the part of the binary DEM that is under the filter (i.e., which would in this case be a 5x5 zone). If completely in the ocean the result is 0, if completely on land the result is 0. If the filter straddles a coastline, the result taken is dependent on the orientation of the coast and how far the pixel being examined is from the coast. Below is a an example for which the filter is right on the coastal pixel.

<table border="1"> <tr><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>-5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>-2</td><td>-1</td><td>0</td></tr> </table>	0	1	2	1	0	0	0	5	0	0	0	0	0	0	0	0	0	-5	0	0	0	-1	-2	-1	0	×	<table border="1"> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	=	<table border="1"> <tr><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	1	2	1	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	=	9
0	1	2	1	0																																																																													
0	0	5	0	0																																																																													
0	0	0	0	0																																																																													
0	0	-5	0	0																																																																													
0	-1	-2	-1	0																																																																													
1	1	1	1	1																																																																													
1	1	1	1	1																																																																													
1	1	1	1	1																																																																													
0	0	0	0	0																																																																													
0	0	0	0	0																																																																													
0	1	2	1	0																																																																													
0	0	5	0	0																																																																													
0	0	0	0	0																																																																													
0	0	0	0	0																																																																													
0	0	0	0	0																																																																													
Filter		DEM		Resultant																																																																													

The value obtained after adding the values in the pixels of the resultant is placed in the pixel of a 'wind effect image'. Below is a second example when the pixel under consideration is one pixel in from the coast:

<table border="1"> <tr><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>-5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>-1</td><td>-2</td><td>-1</td><td>0</td></tr> </table>	0	1	2	1	0	0	0	5	0	0	0	0	0	0	0	0	0	-5	0	0	0	-1	-2	-1	0	×	<table border="1"> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	=	<table border="1"> <tr><td>0</td><td>1</td><td>2</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>-5</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>	0	1	2	1	0	0	0	5	0	0	0	0	0	0	0	0	0	-5	0	0	0	0	0	0	0	=	4
0	1	2	1	0																																																																													
0	0	5	0	0																																																																													
0	0	0	0	0																																																																													
0	0	-5	0	0																																																																													
0	-1	-2	-1	0																																																																													
1	1	1	1	1																																																																													
1	1	1	1	1																																																																													
1	1	1	1	1																																																																													
1	1	1	1	1																																																																													
0	0	0	0	0																																																																													
0	1	2	1	0																																																																													
0	0	5	0	0																																																																													
0	0	0	0	0																																																																													
0	0	-5	0	0																																																																													
0	0	0	0	0																																																																													
Filter		DEM		Resultant																																																																													

The final value has dropped, reflecting a decrease in the potential influence of the ocean on that inland pixel. In the model these values are percentages indicating just how much of the ocean temperature should be used at that land pixel. Thus the final result is not a wind value but a reflection of the effect the wind has on the temperature at that point, with respect to the temperature of the source, that is, the ocean and the temperature of the land.

Figure 4.4: The operation of a wind effect filter.

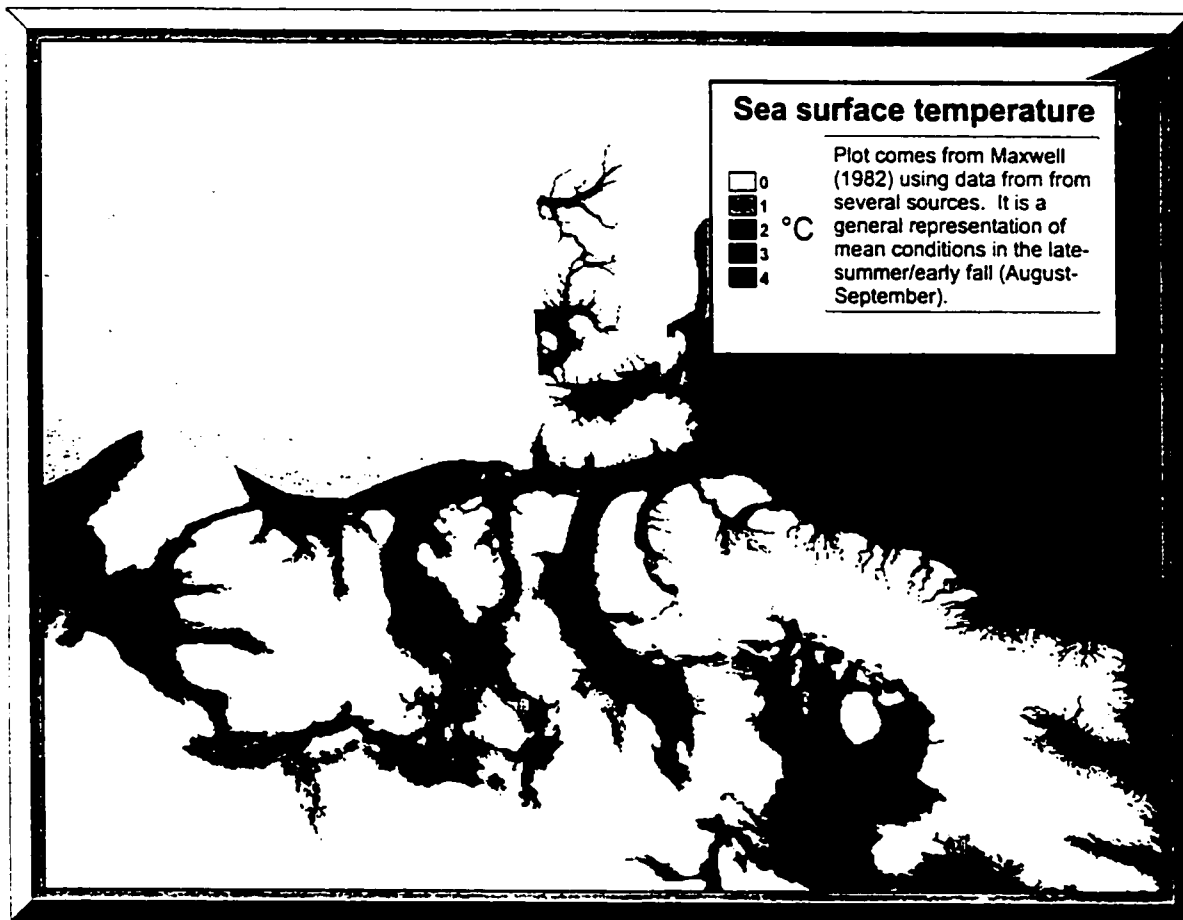


Figure 4.5: Sea-surface temperatures used in the variable SST sensitivity analysis. Source is Maxwell (1982). Plot represents mean late-summer conditions. Original data come from US Navy observations and other sources.



Figure 4.6: Estimates of the mean temperature for the period 4 July – 17 July 1976. Values are rounded to the nearest degree Celsius. Ocean is a uniform 2°C. Warmer areas are light, cooler areas are dark. Values range from a maximum of 7°C to a minimum of -13°C.



Figure 4.7: Estimates of the mean temperature for the period 16 July – 29 July 1988. Values are rounded to the nearest degree Celsius. Ocean is a uniform 2°C. Warmer areas are light, cooler areas are dark. Values range from a maximum of 13°C to a minimum of -5°C.

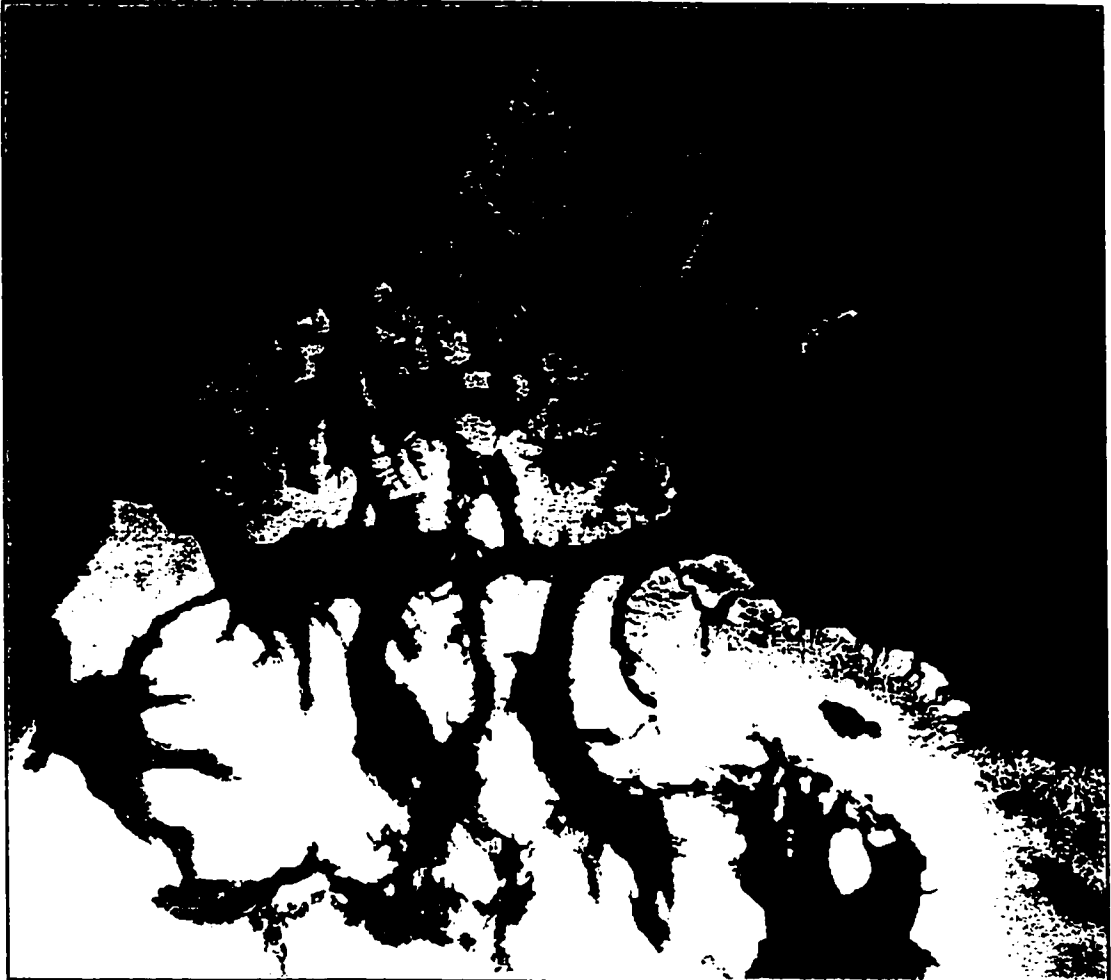


Figure 4.8: Estimates of the mean temperature for the period 9 July – 22 July 1990. Values are rounded to the nearest degree Celsius. Ocean is a uniform 2°C. Warmer areas are light, cooler areas are dark. Values range from a maximum of 13°C to a minimum of –13°C.

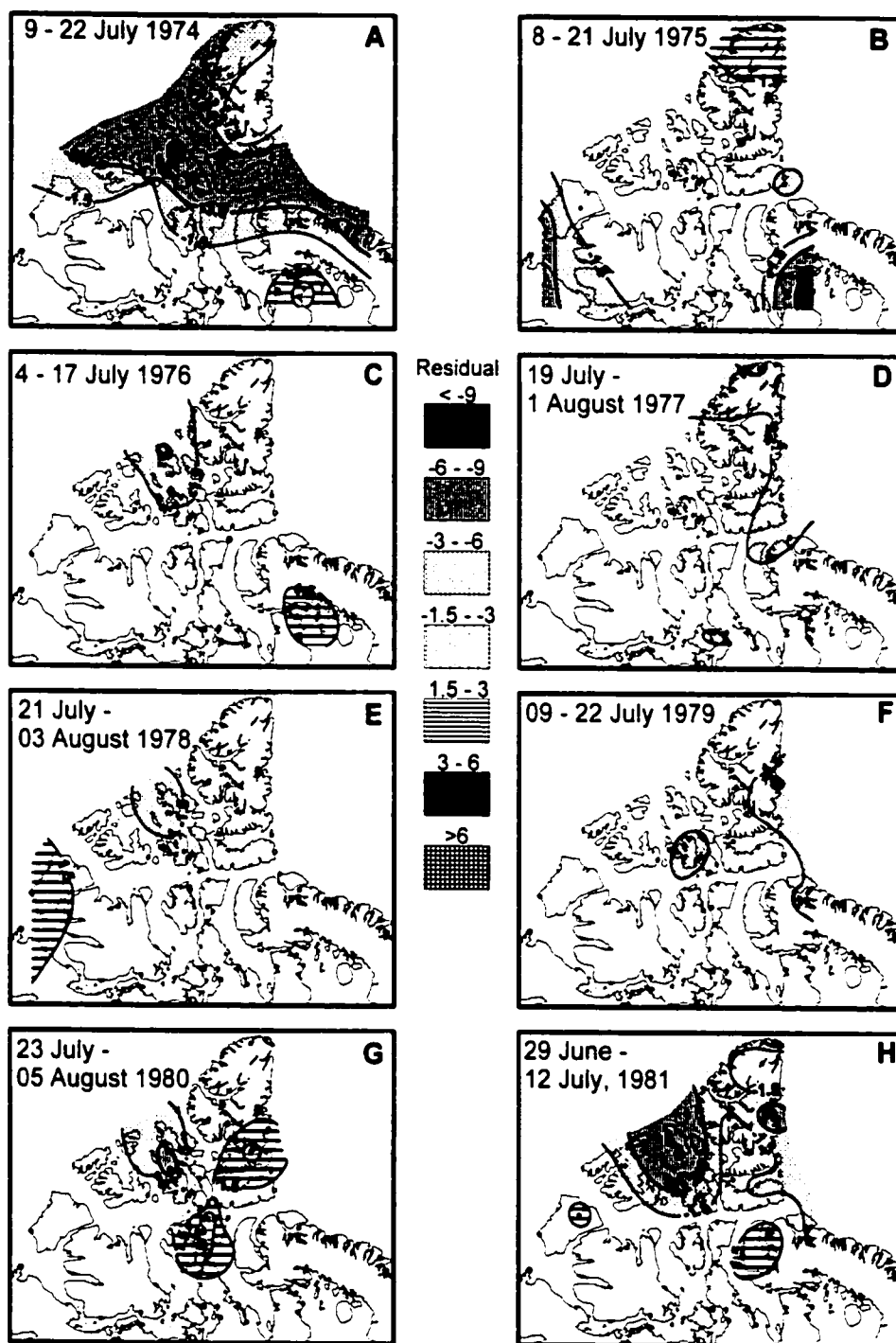


Figure 4.9a-p: Plots of residuals for each of the 14-day model runs. Contours are at 6, 4.5, 3, 1.5, -1.5, -3, -4.5, -6; units are degrees Celsius. Heavy shading are zones of negative residual less than -1.5°C (negative residuals); light shading are zones of positive residual more than $+1.5^{\circ}\text{C}$ (positive residuals). Negative residuals occur where observed values are less than 1.5°C less than the model estimate; positive residuals occur where observed values are more than 1.5°C more than the model estimate. Periods of the runs are indicated on the plots

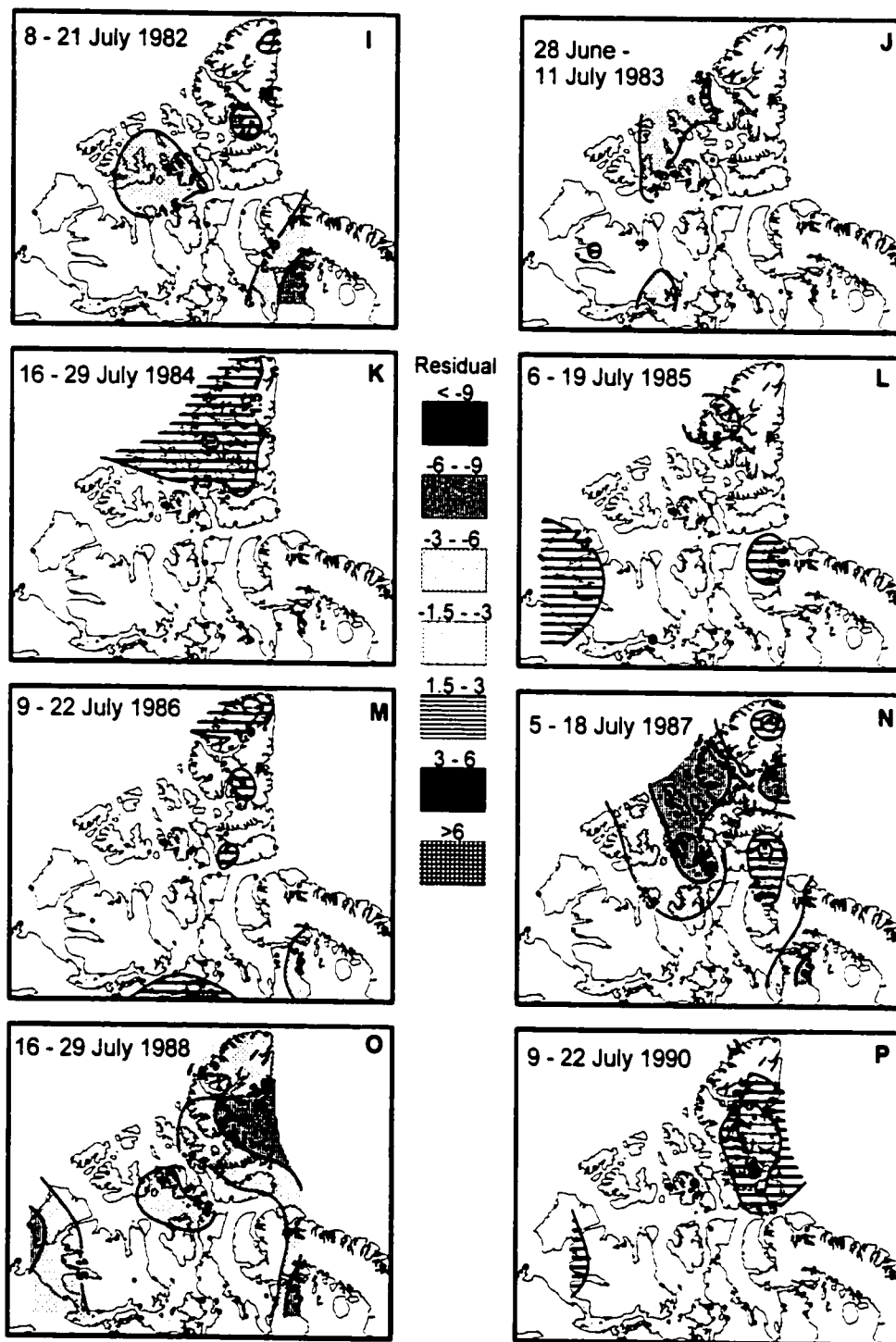


Figure 4.9a-p: Plots of residuals for each of the 14-day model runs. Contours are at 6, 4.5, 3, 1.5, -1.5, -3, -4.5, -6; units are degrees Celsius. Heavy shading are zones of negative residual less than -1.5°C (negative residuals); light shading are zones of positive residual more than $+1.5^{\circ}\text{C}$ (positive residuals). Negative residuals occur where observed values are less than 1.5°C less than the model estimate; positive residuals occur where observed values are more than 1.5°C more than the model estimate. Periods of the runs are indicated on the plots

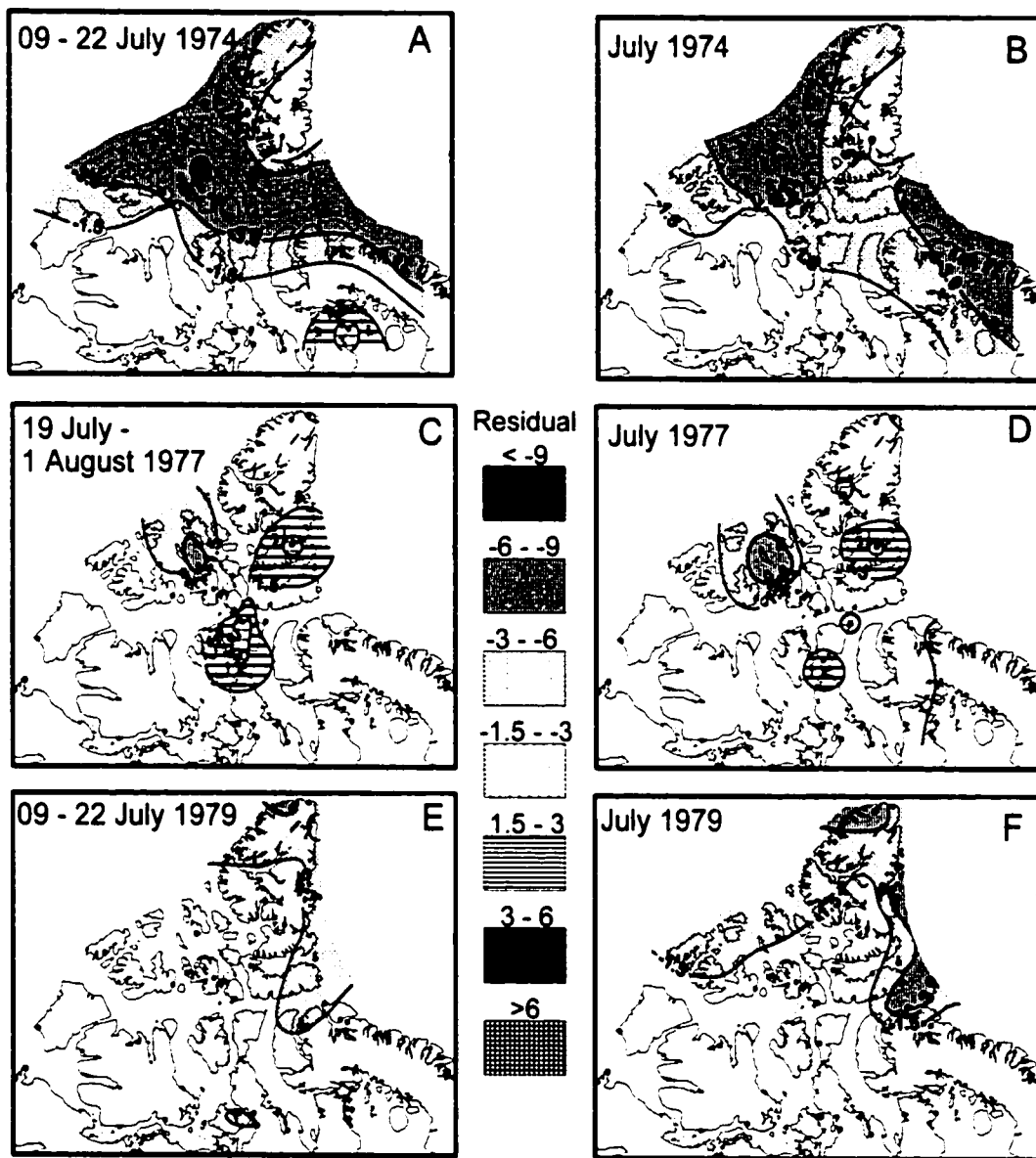


Figure 4.10a-f: Comparison of residuals plots from 14-day model runs and from runs conducted for the month of July of the same year. Contours are at 6, 4.5, 3, 1.5, -1.5, -3, -4.5, -6; units are degrees Celsius. Heavy shading are zones of negative residual less than -1.5°C (negative residuals); light shading are zones of positive residual more than $+1.5^{\circ}\text{C}$ (positive residuals). Negative residuals occur where observed values are less than 1.5°C less than the model estimate; positive residuals occur where observed values are more than 1.5°C more than the model estimate. Periods of the runs are indicated on the plots.

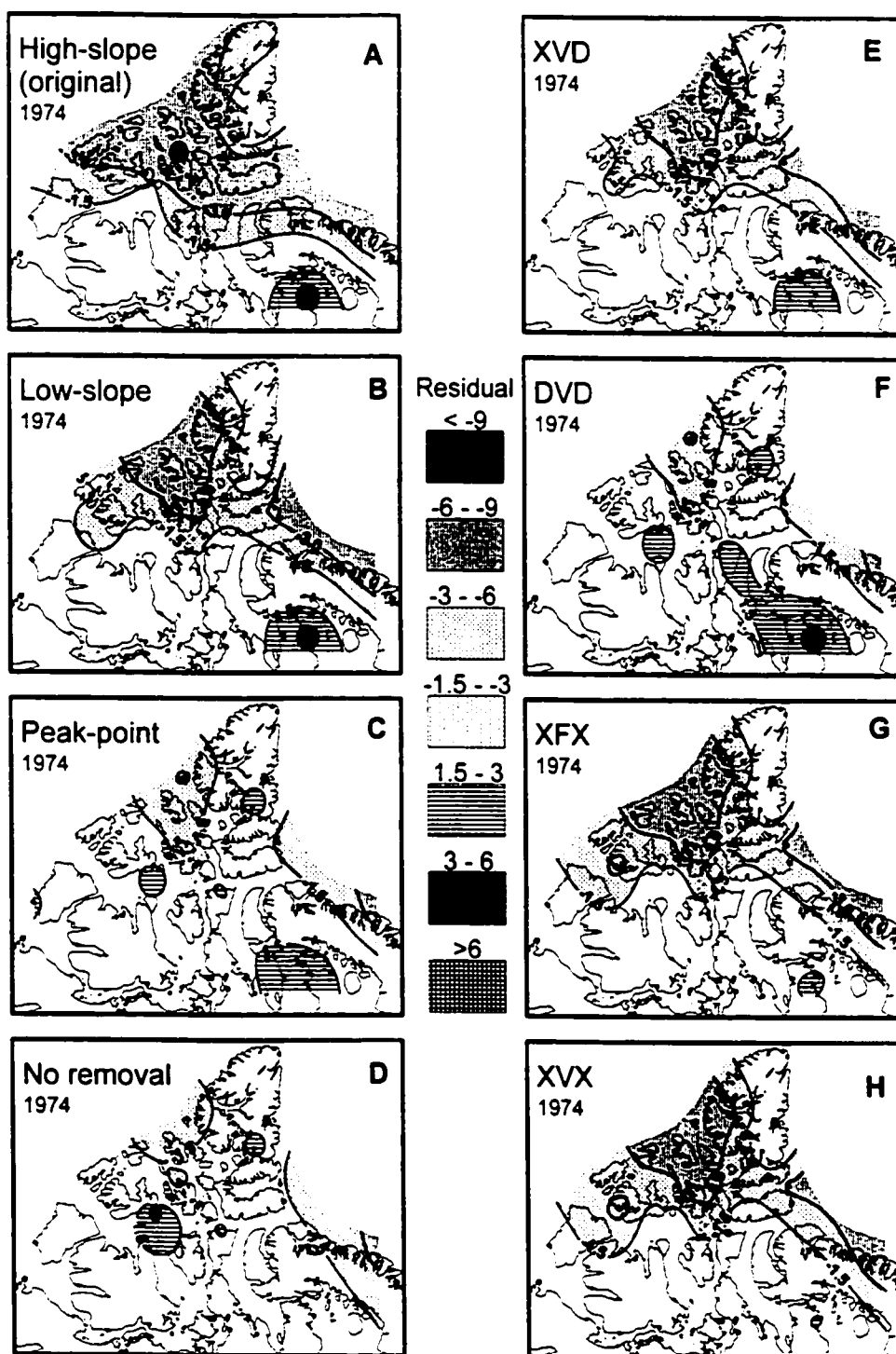


Figure 4.11a-h: Comparisons between different sensitivity runs for the 1974 run period (9 – 22 July, 1974). Sources of the results is as follows: A - original (high slope algorithm) model, B – low-slope algorithm, C – peak-point algorithm, D – no curve modification, E – low-slope algorithm with variable SST, F – peak-point algorithm with variable SST, G – low-slope algorithm, constant SST and maximum wind, H – low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates).

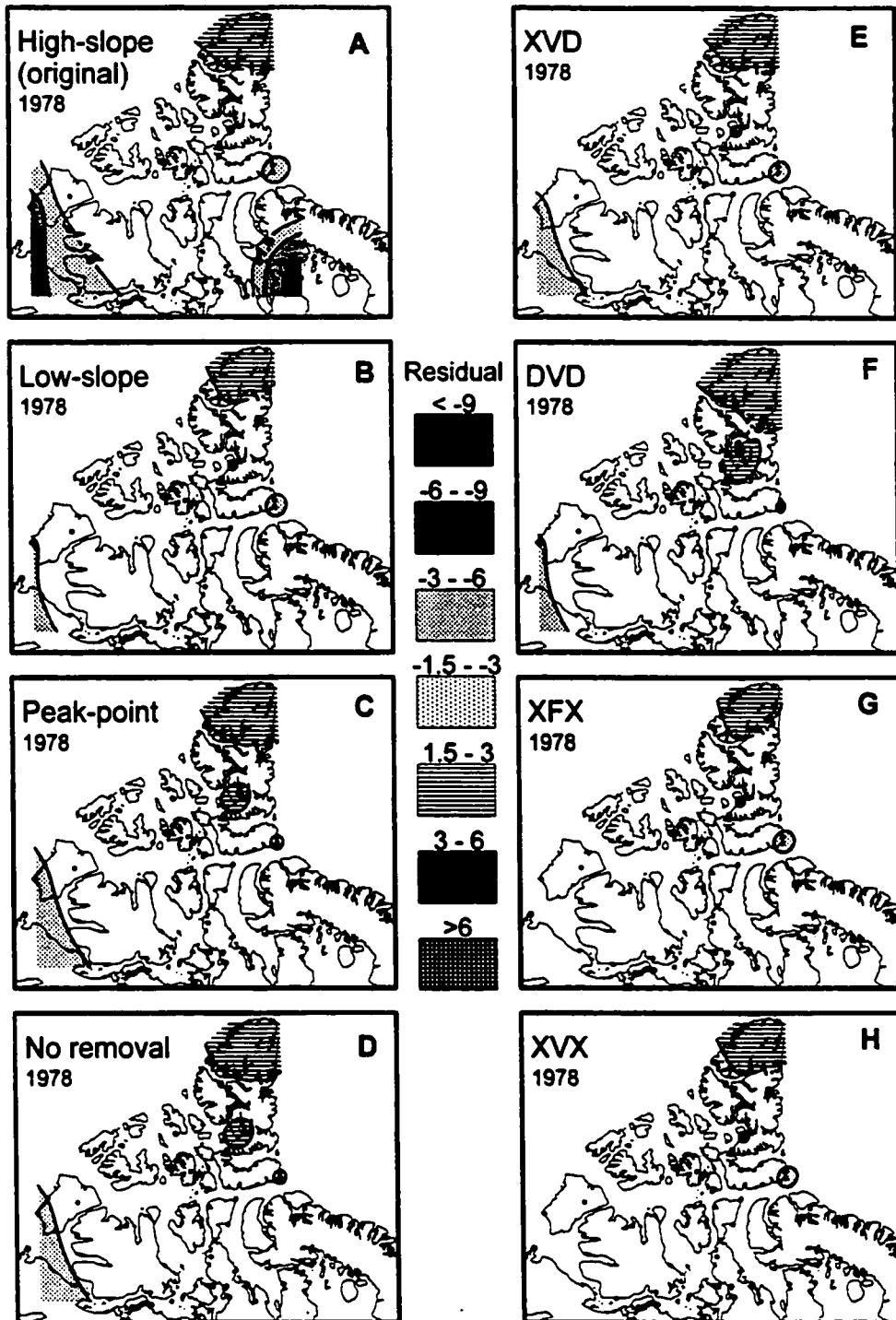


Figure 4.12a-h: Comparisons between different sensitivity runs for the 1978 run period (21 July – 3 August, 1978). Sources of the results is as follows: A - original (high slope algorithm) model, B – low-slope algorithm, C – peak-point algorithm, D – no curve modification, E – low-slope algorithm with variable SST, F – peak-point algorithm with variable SST, G – low-slope algorithm, constant SST and maximum wind, H – low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates).

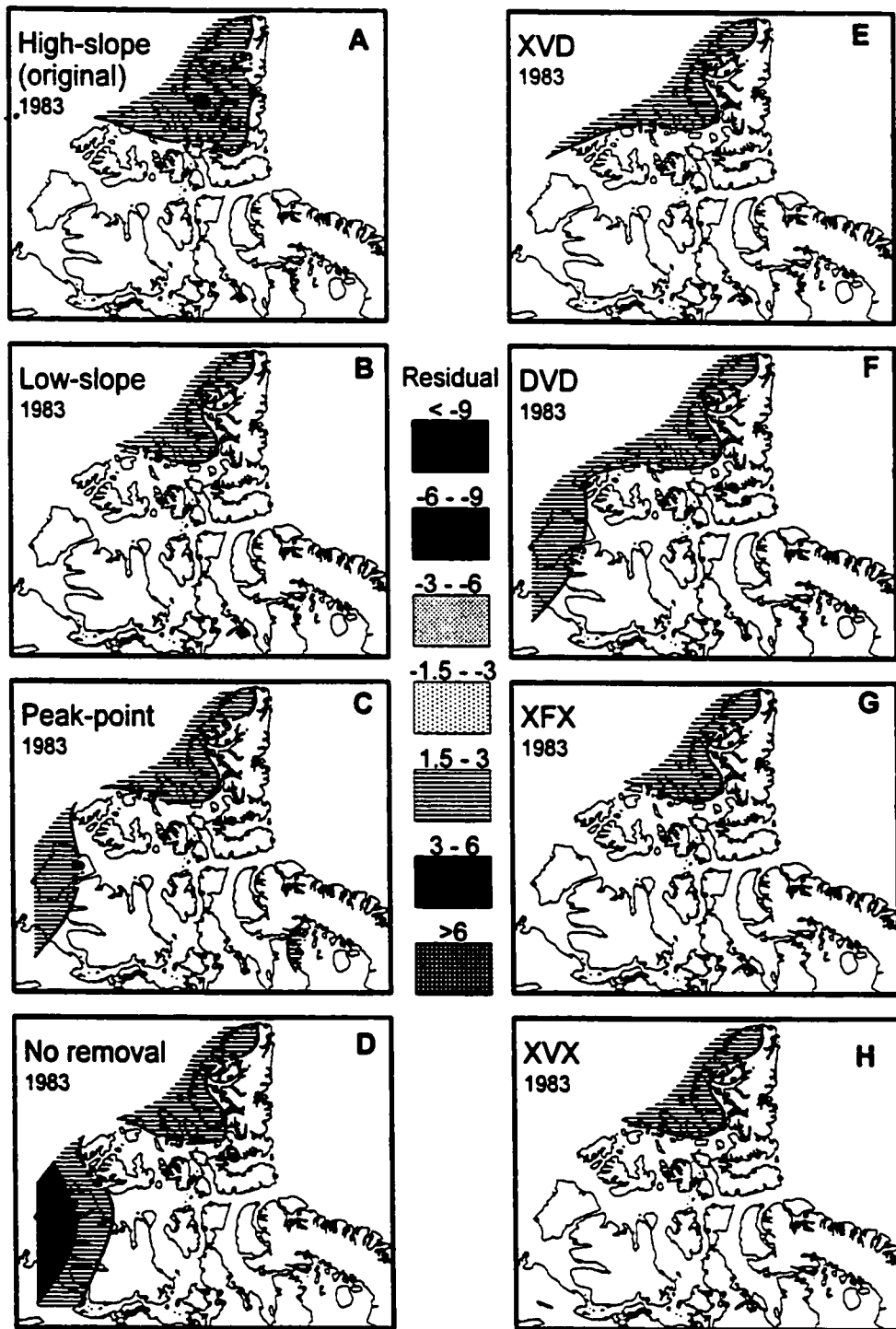


Figure 4.13a-h: Comparisons between different sensitivity runs for the 1983 run period (28 June – 11 July, 1983). Sources of the results is as follows: A - original (high slope algorithm) model, B - low-slope algorithm, C - peak-point algorithm, D - no curve modification, E - low-slope algorithm with variable SST, F - peak-point algorithm with variable SST, G - low-slope algorithm, constant SST and maximum wind, H - low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates).

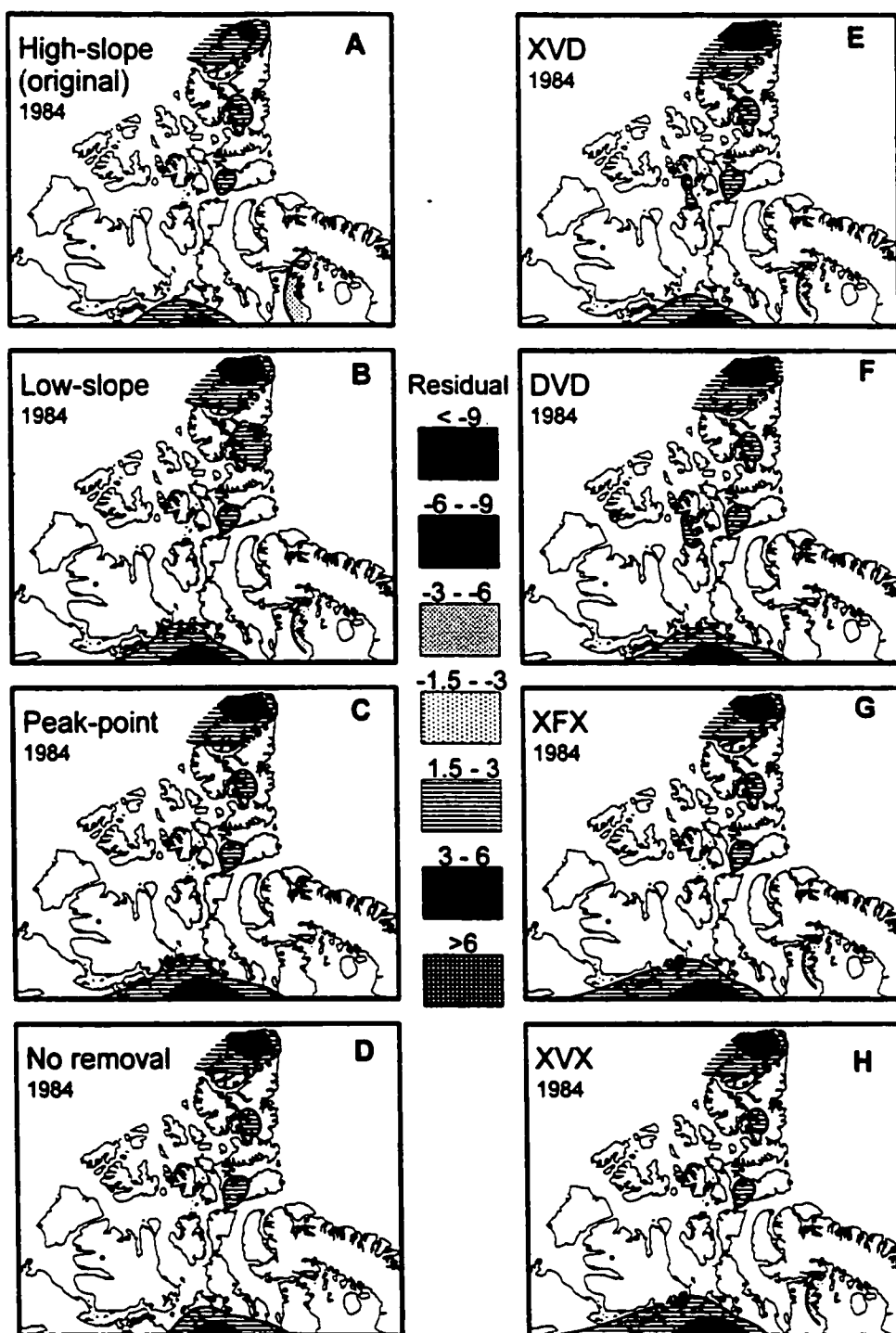


Figure 4.14a-h: Comparisons between different sensitivity runs for the 1984 run period (16 – 29 July, 1984). Sources of the results is as follows: A - original (high slope algorithm) model, B – low-slope algorithm, C – peak-point algorithm, D – no curve modification, E – low-slope algorithm with variable SST, F – peak-point algorithm with variable SST, G – low-slope algorithm, constant SST and maximum wind, H – low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates).

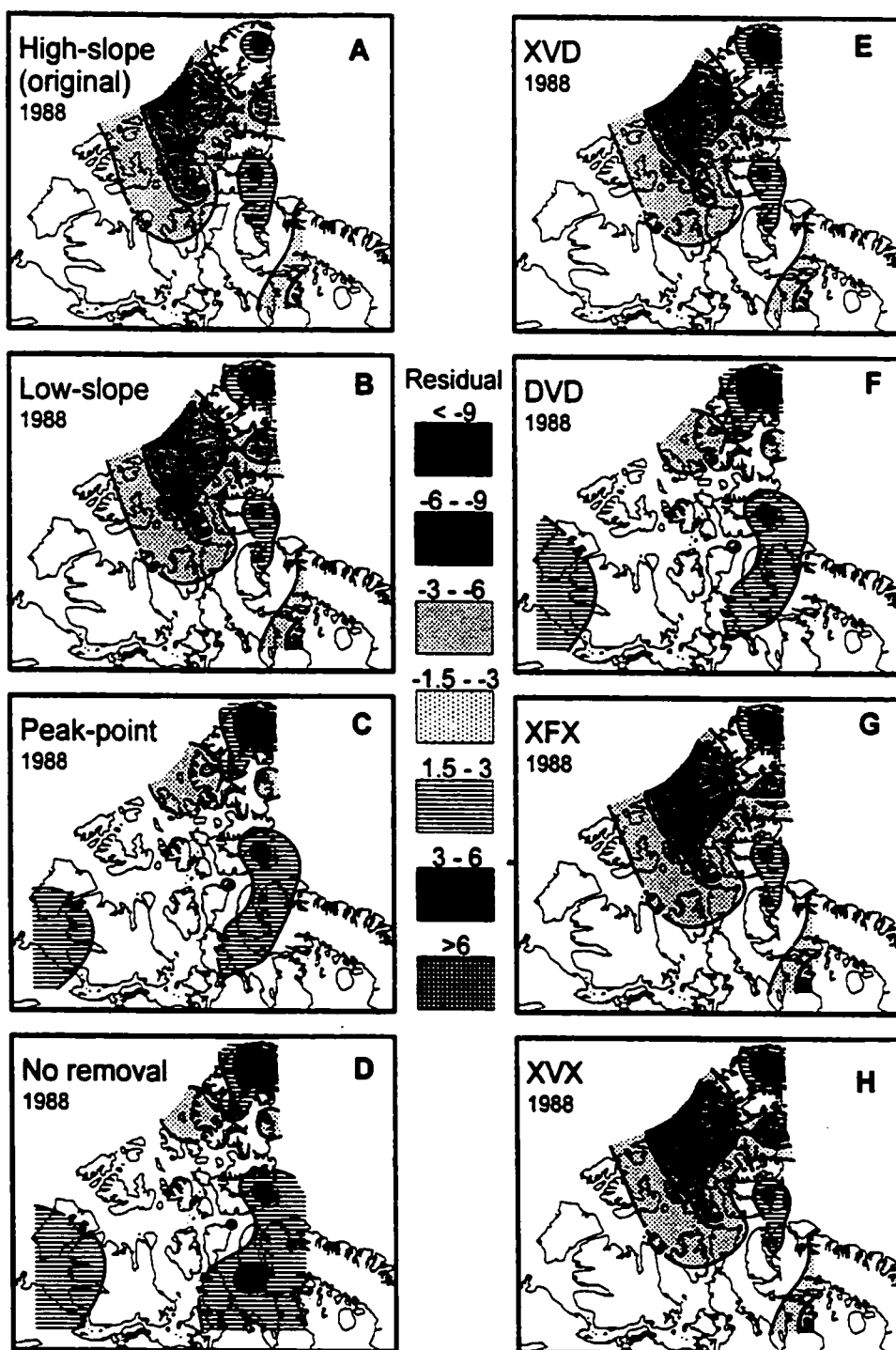


Figure 4.15a-h: Comparisons between different sensitivity runs for the 1988 run period (16 - 29 July, 1988). Sources of the results is as follows: A - original (high slope algorithm) model, B - low-slope algorithm, C - peak-point algorithm, D - no curve modification, E - low-slope algorithm with variable SST, F - peak-point algorithm with variable SST, G - low-slope algorithm, constant SST and maximum wind, H - low-slope algorithm, variable SST and max wind. Shading indicates negative residuals (observed less than model estimates); horizontal line pattern indicates positive residuals (observed greater than model estimates).

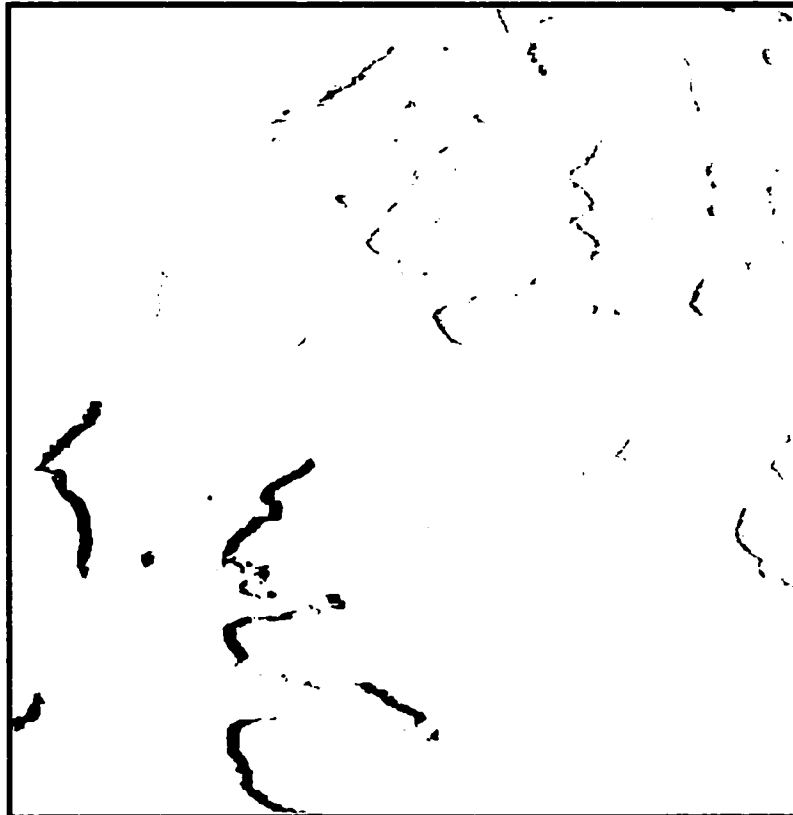


Figure 4.16: Example of net change with introduction of variable SST (variable SST minus constant SST). Dark shading represent areas in which an increase in temperature was observed, light shading represent areas in which a decrease in temperature was observed.

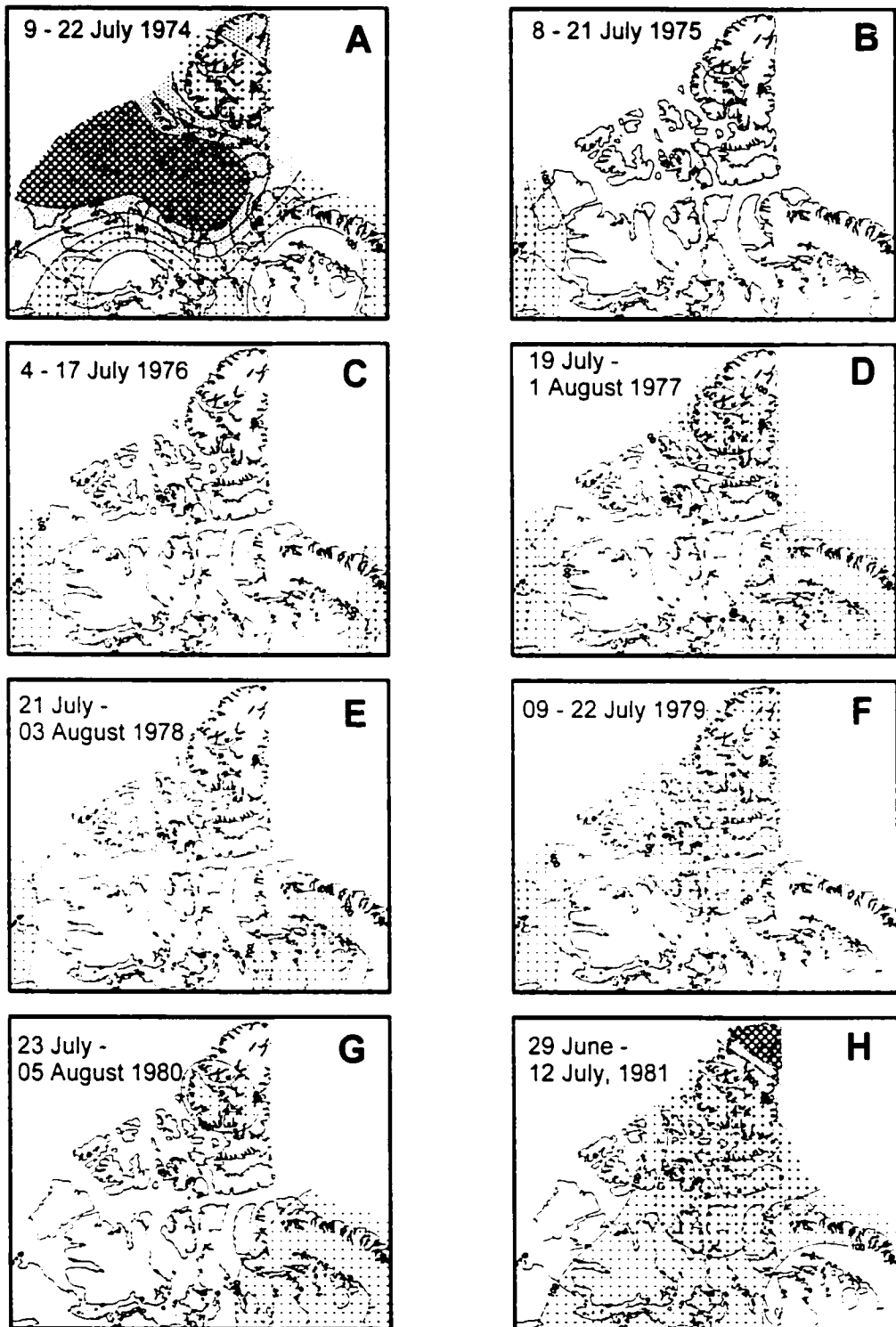


Figure 4.17a-p: Plots of inversion heights at the upper air stations for the periods of the model runs (14-day runs). 'Inversion height' is the distance from the ground to the elevation of maximum temperature on the vertical temperature profile. Elevations are in meters; contour interval is 100 m.

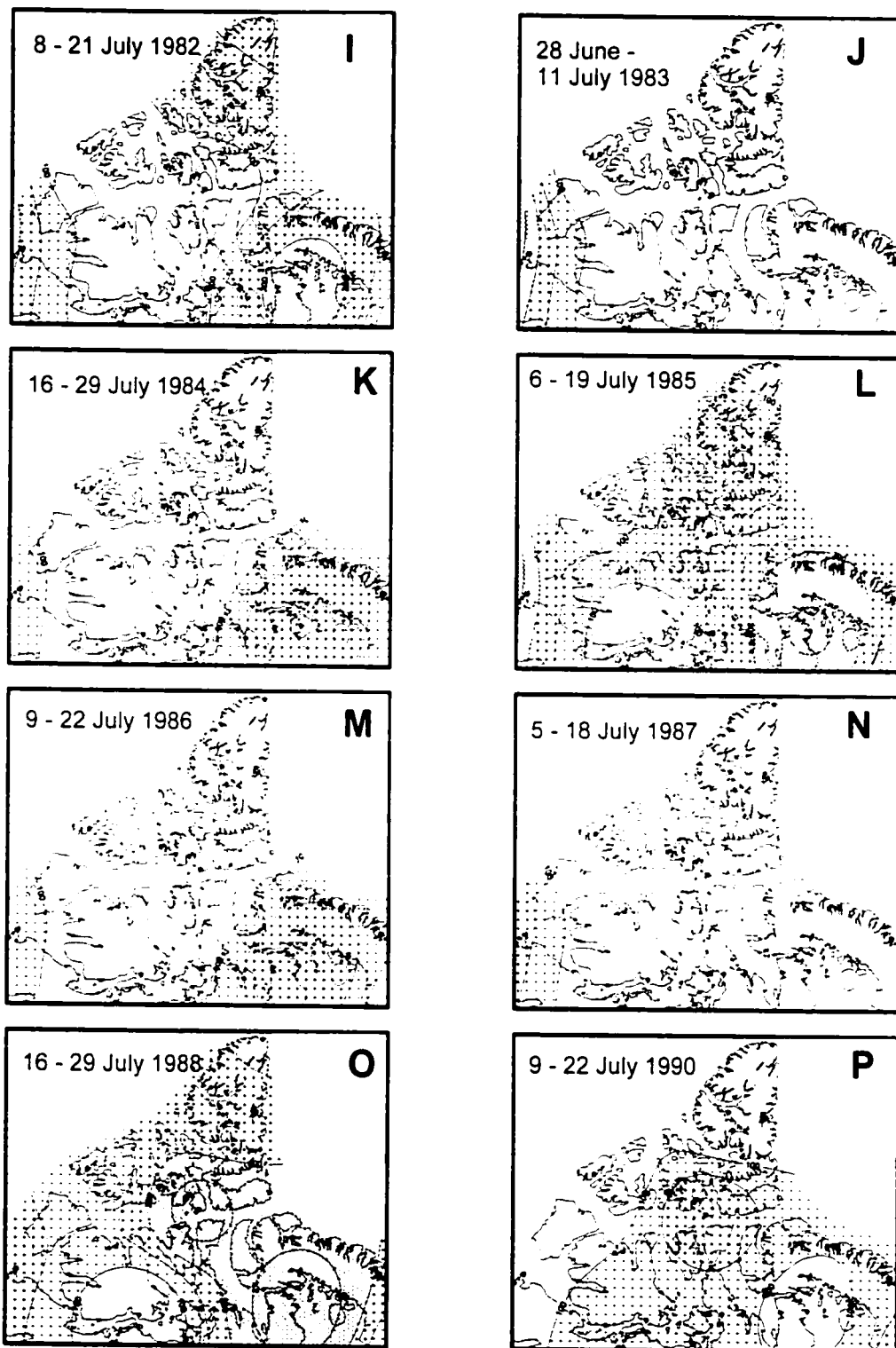


Figure 4.17a-p: Plots of inversion heights at the upper air stations for the periods of the model runs (14-day runs). 'Inversion height' is the distance from the ground to the elevation of maximum temperature on the vertical temperature profile. Elevations are in meters; contour interval is 100 m.

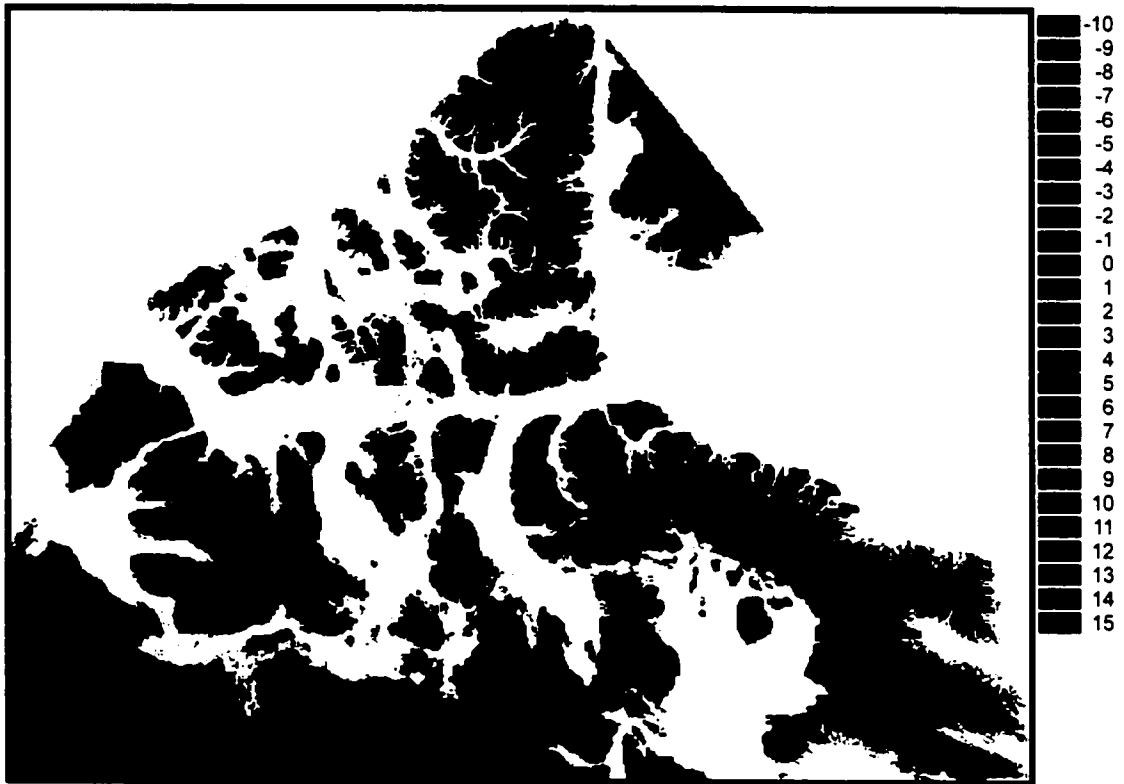


Figure 4.18: Mean temperatures for the period 16 July – 29 July 1984. Model parameters include "peak point" inversion removal, variable SST, and normal wind field. (model estimates).



Figure 4.19: Mean temperatures for the period 16 July – 29 July 1988. Model parameters include "low-slope" inversion removal, variable SST, and normal wind field. (model estimates).

Appendix A: Data structures and computer code

Table of Contents

POLAR CONTINENTAL SHELF PROJECT DATABASE COUNTS BY ELEMENT, YEAR, MONTH	170
OBSERVATIONAL DATA	171
1974 - 1983	171
<i>Original data structure</i>	171
1984 - 1986.....	172
<i>Original data structure</i>	172
1987 - 1993.....	173
<i>Original data structure</i>	173
STATION DATA	176
IDENTIFICATION CROSS REFERENCE STATION DATABASE	176
MAPPING STATION LOCATIONS.....	177
FIRST STATION INFORMATION DATABASE.....	178
SECOND STATION INFORMATION DATABASE	180
SAS PROGRAMS	183
SAS PROGRAM 1: BIG-READ.SAS	183
SAS PROGRAM 2: YEARREAD.SAS	185
SAS PROGRAM 3: READSTNS.SAS	187
SAS PROGRAM 4: COASTS.SAS	188
SAS PROGRAMS FOR 1987 - 1988	205
<i>SAS Program: 8788read.sas</i>	205
<i>SAS Program: 8788ex.sas</i>	207
<i>SAS Program: 8788mrg.sas</i>	210
QUATTRO MACRO 1: M8788.WQM	211
UNIX PROGRAMS (KORN SHELL, AWK).....	214
UNIX PROGRAM 1: TITLECNT.AWK, VERSION 1.....	214
UNIX PROGRAM 2: TITLECNT.AWK, VERSION 2.....	214
UNIX PROGRAM 3: UNDER_SCORE.AWK	215
UNIX PROGRAM 4: UNDER_SCORE2.AWK	216
UNIX PROGRAM 5: PLACER.AWK.....	216

Counts of climatic elements by month and by year for the PCSP database.

	1974	1975	1976	1977	1978	1979	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993
Total	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526
Dry Bulb T	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526	51526
% obs with T	92.5	97.8	98.1	91.9	92.7	97.5	95.3	93.9	97.4	85.9	92	90.8	85.3	87.3	76.8	89.9	81.3	91.1	88.7	90.9
Dew Point T	16875	1647	958	1245	795	1860	1113	1285	1126	961	668	1018	585	686	478	302	86	373	1074	333
Max T (12 hr)	13985	322	1	193	130	1892	2294	2129	2028	1583	1577	0	0	0	651	757	131	129	76	75
Min T(12 hr)	15811	320	1	228	231	2131	2380	2361	2447	1642	1805	0	0	0	610	754	17	204	322	282
Wind vel	56580	5251	4523	3630	3424	3638	3357	3828	3107	2841	3077	3556	3170	3274	1272	1307	440	2416	1967	1526
Wind dir	56610	5218	4522	3629	3419	3638	3361	3826	3095	2851	3065	3554	3149	3264	1272	1409	440	2416	1968	1538
Pressure	21656	3988	3086	1698	1135	1456	991	1064	752	652	1055	675	822	844	736	282	117	721	898	637
Total cloud	55044	5095	4365	3638	3426	3624	3352	3828	3041	2776	2935	3437	2748	3220	1029	1216	412	2406	1990	1547
Cloud obs, ly1	49470	4771	4366	3226	2933	3142	3069	3224	2618	2316	2308	3112	2631	2904	1007	1215	343	2188	1841	1366
Cloud obs, ly2	22385	2281	2338	1483	1279	1498	1197	1435	1203	987	992	1360	939	1140	398	551	168	1104	908	666
Cloud obs, ly3	5347	544	617	429	326	441	246	370	324	181	187	343	151	219	76	124	32	254	220	159
Cloud obs, ly4	535	66	84	76	40	74	21	32	38	3	17	23	8	17	7	5	2	3	11	2
Wx elements	13631	1625	1519	1067	919	1101	1061	1233	768	458	786	545	247	355	464	146	93	471	381	284
Total obs, Feb	72	0	0	2	20	1	0	4	16	0	0	0	0	0	0	0	0	29	0	0
Total obs, Mar	2470	0	0	93	353	304	103	397	315	71	254	112	190	38	151	13	30	46	0	0
Total obs, Apr	3874	0	0	278	301	330	163	606	370	159	464	189	340	343	170	150	9	2	0	0
Total obs, May	4130	22	206	480	216	219	190	169	183	230	319	325	316	330	175	82	158	187	150	113
Total obs, Jun	11617	2020	1524	662	456	265	538	498	520	559	514	565	514	647	218	212	92	595	581	370
Total obs, Jul	23354	1707	1806	1151	1185	1275	1526	1496	1196	1506	1203	1648	1442	1508	367	777	99	1158	913	875
Total obs, Aug	11132	1525	1001	727	811	998	823	639	523	402	350	730	476	433	224	250	56	417	349	243
Total obs, Sep	886	1	0	254	98	272	52	40	27	0	7	12	42	5	8	8	0	32	28	0
Total obs.	57539	5275	4537	3647	3440	3664	3395	3849	3150	2927	3111	3585	3320	3304	1313	1492	444	2466	2021	1601
Total no. of sites	1107	33	39	50	62	65	68	85	71	92	62	113	108	100	20	26	7	42	22	26
Unique locations	553	153	139	166	110	121	118	119	110	106	97	101	163	148	185	66	79	105	108	103
Max days/site	185	40	80	84	34	38	36.5	32	37	24.5	34	19	15.5	24.5	28	42	27	48	70	45
75th qtile days/site	29	54	49	33	30	25	24	22	24	19	24	17	16	20	30	31	27	30	47	32
Mean days/site	20	43	44	16	20	17	11	12	14	11	15	11	9	13	18.5	28	20	13	47	22
Median days/site	10	7	13	7	9	9	8	7	8	7	7	6	6	8	10	20	13	9	21	11
Min days/site	1	1	5	1	5	1	2	2	1	2	3	1	1	1	1	8	7	6	5	6

Observational Data

1974 - 1983

Original data structure

Original data structure consisted of 10 files, one file per year, each possessing several thousand 80 - column lines. Each line represented one observation from one camp at one time interval. Observations of individual elements are not coded with decimal points and negative signs are coded as "1". The coding of a line is as follows:

Table AP- 1: Structure of original data sets, 1974 – 1983 period.

Columns	Element
1 - 4	PCSP station identifier
5 - 6	year
7 - 8	month
9 - 10	day
11 - 12	local time
13 - 14	universal time
15 - 18	visibility
19 - 20	first weather element
21 - 22	second weather element
23 - 24	third weather element
25 - 27	station pressure
28 - 30	dry bulb temperature
31 - 33	dew point temperature
34 - 35	wind direction
36 - 37	wind speed
38 - 40	altimeter setting
41 - 42	total cloud cover (in tenths)
43 - 44	cloud cover of cloud layer #1
45 - 46	cloud type of cloud layer #1
47 - 49	cloud height of cloud layer #1
50 - 51	cloud cover of cloud layer #2
52 - 53	cloud type of cloud layer #2
54 - 56	cloud height of cloud layer #2
57 - 58	cloud cover of cloud layer #3
59 - 60	cloud type of cloud layer #3
61 - 63	cloud height of cloud layer #3
64 - 65	cloud cover of cloud layer #4
66 - 67	cloud type of cloud layer #4
68 - 70	cloud height of cloud layer #4
71 - 73	maximum temperature (previous 12-hr period)
74 - 76	minimum temperature (previous 12-hr period)
77 - 79	precipitation
80	pressure tendency

An example of the data in this format:

```
003474080107 0300      295 39 3634050400504010000104004      057036
003474080110 1600      288 46 373603038010118250
003474080113 1600      274 51 4211020340201182500118280
003474080116 1600      257 53 4310050290403112000110250
003474080119 1600      240 49 420907024050109080020812001181800110200
003474080122 1600      220 46 41000001807010810005091200118180
003474080207 1600      166 43 4100000020908081000109120      053038
003474080210 1600      149 47 38200799707050808001091000118200
003474080213 1600      139 51 41161099408040808003091200118200
```

The files had random numbers of extraneous characters, specifically ASCII code 127, introduced at the beginning of each line, a result of the transfer process to PC from mainframe. Before processing by SAS the files needed the extraneous characters removed. This was done in Quattro for DOS using a search and replace.

The files were then read into SAS using BIG-READ.SAS. Another program, YEARREAD.SAS, was derived from BIG-READ.SAS as a macro-free, simplified variant for purposes of reading single years only.

Table AP- 2: SAS programs to deal with data from the 1974 – 1983 period.

Program name	Function
BIG-READ.SAS	Sequential text file reader. Will read in an entire series of identically structured data files provided they possess names that have a numerical increment of some sort, e.g., file1.bt, file2.bt, file3.bt. Basic program structure, with suitable modification to the INPUT statement, can be used for reading any series of text files that possess identical structure and sequentially numbered filenames.
YEARREAD.SAS	Reads single years in the 1974 – 1983 group. Inputs text file and outputs re-arranged text file and SAS dataset.

1984 – 1986

Original data structure

Data for this period were only available in an Atmospheric Environment Service archive format for hourly data, HLY01. This is a space efficient format that is not in an immediately useable form but which depends upon formatting programs to prepare it for use.

Table AP- 3: Structure of original data sets, 1984 – 1986 period.

Columns	Element
1 – 3	AES region identifier
4 – 7	AES station identifier
8 – 10	3 digit year value
11 – 12	month
13 – 14	day
15 – 17	observed element (see Table AP-4)
18 – 23	data point
24	flag
18 – 24	one complete datum, repeated 24 times

Table AP- 4: Some element identifiers for 1984 – 1986 period data.

Identifier	Climate element	Format
072	visibility	0.1 kilometer
093	ice crystals (weather element)	1 or 0 (observed yes or no)

073	sea level pressure	0.01 kilopascals
078	dry bulb temperature	0.1 degree Celsius
156	wind direction	10's of degrees
076	wind speed	Kilometers/hour
082	total cloud amount	Tenths
108	cloud layer 1 amount	Tenths
109	cloud layer 1 type	code value, 0-16
110	cloud layer 1 elevation	30's of meters

These data were coded without decimal points or negative signs. Formatting for each climatic element was found in an accompanying guide. The number 1 was used to represent the negative sign. For example, with reference to the example of data in the original format listed below, the values for the occurrence of element 078, dry-bulb temperature, on March 08, 1984, are both -39.0 degrees Celsius.

The files had random numbers of extraneous characters, specifically ASCII code 127, introduced at the beginning of each line, a result of the transferal process to PC from mainframe. Before processing by SAS the files needed the extraneous characters removed. This was done in Quattro for DOS using a search and replace.

An example of the data in the original AES HLY01 format:

```

24010189040308072      000113      000080
24010189040308093      000000      000000
24010189040308073      000120      000060
24010189040308078      001390      001390
24010189040308156      000021      000033
24010189040308076      000026      000012
24010189040308082      000002      000003
24010189040308100      000006      000003
24010189040308109      000202      000003
24010189040308110      000048      000101
24010189040309072      000000      000004
24010189040309100      000000      000000
24010189040309103      000000      000000
24010189040309073      000030      000070
24010189040309078      001420      001400
24010189040309156      000030      000027
24010189040309076      000029      000029
24010189040309082      000009      000010
24010189040309100      000001      000009
24010189040309109      000003      000016
24010189040309110      000101      000000
24010189040309112      000008      000001
24010189040309113      000005      000001
24010189040309114      000202      000001
24010189040310072      000113      000032
24010189040310103      000000      000000
24010189040310073      000150      000200
24010189040310078      001410      001300
24010189040310156      000024      000007
24010189040310076      000006      000041
24010189040310082      000002      000010
24010189040310100      000002      000010
24010189040310109      000001      000016
24010189040310110      000101      000000
24010189040311072      000113      000113
24010189040311073      000310      000200

```

1987 – 1993

Original data structure

Here is an example of the original data structure from 1987 – 1988:

STN	LAT	LONG	MONTH	YEAR	DAY	HOUR	HOUR	VIS	WK	PRESS.	DRY	BULB	DEW	PT.	WIND	WIND	ALT.	TOT.	CLOUD	LOW	LOW	2ND	2ND	3RD
					(LST)	(LST)	(GMT)		OBST.	(mb)	TEMP	TEMP	TEMP	(C)	DIR	SPEED	(IN)	AMOUNT	AMOUNT	HEIGHT	HEIGHT	HEIGHT	AMOUNT	HEIGHT
75	74	9047	75	87	31	7		10							27	7		10	10	4	30			
75	74	9047	75	87	31	19		10							30	7		10	3	6	30			
75	74	9047	75	87	1	7		25							30	7		2	1	6	10			
75	74	9047	75	87	1	9		15							27	15		7	2	3	5			
75	74	9047	75	87	1	19		25							27	25								
75	74	9047	75	87	2	19		25							32	20		7	4	8	50			
75	74	9047	75	87	3	7		0.5	1						36	7		10	10	1	0			
75	74	9047	75	87	3	9		Missing																
75	74	9047	75	87	3	19		0.5	1	11					2	25		9	5	3				
75	74	9047	75	87	4	7		0.12	1						16	7		10	10	1	0			
75	74	9047	75	87	4	19		2.5	1	22					4	15		10	10	3	1			
75	74	9047	75	87	5	7			1	22					7	10		10	5	1	0			
75	74	9047	75	87	5	19			22	11					6	15		10	10	1	0			
75	74	9047	75	87	6	7			1						16	11		10	10	1	0			
75	74	9047	75	87	6	19									9	25		9	3	25	3	1		
75	74	9047	75	87	7	7									9	35		10	10	1	0			
75	74	9047	75	87	7	19									9	20		9	7	1	0			
75	74	9047	75	87	8	7			1						9	5		9	7	1	0			
75	74	9047	75	87																				

The first step was a reading into Quattro Pro for DOS, formatting for the Unix system, outputting as an ASCII text file and concatenating with all other files for the same year.

Unix commands issued to clean the 1987 data file for input into SAS. These commands were issued after the 1987 station data were formatted and concatenated in Quattro. The same type of command sequence was also issued for the 1988 data, although the specific text strings targeted were different.

```
grep [A-Z,a-z] OUT87.PRN
#Search for any occurrences of non-numeric data (there should be none). If found, begin replacing any
occurrences with blank spaces.
sed 's/Missing/ /g' OUT87.PRN > OUT87.PRN.sed1
#Based on what the grep found begin issuing sed's to remove extraneous text, replacing it with blanks to
hold the place of other data in the file. In the case of various manifestations of truncated longer words
start with the longest occurrences of the word and work through progressively smaller occurrences. In
this case the word 'Missing' was coded throughout two of the 1987 camp records but because of the
spreadsheet formatting, it was truncated to various manifestations from the full word down to a single
letter 'M'.
sed 's/Missin/ /g' OUT87.PRN.sed1 > OUT87.PRN.sed2
sed 's/Missi/ /g' OUT87.PRN.sed2 > OUT87.PRN.sed3
sed 's/Miss/ /g' OUT87.PRN.sed3 > OUT87.PRN.sed4
sed 's/Mis/ /g' OUT87.PRN.sed4 > OUT87.PRN.sed5
sed 's/Mi/ /g' OUT87.PRN.sed5 > OUT87.PRN.sed6
sed 's/M/ /g' OUT87.PRN.sed6 > OUT87.PRN.sed7
sed 's/TOPO/ /g' OUT87.PRN.sed7 > OUT87.PRN.sed8
sed 's/header/ /g' OUT87.PRN.sed8 > OUT87.PRN.sed9
sed 's/POPOFOG/ /g' OUT87.PRN.sed9 > OUT87.PRN.sed10
grep [a-z,A-Z] OUT87.PRN.sed10
#Once the sed's have been completed re-run the grep, searching for any non-numeric characters that
may have been missed.
f1t
#Perform a directory listing in preparation to removal of the intermediate files created during the sed
process.
mv OUT87.PRN.sed10 OUT87.DONE
#Rename the final output file.
ls OUT87.PRN.s*
#Perform a directory listing using the same filters as will be used in the file remove, to make sure only the
desired files will be erased (safe Unix practice – once a file is removed in Unix it is gone).
rm OUT87.PRN.s*
#If the ls filter was okay proceed with the rm command to remove the files.
f1t
#Perform another directory listing to make sure no unwanted files are left. It is very easy to rapidly
accumulate unwanted files from intermediate steps in a process; it is important to clean as you go.
```

Station data

The original station information database is one that I input in 1994 from binders containing all original field camp information sheets. This file did not contain AES identifiers, so a second database, the identification cross reference station database, was input in December of 1997. Numerous programs in Unix and SAS were written in the process of getting these data into a useable form as a SAS dataset. In several instances a program was created, used and then modified when a better way of performing that task was discovered. All versions of programs that saw use as working programs are presented here in order that the full process involved is chronicled (*i.e.*, no experimental/learning programs are included).

Identification cross reference station database

The fields present include station name, AES station identifier, PCSP station identifier, and, when available, degrees of latitude, minutes of latitude, degrees of longitude, minutes of longitude. Observations in each field for a single record are separated by one space. Here is an example of the original data file:

```
CORY GLACIER 2401009 201
CRANE CITY 240J017 182
CRESWELL BAY 2401018 15
CROKER BAY 240J018 283
CROOKED LAKE 240J0AH 235.1
CROOKED LAKE FISHERIES 240J0JQ 235.2
CUNNINGHAM INLET 2401019 49
DEALY ISLAND 250J0JR 158
DECCA GREEN1 250A0K0 202.1
DECCA GREEN2 250J020 202.2
DECCA GREEN3 240J0B0 202.3
DECCA GREEN4 240J0K0 202.4
DECCA MASTER1 2501021 203.1
DECCA MASTER2 24010B1 203.2
```

Where the first identifier is the camp name, the second identifier is the AES identification and the third identifier is the PCSP number. Note the presence of spaces within the station name field and the fact that stations do not possess names with the same number of words (the count ranged from one to six word in the name - ran the script `titlecnt.awk` to determine the largest number of words in a station name).

Table AP- 5: Unix scripts to deal with the station database.

Program name	Function
titlecnt.awk	Counts the number of words in the station identifier. This was important to know how many times the underscore program should be run.
titlecnt2.awk	A second version that includes a means of displaying which files possess the longest station identifiers.
under_score.awk	Inserts an underscore character between words in the station identifier. This version will only insert one underscore per line per run, thus if the longest name is six words, the program will have to be run five times.
under_score2.awk	Improved version of under_score.awk. This version inserts all necessary underscores on the first run. Does not require information supplied by titlecnt.awk.
placer.awk	Modifies the structure of the station data file.

SAS will accept spaces within a character field if the data are arranged by column. If the data are not arranged by column list input format must be used. In SAS list format, however, fields are delineated by spaces. This would not be a problem if all stations had the same number of words in their names; each word could be read in as a separate variable which could then be concatenated. However there are varying numbers of words in each name, thus spaces appearing within the station name field must be removed so that the station name is one continuous entity (string) that can be read into one variable. Although this could be done in SAS it is much easier to prep the data using awk in a Unix script. It was decided to replace all spaces within a station name with

an underscore character and to convert the underscores back to spaces within SAS. The awk script to insert the underscore character is called `under_score.awk`.

Later modified `under_score.awk` to perform as many substitutions as necessary per line; new program is called `under_score2.awk`.

This data file can now be read into a SAS dataset. The SAS program that reads in these station data and removes the underscores from the name field is called `readstns.sas`.

Mapping station locations

Initial attempts to work with and map the station location data were performed in SAS using PROC GMAP. It was felt that using SAS in-house mapping packages would be a logical extension to work already being conducted in SAS. The SAS program `coasts.sas` is the result. This line was abandoned and all mapping was consolidated in IDRISI GIS, with SAS programs written to output datasets in IDRISI format.

Station Extra

```
# byecomm.awk <filename> used to strip off comments that appeared after
# data fields. Script depends upon the fact that
# data fields contain numeric (and a leading -)
# and comments start with a non-numeric or -.
#
# David Atkinson, Jan 9, 1998
# Examples of Unix "awk" language - no, awk doesn't stand for "awkward", it stands for the
# last names of its developers: Aho, Weinegar(?) and Kernigan. - I would suggest learning
# PERL nowadays, instead of awk, if you are going to invest time in it.
```

```
awk '{
    for (i=1;i<=NF;++i) {
        if ( i > 3 ) {
            if ( $i !~ /^[0-9-]/ ) break
        }
        printf $i" "
    }
    printf "\n"
}' $1
```

```
# lastones.awk "manually" add underscore to three stn names that had numbers in them
```

```
awk '{
    if ($2 ~ /^XMH/) {
        printf $1" "$2" "$3
        for (i=4;i<NF;i++) printf " "$i
        printf "\n"
    }
    else print $0
}' $1
```

stations.doc write up in progress, place here

Work on Stations files

The original station database is one that I input in 1994 from the binders containing all original field camp information sheets. This file does not contain AES identifiers, so a second database was input in December of 1997.

First station information database

The original datafile contains these fields: PCSP identifier, station name, latitude degrees, latitude minutes, latitude seconds, longitude degrees, longitude minutes, longitude seconds, year, number of observations, decimal latitude, decimal longitude, julian start day and julian end day. The observations within a record are separated by one or more spaces. If there is more than one word in the station name it is separated by a single space. Not all fields possess observations for all records. Some observations possess a comments field.

This dataset will serve as a backup to the second station database by providing positional information and any missing stations. The following is an example of the unprocessed data set:

144	Dyer Bay	76	8	121	49	?	76.133	-121.817		
174	Dyke Ackland	75	1	75	45	1979	75.017	-75.750		
53	Eastwind Lake	80	6	85	39	1977	80.100	-85.650		
999	Eden Point	?		?		1976				
999	Eglington Island	75	53		118	8	1974	75.883 -		
118.133										
54	Eides Fiord	77	25	86	26	1974	77.417	-86.433		
173.1	Eidsbotn Fiord	76	9	91	32	1979	76.150	-91.533		
173.2	Eidsbotn Fiord	76	9	36	91	30	50	1980	76.150	-91.500
315	Ella Bay	81	6	69	0	1984	81.100	-69.000		
393	Emma Fiord	?		?						
177	Erebus Bay	74	43	91	50	1979	74.717	-91.833		
379	Esayoo Bay	80	50	81	52	1985	80.833	-81.867		
367.1	Eureka Sound	79	37	84	50	1985	10	79.617	-84.833	
367.2	Eureka (GSC)	80	0	85	56	1985	120	80.000	-85.933 -	
55	Expedition Fiord	79	25	90	35	Multi yr	500+	79.417		
-90.583										

The datafile came from MS-DOS operating system. Before further processing it must be converted to a Unix file structure by using the tr command to translate DOS carriage return/line feeds into Unix line feeds. Executing the command

```
tr '\015' '\012' < DOS.file > Unix.file
```

will perform the conversion and

```
tr -s '\012' < Unix.file
```

will strip any occurrences of extra line feed characters.

Place underscores in the station names. This process was the same as described for the second station information database. Again, the program was run until there were no more changes performed. The script `under_score.awk` also removed extra spaces.

Unmatched station names were then dealt with. `under_score.awk` was modified accordingly and run. In all three cases unmatched stations occurred when one of the words in the station name was a number, such as Camp 5 Creek.

The next step was to clean up the file. This consisted of removing extraneous spaces, question marks, periods, the comments field and moving the name fields such that they all began at a certain column.

The first step to repositioning the second field was to determine the widest PCSP station number observation.

The awk command line

```
awk '{print length($1)}' < filename | sort | uniq -c
```

printed out a sorted list showing number of occurrences of each possible length value.

output:

```
2 1
73 2
279 3
60 4
108 5
26 6
```

In other words, two stations have PCSP identifiers that consist of a one-digit number, while 73 possess two digit values.

The greatest length was 6. It was decided to place the beginning of the next field at least two blank spaces beyond the widest observation in the first field, that is, at column 9. As a double check that the command was picking up on valid data, various observations can be analyzed at random using the following awk command:

```
awk '{if (length($1) == 1) print $0}' < filename
```

to examine the lines possessing the shortest field \$1 (i.e. one digit):

output:

```
5 Abbott River 75 17 95 55 75.283 -95.917 Unissued
1 Brock Island 77 51 114 8 77.850 -114.133
```

and to examine the lines possessing 6 digits (the largest width):

```
awk '{if (length($1) == 6) print $0}' < filename
```

output:

```
203.41 Decca Master 80 48 95 24 80.800 -95.400
203.42 Decca Master 69 33 90 18 69.550 -90.300
204.41 Decca Red 68 42.8 85 38 68.713 -85.633
204.42 Decca Red 81 42 91 57 81.700 -91.950
118.41 Fort Conger 81 45 64 48 1981 30 81.750 -64.800
118.42 Fort Conger 81 35 64 55 1981 24 81.583 -64.917
118.51 Fort Conger 81 45 64 53 1982 16 81.750 -64.883
118.52 Fort Conger 81 45 75 45 1982 6 81.750 -75.750
127.31 Freemans Cove 75 9 98 6 1979 14 75.150 -98.100
127.32 Freemans Cove 75 14 98 5 1979 10 75.233 -98.083
127.33 Freemans Cove 75 14 98 5 1979 4 75.233 -98.083
252.11 Gale Point 78 14 75 26 1982 40 78.233 -75.433
252.12 Gale Point II 77 26 72 15 1982 8 77.433 -72.250
319.21 Hare Fiord 80 40 86 10 1985 30 80.667 -86.167
319.22 Hare Fiord 80 42 86 5 1985 18 80.700 -86.083
323.11 Idlout Point 72 42 55 93 56 28 1984 38 72.700 -93.933
323.12 Idlout Point 72 43 93 50 1984 36 72.717 -93.833
62.101 King Christian Island 77 45 101 5 1981 80 77.750 -101.083
62.102 King Christian Island 77 45 101 2 1982 88 77.750 -101.033
381.11 Lightfoot River 80 31 30 92 15 1985 10 80.517 -92.250
381.12 Lightfoot River 80 40 92 22 1985 12 80.667 -92.367
180.11 Loughheed Island 77 9 104 40 1979 24 77.150 -104.667
180.12 Loughheed Island 77 30 105 38 1979 14 77.500 -105.633
214.21 Muskox Fiord 76 37 87 30 1982 8 76.617 -87.500
214.22 Muskox Fiord 76 41 30 87 41 1982 10 76.683 -87.683
```

The scripts look like they have worked properly. The script to reposition the second field can now be run. It is an awk script called placer.awk:

```
### placer.awk <filename> positions the second field at column 9
```

```
awk '{
```

```

printf $1
for (i=1;i<(8-length($1));++i) printf " "
for (i=2; i<=NF; ++i) printf " "$i
printf "\n"
}
' $1

```

output:

Note that two spaces have now been inserted between fields.

Second Station Information Database

The fields present include station name, AES station identifier, PCSP station identifier, and, when available, degrees of latitude, minutes of latitude, degrees of longitude, minutes of longitude. Observations in each field for a single record are separated by one space. Here is an example of the original data file:

```

CORY GLACIER 2401009 201
CRANE CITY 240J017 182
CRESWELL BAY 2401018 15
CROKER BAY 240J018 283 74 42 83 41
CROOKED LAKE 240J0AH 235.1 72 34 98 34
CROOKED LAKE FISHERIES 240J0JQ 235.2 72 41 98 52
CUNNINGHAM INLET 2401019 49
DEALY ISLAND 250J0JR 158
DECCA GREEN1 250AOK0 202.1 73 22 104 36
DECCA GREEN2 250J020 202.2 71 29 104 21
DECCA GREEN3 240J0B0 202.3 81 49 90 38
DECCA GREEN4 240J0K0 202.4 71 21 93 03
DECCA MASTER1 2501021 203.1 71 29 104 21
DECCA MASTER2 24010B1 203.2 70 01 101 11

```

Note not all stations possess positional information.

Note as well the presence of spaces within the station name field and the fact that stations do not possess names with the same number of words (the count ranged from one to six word in the name - ran the script titlecnt.awk to determine the largest number of words in a station name).

titlecnt.awk:

```

## titlecnt.awk
## count how many words are in the station identifier

awk '{
    count=0
    for (i=1;i<NF;i++) { if ($i != /^[0-9]/) count += 1 }
    print count
}' $1 | sort -u

```

output:

```

1
2
3
4
6

```

The addition of the line `if (count == 6) print $0` will print out which stations possess the longest names:

```

## titlecnt.awk
## count how many words are in the station identifier
## and print out those stations possessing 6 words in their name

awk '{
    count=0

```

```

for (i=1;i<NF;i++) { if ($i != /^[0-9]/) count += 1}
if (count == 6) print $0
print count

```

```
} ' $1 | sort -u
```

output:

```

1
2
3
4
6
MER DE GLACE (AGASSIZ ICE CAP) 240K6HR 116

```

SAS will accept spaces within a character field if the data are arranged by column. If the data are not arranged by column list input format must be used. In SAS list format, however, fields are delineated by spaces, thus spaces appearing within the station name field must be removed. Although this could be done in SAS it is much easier to prep the data using awk. It was decided to replace all spaces within a station name with an underscore character and to convert the underscores back to spaces within SAS. The awk script to insert the underscore character is:

```

### under_score.awk <filename> inserts an underscore character
### between the first two non-numeric fields,
### the assumption being that they are identifiers.

```

```

awk 'BEGIN {counter=0}
{
  if ($2 - /^[0-9]/) {
    printf $1
    for (i=2; i<=NF; ++i) printf " $i
    printf "\n"
  }

  if ($2 != /^[0-9]/) {
    printf $1" "$2
    for (i=3; i <= NF; ++i) printf " $i
    printf "\n"
    counter += 1
  }
}
END {# print a count of the number of conversions, so we know when to stop!
  printf counter" changes made\n" > "/dev/tty" # this directs o/p to screen
} ' $1

```

output:

```

LOWTHER_ICE_(II) 240268E 66.1 74 33 97 10
LOWTHER_ISLAND 240268N 66.2 74 33 97 29
MACKAR_INLET 2402F8E 27
MACKINSON_INLET 2402F8F 120
MALLOCK_DOME 240BFQF 2
MARSHALL_PENINSULA 240KF86 133
MARVIN_PENINSULA 240KFQF 369
MAXWELL_BAY 240B68Q 375
MCCLINNTOCK_INLET 250KFH8 243
MCKINLEY_BAY 240KFQQ 358
MEIGHEN_ISLAND/ICE_CAP 2402F8R 67
MELVILLE_ICE_CAP 250K689 68
MER_DE_GLACE_(AGASSIZ_ICE_CAP) 240K6HR 116
MERCY_BAY 250K6H9 279
MIDDLE_ISLAND 2502693 324
MINTO_INLET 250K698 297
MOCKLIN_POINT 2502699 149
MOKKA_FIORD 240K699 69
MOUNT_BEAUFORT 2402702 143
MOUNT_BOMPASS 2502704 264
MUSKOX_BAY 2402712 233

```

The script prints a count of how many conversions it performed. The count is used as a guide to know when to stop. When the script performs 0 conversions, it has done them all.

This data file can now be read into a SAS dataset. The SAS program that reads in these station data and removes the underscores from the name field is:

```
/* readstns.sas reads in AES station list ascii file, strips off
underscores and stores it in a SAS datafile.

David Atkinson
January 6, 1998.
*/

/* specify LIBNAME and add libref to DATA name to make permanent */

DATA stations;
  infile "/u/atkinson/sas/database/stations/stns.prn"
         MISSOVER;
  /* missover because not all stns have position data */

  /* the colon for variable name forces a read until the first blank */
  input name :$
        AESID $7.
        ID
        LatD
        LatM
        LongD
        LongM
        ;

  format ID 4. name $35. ;

  /* The underscores in the name field are necessary for
  convenient SAS input variable coding, but they are
  ugly for any sort of output. The following loop
  replaces any occurrence of an underscore with a blank.
  */

  DO i = 1 to length(name);
    if (substr(name,i,1)="_") then substr(name,i,1)=" ";
  END;

  /* ...and don't bother ouputting the counter. */

  DROP I;
  RETURN;
  RUN;
```

This puts the station database into a SAS dataset specified in the DATA statement.

SAS Programs

SAS Program 1: BIG-READ.SAS

```
/*.....*/
/* BIG-READ.sas is a Sequential Text File Reader. */
/* The initial screen prompts for the years to read in. */
/* */
/* Changes to the INPUT and PUT statements, as well as */
/* to the DATA statement must be made manually. */
/* */
/* Program both arranges date properly and then reads in */
/* all text files into SAS dataset. Program also performs */
/* some QC processing by rejecting missing records and */
/* extreme temperature values. It also substitutes */
/* negative values for temperature values above 100. */
/* */
/* */
/* D.E. Atkinson */
/* July 22, 1997. */
/*.....*/

LIBNAME PCSP '/u/atkinson/sas/database/74-83/output';

OPTIONS ERRORS = 10;

%let start=73;
%let finish=73;

%window reader color=gray columns=65 rows=30 icolumn=25 irow=20
%2 %2 "Good Morning, Human Master"
%3 %5 "%i humbly submit a "
%4 %5 "Sequential Text File Reader"
%5 %5 "for Your blessed use."
%7 %5 "This slave was accessed on %sysday, %sysdate.."
%9 %5 "Enter the start and finish years for your files, or"
%10 %5 "accept the default values listed."
%13 %5 "Current start year is 19"
%29 start 2 attr=underline
%14 %5 "and the current end year is 19"
%35 finish 2 attr=underline
%16 %5 "Enter a new value if needed."
%17 %5 "ENTER advances to the next field."

%25 %5 "Press ENTER to continue.:"
%display reader;

%let yr = %start;

/** BEGIN DEFINE MACROS; START WITH LAST CALLED DEFINED FIRST **/

%macro DSBUILD; /* 4th Combine each year's resultant dataset into one large DS*/

DATA PCSP.MAIN;
%IF %yr = %start %THEN
%DO;
SET PCSP.PCSP%yr;
%END; /* create the new dataset because it won't exist on the first run */

SET PCSP.MAIN PCSP.PCSP%yr;
RETURN;

%mend DSBUILD;

%macro READIN; /* 3rd */
DATA PCSP.PCSP%yr; /* NOTE DOUBLE quotes must be used */
/* for external file refs with macro vars */

INFILE '/u/atkinson/sas/database/74-83/data/B19%yr..txt' MISSOVER;

INPUT @1 ID 4.
@5 DATE DATETIME13.
@18 VIS 4.
@22 WK1 2.
@24 WK2 2.
@26 WK3 2.
@28 SLP 3.
@31 DBTEMP 3.
@34 DEWTEMP 3.
@37 WDIR 2.
@39 WSPD 2.
@41 ALTSET 3.
@44 TOTCLD 2.
@46 CLDAMT1 2.
@48 CLDTYP1 2.
@50 CLDHGT1 3.
@53 CLDAMT2 2.
@55 CLDTYP2 2.
@56 CLDHGT2 3.
```

```

060 CLDAMT3 2.
062 CLDTYP3 2.
064 CLDHGT3 3.
067 CLDAMT4 2.
069 CLDTYP4 2.
071 CLDHGT4 3.
074 MAXTEMP 3.
077 MINTEMP 3.
080 PRECIP 3.
083 PTEENCY
:

IF DBTEMP > 100 THEN
  IF DBTEMP > 200 THEN DBTEMP = .;
  ELSE DBTEMP = -1*(DBTEMP-100);

IF DEWTEMP > 100 THEN
  IF DEWTEMP > 200 THEN DEWTEMP = .;
  ELSE DEWTEMP = -1*(DEWTEMP-100);

/* reject extreme temperature values */

IF VIS=. AND SLP=. AND DBTEMP=. AND DEWTEMP=. AND WDIR=.
AND WSPD=. AND ALTSET=. AND TOTCLD=. AND CLDAMT1=. AND CLDTYP1=.
AND CLDHGT1=. AND CLDAMT2=. AND CLDTYP2=. AND CLDHGT2=. AND CLDAMT3=.
AND CLDTYP3=. AND CLDHGT3=. AND CLDAMT4=. AND CLDTYP4=. AND CLDHGT4=.
AND MAXTEMP=. AND MINTEMP=. AND PRECIP=. AND PTEENCY=. THEN RETURN;
/* This checks that there is an observational record accompanying the station
identification and date header. If not then throw it out.*/

/* These folling two lines also create a text file that is
identical to the SAS datafile */

/* file '/u/atkinson/sas/text.op.test'; */
/* put _infile_; */

RETURN;
%mend READIN;

%macro DSTPEP; /* 2nd Rearrange the data, throw out missing dates, correct any missing
OCT times and rewrite in a form that SAS can handle. */
PROC FORMAT;
  VALUE MONTH (DEFAULT=3)
    1 = 'JAN'
    2 = 'FEB'
    3 = 'MAR'
    4 = 'APR'
    5 = 'MAY'
    6 = 'JUN'
    7 = 'JUL'
    8 = 'AUG'
    9 = 'SEP'
    10 = 'OCT'
    11 = 'NOV'
    12 = 'DEC'
    OTHER = ''
;
RUN;

DATA _NULL_;
  INFILE '/u/atkinson/sas/database/74-83/data/196yr..txt*' MISSOVER SHAREBUFFERS;

  INPUT 01 ID 4.
        05 MONTH 2.
        07 YEAR 2.
        09 DAY 2.
        011 LCLTIME 2.
        013 UCLTIME 2.
        015 VIS 4.
        019 WK1 2.
        021 WK2 2.
        023 WK3 2.
        025 STWPRESS 3.
        028 DBTEMP 3.
        031 DEWTEMP 3.
        034 WDIR 2.
        036 WSPD 2.
        038 ALTSET 3.
        041 TOTCLD 2.
        043 CLDAMT1 2.
        045 CLDTYP1 2.
        047 CLDHGT1 3.
        050 CLDAMT2 2.
        052 CLDTYP2 2.
        054 CLDHGT2 3.
        057 CLDAMT3 2.
        059 CLDTYP3 2.
        061 CLDHGT3 3.
        064 CLDAMT4 2.
        066 CLDTYP4 2.
        068 CLDHGT4 3.
        071 MAXTEMP 3.
        074 MINTEMP 3.
        077 PRECIP 3.

```

```

        @80 PTERDCY
        ;

IF ID =. AND MONTH =. AND YEAR =. THEN RETURN;
/* ditch any blank lines */

IF LCLTIME NE . AND UCTTIME = . THEN UCTTIME=LCLTIME+5;
IF UCTTIME = 24 THEN UCTTIME = 00;

IF UCTTIME > 24 THEN DO;
    UCTTIME = UCTTIME-24;
    DAY=DAY+1;
END;

DROP LCLTIME;

FILE '/u/atkinson/sas/database/74-83/data/B19&yr..txt';

PUT
  01 ID
  05 DAY          Z2. -R
  07 MONTH        MON3. /* format defined above */

  010 YEAR
  012 ':'
  013 UCTTIME     2.
  015 ':00'
  018 VIS         4. -R
  022 W01         2.
  024 W02         2.
  026 W03         2.
  029 STMPRESS   3.
  032 DSTEMP     3.
  035 DENTEMP     3.
  037 WDIR        2.
  039 WSPD        2.
  042 ALTSET      3.
  044 TOTCLD      2.
  046 CLDAMT1    2.
  048 CLDTYP1    2.
  050 CLDHGT1    3.
  053 CLDAMT2    2.
  055 CLDTYP2    2.
  057 CLDHGT2    3.
  060 CLDAMT3    2.
  062 CLDTYP3    2.
  064 CLDHGT3    3.
  067 CLDAMT4    2.
  069 CLDTYP4    2.
  071 CLDHGT4    3.
  074 MAXTEMP    3.
  077 MINTEMP    3.
  080 PRECIP     3.
  083 PTERDCY    3.
;

RETURN;
tend DSTEPER;

macro mainmac;
  %do %while (%yr < %eval(%finish+1));
    %DSTEPER
  /* %READIN
    %DSBUILD
  /* %LET yr = %eval(%yr + 1); /* increment the year value for the filenames */
  %end;
tend mainmac;

mainmac

```

SAS Program 2: YEARREAD.SAS

```

/*****
/* YEARREAD.sas reads single PCSP files for the years
/* 1974 to 1983. Years must be changed in the program.
/*
/*
/* Program both arranges date properly and then reads in
/* all text files into SAS dataset. Program also performs
/* some QC processing by rejecting missing records and
/* extreme temperature values. It also substitutes
/* negative values for temperature values above 100.
/*
/*
/* D.E. Atkinson
/* November 4, 1997.
*****/

LIBNAME PCSP '/u/atkinson/sas/database/74-83/output';

```

OPTIONS ERRORS = 10;

/* 2nd Rearrange the data, throw out missing dates, correct any missing
UCT times and rewrite in a form that SAS can handle. */

PROC FORMAT;

VALUE MON (DEFAULT=3)

1 = 'JAN'
2 = 'FEB'
3 = 'MAR'
4 = 'APR'
5 = 'MAY'
6 = 'JUN'
7 = 'JUL'
8 = 'AUG'
9 = 'SEP'
10 = 'OCT'
11 = 'NOV'
12 = 'DEC'
OTHER = ''

;

RUN;

DATA _NULL_;

INFILE "/u/atkinson/sas/database/74-83/data/1973.txt" MISSEVER;

INPUT @1 ID 4.
@5 year 2.
@7 month 2.
@9 DAY 2.
@11 LCLTIME 2.
@13 UCTTIME 2.
@15 VIS 4.
@19 WX1 2.
@21 WX2 2.
@23 WX3 2.
@25 STMPRESS 3.
@28 DBTEMP 3.
@31 DBWTEMP 3.
@34 WDIR 2.
@36 WSPD 2.
@38 ALTSET 3.
@41 TOTCLD 2.
@43 CLDAMT1 2.
@45 CLDTYP1 2.
@47 CLONGT1 3.
@50 CLDAMT2 2.
@52 CLDTYP2 2.
@54 CLONGT2 3.
@57 CLDAMT3 2.
@59 CLDTYP3 2.
@61 CLONGT3 3.
@64 CLDAMT4 2.
@66 CLDTYP4 2.
@68 CLONGT4 3.
@71 MAXTEMP 3.
@74 MINTEMP 3.
@77 PRSCIP 3.
@80 PTREDCY

;

/* The last element, pressure tendency, does not have a fixed width specified. This format causes SAS to look at column 90 and grab whatever is there, no matter its length. I found that if a fixed width was specified for this last element, that SAS would screw up the read because it was grabbing the end of line characters. This may have been something to do with SAS Unix having trouble with the text files coming from the PC environment, where an end of line has a CR and LF; Unix just has an LF.

*/

IF ID = . AND MONTH = . AND YEAR = . THEN RETURN;
/* ditch any blank lines */

IF LCLTIME NE . AND UCTTIME = . THEN UCTTIME=LCLTIME+5;
IF UCTTIME = 24 THEN UCTTIME = 00;

IF UCTTIME > 24 THEN DO;
UCTTIME = UCTTIME-24;
DAY=DAY+1;
END;

DROP LCLTIME;

FILE "/u/atkinson/sas/database/74-83/data/1973B.txt";

PUT @1 ID
@5 DAY 22. -R
@7 MONTH MON3.
@10 YEAR
@12 " : "
@13 UCTTIME 22. -R
@15 " : 00 "
@18 VIS 24. -R
@22 WX1 2.
@24 WX2 2.
@26 WX3 2.
@29 STMPRESS 3.
@32 DBTEMP 3.

```

035 DEWTEMP 3.
038 WDIR 2.
040 WSPD 2.
043 ALTSET 3.
045 TOTCLD 2.
047 CLDAMT1 2.
049 CLDTYP1 2.
051 CLDHGT1 3.
054 CLDAMT2 2.
056 CLDTYP2 2.
058 CLDHGT2 3.
061 CLDAMT3 2.
063 CLDTYP3 2.
065 CLDHGT3 3.
068 CLDAMT4 2.
070 CLDTYP4 2.
072 CLDHGT4 3.
075 MAXTEMP 3.
078 MINTEMP 3.
081 PRECIP 3.
084 PTENCY 3.
;

```

```
RETURN;
```

```
DATA PCSP.PCSP73;
```

```
INFILE '/u/atkinson/sas/database/74-83/data/1973B.txt' MISCOVER;
```

```

INPUT 01 ID 4.
05 DATE DATETIME13.
018 VIS 4.
022 WK1 2.
024 WK2 2.
026 WK3 2.
028 STMPRESS 3.
032 DBTEMP 3.
035 DEWTEMP 3.
038 WDIR 2.
040 WSPD 2.
042 ALTSET 3.
045 TOTCLD 2.
047 CLDAMT1 2.
049 CLDTYP1 2.
051 CLDHGT1 3.
054 CLDAMT2 2.
056 CLDTYP2 2.
058 CLDHGT2 3.
061 CLDAMT3 2.
063 CLDTYP3 2.
065 CLDHGT3 3.
068 CLDAMT4 2.
070 CLDTYP4 2.
072 CLDHGT4 3.
075 MAXTEMP 3.
078 MINTEMP 3.
081 PRECIP 3.
084 PTENCY 3.
;

```

```

IF DBTEMP > 100 THEN
  IF DBTEMP < 200 THEN DBTEMP = -1*(DBTEMP-100);

```

```

IF DEWTEMP > 100 THEN
  IF DEWTEMP < 200 THEN DEWTEMP = -1*(DEWTEMP-100);

```

```
/* convert temperature values from 100 to negative */
```

```

IF VIS=. AND STMPRESS =. AND DBTEMP=. AND DEWTEMP=. AND WDIR=. AND WSPD=. AND ALTSET=. AND TOTCLD=. AND CLDAMT1=. AND
CLDTYP1=. AND CLDHGT1=. AND CLDAMT2=. AND CLDTYP2=. AND CLDHGT2=. AND CLDAMT3=. AND CLDTYP3=. AND CLDHGT3=. AND
CLDAMT4=. AND CLDTYP4=. AND CLDHGT4=. AND MAXTEMP=. AND MINTEMP=. AND PRECIP=. AND PTENCY=. THEN RETURN;

```

```
/* This checks that there is an observational record accompanying the station identification and date header. */
```

```

/* These folling two lines also create a text file that is
identical to the SAS datafile */

```

```

/* file '/u/atkinson/sas/text.op.test': */
/* put _infile_; */

```

```
RETURN;RUN;QUIT;
```

SAS Program 3: READSTNS.SAS

```

/* readstns.sas reads in AES station list ascii file, strips off
underscores and stores it in a SAS datafile.

```

David Atkinson

```

January 6, 1998.
*/

/* specify LIBNAME and add libref to DATA name to make permanent */

DATA stations;
  infile "/u/atkinson/sas/database/stations/stns.prn"
    MISSOVER;
/* missover because not all stns have position data */

/* the colon for variable name forces a read until the first blank.
   This conveniently deals with variable length data. */
input name :$
      AESID $7.
      ID
      LatD
      LatM
      LongD
      LongM
      ;

format ID 4. name $35. ;

/* The underscores in the name field are necessary for
   convenient SAS input variable coding, but they are
   ugly for any sort of output. The following loop
   replaces any occurrence of an underscore with a blank.
*/

DO i = 1 to length(name);
  if (substr(name,i,1)="_") then substr(name,i,1)=" ";
END;

/* ...and don't bother ouputting the counter. */

DROP I;
RETURN;
RUN; QUIT;

```

This puts the station database into a SAS dataset specified in the DATA statement.

SAS Program 4: coasts.sas

```

/****
coasts.sas reads in coastlines.vec file and
writes a sas mapfile dataset for GWAP. The
original file was created by IDRISI and its
source is the GSC Bedrock Geology CDROM.
SAS is sensitive to carriage return characters:
anytime a file comes from DOS run the command line
* tr -d '\015' < doefile > new.unix.file * to
correct so SAS can read.

DE Atkinson May 12, 1998.
****/

data coasts;
  infile "-/sas/database/stations/coastlines.vec";
  input first @;
  retain segment id;
  keep id segment x y;
  if first > 0 then do;
    segment=first;
    id=segment;
    return;
  end;
  x=first;
  input @26 y;
  output;
  return;
run;

quit;

```

```

/*****/
/*
/* READ84.sas to read AES HLY01 format data.
/* For the year 1984 in particular.
/*
/* This reads in region, station number, year,
/* month, element, and data values and flags
/* for each hour. This program organizes
/* the data into 26 columns, one for each
/* element, using one row for each hour.
/*
/* Identical text file and SAS DS generated.
/*
/* DEC 15, 1997
/* D.E. Atkinson.
/*
/*****/
/* Program operates in several steps:
/*
/* 1) First DATA step initiates a DO loop
/* that reads across each line of input
/* and writes out to a text file one line
/* of output for every observation spot
/* of every element, whether or not a datum
/* is present. Extra information is written
/* to construct a date/time field that SAS
/* can handle. Note the use of a user
/* defined format, MON.
/* The IF statements dealing with MONTH and
/* DAY are to drop inappropriate the 31st of
/* months that do not have them and to drop
/* Feb 29, 30 and 31 for normal years and
/* Feb 30 and 31 for leap years. The
/* original HLY01 data has 31 value spots
/* for all months; such missing days are
/* problematic in time series analyses and
/* the non-existent dates confuse SAS.
/*
/* Trace precip values are assigned 8888;
/* actual missing values are assigned 9999.
/*
/* 2) A temporary SAS dataset is created in the
/* image of the text file of the first step.
/* The main reason for doing this is to
/* give SAS a proper date field. This DS
/* is then sorted by Station, Date/Time
/* and element using PROC SORT.
/*
/* 3) The sorted SAS ds is written to a text
/* file.
/*
/*****/
LIBNAME PCSP '/u/atkinson/sas/database/84-86/data';

PROC FORMAT;
  VALUE MON (DEFAULT=3)
    1 = 'JAN'
    2 = 'FEB'
    3 = 'MAR'
    4 = 'APR'
    5 = 'MAY'
    6 = 'JUN'
    7 = 'JUL'
    8 = 'AUG'
    9 = 'SEP'
    10 = 'OCT'
    11 = 'NOV'
    12 = 'DEC'
    OTHER = ' '
  ;

  VALUE RAIN (DEFAULT=2)
    1='21'
    2='22'
    3='23'
    OTHER=' '
  ;

  VALUE RAINSNOW (DEFAULT=2)
    1='61'
    2='62'
    3='63'
    OTHER=' '
  ;

  VALUE DRIZZLE (DEFAULT=2)
    1='31'
    2='32'
    3='33'
    OTHER=' '
  ;

  VALUE ZRAIN (DEFAULT=2)
    1='25'
    2='26'
    3='27'
    OTHER=' '
  ;

  VALUE ZDRIZZLE (DEFAULT=2)
    1='35'
    2='36'
    3='37'
    OTHER=' '
  ;

  VALUE SNOW (DEFAULT=2)
    1='11'
    2='12'
    3='13'
    OTHER=' '
  ;

  VALUE SNOWGRAN (DEFAULT=2)
    1='15'
    2='16'
    3='17'
    OTHER=' '
  ;

  VALUE ICECRYS (DEFAULT=2)
    1='05'
    OTHER=' '
  ;

  VALUE ICEPELT (DEFAULT=2)
    1='51'
    2='52'
    3='53'
    OTHER=' '
  ;

  VALUE WICEPELT (DEFAULT=2)
    1='55'
    2='56'
    3='57'
    OTHER=' '
  ;

  VALUE SNOWSNOW (DEFAULT=2)
    1='45'
    2='46'
    3='47'
    OTHER=' '
  ;

  VALUE SNOWPELT (DEFAULT=2)
    1='41'
    2='42'
    3='43'
    OTHER=' '
  ;

  VALUE HAIL (DEFAULT=2)
    1='65'
    2='66'
    3='67'
    OTHER=' '
  ;

  VALUE FOG (DEFAULT=2)
    1='01'
    OTHER=' '
  ;

  VALUE ICEFOG (DEFAULT=2)
    1='02'
    OTHER=' '
  ;

  VALUE SNOKE (DEFAULT=2)
    1='06'
    OTHER=' '
  ;

  VALUE HAZE (DEFAULT=2)
    1='07'
    OTHER=' '
  ;

  VALUE BLOWSNOW (DEFAULT=2)
    1='03'
    OTHER=' '
  ;

```

```

VALUE BLOMSAND (DEFAULT=2)
  1='68'
  OTHER=' '
;

VALUE BLOWDUST (DEFAULT=2)
  1='04'
  OTHER=' '
;

VALUE DUST (DEFAULT=2)
  1='08'
  OTHER=' '
;

RUN;
DATA _NULL_;
  INFILE '/u/atkinson/sas/database/84-86/data/pcspy84.prn'
         MISSEVER SHAREBUFFERS FIRSTOBS=9964;
  ATTRIB REGION FORMAT=3. ;
  ATTRIB STN LENGTH=54 ;
  ATTRIB YEAR FORMAT=3. ;
  ATTRIB MONTH FORMAT=2. ;
  ATTRIB ELEMENT FORMAT=3. ;
  ATTRIB VALUE FORMAT=6. ;
  ATTRIB FLAG LENGTH=51 ;
  ATTRIB HOURNUM FORMAT=2. ;

  RETAIN REGION STN YEAR MONTH ELEMENT;

INPUT @1 REGION 3.
      @4 STN $ 4.
      @9 YEAR 2.
      @11 MONTH 2.
      @13 DAY 2.
      @15 ELEMENT 3. @
;

DO HOURNUM=1 TO 24;
  INPUT @((HOURNUM-1)*7+18) VALUE 6.
        @((HOURNUM-1)*7+18+6) FLAG $1. @ ;

  IF FLAG='T' THEN VALUE=8888;
  IF FLAG='M' THEN VALUE=.; /* ditch missing data */

  IF (MONTH=09 OR MONTH=04 OR MONTH=06 OR MONTH=11)
     AND (DAY=31) THEN RETURN;

  IF (MONTH=02 AND DAY GE 29 AND
      (YEAR NOTIN (92,88,84,80,76,72,68,64,60,56,52,48)))
     THEN RETURN;
  ELSE IF (MONTH=02 AND DAY=30) THEN RETURN;

  FILE '/usr/local2/aes/pcspstest.txt';

  PUT @1 REGION
      @5 STN $
      @13 DAY 22. -R
      @15 MONTH MON3. /* format defined above */
      @18 YEAR
      @20 ':'
      @21 HOURNUM
      @23 ':00'
      @28 ELEMENT 3. -R
      @32 VALUE 6. -R
      FLAG $ 45
;
END;
RUN;

/* New library assigned because program output was
exceeding the limit of /usr/tmp (/var/tmp), set
at 32MB. Temporary output is directed to /usr/local2
with more space available. DATA steps submitted one
at a time to ensure /var/tmp does not get overloaded.
*/

libname PCSPTMP '/usr/local2/aes/SAS';

DATA PCSPTMP.Y1984b;
  INFILE '/usr/local2/aes/pcspstest.txt' MISSEVER;
  INPUT STN $ 1-9
        @13 DATE DATETIME13.
        ELEMENT
        VALUE
;
RETURN;
RUN;

PROC SORT DATA=PCSPTMP.Y1984b;
  BY STN DATE ELEMENT;
RUN;

```

```

DATA _NULL_ ;
SET PCSPTMP.Y1984b;
FILE '/usr/local2/aes/pcsptest.sort';
PUT STN DATE ELEMENT VALUE;
RETURN;
RUN;

```

```

DATA _NULL_ ;
INFILE '/usr/local2/aes/pcsptest.sort' MISSOVER;
ARRAY VAL(44);

/* Assign the first OLDDATE value - perform only on the */
/* first iteration of this DATA step. */

IF _N_=1 THEN DO;
INPUT RGN STN $CHAR4. DATE @;
OLDDATE=date;
OPFLAG=1; /* set high to start 1st output */
END;

/* Check for the occurrence of a new date. */
/* If it is a new date, then set up a new output line */
/* and restart the loop that reads through the elements */

INPUT @1 RGN STN $ DATE @; /* read first parameters for */
/* testing purposes */

IF DATE NE OLDDATE THEN DO;
OLDDATE=DATE;
OPFLAG=1;

FILE '/usr/local2/aes/pcsptest.fin'; /* new output line */
PUT;

END;

INPUT @1 RGN STN $ DATE ELEM @;

SELECT(ELEM); /* convert element numbers */
WHEN(71) BELEM = 1;
WHEN(72) BELEM = 2; /* to useable array subscripts */
WHEN(73) BELEM = 3;
WHEN(74) BELEM = 4;
WHEN(76) BELEM = 5;
WHEN(77) BELEM = 6;
WHEN(78) BELEM = 7;
WHEN(79) BELEM = 8;
WHEN(80) BELEM = 9;
WHEN(82) BELEM = 10;
WHEN(86) BELEM = 11;
WHEN(87) BELEM = 12;
WHEN(88) BELEM = 13;
WHEN(89) BELEM = 14;
WHEN(90) BELEM = 15;
WHEN(91) BELEM = 16;
WHEN(92) BELEM = 17;
WHEN(93) BELEM = 18;
WHEN(94) BELEM = 19;
WHEN(95) BELEM = 20;
WHEN(96) BELEM = 21;
WHEN(97) BELEM = 22;
WHEN(98) BELEM = 23;
WHEN(99) BELEM = 24;
WHEN(100) BELEM = 25;
WHEN(101) BELEM = 26;
WHEN(102) BELEM = 27;
WHEN(103) BELEM = 28;
WHEN(104) BELEM = 29;
WHEN(105) BELEM = 30;
WHEN(106) BELEM = 31;
WHEN(108) BELEM = 32;
WHEN(109) BELEM = 33;
WHEN(110) BELEM = 34;
WHEN(112) BELEM = 35;
WHEN(113) BELEM = 36;
WHEN(114) BELEM = 37;
WHEN(116) BELEM = 38;
WHEN(117) BELEM = 39;
WHEN(118) BELEM = 40;
WHEN(120) BELEM = 41;
WHEN(121) BELEM = 42;
WHEN(122) BELEM = 43;
WHEN(156) BELEM = 44;
OTHERWISE;
END;

INPUT VAL{BELEM};

IF OPFLAG THEN DO;
FILE '/usr/local2/aes/pcsptest.fin';
PUT RGN STN $CHAR5. -L
OLDDATE DATETIME13.

```

```

@:
OPFLAG=0;
END;

IF VAL{BELEM}=' ' THEN DO; /* Use proper missing tag. */
VAL{BELEM}=.;
END;

IF ELEM=78 OR ELEM=74 THEN DO; /* convert 1 to negative for Temp */
IF VAL{BELEM} > 999 AND VAL{BELEM} NE 9999 THEN DO;
VAL{BELEM} = (VAL{BELEM}-1000) * -1;
END;
END;

FILE '/usr/local2/aes/pcsptest.fin';
PUT @((BELEM-1)*5+26) VAL{BELEM} @;

RETAIN OLDDATE DATE;

RETURN;
RUN;

DATA PCSP.Y1984b;
INFILE '/usr/local2/aes/pcsptest.fin' MISSOEVER;

/** REFER TO PINK MANUAL FOR FURTHER CODING OF HLY01 DATA **/

INPUT @5 STN $ 4.
@10 DATE DATETIME13.
@26 CIG 5. /* HLY01 71 ceiling */
@31 VIS 5.1 /* HLY01 72 0.1 km */
@36 SLP 5. /* HLY01 73 0.01 kPa */
@41 DBWT 5.1 /* HLY01 74 0.1 C */
/* not included */ /* HLY01 75 pre 1970 wind dir */
@46 WSPD 5. /* HLY01 76 km/hr */
@51 STMP 5. /* HLY01 77 station pressure */
@56 DBTEMP 5.1 /* HLY01 78 0.1 C */
@61 WBTEMP 5.1 /* HLY01 79 wet bulb temp */
@66 RH 5. /* HLY01 80 relative humidity */
/* not included */ /* HLY01 81 total opacity */
@71 TOTCLD 5. /* HLY01 82 tenths */
/* not included */ /* HLY01 83 WX indicator */
/* not included */ /* HLY01 84 tornado */
/* not included */ /* HLY01 85 thunderstorms */
@76 WXR 5. /* HLY01 86 rain -. . */
@81 WXRW 5. /* HLY01 87 rainshowers -. . */
@86 WXL 5. /* HLY01 88 drizzle -. . */
@91 WXER 5. /* HLY01 89 frzng rain -. . */
@96 WXEL 5. /* HLY01 90 frzng drzl -. . */
@101 WXS 5. /* HLY01 91 snow -. . */
@106 WXSG 5. /* HLY01 92 snow grains -. . */
@111 WXIC 5. /* HLY01 93 ice crystals y/n */
@116 WXIP 5. /* HLY01 94 ice pellets -. . */
@121 WXIPW 5. /* HLY01 95 IP showers -. . */
@126 WXSX 5. /* HLY01 96 snow show -. . */
@131 WXSP 5. /* HLY01 97 snow pellet -. . */
@136 WXA 5. /* HLY01 98 hail -. . */
@141 WXP 5. /* HLY01 99 fog y/n */
@146 WXIF 5. /* HLY01 100 ice fog y/n */
@151 WXC 5. /* HLY01 101 smoke y/n */
@156 WCH 5. /* HLY01 102 haze y/n */
@161 WXS 5. /* HLY01 103 blowing snow y/n */
@166 WXB 5. /* HLY01 104 blowing sand y/n */
@171 WXB 5. /* HLY01 105 blowing dust y/n */
@176 WXD 5. /* HLY01 106 dust y/n */
/* not included */ /* HLY01 107 layer1 opacity */
@181 CLDAMT1 5. /* HLY01 108 layer1 cld amount */
@186 CLDTYP1 5. /* HLY01 109 layer1 cld type */
@191 CLDHGT1 5. /* HLY01 110 layer1 cld height */
/* not included */ /* HLY01 111 layer2 opacity */
@196 CLDAMT2 5. /* HLY01 112 layer2 cld amount */
@201 CLDTYP2 5. /* HLY01 113 layer2 cld type */
@206 CLDHGT2 5. /* HLY01 114 layer2 cld height */
/* not included */ /* HLY01 115 layer3 opacity */
@211 CLDAMT3 5. /* HLY01 116 layer3 cld amount */
@216 CLDTYP3 5. /* HLY01 117 layer3 cld type */
@221 CLDHGT3 5. /* HLY01 118 layer3 cld height */
/* not included */ /* HLY01 119 layer4 opacity */
@226 CLDAMT4 5. /* HLY01 120 layer4 cld amount */
@231 CLDTYP4 5. /* HLY01 121 layer4 cld type */
@236 CLDHGT4 5. /* HLY01 122 layer4 cld height */
@241 WDIR /* HLY01 156 tens of degrees */

;

/* NOTE WDIR input CANNOT have a pre-defined width. */
/* If it gets width 5 it is forced to read beyond */
/* the $OL marker. It is thus reaching into 'null' */
/* space and the MISSOEVER option treats the value as */
/* missing because it is not a proper value. Avoid */
/* this problem by using a column positioner and */
/* letting SAS grab the data point up to the $OL mark. */

/* The other INPUT variables must have widths specified. */
/* If not, because the original text file is written with */

```

```

/* spaces for missing data, an INPUT variable without a */
/* defined width will search until it finds more data. */
/* no matter how many columns away it is. The datafile */
/* the above sequence writes out if the field widths are */
/* not included has many recurrences of the same data */
/* value as variable after variable reaches from its */
/* positioned column over to the next available data */
/* value. */

FORMAT DATE DATETIME10. /* output date in a palatable manner */
DBTEMP BEST4.1
WXR RAIN2. /* Format various WX phenomena according */
WXRW RAINSHOW2. /* to standard PCSP format */
WXL DRIZZLE2.
WXZR ZRAIN2.
WXZL ZDRIZZLE2.
WXS SNOW2.
WXSG SNOWGRAN2.
WXIC ICECRYS2.
WXIP ICEPELLT2.
WXIPW WICEPELLT2.
WXSW SNOWSHOW2.
WXSP SNOWPELT2.
WXA HAIL2.
WXP FOG2.
WXIP ICEFOG2.
WXX SMOKE2.
WXH HAZE2.
WXBS BLOWSNOW2.
WXBN BLOWSAND2.
WXBD BLOWDUST2.
WXD DUST2.
;

/* Combine weather phenomena observations into */
/* a single string. Drop individual wx vars. */

DROP WXR WXRW WXL WXSW WXP WXIC WXBS WXIP WXL WXZR WXZL
WXSG WXIP WXIPW WXSP WXA WXX WXH WXBN WXBD WXD;

/* Remove observations that possess no data. */
IF VIS=. AND SLP=. AND DBTEMP=. AND WDIR=.
AND WSPD=. AND TOTCLD=. AND CLDANT1=. AND CLDTYP1=.
AND CLDHGT1=. AND CLDANT2=. AND CLDTYP2=. AND CLDHGT2=.
AND CLDANT3=. AND CLDTYP3=. AND CLDHGT3=. AND CLDANT4=.
AND CLDTYP4=. AND CLDHGT4=. AND WXR=. AND WXRW=. AND WXL=.
AND WXIC=. AND WXSW=. AND WXP=. AND WXIP=. AND WXBS=.
AND WXL=. AND WXZR=. AND WXA=. AND WXZL=. AND WXSG=.
AND WXSP=. AND WXBN=. AND WXD=. AND WXBD=. AND WXIP=.
AND WXIPW=. AND WXX =. AND WXH=.
THEN RETURN;

/* and output them instead to a text file for later merging */
FILE '/u/atkinson/sas/database/84-86/data/WX';
PUT STN $ 4.
DATE DATETIME13.
WXR RAIN2.
WXRW RAINSHOW2.
WXL DRIZZLE2.
WXZR ZRAIN2.
WXZL ZDRIZZLE2.
WXS SNOW2.
WXSG SNOWGRAN2.
WXIC ICECRYS2.
WXIP ICEPELLT2.
WXIPW WICEPELLT2.
WXSW SNOWSHOW2.
WXSP SNOWPELT2.
WXA HAIL2.
WXP FOG2.
WXIP ICEFOG2.
WXX SMOKE2.
WXH HAZE2.
WXBS BLOWSNOW2.
WXBN BLOWSAND2.
WXBD BLOWDUST2.
WXD DUST2.
;

OUTPUT:
RETURN;
RUN;

/******
/* Compress the WX indicators **/

LIBNAME PCSP '/u/atkinson/sas/database/84-86/data';

DATA PCSP.WX;

```

```

INFILE '/u/atkinson/sas/database/84-86/data/WX';
INPUT STN $ 1-4
      @5 DATE DATETIME13.
      WX $ 18-59;

nwx=compress(wx);
drop wx;

FILE '/u/atkinson/sas/database/84-86/data/WX.comp';
PUT nwx 6.;

RETURN;

RUN;

/*-----*/
/*                merge the datasets                */
/*-----*/

DATA PCSP.FIM1984b;
MERGE PCSP.Y1984b PCSP.WX;

RUN;

QUIT;

/*-----*/
/* READ85.sas to read AES HLY01 format data. */
/* For the year 1986 in particular. */
/*
/* This reads in region, station number, year,
/* month, element, and data values and flags
/* for each hour. This program organizes
/* the data into 26 columns, one for each
/* element, using one row for each hour.
/*
/* Identical text file and SAS DS generated.
/*
/*
/* DEC 15, 1997
/* D.E. Atkinson.
/*
/*-----*/
/* Program operates in several steps:
/*
/* 1) First DATA step initiates a DO loop
/* that reads across each line of input
/* and writes out to a text file one line
/* of output for every observation spot
/* of every element, whether or not a datum
/* is present. Extra information is written
/* to construct a date/time field that SAS
/* can handle. Note the use of a user
/* defined format, MONM.
/* The IF statements dealing with MONTH and
/* DAY are to drop inappropriate the 31st of
/* months that do not have them and to drop
/* Feb 29, 30 and 31 for normal years and
/* Feb 30 and 31 for leap years. The
/* original HLY01 data has 31 value spots
/* for all months; such missing days are
/* problematic in time series analyses and
/* the non-existent dates confuse SAS.
/*
/* Trace precip values are assigned 8888.
/* actual missing values are assigned 9999.
/*
/* 2) A temporary SAS dataset is created in the
/* image of the text file of the first step.
/* The main reason for doing this is to
/* give SAS a proper date field. This DS
/* is then sorted by Station, Date/Time
/* and element using PROC SORT.
/*
/* 3) The sorted SAS ds is written to a text
/* file.
/*
/*-----*/

LIBNAME PCSP '/u/atkinson/sas/database/84-86/data';

PROC FORMAT;
VALUE MONM (DEFAULT=3)
1 = 'JAN'
2 = 'FEB'
3 = 'MAR'
4 = 'APR'
5 = 'MAY'
6 = 'JUN'
7 = 'JUL'
8 = 'AUG'
9 = 'SEP'
10 = 'OCT'
11 = 'NOV'
12 = 'DEC'
OTHER = ' ';
;
```



```

IF (MONTH=02 AND DAY GE 29 AND
    (YEAR NOTIN (92,88,84,80,76,72,68,64,60,56,52,48)))
THEN RETURN;
ELSE IF (MONTH=02 AND DAY=30) THEN RETURN;

FILE '/usr/local2/aes/pcsptest.txt';

PUT  @1 REGION
     @5 STN $
     @13 DAY Z2. -R
     @15 MONTH MON3. /* format defined above */
     @18 YEAR
     @20 ':'
     @21 HOURNUM
     @23 ':00'
     @28 ELEMENT 3. -R
     @32 VALUE 6. -R
     FLAG $ 45
;

END;

RUN;

/* New library assigned because program output was
exceeding the limit of /usr/tmp (/var/tmp), set
at 32MB. Temporary output is directed to /usr/local2
with more space available. DATA steps submitted one
at a time to ensure /var/tmp does not get overloaded.
*/

libname PCSPTMP '/usr/local2/aes/SAS';

DATA PCSPTMP.Y1985;
  INFILE '/usr/local2/aes/pcsptest.txt' MISSOVER;
  INPUT  STN $ 1-9
         @13 DATE DATETIME13.
         ELEMENT
         VALUE
;

RETURN;
RUN;

PROC SORT DATA=PCSPTMP.Y1985;
  BY STN DATE ELEMENT;
RUN;

DATA _NULL_;
  SET PCSPTMP.Y1985;
  FILE '/usr/local2/aes/pcsptest.sort';
  PUT STN DATE ELEMENT VALUE;
RETURN;
RUN;

DATA _NULL_;
  INFILE '/usr/local2/aes/pcsptest.sort' MISSOVER;
  ARRAY VAL(44);

/* Assign the first OLDDATE value - perform only on the */
/* first iteration of this DATA step. */

IF _N_=1 THEN DO;
  INPUT RGN STN $ CHAR4. DATE @;
  OLDDATE=DATE;
  OPFLAG=1; /* set high to start 1st output */
END;

/* Check for the occurrence of a new date. */
/* If it is a new date, then set up a new output line */
/* and restart the loop that reads through the elements */

INPUT @1 RGN STN $ DATE @; /* read first parameters for */
                          /* testing purposes */

IF DATE NE OLDDATE THEN DO;
  OLDDATE=DATE;
  OPFLAG=1;

  FILE '/usr/local2/aes/pcsptest.fin'; /* new output line */
  PUT;

END;

INPUT @1 RGN STN $ DATE ELEM @;

SELECT(ELEM); /* convert element numbers */
  WHEN(71) BELEM = 1;
  WHEN(72) BELEM = 2; /* to useable array subscripts */
  WHEN(73) BELEM = 3;
  WHEN(74) BELEM = 4;
  WHEN(76) BELEM = 5;
  WHEN(77) BELEM = 6;

```

```

WHEN(78) BELEM = 7;
WHEN(79) BELEM = 8;
WHEN(80) BELEM = 9;
WHEN(82) BELEM = 10;
WHEN(86) BELEM = 11;
WHEN(87) BELEM = 12;
WHEN(88) BELEM = 13;
WHEN(89) BELEM = 14;
WHEN(90) BELEM = 15;
WHEN(91) BELEM = 16;
WHEN(92) BELEM = 17;
WHEN(93) BELEM = 18;
WHEN(94) BELEM = 19;
WHEN(95) BELEM = 20;
WHEN(96) BELEM = 21;
WHEN(97) BELEM = 22;
WHEN(98) BELEM = 23;
WHEN(99) BELEM = 24;
WHEN(100) BELEM = 25;
WHEN(101) BELEM = 26;
WHEN(102) BELEM = 27;
WHEN(103) BELEM = 28;
WHEN(104) BELEM = 29;
WHEN(105) BELEM = 30;
WHEN(106) BELEM = 31;
WHEN(108) BELEM = 32;
WHEN(109) BELEM = 33;
WHEN(110) BELEM = 34;
WHEN(112) BELEM = 35;
WHEN(113) BELEM = 36;
WHEN(114) BELEM = 37;
WHEN(116) BELEM = 38;
WHEN(117) BELEM = 39;
WHEN(118) BELEM = 40;
WHEN(120) BELEM = 41;
WHEN(121) BELEM = 42;
WHEN(122) BELEM = 43;
WHEN(156) BELEM = 44;
OTHERWISE:
END;

INPUT VAL{BELEM};

IF OPFLAG THEN DO;
  FILE '/usr/local2/aes/pcspstest.fin';
  PUT RMN STN $CHAR5. -L
      OLDDATE DATETIME13.
      @;

  OPFLAG=0;
END;

IF VAL{BELEM}=' ' THEN DO; /* Use proper missing tag. */
  VAL{BELEM}=-.;
END;

IF ELEM=78 OR ELEM=74 THEN DO; /* convert 1 to negative for Temp */
  IF VAL{BELEM} > 999 AND VAL{BELEM} NE 9999 THEN DO;
    VAL{BELEM} = (VAL{BELEM}-1000) * -1;
  END;
END;

FILE '/usr/local2/aes/pcspstest.fin';
PUT @((BELEM-1)*5+26) VAL{BELEM} @;

RETAIN OLDDATE DATE;

RETURN;
RUN;

DATA PCSP.Y1985;
  IMFILE '/usr/local2/aes/pcspstest.fin' MISSEVER;

/* REFER TO PINK MANUAL FOR FURTHER CODING OF HLY01 DATA */

INPUT @5 STN $ 4.
@10 DATE DATETIME13.
@26 CIG 5. /* HLY01 71 ceiling */
@31 VIS 5.1 /* HLY01 72 0.1 km */
@36 SLP 5. /* HLY01 73 0.01 KPa */
@41 DEWT 5.1 /* HLY01 74 0.1 C */
/* not included */ /* HLY01 75 pre 1970 wind dir */
@46 NSPD 5. /* HLY01 76 km/hr */
@51 STWP 5. /* HLY01 77 station pressure */
@56 DBTEMP 5.1 /* HLY01 78 0.1 C */
@61 WBTEMP 5.1 /* HLY01 79 wet bulb temp */
@66 RH 5. /* HLY01 80 relative humidity */
/* not included */ /* HLY01 81 total opacity */
@71 TOTCLD 5. /* HLY01 82 tenths */
/* not included */ /* HLY01 83 WX indicator */
/* not included */ /* HLY01 84 tornado */
/* not included */ /* HLY01 85 thunderstorms */
@76 WXR 5. /* HLY01 86 rain -. . */
@81 WXRN 5. /* HLY01 87 rainshowers -. . */

```

```

086 WXL      5.      /* HLY01 88 drizzle - . . * */
091 WXZR     5.      /* HLY01 89 frzng rain - . . * */
096 WXZL     5.      /* HLY01 90 frzng drzl - . . * */
0101 WXS     5.      /* HLY01 91 snow - . . * */
0106 WXSG     5.      /* HLY01 92 snow grains - . . * */
0111 WXIC     5.      /* HLY01 93 ice crystals y/n */
0116 WXIP     5.      /* HLY01 94 ice pellets - . . * */
0121 WXIPW    5.      /* HLY01 95 IP showers - . . * */
0126 WXSW     5.      /* HLY01 96 snow show - . . * */
0131 WXSP     5.      /* HLY01 97 snow pellet - . . * */
0136 WXA      5.      /* HLY01 98 hail - . . * */
0141 WXP      5.      /* HLY01 99 fog y/n */
0146 WXIF     5.      /* HLY01 100 ice fog y/n */
0151 WVK      5.      /* HLY01 101 smoke y/n */
0156 WXH      5.      /* HLY01 102 haze y/n */
0161 WXBS     5.      /* HLY01 103 blowing snow y/n */
0166 WXBN     5.      /* HLY01 104 blowing sand y/n */
0171 WXBD     5.      /* HLY01 105 blowing dust y/n */
0176 WXD      5.      /* HLY01 106 dust y/n */
/* not included */ /* HLY01 107 layer1 opacity */
0181 CLDAMT1  5.      /* HLY01 108 layer1 cid amount */
0186 CLDTYP1  5.      /* HLY01 109 layer1 cid type */
0191 CLDHGT1  5.      /* HLY01 110 layer1 cid height */
/* not included */ /* HLY01 111 layer2 opacity */
0196 CLDAMT2  5.      /* HLY01 112 layer2 cid amount */
0201 CLDTYP2  5.      /* HLY01 113 layer2 cid type */
0206 CLDHGT2  5.      /* HLY01 114 layer2 cid height */
/* not included */ /* HLY01 115 layer3 opacity */
0211 CLDAMT3  5.      /* HLY01 116 layer3 cid amount */
0216 CLDTYP3  5.      /* HLY01 117 layer3 cid type */
0221 CLDHGT3  5.      /* HLY01 118 layer3 cid height */
/* not included */ /* HLY01 119 layer4 opacity */
0226 CLDAMT4  5.      /* HLY01 120 layer4 cid amount */
0231 CLDTYP4  5.      /* HLY01 121 layer4 cid type */
0236 CLDHGT4  5.      /* HLY01 122 layer4 cid height */
0241 WDIR     /* HLY01 156 tens of degrees */
;
/* NOTE WDIR input CANNOT have a pre-defined width. */
/* If it gets width 5 it is forced to read beyond */
/* the EOL marker. It is thus reaching into 'null' */
/* space and the MISSEVER option treats the value as */
/* missing because it is not a proper value. Avoid */
/* this problem by using a column positioner and */
/* letting SAS grab the data point up to the EOL mark. */

/* The other INPUT variables must have widths specified. */
/* If not, because the original text file is written with */
/* spaces for missing data, an INPUT variable without a */
/* defined width will search until it finds more data, */
/* no matter how many columns away it is. The datafile */
/* the above sequence writes out if the field widths are */
/* not included has many recurrences of the same data */
/* value as variable after variable reaches from its */
/* positioned column over to the next available data */
/* value.

FORMAT DATE DATETIME10. /* output date in a palatable manner */
          DBTEMP  BEST4.1
          WXR     RAIN2. /* Format various WX phenomena according */
          WXRW    RAINSHOW2. /* to standard PCSP format */
          WXL     DRIZZLE2.
          WXZR    ZRAIN2.
          WXZL    ZDRIZZLE2.
          WXS     SNOW2.
          WXSG    SNOWGRAM2.
          WXIC    ICECRYS2.
          WXIP    ICEPELLT2.
          WXIPW   WICEPELLT2.
          WXSW    SNOWSHOW2.
          WXSP    SNOWFELT2.
          WXA     HAIL2.
          WXP     FOG2.
          WXIF    ICEFOG2.
          WVK     SMOKE2.
          WXH     HAZE2.
          WXBS    BLONSNOW2.
          WXBN    BLONSAND2.
          WXBD    BLOWDUST2.
          WXD     DUST2.
;

/* Combine weather phenomena observations into */
/* a single string. Drop individual wx vars. */

DROP WXR WXRW WXS WXSX WXP WXIC WXBS WXIP WXL WXZR WXZL
      WXSG WXIP WXIPW WXSP WXA WVK WXH WXBN WXBD WXD;

/* Remove observations that possess no data. */
IF VIS=. AND SLP=. AND DBTEMP=. AND WDIR=.
   AND MSPD=. AND TOTCLD=. AND CLDAMT1=. AND CLDTYP1=.
   AND CLDHGT1=. AND CLDAMT2=. AND CLDTYP2=. AND CLDHGT2=.
   AND CLDAMT3=. AND CLDTYP3=. AND CLDHGT3=. AND CLDAMT4=.
   AND CLDTYP4=. AND CLDHGT4=. AND WXR=. AND WXRW=. AND WXS=.
   AND WXIC=. AND WXSW=. AND WXP=. AND WXIP=. AND WXBS=.

```

```

AND WXL=. AND WXZR=. AND WXA=. AND WXZL=. AND WXSG=.
AND WXSP=. AND WXBN=. AND WXD=. AND WXBD=. AND WXIP=.
AND WXIPW=. AND WXX =. AND WGH=.
THEN RETURN;

/* and output them instead to a text file for later merging */
FILE '/u/atkinson/sas/database/84-86/data/WX';
PUT STN $ 4.
DATE DATETIME13.
WXR RAIN2.
WXRW RAINSHOW2.
WXL DRIZZLE2.
WXZR ZRAIN2.
WXZL ZDRIZZLE2.
WXS SNOW2.
WXSG SNOWGRAM2.
WXIC ICECRYS2.
WXIP ICEPELT2.
WXIPW WICEPELT2.
WXSX SNOWSHOW2.
WXSP SNOWPELT2.
WXA HAIL2.
WXP FOG2.
WXIP ICEFOG2.
WXX SMOKE2.
WGH HAZE2.
WXS BLOWSNOW2.
WXBN BLOWSNOW2.
WXBD BLOWDUST2.
WXD DUST2.
;

OUTPUT;

RETURN;

RUN;

/******
/* Compress the WX indicators **/

LIBNAME PCSP '/u/atkinson/sas/database/84-86/data';

DATA PCSP.WX;
  INFILE '/u/atkinson/sas/database/84-86/data/WX';
  INPUT STN $ 1-4
        @5 DATE DATETIME13.
        WX $ 18-59;

  nwx=compress(wx);
  drop wx;

  FILE '/u/atkinson/sas/database/84-86/data/WX.comp';
  PUT nwx 6.;

RETURN;

RUN;

/******
/* merge the datasets **/

DATA PCSP.FIM1985;
  MERGE PCSP.Y1985 PCSP.WX;

RUN;

QUIT:

/*****
/*
/* READ86.sas to read AES HLY01 format data. */
/* For the year 1986 in particular. */
/*
/* This reads in region, station number, year, */
/* month, element, and data values and flags */
/* for each hour. This program organizes */
/* the data into 26 columns, one for each */
/* element, using one row for each hour. */
/*
/* Identical text file and SAS DS generated. */
/*
/*
/* DEC 09, 1997 */
/* D.E. Atkinson. */
/*
/*
/*****
/* Program operates in several steps: */
/*

```



```

DATA _NULL_;
  INFILE '/u/atkinson/sas/database/84-86/data/pcspy86.prm' MISSEVER SHAREBUFFERS;
  ATTRIB REGION FORMAT=3. ;
  ATTRIB STN LENGTH=54 ;
  ATTRIB YEAR FORMAT=3. ;
  ATTRIB MONTH FORMAT=2. ;
  ATTRIB ELEMENT FORMAT=3. ;
  ATTRIB VALUE FORMAT=6. ;
  ATTRIB FLAG LENGTH=51 ;
  ATTRIB HOURNUM FORMAT=2. ;

  RETAIN REGION STN YEAR MONTH ELEMENT;

  INPUT @1 REGION 3.
        @4 STN $ 4.
        @9 YEAR 2.
        @11 MONTH 2.
        @13 DAY 2.
        @15 ELEMENT 3. @
        ;

  DO HOURNUM=1 TO 24;
    INPUT @((HOURNUM-1)*7+18) VALUE 6.
          @((HOURNUM-1)*7+18+6) FLAG $1. @ ;

    IF FLAG='Z' THEN VALUE=8888;
    IF FLAG='M' THEN VALUE=.; /* ditch missing data */

    IF (MONTH=09 OR MONTH=04 OR MONTH=06 OR MONTH=11) .
      AND (DAY=31) THEN RETURN;

    IF (MONTH=02 AND DAY GE 29 AND
        (YEAR MODIN (92,88,84,80,76,72,68,64,60,56,52,48)))
      THEN RETURN;
    ELSE IF (MONTH=02 AND DAY=30) THEN RETURN;

    FILE '/usr/local2/aes/pcspstest.txt';

    PUT @1 REGION
        @5 STN $
        @13 DAY 22. -R
        @15 MONTH MON3. /* format defined above */
        @18 YEAR
        @20 ':'
        @21 HOURNUM
        @23 ':00'
        @28 ELEMENT 3. -R
        @32 VALUE 6. -R
          FLAG $ 45
        ;
  END;
RUN;

/* New library assigned because program output was
exceeding the limit of /usr/tmp (/var/tmp), set
at 32MB. Temporary output is directed to /usr/local2
with more space available. DATA steps submitted one
at a time to ensure /var/tmp does not get overloaded.
*/

libname PCSPTMP '/usr/local2/aes/SAS';

DATA PCSPTMP.Y1986;
  INFILE '/usr/local2/aes/pcspstest.txt' MISSEVER;
  INPUT STN $ 1-9
        @13 DATE DATETIME13.
        ELEMENT
        VALUE
        ;
RETURN;
RUN;

PROC SORT DATA=PCSPTMP.Y1986;
  BY STN DATE ELEMENT;
RUN;

DATA _NULL_;
  SET PCSPTMP.Y1986;
  FILE '/usr/local2/aes/pcspstest.sort';
  PUT STN DATE ELEMENT VALUE;
RETURN;
RUN;

DATA _NULL_;
  INFILE '/usr/local2/aes/pcspstest.sort' MISSEVER;
  ARRAY VAL(44);

/* Assign the first OLDDATE value - perform only on the */
/* first iteration of this DATA step. */

  IF _N_=1 THEN DO;
    INPUT RGN STN SCHAR4. DATE @;

```

```

OLDDATE=DATE;
OPFLAG=1; /* set high to start 1st output */
END;

/* Check for the occurrence of a new date. */
/* If it is a new date, then set up a new output line */
/* and restart the loop that reads through the elements */

INPUT @1 RGN STN $ DATE @; /* read first parameters for */
/* testing purposes */

IF DATE NE OLDDATE THEN DO;
  OLDDATE=DATE;
  OPFLAG=1;

  FILE '/usr/local2/aes/pcsptest.fin'; /* new output line */
  PUT;

  END;

INPUT @1 RGN STN $ DATE ELEM @;

SELECT(ELEM); /* convert element numbers */
WHEN(71) BELEM = 1;
WHEN(72) BELEM = 2; /* to useable array subscripts */
WHEN(73) BELEM = 3;
WHEN(74) BELEM = 4;
WHEN(76) BELEM = 5;
WHEN(77) BELEM = 6;
WHEN(78) BELEM = 7;
WHEN(79) BELEM = 8;
WHEN(80) BELEM = 9;
WHEN(82) BELEM = 10;
WHEN(86) BELEM = 11;
WHEN(87) BELEM = 12;
WHEN(88) BELEM = 13;
WHEN(89) BELEM = 14;
WHEN(90) BELEM = 15;
WHEN(91) BELEM = 16;
WHEN(92) BELEM = 17;
WHEN(93) BELEM = 18;
WHEN(94) BELEM = 19;
WHEN(95) BELEM = 20;
WHEN(96) BELEM = 21;
WHEN(97) BELEM = 22;
WHEN(98) BELEM = 23;
WHEN(99) BELEM = 24;
WHEN(100) BELEM = 25;
WHEN(101) BELEM = 26;
WHEN(102) BELEM = 27;
WHEN(103) BELEM = 28;
WHEN(104) BELEM = 29;
WHEN(105) BELEM = 30;
WHEN(106) BELEM = 31;
WHEN(108) BELEM = 32;
WHEN(109) BELEM = 33;
WHEN(110) BELEM = 34;
WHEN(112) BELEM = 35;
WHEN(113) BELEM = 36;
WHEN(114) BELEM = 37;
WHEN(116) BELEM = 38;
WHEN(117) BELEM = 39;
WHEN(118) BELEM = 40;
WHEN(120) BELEM = 41;
WHEN(121) BELEM = 42;
WHEN(122) BELEM = 43;
WHEN(156) BELEM = 44;
OTHERWISE;
END;

INPUT VAL{BELEM};

IF OPFLAG THEN DO;
  FILE '/usr/local2/aes/pcsptest.fin';
  PUT RGN STN $CHAR5. -L
    OLDDATE DATETIME13.
    @;

  OPFLAG=0;
  END;

IF VAL{BELEM}=' ' THEN DO; /* Use proper missing tag. */
  VAL{BELEM}=.;
  END;

IF ELEM=78 OR ELEM=74 THEN DO; /* convert 1 to negative for Temp */
  IF VAL{BELEM} > 999 AND VAL{BELEM} NE 9999 THEN DO;
    VAL{BELEM} = (VAL{BELEM}-1000) * -1;
  END;
  END;

FILE '/usr/local2/aes/pcsptest.fin';
PUT @((BELEM-1)*5+26) VAL{BELEM} @;

```

RETAIN OLDDATE DATE;

RETURN;
RUN;

DATA PCSP.Y1986;

INFILE '/usr/local2/aes/pcspstest.fin' MISSEVER;

/** REFER TO PINK MANUAL FOR FURTHER CODING OF HLY01 DATA **/

```
INPUT @5 STN $ 4.
@10 DATE DATETIME11.
@26 CIC 5. /* HLY01 71 ceiling */
@31 VIS 5.1 /* HLY01 72 0.1 km */
@36 SLP 5. /* HLY01 73 0.01 KPa */
@41 DEWT 5.1 /* HLY01 74 0.1 C */
/* not included */ /* HLY01 75 pre 1970 wind dir */
@46 WSPD 5. /* HLY01 76 km/hr */
@51 STNP 5. /* HLY01 77 station pressure */
@56 DBTEMP 5.1 /* HLY01 78 0.1 C */
@61 WBTEMP 5.1 /* HLY01 79 wet bulb temp */
@66 RH 5. /* HLY01 80 relative humidity */
/* not included */ /* HLY01 81 total opacity */
@71 TOTCLD 5. /* HLY01 82 tenths */
/* not included */ /* HLY01 83 WX indicator */
/* not included */ /* HLY01 84 tornado */
/* not included */ /* HLY01 85 thunderstorms */
@76 WGR 5. /* HLY01 86 rain - , , */
@81 WGRW 5. /* HLY01 87 rainshowers - , , */
@86 WKL 5. /* HLY01 88 drizzle - , , */
@91 WZR 5. /* HLY01 89 frzng rain - , , */
@96 WZL 5. /* HLY01 90 frzng drzl - , , */
@101 WKS 5. /* HLY01 91 snow - , , */
@106 WKSG 5. /* HLY01 92 snow grains - , , */
@111 WKIC 5. /* HLY01 93 ice crystals y/n */
@116 WKIP 5. /* HLY01 94 ice pellets - , , */
@121 WKIPW 5. /* HLY01 95 IP showers - , , */
@126 WKSW 5. /* HLY01 96 snow show - , , */
@131 WKSP 5. /* HLY01 97 snow pellet - , , */
@136 WXA 5. /* HLY01 98 hail - , , */
@141 WXP 5. /* HLY01 99 fog y/n */
@146 WKIP 5. /* HLY01 100 ice fog y/n */
@151 WKK 5. /* HLY01 101 smoke y/n */
@156 WGH 5. /* HLY01 102 haze y/n */
@161 WKBS 5. /* HLY01 103 blowing snow y/n */
@166 WKBN 5. /* HLY01 104 blowing sand y/n */
@171 WKBD 5. /* HLY01 105 blowing dust y/n */
@176 WDD 5. /* HLY01 106 dust y/n */
/* not included */ /* HLY01 107 layer1 opacity */
@181 CLDAMT1 5. /* HLY01 108 layer1 cld amount */
@186 CLDTYP1 5. /* HLY01 109 layer1 cld type */
@191 CLDHGT1 5. /* HLY01 110 layer1 cld height */
/* not included */ /* HLY01 111 layer2 opacity */
@196 CLDAMT2 5. /* HLY01 112 layer2 cld amount */
@201 CLDTYP2 5. /* HLY01 113 layer2 cld type */
@206 CLDHGT2 5. /* HLY01 114 layer2 cld height */
/* not included */ /* HLY01 115 layer3 opacity */
@211 CLDAMT3 5. /* HLY01 116 layer3 cld amount */
@216 CLDTYP3 5. /* HLY01 117 layer3 cld type */
@221 CLDHGT3 5. /* HLY01 118 layer3 cld height */
/* not included */ /* HLY01 119 layer4 opacity */
@226 CLDAMT4 5. /* HLY01 120 layer4 cld amount */
@231 CLDTYP4 5. /* HLY01 121 layer4 cld type */
@236 CLDHGT4 5. /* HLY01 122 layer4 cld height */
@241 WDIR /* HLY01 156 tens of degrees */

/* NOTE WDIR input CANNOT have a pre-defined width. */
/* If it gets width 5 it is forced to read beyond */
/* the EOL marker. It is thus reaching into 'null' */
/* space and the MISSEVER option treats the value as */
/* missing because it is not a proper value. Avoid */
/* this problem by using a column positioner and */
/* letting SAS grab the data point up to the EOL mark. */

/* The other INPUT variables must have widths specified. */
/* If not, because the original text file is written with */
/* spaces for missing data, an INPUT variable without a */
/* defined width will search until it finds more data, */
/* no matter how many columns away it is. The datafile */
/* the above sequence writes out if the field widths are */
/* not included has many recurrences of the same data */
/* value as variable after variable reaches from its */
/* positioned column over to the next available data */
/* value.

FORMAT DATE DATETIME10. /* output date in a palatable manner */
DBTEMP BEST4.1
WXR RAIN2. /* Format various WX phenomena according */
WGRW RAINSHOW2. /* to standard PCSP format */
WKL DRIZZLE2.
WZR ZRAIN2.
WZL ZDRIZZLE2.
WKS SNOW2.
```

```

WXSG SNOWGRAM2.
WXIC ICECRYS2.
WXIP ICEPELLT2.
WXIPW WICEPELLT2.
WXSM SNOWSHOW2.
WXSP SNOWFELT2.
WXA HAIL2.
WXP FOG2.
WXIF ICEFOG2.
WXX SMOKE2.
WXH HAZE2.
WXBS BLOWSNOW2.
WXBW BLOWSAND2.
WXBD BLOWDUST2.
WXD DUST2.
;

/* Combine weather phenomena observations into */
/* a single string. Drop individual wx vars. */

DROP WXR WXRW WXS WXSX WXP WXIC WXBS WXIF WXL WXZR WXZL
      WXSG WXIP WXIPW WXSP WXA WXX WXH WXBW WXBD WXD;

/* Remove observations that possess no data. */
IF VIS= . AND SLP= . AND DBTEMP= . AND WDIR= .
   AND WSPD= . AND TOTCLD= . AND CLDAMT1= . AND CLDTYP1= .
   AND CLDHGT1= . AND CLDAMT2= . AND CLDTYP2= . AND CLDHGT2= .
   AND CLDAMT3= . AND CLDTYP3= . AND CLDHGT3= . AND CLDAMT4= .
   AND CLDTYP4= . AND CLDHGT4= . AND WXR= . AND WXRW= . AND WXS= .
   AND WXIC= . AND WXSX= . AND WXP= . AND WXIF= . AND WXBS= .
   AND WXL= . AND WXZR= . AND WXA= . AND WXZL= . AND WXSG= .
   AND WXSP= . AND WXBW= . AND WXBD= . AND WXIF= .
   AND WXIPW= . AND WXX = . AND WXH= .
   THEN RETURN;

/* and output them instead to a text file for later merging */
FILE '/u/atkinson/sas/database/84-86/data/WX';
PUT STN $ 4.
   DATE DATETIME13.
   WXR RAIN2.
   WXRW RAINSHOW2.
   WXL DRIZZLE2.
   WXZR ZRAIN2.
   WXZL ZDRIZZLE2.
   WXS SNOW2.
   WXSG SNOWGRAM2.
   WXIC ICECRYS2.
   WXIP ICEPELLT2.
   WXIPW WICEPELLT2.
   WXSM SNOWSHOW2.
   WXSP SNOWFELT2.
   WXA HAIL2.
   WXP FOG2.
   WXIF ICEFOG2.
   WXX SMOKE2.
   WXH HAZE2.
   WXBS BLOWSNOW2.
   WXBW BLOWSAND2.
   WXBD BLOWDUST2.
   WXD DUST2.
;

OUTPUT;

RETURN;

RUN;

/******
/* Compress the WX indicators */

LIBNAME PCSP '/u/atkinson/sas/database/84-86/data';

DATA PCSP.WX;
  INFILE '/u/atkinson/sas/database/84-86/data/WX';
  INPUT STN $ 1-4
        @5 DATE DATETIME13.
        WX $ 18-59;

  nwx=compress(wx);
  drop wx;

  FILE '/u/atkinson/sas/database/84-86/data/WX.comp';
  PUT nwx 6.;

RETURN;

RUN;

/******

```

```

/**          merge the datasets          **/

DATA PCSP.FIM1986;
MERGE PCSP.Y1986 PCSP.WX;
RUN; QUIT;

```

SAS Programs for 1987 - 1988

SAS Program: 8788read.sas

Purpose: To read data for 1987 and 1988 into a SAS dataset. Data must be prepared using the Quattro macro M8788.wqm.

```

/*****
/* 8788read.sas reads single PCSP files for the years
/* 1987 to 1988. Years must be changed in the program.
/*
/*
/* Program both arranges date properly and then reads in
/* all text files into SAS dataset. Program also performs
/* some QC processing by rejecting missing records and
/* extreme temperature values. It also substitutes
/* negative values for temperature values above 100.
/*
/*
/* D.E. Atkinson
/* November 14, 1997.
*****/

LIBNAME PCSP '/u/atkinson/sas/database/87-88/output';

OPTIONS ERRORS = 10;

/* 2nd Rearrange the data, throw out missing dates, correct any missing
   UCT times and rewrite in a form that SAS can handle.
*/

PROC FORMAT;
  VALUE MON (DEFAULT=3)
    1 = 'JAN'
    2 = 'FEB'
    3 = 'MAR'
    4 = 'APR'
    5 = 'MAY'
    6 = 'JUN'
    7 = 'JUL'
    8 = 'AUG'
    9 = 'SEP'
   10 = 'OCT'
   11 = 'NOV'
   12 = 'DEC'
   OTHER = ' '
  ;
RUN;

DATA _NULL_;
  INFILE '/u/atkinson/sas/database/87-88/data/out87.prm' MISSEVER FIRSTOBS=2;

  INPUT
    @1 ID 4.
    @5 DAY 2.
    @7 MONTH 2.
    @9 YEAR 2.
    @11 LCLTIME 2.
    @13 UCLTIME 2.
    @15 VIS 4.
    @19 WX1 2.
    @21 WX2 2.
    @23 WX3 2.
    @25 STNPRESS 3.
    @28 DBTEMP 3.
    @31 DEWTEMP 3.
    @34 WDIR 2.
    @36 WSPD 2.
    @38 ALTSBT 3.
    @41 TOTCLD 2.
    @43 CLDAMT1 2.
    @45 CLDTYP1 2.
    @47 CLDHGT1 1.
    @50 CLDAMT2 2.
    @52 CLDTYP2 2.
    @54 CLDHGT2 1.
    @57 CLDAMT3 2.
    @59 CLDTYP3 2.
    @61 CLDHGT3 1.
    @64 CLDAMT4 2.
    @66 CLDTYP4 2.
    @68 CLDHGT4 1.
    @71 MAXTEMP 3.
    @74 MINTEMP 3.

```

```

      @77 PRECIP    3.
      @80 PTENDCY
      ;

IF ID = . AND MONTH = . AND YEAR = . THEN RETURN;
/* ditch any blank lines */

IF LCLTIME NE . AND UCTIME = . THEN UCTIME=LCLTIME+5;
IF UCTIME = 24 THEN UCTIME = 00;

IF UCTIME > 24 THEN DO;
  UCTIME = UCTIME-24;
  DAY=DAY+1;
  END;

DROP LCLTIME;

FILE "/u/atkinson/sas/database/87-88/data/out87B.txt";

PUT   @1 ID          4.
      @5 DAY        22. -R
      @7 MONTH      MON3.
      @10 YEAR
      @12 " : "
      @13 UCTIME    22. -R
      @15 " :00"
      @18 VIS       24. -R
      @22 WX1       2.
      @24 WX2       2.
      @26 WX3       2.
      @29 STWPRESS  3.
      @32 DSTEMP   3.
      @35 DEWTEMP  3.
      @38 MDIR     2.
      @40 WSPD     2.
      @43 ALTSET   3.
      @45 TOTCLD   2.
      @47 CLDAMT1  2.
      @49 CLDTYP1  2.
      @51 CLDHGT1  3.
      @54 CLDAMT2  2.
      @56 CLDTYP2  2.
      @58 CLDHGT2  3.
      @61 CLDAMT3  2.
      @63 CLDTYP3  2.
      @65 CLDHGT3  3.
      @68 CLDAMT4  2.
      @70 CLDTYP4  2.
      @72 CLDHGT4  3.
      @75 MAXTEMP  3.
      @78 MINTEMP  3.
      @81 PRECIP   3.
      @84 PTENDCY  3.
      ;

RETURN;

DATA PCSP.PCSP87;

INFILE "/u/atkinson/sas/database/87-88/data/out87B.txt" MISSEVER;

INPUT  @1 ID          4.
      @5 DATE        DATETIME13.
      @18 VIS        4.
      @22 WX1       2.
      @24 WX2       2.
      @26 WX3       2.
      @28 SLP       3.
      @32 DSTEMP   3.
      @35 DEWTEMP  3.
      @38 MDIR     2.
      @40 WSPD     2.
      @42 ALTSET   3.
      @45 TOTCLD   2.
      @47 CLDAMT1  2.
      @49 CLDTYP1  2.
      @51 CLDHGT1  3.
      @54 CLDAMT2  2.
      @56 CLDTYP2  2.
      @58 CLDHGT2  3.
      @61 CLDAMT3  2.
      @63 CLDTYP3  2.
      @65 CLDHGT3  3.
      @68 CLDAMT4  2.
      @70 CLDTYP4  2.
      @72 CLDHGT4  3.
      @75 MAXTEMP  3.
      @78 MINTEMP  3.
      @81 PRECIP   3.
      @84 PTENDCY  3.
      ;

```

```

IF DBTEMP > 100 THEN
  IF DBTEMP > 200 THEN DBTEMP = .;
  ELSE DBTEMP = -1*(DBTEMP-100);

IF DEWTEMP > 100 THEN
  IF DEWTEMP > 200 THEN DEWTEMP = .;
  ELSE DEWTEMP = -1*(DEWTEMP-100);

/* reject extreme temperature values */

IF VIS=. AND SLP=. AND DBTEMP=. AND DEWTEMP=. AND WDIR=.
  AND WSPD=. AND ALTSET=. AND TOTCLD=. AND CLDAMT1=. AND CLDTYP1=.
  AND CLDHGT1=. AND CLDAMT2=. AND CLDTYP2=. AND CLDHGT2=. AND CLDAMT3=.
  AND CLDTYP3=. AND CLDHGT3=. AND CLDAMT4=. AND CLDTYP4=. AND CLDHGT4=.
  AND MAXTEMP=. AND MINTEMP=. AND PRECIP=. AND PTENDCY=. THEN RETURN;
/* This checks that there is an observational record accompanying the station
identification and date header. */

/* These following two lines also create a text file that is
identical to the SAS datafile */

/* file '/u/atkinson/sas/text.op.test'; */
/* put _infile_; */

RETURN;

RUN;

QUIT;

```

SAS Program: 8788ex.sas

Purpose: Some data in 1987 and 1988 were not included in the batch processing due to discrepancies in formatting. The stations involved were processed by hand, concatenated and then the following program was run to read them into a SAS dataset. Program run once for extra data in 1987 and again in 1988; program is essentially identical to 8788read.sas.

```

/*****/
/* 8788ex.sas reads single PCSP files for the years */
/* 1987 to 1988. Years must be changed in the program. */
/* */
/* Program both arranges date properly and then reads in */
/* all text files into SAS dataset. Program also performs */
/* some QC processing by rejecting missing records and */
/* extreme temperature values. It also substitutes */
/* negative values for temperature values above 100. */
/* */
/* D.E. Atkinson */
/* July 7, 1988 */
/*****/

LIBNAME PCSP '/u/atkinson/sas/database/87-88/output';

OPTIONS ERRORS = 10;

/* 2nd Rearrange the data, throw out missing dates, correct any missing
UCT times and rewrite in a form that SAS can handle. */

PROC FORMAT;
  VALUE MON (DEFAULT=3)
    1 = 'JAN'
    2 = 'FEB'
    3 = 'MAR'
    4 = 'APR'
    5 = 'MAY'
    6 = 'JUN'
    7 = 'JUL'
    8 = 'AUG'
    9 = 'SEP'
   10 = 'OCT'
   11 = 'NOV'
   12 = 'DEC'
    OTHER = ' '
  ;
RUN;

DATA _NULL_;

```

INFILE '/u/atkinson/sas/database/87-88/data/P88OTHR.PRN' MISSEVER FIRSTOBS=2;

```
INPUT 01 ID 4.
      05 DAY 2.
      07 MONTH 2.
      09 YEAR 2.
      011 LCLTIME 2.
      013 UCTTIME 2.
      015 VIS $ 4.
      019 WX1 2.
      021 WX2 2.
      023 WX3 2.
      025 STMPRESS 3.
      028 DBTEMP 3.
      031 DEWTEMP 3.
      034 WDIR 2.
      036 WSPD 2.
      038 ALTSET 3.
      041 TOTCLD 2.
      043 CLDAMT1 2.
      045 CLDTYP1 2.
      047 CLDHGT1 3.
      050 CLDAMT2 2.
      052 CLDTYP2 2.
      054 CLDHGT2 3.
      057 CLDAMT3 2.
      059 CLDTYP3 2.
      061 CLDHGT3 3.
      064 CLDAMT4 2.
      066 CLDTYP4 2.
      068 CLDHGT4 3.
      071 MAXTEMP 3.
      074 MINTEMP 3.
      077 PRECIP 3.
      080 PWDNCY 3.
      ;
```

```
IF ID =. AND MONTH =. AND YEAR =. THEN RETURN;
/* ditch any blank lines */
```

```
IF LCLTIME NE . AND UCTTIME = . THEN UCTTIME=LCLTIME+5;
IF UCTTIME = 24 THEN DO;
  UCTTIME = 00;
  DAY=DAY+1;
END;
```

```
IF UCTTIME > 24 THEN DO;
  UCTTIME = UCTTIME-24;
  DAY=DAY+1;
END;
```

```
IF (MONTH=03 OR MONTH=05 OR MONTH=07 OR MONTH=08 OR MONTH=10)
AND (DAY=32) THEN DO;
  MONTH=MONTH+1;
  DAY=01;
END;
```

```
IF (MONTH=09 OR MONTH=04 OR MONTH=06 OR MONTH=11)
AND (DAY=31) THEN DO;
  MONTH=MONTH+1;
  DAY=01;
END;
```

```
IF (MONTH=02 AND DAY GE 29 AND
(YEAR NOT IN (92,88,84,80,76,72,68,64,60,56,52,48)))
THEN DO;
  MONTH=MONTH+1;
  DAY=01;
END;
```

```
ELSE IF (MONTH=02 AND DAY=30) THEN DO;
  MONTH=MONTH+1;
  DAY=01;
END;
```

```
DROP LCLTIME;
```

```
vis=vis*1;
FILE '/u/atkinson/sas/database/87-88/data/P88OTHR.txt';
```

```
PUT 01 ID 4.
    05 DAY Z2. -R
    07 MONTH MON3.
    010 YEAR
    012 " : "
    013 UCTTIME Z2. -R
    015 " :00"
    018 VIS 4.1 -R
    022 WX1 2.
    024 WX2 2.
    026 WX3 2.
    029 STMPRESS 3.
    032 DBTEMP 3.
    035 DEWTEMP 3.
    038 WDIR 2.
```

```

040 WSPD      2.
043 ALTSET   3.
045 TOTCLD   2.
047 CLDAMT1  2.
049 CLDTYP1  2.
051 CLDHGT1  3.
054 CLDAMT2  2.
056 CLDTYP2  2.
058 CLDHGT2  3.
061 CLDAMT3  2.
063 CLDTYP3  2.
065 CLDHGT3  3.
068 CLDAMT4  2.
070 CLDTYP4  2.
072 CLDHGT4  3.
075 MAXTEMP  3.
078 MINTEMP  3.
081 PRECIP   3.
084 PTENCY   3.

```

```
RETURN;
```

```
DATA PCSP2.P88othr;
```

```
INFILE '/u/atkinson/sas/database/87-88/data/P88OTHR.txt' MISSEVER;
```

```

INPUT 01 ID          4.
      05 DATE        DATETIME13.
      018 VIS        4.
      022 WX1        2.
      024 WX2        2.
      026 WX3        2.
      028 SLP        3.
      032 DBTEMP     3.
      035 DEWTEMP    3.
      038 WDIR       2.
      040 WSPD       2.
      042 ALTSET     3.
      045 TOTCLD     2.
      047 CLDAMT1    2.
      049 CLDTYP1    2.
      051 CLDHGT1    3.
      054 CLDAMT2    2.
      056 CLDTYP2    2.
      058 CLDHGT2    3.
      061 CLDAMT3    2.
      063 CLDTYP3    2.
      065 CLDHGT3    3.
      068 CLDAMT4    2.
      070 CLDTYP4    2.
      072 CLDHGT4    3.
      075 MAXTEMP    3.
      078 MINTEMP    3.
      081 PRECIP     3.
      084 PTENCY     3.

```

```

IF DBTEMP > 100 THEN
  IF DBTEMP > 200 THEN DBTEMP = .;
  ELSE DBTEMP = -1*(DBTEMP-100);

```

```

IF DEWTEMP > 100 THEN
  IF DEWTEMP > 200 THEN DEWTEMP = .;
  ELSE DEWTEMP = -1*(DEWTEMP-100);

```

```
/* reject extreme temperature values */
```

```

IF VIS=. AND SLP=. AND DBTEMP=. AND DEWTEMP=. AND WDIR=.
AND WSPD=. AND ALTSET=. AND TOTCLD=. AND CLDAMT1=. AND CLDTYP1=.
AND CLDHGT1=. AND CLDAMT2=. AND CLDTYP2=. AND CLDHGT2=. AND CLDAMT3=.
AND CLDTYP3=. AND CLDHGT3=. AND CLDAMT4=. AND CLDTYP4=. AND CLDHGT4=.
AND MAXTEMP=. AND MINTEMP=. AND PRECIP=. AND PTENCY=. THEN RETURN;
/* This checks that there is an observational record accompanying the station
identification and date header. */

```

```

/* These following two lines also create a text file that is
identical to the SAS datafile */

```

```

/* file '/u/atkinson/sas/text.op.test'; */
/* put _infile_; */

```

```
RETURN;
```

```
RUN;
```

```
QUIT;
```

```

data p8788.p88fin;
  set p8788.p88new pcsp2.p88othr;

```

run;

SAS Program: 8788mrg.sas

Purpose: Program prepares extra data and then merges it into the main database.
Verification check performed when complete.

```
libname pcsp2 '/usr/local2/pcsp/';
libname p8788 '/u/atkinson/sas/database/87-88/data/';

/* this data step adds the fields necessary to make the dataset compatible with the main database.*/

data p8788.p88fin2;
  length CHARID $ 20;
  length AESID $ 4;
  set p8788.p88fin;
  CHARID=" ";
  AESID=" ";
  PCSPID=ID;
  format PCSPID 4.;
  drop ID;
  stnpress=.;
  RH=.;
  format date date14.4;
run;

/* Sort the new dataset to match the main database.*/
PROC SORT DATA=p8788.p88fin2;
  by CHARID AESID PCSPID DATE;
run;

/* Throw out all existing obs for the year being updated, 1988 in this case. We do this because in adding extra data a
complete dataset for the year 1988 has been created. If this is merged with the main database there will be duplicate
records to deal with. */

data pcsp2.done;
  set pcsp2.done;
  if year(datepart(date))=1988 then return;
  /* Remember function datepart must be used to extract the date portion of datetime value, which the variable
  "date" is stored as; only then can the "year" function be used to extract the year value. Using the function
  year on the variable date will not work because it is stored as a datetime variable. Remember the distinction
  between datetime and date formats for variables.
  */
  output;
  /* Explicitly coding an OUTPUT statement in a data step forces output to the new dataset to occur at that
  point. If an OUTPUT statement does not appear then its presence is implied at the end of the DATA step and
  that is when output will occur. In this way can certain observations be dropped, as in this case all
  observations with a year of 1988. This is done by including a condition that, upon testing positive, executes
  a RETURN statement that returns control of the loop to the top of the DATA step, bypassing the OUTPUT
  statement and thus excluding the current observation from the new dataset being formed.
  */
run;

/* Merge the dataset containing the extra observations into the main database. */

DATA pcsp2.done;
  merge pcsp2.done (in=a) p8788.p88fin2 (in=b);
  by CHARID AESID PCSPID DATE;
  if a or b;
  /* The structures (in=a) specify a variable to be assigned to the contribution of each dataset. Use of
  this, with a the statement "if a or b;" forces all observations of both datasets to be put into the new
  output dataset being created.
  */
run;

/** The following part allows a quick verification of the data possessed by the main database for the years that have
just been updated. Verification consists of checking that 1) the previously existing data had been overwritten, and 2)
that the data expected to be there were in fact present.
First a subset of the data containing only the years in question is created (in this case 1987 and 1988). The subset is
then sorted by the variables that will be specified as BY groups in the frequency analysis procedure (necessary or the
frequency procedure will not operate). Then PROC FREQ is run to return a count of observations by identification number
(whatever is specified in the TABLES statement), grouped BY year.
*/

data subset;
  set pcsp2.done;
  year=year(datepart(date));
  drop date;
  if year(datepart(date)) in (1987,1988) then output;
run;

proc sort data=subset;
  by year pcspid;
run;

proc freq data=subset;
  tables pcspid ;
  by year;
```

run:

Quattro macro 1: M8788.wqm

insert macro that formats the 8788 data

```
first      (: Only needs to be executed once)
           (/ View:NewFileMgr)
           (UP 2)
           (:erase current entry)
           {ESC}\atkinson\DATA\Y1988\{D}{ESC}output*.w*-
           *{FOR downcount.0.67.1.BodyLoop}*

           downcount 19

BodyLoop   (: move down to the correct file)
           {DOWN}      (: position on first file in list)
           {DOWN}      (: and get the file (use the -))
           -

move lat/lon      (/ Block:Move){ESC}B1..C600-AG1-
                 (/ Column:Delete)B1-
move day over     (/ Block:Move){ESC}E1..E600-B1-
                 (/ Column:Delete)E1-
                 {HOME}
                 (/ Publish:AlignLeft)
                 {END}{R}{D 600}{R 2}-
                 {HOME}

id              (/ Column:Width)4-(R)
day             (/ Column:Width)2-(R)
mon            (/ Column:Width)2-(R)
year           (/ Column:Width)2-(R)
LCL            (/ Column:Width)2-(R)
UCT            (/ Column:Width)2-(R)
vis            (/ Column:Width)4-(R)
*wx1,wx2,wx3*   (/ Column:Width)6-(R)
stmpress       (/ Column:Width)3-(R)
dbtemp         (/ Column:Width)3-(R)
dewtemp        (/ Column:Width)3-(R)
wdir           (/ Column:Width)2-(R)
wspd           (/ Column:Width)2-(R)
altset         (/ Column:Width)3-(R)
totcld         (/ Column:Width)2-(R)
cldamt1        (/ Column:Width)2-(R)
cldamt1        (/ Column:Width)2-(R)
cldbgt1        (/ Column:Width)3-(R)
cldamt2        (/ Column:Width)2-(R)
cldamt2        (/ Column:Width)2-(R)
cldamt2        (/ Column:Width)3-(R)
cldamt3        (/ Column:Width)2-(R)
cldamt3        (/ Column:Width)2-(R)
cldamt3        (/ Column:Width)3-(R)
cldamt4        (/ Column:Width)2-(R)
cldamt4        (/ Column:Width)2-(R)
cldamt4        (/ Column:Width)3-(R)
cldamt4        (/ Column:Width)3-(R)
maxtemp        (/ Column:Width)3-(R)
mintemp        (/ Column:Width)3-(R)
precip         (/ Column:Width)3-(R)
               {HOME}
               {formater}
               {printset}
               {kill}

kill          {CHOOSE}c
              {D}
              {DEL}Y
              {U 2}

formater      {HOME}{MakeChar1} stn
              {HOME}{R}{AddZeroDate}      day
              {HOME}{R 2}{AddZeroDate}    month
              {HOME}{R 3}{MakeChar}       year
              {HOME}{R 4}{AddZeroDate}    LCL
              {HOME}{R 5}{AddZeroDate}    UTC
              {HOME}{R 7}{AddZeroDate}    wx
              {HOME}{R 8}{MakeChar}       stn
              {HOME}{R 9}{MakeChar}
              {HOME}{R 10}{MakeChar}
              {HOME}{R 11}{MakeChar}
              {HOME}{R 12}{MakeChar}
              {HOME}{R 13}{MakeChar}
              {HOME}{R 14}{MakeChar}
              {HOME}{R 15}{MakeChar}
              {HOME}{R 16}{MakeChar}
              {HOME}{R 17}{MakeChar}
              {HOME}{R 18}{MakeChar}
              {HOME}{R 19}{MakeChar}
              {HOME}{R 20}{MakeChar}
```

```

(HOME){R 21}{MakeChar}
(HOME){R 22}{MakeChar}
(HOME){R 23}{MakeChar}
(HOME){R 24}{MakeChar}
(HOME){R 25}{MakeChar}
(HOME){R 26}{MakeChar}
(HOME){R 27}{MakeChar}
(HOME){R 28}{MakeChar}
(HOME){R 29}{MakeChar}

```

MakeChar routine converts numeric to character.
RowTotal provides a tally of rows.

```
MakeChar *{D}{FOR CharCount,1,RowTotal,1,goChar}*

```

```
goChar {DOWN}
*{IF @CELLPOINTER(**type**)=**b**}{RETURN}*
*{IF @CELLPOINTER(**type**)=**v**}{EDIT}{HOME}'-'*

```

AddZeroDate duplicate of add zero except it exits the loop
when it runs out of rows: provides RowTotal

```
AddZeroDate *{D}{FOR CharCount,1,RowTotal,1,goZeroDate}*

```

```
goZeroDate {DOWN} CelVal
*{IF @CELLPOINTER(**type**)=**b**}{RETURN}*
*{IF @CELLPOINTER(**type**)=**v**}{EDIT}{HOME}'-'*
*{PUT CelVal,0,0,@MID(CELLPOINTER(**contents**),1,1)}*
*{IF @CELL(**contents**,CelVal)=**#AND#@CELLPOINTER(**contents**)<>***}{EDIT}{HOME}{R}0-*

```

```
CharCount 147
RowTotal 146

```

MakeChar1 routine converts numeric to character.
RowTotal provides a tally of rows.
Run on first iteration for station ID to get total # obs

```
MakeChar1 *{D}{FOR CharCount,1,2000,1,goChar1}*

```

```
goChar1 {DOWN}
*{IF @CELLPOINTER(**type**)=**b**}{FORBREAK}*
*{IF @CELLPOINTER(**type**)=**v**}{EDIT}{HOME}'-'*
*{PUT RowTotal,0,0,CharCount}*

```

CpMon: Lookup table below,

CpMon routine uses lookup to convert month # to month alpha values SASfmt single comma means
CpMon *{D}{FOR CharCount,1,RowTotal,1,goCpMon}* destination is wherever cursor is

```
goCpMon *{D}{RECALC @CELLPOINTER(**address**)}{BlockValues SASfmt:014,}*
*VALUE! <formula here

```

ADDCOLON converts day numbers into proper format

```
AddColon *{FOR CharCount,1,RowTotal,1,goColon}*

```

```
{DOWN} CelVal 1
*{IF @CELLPOINTER(**type**)=**b**}{RETURN}*
*{IF @CELLPOINTER(**type**)=**v**}{EDIT}{HOME}'-'*
*{PUT CelVal,0,0,@MID(CELLPOINTER(**contents**),1,1)}*
*{IF @CELLPOINTER(**type**)=**1**#@AND#@CELLPOINTER(**contents**)<>***}{EDIT}:-*
*{IF @CELL(**contents**,CelVal)=**#AND#@CELLPOINTER(**contents**)<>***}{EDIT}{HOME}{R}0-*

```

converts weather elements into character

```
*{FOR CharCount,1,RowTotal,1,goZero}*
{HOME}

```

```
goZero {DOWN} CelVal 1
*{IF @CELLPOINTER(**type**)=**b**}{RETURN}*
*{IF @CELLPOINTER(**type**)=**v**}{EDIT}{HOME}'-'*
*{PUT CelVal,0,0,@MID(CELLPOINTER(**contents**),1,1)}*
*{IF @CELL(**contents**,CelVal)=**#AND#@CELLPOINTER(**contents**)<>***}{EDIT}{HOME}{R}0-*

```


Unix Programs (korn shell, awk)

Unix program 1: titlecnt.awk, version 1

```
## titlecnt.awk
## count how many words are in the station identifier

awk '{
    count=0
    for (i=1;i<NF;i++) { if ($i != /^[0-9]/) count += 1}
    print count
}' $1 | sort -u
```

output:

```
1
2
3
4
6
```

Unix program 2: titlecnt.awk, version 2

```
## titlecnt.awk
## count how many words are in the station identifier
## and print out those stations possessing 6 words in their name

awk '{
    count=0
    for (i=1;i<NF;i++) { if ($i != /^[0-9]/) count += 1}
    if (count == 6) print $0 # This prints out which stations have 6 words
    print count
}' $1 | sort -u
```

output:

```
1
2
3
4
6
MER DE GLACE (AGASSIZ ICE CAP) 240K6HR 116
```

Unix program 3: under_score.awk

```
### under_score.awk <filename> inserts an underscore character
### between the first two non-numeric fields,
### the assumption being that they are identifiers.
```

```
awk 'BEGIN {counter=0}
{
  if ($2 ~ /^[0-9]/) {
    printf $1
    for (i=2; i<=NF; ++i) printf " %s" $i
    printf "\n"
  }

  if ($2 !~ /^[0-9]/) {
    printf $1"_"$2
    for (i=3; i <= NF; ++i) printf " %s" $i
    printf "\n"
    counter += 1
  }
}
END {# print a count of the number of conversions, so we know when to stop!
printf counter" changes made\n" > "/dev/tty" # this directs o/p to screen
}' $1
```

Output:

```
LOWTHER_ICE_(II) 240268E 66.1 74 33 97 10
LOWTHER_ISLAND 240268N 66.2 74 33 97 29
MACKAR_INLET 2402F8E 27
MACKINSON_INLET 2402F8F 120
MALLOCK_DOME 240BFQF 2
MARSHALL_PENINSULA 240KF86 133
MARVIN_PENINSULA 240KFQF 369
MAXWELL_BAY 240B68Q 375
MCCLINTOCK_INLET 250KPH8 243
MCKINLEY_BAY 240KFQQ 358
MEIGHEN_ISLAND/ICE_CAP 2402F8R 67
MELVILLE_ICE_CAP 250K689 68
MER_DE_GLACE_(AGASSIZ_ICE_CAP) 240K6HR 116
MERCY_BAY 250K6H9 279
MIDDLE_ISLAND 2502693 324
MINTO_INLET 250K698 297
MOCKLIN_POINT 2502699 149
MOKKA_FIORD 240K699 69
MOUNT_BEAUFORT 2402702 143
MOUNT_BOMPASS 2502704 264
MUSKOX_BAY 2402712 233
```

The script prints a count of how many conversions it performed. The count is used as a guide to know when to stop. When the script performs 0 conversions, it has done them all.

Unix program 4: *under_score2.awk*

This script is an improved version of *under_score.awk*. It only needs to be run once, inserting all the underscores each station identifier needs in the first pass. Use of this program also makes *titlecnt.awk* redundant, because the user no longer needs to know when to stop – this program will take care of it. Note that it will not insert an underscore if a word in the station identifier begins with a non-alpha character.

```
### under_score2.awk <filename> inserts an underscore character
###                               between the first two non-numeric fields,
###                               the assumption being that they are identifiers.
###   this leaves all data in their original spots so that
###   date-get.awk can extract lat/long along with the start/end dates
###   (it has to use substr($0,lat.place.length) so lat/long must be at
###   a specific column.

awk '{
    stnname = ""
    printf $1
    for (i=2;i<NF;i++) {
        if ($i !~ /^[0-9]/) stnname = stnname "_" $i
        else {
            printf stnname
            start = length(stnname)
            for (x=(start+1);x<200;x++) {
                printf substr($0,x,1)
            }
            i = NF
            printf "\n"
        }
    }
} #awk end
END {# print a count of the number of conversions, so we know when to stop!
    printf counter" changes made\n" > "/dev/tty" # this directs o/p to screen
}

' $1
```

Unix program 5: *placer.awk*

placer.awk <filename> positions the second field at column 9

```
awk '{
    printf $1
    for (i=1;i<(8-length($1));++i) printf " "
    for (i=2; i<=NF; ++i) printf " "$i
    printf "\n"
}

' $1
```

Appendix B:

SAS code to perform, plot and map the principal components analyses

HumpRemove.sas

This subroutine removes the seasonal mean from all data sets to be processed in the analysis. For each day it takes the mean from all stations and then subtracts that value from each station. This routine must be called from another routine.

```
MACRO HUMP(cols);
proc transpose data=donezet out=tdoneset; run;
proc means data=tdoneset noprint;
  var coll-coll&cols;
  output out=donezmean mean(coll-coll&cols)=m1-m&cols;
run;
data donezmean;
  set donezmean;
  drop _type_ _freq_;
run;
proc transpose data=donezmean out=tdonezm; run;
data tdonezm; set tdonezm; drop _name_ coll; mean=coll; run;
proc transpose data=donezet out=stnname; run;
data stnname; set stnname; keep _name_; run;

proc iml;
  start getdata;
    use donezet;
    read all var _num_ into data;
    use tdonezm;
    read all var _num_ into means;
    c=ncol(data);
    r=row(data);
    newdat=j(r,c,0);
    do cnt=1 to c; newdat[,cnt]=data[,cnt]-means; end;
    create meandat from newdat /*[colname={x y contour}] */;
    append from newdat;
  finish;
run getdata; quit;

proc transpose data=meandat out=tmmandat; run;
data tmmandat; set tmmandat; drop _name_; run;
data meandone; merge stnname tmmandat; run;
proc transpose data=meandone out=donezstm; run;
data donezstm; set donezstm; drop _name_; run;

MEND HUMP;

/* See if it worked
options reset=all;
symbol i=j;
proc gplot data=donezstm;
  plot mean*ob; run;quit;
*/
```

cline-4.sas

I had to write this routine because, although SAS will contour data, it won't write the contour data to a datafile. Without that, there is no way to reproject the contour lines the way you want; you are stuck with SAS's contour output.

This program contours a point field arranged as a lattice. It writes the resultant coordinates of all contour line segments and the original data points to a SAS dataset. The data set can be reprojected in any way necessary. I developed this algorithm in Visual Basic 5.0. This was advantageous for two reasons: 1), VB has a very nice debugging environment, and 2), VB code is very similar to SAS/IML (interactive matrix language) code, so a finished program can be ported over with little modification.

```
/** CLINE-IM.sas to 'draw' contour lines in a regular grid **/

/** display original point field **/
macro PCASTMS;
  data origdot ;
    length function color style $8 text $20;
    set eq1.finfo1 end=last;
    xsys='2'; ysys='2'; hsys='3'; when='a'; function='label'; style='special';
```

```

text='J'; size=2; position='e'; color='black'; province='-2';
x=long*d2r*-1; y=lat*d2r;
output ;

ysys='2'; hsys='3'; /*y=-.003;*/ x=x-.008;
function='label'; style='SWISSB'; size=2;
IF trim(left(PID))='CLYDE' THEN position='1';
ELSE position='6';
color='black';
if *factor* ne *MnT* then text=left(trim(int(&factor*100)));
else text=left(trim(&factor));
province='-1';
output;

/* all the following to put underlines on AES station factor scores */
if type ne 'P' then do;
if subtr(pid,1,1) not in ('0','1','2','3','4','5','6','7','8','9') then do; */

    ysys='2'; hsys='3'; y=y-.00001;
    function='label'; style='SWISSB'; size=2;
    IF trim(left(PID))='CLYDE' THEN position='1';
    ELSE position='6';
    color='black'; province='-1';
    if *factor* ne *MnT* then chfact=left(trim(int(&factor*100)));
    else chfact=left(trim(&factor));
    if length(chfact)=4 then do;
        if subtr(chfact,1,1)='-' then text="____";
        else text="___";
        end;
    if length(chfact)=3 then do;
        if subtr(chfact,1,1)='-' then text="___";
        else text="__";
        end;
    if length(chfact)=2 then do;
        if subtr(chfact,1,1)='-' then text="__";
        else text="_";
        end;
    if length(chfact)=1 then text="_";
    output;
end;

drop pid long lat factor1 factor2 factor3 factor4 mnT;
/* reverse vertical axis for display and store some values for grid setup */
run;
tmend PCASTNS;

%MACRO contour;
PROC IML;
rows=100;
cols=100;
grid=j(rows,cols,0);
gridlat=j(rows,cols,0);
gridlong=j(rows,cols,0);

start getgrid(grid,gridlat,gridlong,rows,cols);
use pcagrid1;
read all var {&factor} into a;
grid=shape(a,rows);
read all var {lat} into b;
gridlat=shape(b,rows);
read all var {long} into c;
gridlong=shape(c,rows);
finish;

start MAINBODY(matrix,matlat,matlong,rows,cols);
/*
    LoMin=symget('loogmin');
    LoMax=symget('longmax');
    LaMin=symget('latmin');
    LaMax=symget('latmax');

    LongMin=NUN(LoMin)*-1;
    LongMax=NUN(LoMax)*-1;
    LatMin=NUN(LaMin);
    LatMax=NUN(LaMax);
*/

    LongMin=126;
    LongMax=60.66;
    LatMin=67;
    LatMax=83.83;

    gridlat=matlat;
    gridlong=matlong;
    grid=matrix;
    found1 = 0;
    found2 = 0;

    xinc = .66;
    yinc = .17 ;

    X1 = 0;
    X2 = 0;
    Y1 = 0;
    Y2 = 0;
    s1 = 0;
    s2 = 0;
    s3 = 0;

```

```

s4 = 0;
c1 = 0;
c2 = 0;
c3 = 0;
c4 = 0;
reverse=0;
db=0;
secondRn=0;

!if &factor ne MNT !then !do;
  cStart = -2.4;
  cEnd = 2.6;
  cStep = .2 ;
  !end;
!else !do;
  cStart = -10;
  cEnd = 16;
  cStep = 1;
  !end;

Do contour = cStart To cEnd by cStep;
  Do i = 1 to (rows-1);
    Do j = 1 to (cols-1);
      ULv = grid[i, j];
      URv = grid[i, j + 1];
      LRv = grid[i + 1, j + 1];
      LLv = grid[i + 1, j];

/*
' skip check - if all vertices are less than or greater than
' the contour being checked then skip the checking process
*/
If contour > ULv & contour > URv & contour > LRv & contour > LLv Then GoTo skipit;
If contour < ULv & contour < URv & contour < LRv & contour < LLv Then GoTo skipit;

/*
'double line check - if diagonal vertices bracket the contour
'then there are double contour lines. This situation must
'be dealt with individually
*/
If ((ULv < contour) & (LRv < contour) & (URv > contour) & (LLv > contour)) |
((ULv > contour) & (LRv > contour) & (URv < contour) & (LLv < contour)) Then do;
  db = 1;
  reverse = 0;
  secondRn = 1;
  End;
If ((ULv > contour) & (LRv > contour) & (URv < contour) & (LLv < contour)) |
((ULv < contour) & (LRv < contour) & (URv > contour) & (LLv > contour)) Then do;
  db = 1;
  reverse = 1;
  secondRn = 1;
  End;

xpos = gridlong[i,j];
ypos = gridlat[i,j];

```

reptRun:

```

Do side = 1 To 4;
  If reverse = 1 & db = 1 Then side = 3;

  If found1 = 0 Then do;
    If side = 1 & found1 = 0 Then do;
      If ((ULv < contour) & (URv > contour)) Then do;
        found1 = 1;
        s1 = 1;
        X1 = xpos - (((contour - ULv) / (URv - ULv)) * xinc);
        Y1 = ypos;
      End;
      If ((ULv > contour) & (URv < contour)) Then do;
        found1 = 1;
        s1 = 1;
        X1 = xpos - (((ULv - contour) / (ULv - URv)) * xinc);
        Y1 = ypos;
      End;
    End;

    If side = 2 & found1 = 0 Then do;
      If ((URv < contour) & (LRv > contour)) Then do;
        found1 = 1;
        s2 = 1;
        X1 = xpos - xinc;
        Y1 = ypos - (((contour - URv) / (LRv - URv)) * yinc);
      End;
      If ((URv > contour) & (LRv < contour)) Then do;
        found1 = 1;
        s2 = 1;
        X1 = xpos - xinc;
        Y1 = ypos - (((URv - contour) / (URv - LRv)) * yinc);
      End;
    End;

    If side = 3 & found1 = 0 Then do;
      If ((LLv < contour) & (LRv > contour)) Then do;
        found1 = 1;
        s3 = 1;
        X1 = xpos - (((contour - LLv) / (LRv - LLv)) * xinc);
        Y1 = ypos - yinc;
      End;
      If ((LLv > contour) & (LRv < contour)) Then do;
        found1 = 1;
        s3 = 1;
      End;
    End;
  End;

```

```

        X1 = xpos - (((LLv - contour) / (LLv - LRv)) * xinc);
        Y1 = ypos - yinc;
    End;
End;

If side = 4 & found1 = 0 Then do;
    If ((ULv < contour) & (LLv > contour)) Then do;
        found1 = 1;
        s4 = 1;
        X1 = xpos;
        Y1 = ypos - (((contour - ULv) / (LLv - ULv)) * yinc);
    End;
    If ((ULv > contour) & (LLv < contour)) Then do;
        found1 = 1;
        s4 = 1;
        X1 = xpos;
        Y1 = ypos - (((ULv - contour) / (ULv - LLv)) * yinc);
    End;
End;

If contour = ULv & found1 = 0 Then do;
    found1 = 1;
    c1 = 1;
    X1 = xpos;
    Y1 = ypos;
End;

If contour = URv & found1 = 0 Then do;
    found1 = 1;
    c2 = 1;
    X1 = xpos - xinc;
    Y1 = ypos;
End;

If contour = LLv & found1 = 0 Then do;
    found1 = 1;
    c3 = 1;
    X1 = xpos;
    Y1 = ypos - yinc;
End;

If contour = LRv & found1 = 0 Then do;
    found1 = 1;
    c4 = 1;
    X1 = xpos - xinc;
    Y1 = ypos - yinc;
End;

End; /* FOUND1 = 0 */

If found1 = 1 Then do;
    If contour = ULv & c1 = 0 & found2 = 0 Then do;
        found2 = 1;
        X2 = xpos;
        Y2 = ypos;
    End;
    If contour = URv & c2 = 0 & found2 = 0 Then do;
        found2 = 1;
        X2 = xpos - xinc;
        Y2 = ypos;
    End;
    If contour = LLv & c3 = 0 & found2 = 0 Then do;
        found2 = 1;
        X2 = xpos;
        Y2 = ypos - yinc;
    End;
    If contour = LRv & c4 = 0 & found2 = 0 Then do;
        found2 = 1;
        X2 = xpos - xinc;
        Y2 = ypos - yinc;
    End;
    If side = 1 & s1 = 0 & found2 = 0 Then do;
        If ((ULv < contour) & (URv > contour)) Then do;
            found2 = 1;
            X2 = xpos - (((contour - ULv) / (URv - ULv)) * xinc);
            Y2 = ypos;
        End;
        If ((ULv > contour) & (URv < contour)) Then do;
            found2 = 1;
            X2 = xpos - (((ULv - contour) / (ULv - URv)) * xinc);
            Y2 = ypos;
        End;
    End;
    If side = 2 & s2 = 0 & found2 = 0 Then do;
        If ((URv < contour) & (LRv > contour)) Then do;
            found2 = 1;
            X2 = xpos - xinc;
            Y2 = ypos - (((contour - URv) / (LRv - URv)) * yinc);
        End;
        If ((URv > contour) & (LRv < contour)) Then do;
            found2 = 1;
            X2 = xpos - xinc;
            Y2 = ypos - (((URv - contour) / (URv - LRv)) * yinc);
        End;
    End;
    If side = 3 & s3 = 0 & found2 = 0 Then do;
        If ((LLv < contour) & (LRv > contour)) Then do;
            found2 = 1;
            X2 = xpos - (((contour - LLv) / (LRv - LLv)) * xinc);
            Y2 = ypos - yinc;
        End;
    End;
End;

```

```

End;
If ((LLv > contour) & (LRv < contour)) Then do;
  found2 = 1;
  X2 = xpos - (((LLv - contour) / (LLv - LRv)) * xinc);
  Y2 = ypos - yinc;
End;
End;

If side = 4 & s4 = 0 & found2 = 0 Then do;
  If ((ULv < contour) & (LLv > contour)) Then do;
    found2 = 1;
    X2 = xpos;
    Y2 = ypos - (((contour - ULv) / (LLv - ULv)) * yinc);
  End;
  If ((ULv > contour) & (LLv < contour)) Then do;
    found2 = 1;
    X2 = xpos;
    Y2 = ypos - (((ULv - contour) / (ULv - LLv)) * yinc);
  End;
End;

End; /* FOUND1=1 */
db = 0;
End; /* SIDE */

/*
If (X1 = 0 & Y1 = 0) | (X2 = 0 & Y2 = 0) Then Goto skipPlot;
'if this if is true then the condition is a contour value equals
'a vertex and the current grid cell under investigation does not
'have any other occurrences of the contour value in it, either
'in other verticies or on any sides. The simple remedy is to just
'skip the plotting of that line. A more complex solution is to set
'more check variables. Note that the solution implemented here
'will only work if the origin (0,0) is not in the plotting space.
'that is, and therefore not a valid point for plotting.
*/

xout=xout//x1;
yout=yout//y1;
fn=fn/'move';
cont=cont//contour;

xout=xout//x2;
yout=yout//y2;
fn=fn/'draw';
cont=cont//contour;

If secondRn = 1 Then do;
  reverse = 0;
  db = 0;
  secondRn = 0;
  found1 = 0;
  found2 = 0;
  Goto reptRun;
End;

skipPlot:
  found1 = 0;
  found2 = 0;
  X1 = 0;
  X2 = 0;
  Y1 = 0;
  Y2 = 0;
  s1 = 0;
  s2 = 0;
  s3 = 0;
  s4 = 0;
  c1 = 0;
  c2 = 0;
  c3 = 0;
  c4 = 0;

skipit:
  End; /* j */
End; /* i */
End; /* contour */

/*
latrange=latmax-latmin;
longrang=longmax-longmin;

xout=(xout/100)*longrang+longmin;
yout=latmax-(yout/100)*latrange;
*/

contout=fuzz(cont);

done=xout||yout||contout;

create annoraw1 from done[colname={x y contour}];
append from done;

create annoraw2 from fn[colname={function}];
append from fn;

finish;

run getgrid(grid,gridlat,gridlong,rows,cols);
run mainbody(grid,gridlat,gridlong,rows,cols);

QUIT;

```

```

data annoraw;
    merge annoraw1 annoraw2;
run;

/*
tgridplot
*/
/**** FINAL DISPLAY ****/

data plot;
    set annoraw;
        color='black'; hsys='1'; xsys='2'; ysys='2'; size=0.1; line=1;
        when='a'; province='-3';
        x=6d2r*x;
        y=6d2r*y;
    if contour=0 then do;
        size=.7;
    end;
    else do;
        size=.3;
    end;
run;

tpeastns

data pcaplot;
    length function $8;
    set plot origdot;
run;
tend contour;
/*
goptions hsize = 6 IN;
proc GMMO anno=plot datasys; run;
*/

```

mapit.sas

This program generates the graticule, labels, coastlines and shading necessary for the basemap. It combines this information with the contouring, point and point label information and reprojects into the final form of the individual maps.

```

/**** MAPSTART.sas to plot a basemap ****/

tmacro GRATICUL;
goptions reset=all DEV=WIN target=winPRtg /*nopolygonfill nofill nopclip */;

tlet xmin=50;
tlet xmax=160;
tlet ymin=60;
tlet ymax=88;

tlet stlong=6xmin;
tlet stlat=6ymin;

tlet edlong=6xmax;
tlet edlat=6ymax;

data latlong;
    do long = 6stlong to 6edlong by 5;
        lat=6stlat;
        function='move';
        output;
        lat=6edlat;
        function='draw';
        output;
    end;

    do lat = 6stlat to 6edlat by 5;
        do long = 6stlong to 6edlong by .5;
            if long = 6stlong then function='move';
            else function='draw';
            output;
        end;
    end;
run;

data grid;
    length color $8;
    set latlong;
    retain xsys ysys '2' hsys '3' color 'black' /*line 2*/ line 1 size .05 /* when 'a'*/ province '-1' ;
    x=long*6d2r;
    y=lat*6d2r;
run;

data gratlabl;
    length function style $8 text $20;
    retain province '94'; angle=-12;

    /* latitude labels */

```

```

x=124*d2r; y=70*d2r; position='0';
style='SWISSB'; text='70'; output;
x.; y.;
style='special'; text='0'; output;

x=124*d2r; y=75*d2r; position='0';
style='SWISSB'; text='75'; output;
x.; y.;
style='special'; text='0'; output;

x=124*d2r; y=80*d2r; position='0';
style='SWISS'; text='80'; output;
x.; y.;
style='special'; text='0'; output;

x=124*d2r; y=85*d2r; position='0';
style='SWISS'; text='85'; output;
x.; y.;
style='special'; text='0'; output;

/* longitude labels */
x=131*d2r; y=84.5*d2r; position='0';
style='SWISS'; text='130'; angle='-20';output;
x.; y.;
style='special'; text='0'; output;

x=121*d2r; y=84.5*d2r; position='0';
style='SWISS'; text='120'; angle='-10';output;
x.; y.;
style='special'; text='0'; output;

x=111*d2r; y=84.5*d2r; position='0';
style='SWISS'; text='110'; angle='0';output;
x.; y.;
style='special'; text='0'; output;

x=101*d2r; y=84.5*d2r; position='0';
style='SWISS'; text='100'; angle='10';output;
x.; y.;
style='special'; text='0'; output;

x= 91*d2r; y=84.5*d2r; position='0';
style='SWISS'; text='90'; angle='20';output;
x.; y.;
style='special'; text='0'; output;

x= 81*d2r; y=84.5*d2r; position='0';
style='SWISS'; text='80'; angle='30';output;
x.; y.;
style='special'; text='0'; output;

x= 71*d2r; y=83.8*d2r; position='0';
style='SWISS'; text='70'; angle='40';output;
x.; y.;
style='special'; text='0'; output;

run;
twend GRATICUL;

tmacro frame;

data frame;
length function color style $8 text $20;
retain xsys ysys '2' hsys '3' color 'black' line 1 style 'empty' size .4 when 'a'
province '-5';
function='move'; x=&xmincon; y=&ymincon; output;
function='label'; style='special'; text='J'; position='.'; output;
function='draw'; x=&xmincon; y=&ymaxcon; output;
function='label'; style='special'; text='J'; position='.'; output;
function='draw'; x=&xmaxcon; y=&ymaxcon; output;
function='label'; style='special'; text='J'; position='.'; output;
function='draw'; x=&xmaxcon; y=&ymincon; output;
function='label'; style='special'; text='J'; position='.'; output;
function='draw'; x=&xmincon; y=&ymincon; output;

run;

twend frame;

tmacro basenaps;
data canarea;
set maps.canada4(where=(province in ('60' '61') and density < 6));
myid=1;

run;

data GRNarea;
set maps.worldmap(where=(id = 405));
province='98';
myid=2;

run;

data ALarea;
set maps.states(where=(state = 2 and density < 6));
province='99';
myid=3;

run;

data colors;

```

```

myid=1;colors=1;output;
myid=2;colors=2;output;
myid=3;colors=2;output;
/* color list is specified during final GREPLAY output in a GOPTIONS statement */
run;
%mend basemaps;

%macro mapit;

data gridfeed;
  set sql.finloc1;
  long=-1*long;
run;

PROC G3GRID data=gridfeed out=pcagrid1;
  grid long*lat=factor /

                               spline smooth=.0001
                               axis1=126 to 60.66 by -.66 /*120 to 60.6 by -.6*/
                               axis2=83.83 to 67 by -.17 /*83.86 to 70 by -.14*/;
run;

data _null_;
  set pcagrid1;
  file "d:\atkinson\raobdata\junkgrid.ieg";
  put ffactor;
run;

%contour

data all;
  length function $8;
  set pcpplot canarea GRMarea grid gratlab1;
  if xsys='1' then xsys='2';
  if ysys='1' then ysys='2';
  drop factor1 factor2 factor3 factor4 factor5 density long lat;
  output;
run;

PROC GPROJECT data=all out=setone
  project=gnomon
  polelong=110;
  id province;
run;

PROC means data=setone min max noprint;
  var x y;
  output out=stats min=minx miny max=maxx maxy;
run;

data _null_;
  set stats;
  xrange=maxx-minx;
  yrange=maxy-miny;

  newmaxx=maxx*(.10*xrange);
  newminx=minx*(.14*xrange);

  newmaxy=maxy*(.17*yrange);
  newminy=miny*(.30*yrange);

  call symput('xmincon',newminx);
  call symput('xmaxcon',newmaxx);
  call symput('ymincon',newminy);
  call symput('ymaxcon',newmaxy);
run;

/* The following is to correct a problem whereby the clipping PROC GPROJECT
for some unknown reason chops off the first MOVE command in
the annotate dataset. To remedy this the first MOVE command
observation is stored in dataset DUMB, the entire dataset
is clipped (where the first value gets chopped off), and the
chopped value is then re-added after the clipping. If the
first MOVE command is missing ANNOTATE starts the line at the
procedure area origin in the lower left corner, connecting the
origin with the first contour vertex, which results in a contour
line moving to a point outside the map box frame.
*/
data dumb;
  set setone;
  if _n_=1;
run;

PROC GPROJECT data=setone out=set2
  latmax=&yymaxcon
  latmin=&ymincon
  longmin=&xmincon
  longmax=&xmaxcon
  project=NONE
  ;
  id province;
run;

/* There is a problem whereby the last contour move/draw does not have
a completeing draw. This causes a line to protrude beyond the border
of the map. The following searches through the contour DS and removes
this occurrence (seems to work okay now).
*/

```

```

data set3;
  set set2 end=last;
  ahead_n_1;
  oldfunc=function;
  if last then stop;

  set set2 point=ahead;
  newfunc=function;
  if oldfunc="move" and not (newfunc = "move" OR newfunc = "draw") then return;
  else output;

run;

data settwo;
  set dumb set3 /*set2*/;

run;

data mapproj annoproj; /* peel out annotate items (graticule, points) */
  set settwo;
  if province in ('-1' '-2' '-3' '-4' '-5' '93' '94' '95') then output annoproj;
  else output mapproj;

run;

%frame

data annc;
  length function color style $8 text $20;
  set annoproj /*points */;

  if province in ('-1' '-2' '-3' '-4' '-5') then do;
    output;
  end;

  if province = '94' then do;
    xsys='2'; ysys='2'; hsys='3'; when='a';
    function='label'; size=1; color='black';
    output;
  end;

run;

data maint1;
  length stmoval edmoval $9 text $20;
  stdate=ststartm; eddate=sendtm; fact="&factor";
  stmonth=month(stdate);
  edmonth=month(eddate);
  if stmonth=1 then stmoval="Jan";
  if stmonth=2 then stmoval="Feb";
  if stmonth=3 then stmoval="Mar";
  if stmonth=4 then stmoval="Apr";
  if stmonth=5 then stmoval="May";
  if stmonth=6 then stmoval="Jun";
  if stmonth=7 then stmoval="Jul";
  if stmonth=8 then stmoval="Aug";
  if stmonth=9 then stmoval="Sep";
  if stmonth=10 then stmoval="Oct";
  if stmonth=11 then stmoval="Nov";
  if stmonth=12 then stmoval="Dec";
  if edmonth=1 then edmoval="Jan";
  if edmonth=2 then edmoval="Feb";
  if edmonth=3 then edmoval="Mar";
  if edmonth=4 then edmoval="Apr";
  if edmonth=5 then edmoval="May";
  if edmonth=6 then edmoval="Jun";
  if edmonth=7 then edmoval="Jul";
  if edmonth=8 then edmoval="Aug";
  if edmonth=9 then edmoval="Sep";
  if edmonth=10 then edmoval="Oct";
  if edmonth=11 then edmoval="Nov";
  if edmonth=12 then edmoval="Dec";
  retain function 'label' style 'SWISS' size 2.5 xsys ysys '5' hsys '3' color 'black' when 'a' position '>';
  text='Start: '||trim(stmoval)||' '||trim(left(day(stdate)))||' '||trim(left(year(stdate)));
  y=92; x=70; output;
  text='End: '||trim(edmoval)||' '||trim(left(day(eddate)))||' '||trim(left(year(eddate)));
  y=89; x=70; output;
  if &factor = 'MNT' then text="Mean temperature";
  else if &factor = 'factor1' then text="Component One";
  else if &factor = 'factor2' then text="Component Two";
  else if &factor = 'factor3' then text="Component Three";
  else if &factor = 'factor5' then text="Component Four";
  else text="Component Five";
  y=85; x=70; output;
  drop stdate eddate fact stmonth stmoval edmonth edmoval;

run;

data plotper;
  length fact $2 style color $8 text $20 ;
  set percent;
  fact=trim(left(&factor));
  if &factor=FACTOR5 then fact=trim(left(factor4));
  if &factor=FACTOR6 then fact=trim(left(factor5));
  numstn=trim(left(stotlstns));
  retain function 'label' style 'SWISS' size 2 xsys ysys '5' hsys '3' color 'black' when 'a' position '>';
  if &factor ne 'MNT' then do;
    text='Explained var: '||fact||'%'; y=82; x=70; output;
  end;
  text='Stations used: '||numstn; y=80; x=70; output;
  if &factor ne 'MNT' then do;

```

```

        text='Contour interval: 20'; y=77; x=70; output;
        text='heavy line = 0'; y=75; x=70; output;
    end;
else do;
    text='Contour interval: 1'; y=77; x=70; position='0'; output;
    style='special'; text='0'; y=.; x=.; output;
    style='SWISS'; text=' C'; y=.; x=.; output;
end;
drop factor1 factor2 factor3 MnT factor4 factors5 pid fact;

/* Figure label - only for final printing */

if (%letters='YES' AND %factor = 'factor1') then do;
    function='move'; color='white'; x=8; y=85; output;
    function='bar'; x=16; y=93; output;
    function='label'; x=10; y=98; text='A'; xsys='3'; ysys='3';
        position='9'; style='swissb'; color='black'; size=6; output;
end;
if (%letters='YES' AND %factor = 'factor2') then do;
    function='move'; color='white'; x=8; y=85; output;
    function='bar'; x=16; y=93; output;
    function='label'; x=10; y=98; text='C'; xsys='3'; ysys='3';
        position='9'; style='swissb'; color='black'; size=6; output;
end;
if (%letters='YES' AND %factor = 'factor3') then do;
    function='move'; color='white'; x=8; y=85; output;
    function='bar'; x=16; y=93; output;
    function='label'; x=10; y=98; text='E'; xsys='3'; ysys='3';
        position='9'; style='swissb'; color='black'; size=6; output;
end;
if (%letters='YES' AND %factor = 'factor4') then do;
    function='move'; color='white'; x=8; y=85; output;
    function='bar'; x=16; y=93; output;
    function='label'; x=10; y=98; text='G'; xsys='3'; ysys='3';
        position='9'; style='swissb'; color='black'; size=6; output;
end;
if (%letters='YES' AND %factor = 'factor5') then do;
    function='move'; color='white'; x=8; y=85; output;
    function='bar'; x=16; y=93; output;
    function='label'; x=10; y=98; text='I'; xsys='3'; ysys='3';
        position='9'; style='swissb'; color='black'; size=6; output;
end;

if (%letters='YES' AND %factor = 'MnT') then do;
    function='move'; color='white'; x=8; y=85; output;
    function='bar'; x=16; y=93; output;
    function='label'; x=10; y=98; text='K'; xsys='3'; ysys='3';
        position='9'; style='swissb'; color='black'; size=6; output;
end;

/* box around the title section to mask the contour lines */
function='move'; color='GRAYF9'; x=69; y=73; output;
function='bar'; x=93; y=90; output;

run;

data annoall;
    set anno frame maintit1 plotper;
run;

options hsize=3.5 in vsize=3.5 in colors=(grayE9 grayF9 grayF9) noprompt;

pattern1 v=solid;
pattern2 v=solid;
pattern3 v=solid;

proc GMAP map=mapproj data=colors anno=annoall
    %if %factor ne MnT %then GOUT=plots.plots;
    %else GOUT=plots.plots; /* change to plots.means if it needs to be permanent */
    ;
    id wmid;

    choro colors / %if %factor ne MnT %then name=%&pltname%;
        %else name=%&pltname%;
        nolegend
        coutline=black
        ;

run;
quit;
%mend mapit;

/*
data points;
    length name $ 20 province $ 2;
    input name $ & latdeg latmin longdeg longmin province ;
    x=&d2r*(longdeg-longmin/60);
    y=&d2r*(latdeg-latmin/60);
    keep name x y province;
cards;
Eureka            80 00 85 56 95
Resolute Bay     74 43 94 59 95
Alert            82 20 62 30 95
Mould Bay        76 14 119 20 95
Cambridge Bay   69 07 105 01 95
Isachsen         78 47 103 32 95
Iqaluit          63 45 68 33 95

```

```

Hall Beach      68 47 81 15 95
Barter Island   68 00 140 00 95
Barrow          68 00 150 00 95
Sachs Harbour   71 59 125 17 93
Pond Inlet      72 40 78 00 93
Rea Point       75 21 105 43 93
;
run;
*/
/*
%graticul
%basemaps
%aspic
*/

```

running-mean.sas

This is a function that will determine a running mean. SAS does not have a readily accessible running mean function and I found their programmed solution to be limited. I made this one which can handle any number of input columns and it can be told not to process certain columns, such as a date or label field; the bandwidth can also be user specified, as can a weighting option.

```

/** Smoothing option **/
/** pass in parameters: (original DS, smoothlevel, output DS, weight option, name of var to skip) **/
/** where original DS has one or more columns of numeric data
    smoothlevel is 3 5 7 9 11 etc and is the number of points used to compute the local mean
    output DS is the output dataset; can be the same name as original for replacement
    weight option is 0 1 2 3 etc and determines the magnitude of the weighting
    name of var to skip is the name of a variable column to NOT apply smoothing to
    (eg used to skip a data variable, for example
**/

macro smooth(origDS,smlevel,outDS,weight,keeplist);

    data cutDS; set sorigDS; keep skeeplist; run;

    proc transpose data=cutDS out=donetran; run;
    data donetran; set donetran; stn= name; stnno= n_; keep stn stnno; run;
    proc univariate data=donetran noprint; output out=count n=n; run;
    proc univariate data=sorigDS noprint; output out=count2 n=n; run;
    data _null_; set count; call symput('numstns',trim(left(n)));
        set count2; call symput('numobs',trim(left(n)));
        call symput('stnlist',' ');
    run;

    %do stnct=1 %to &numstns;
        data _null_; set donetran; if _n_=%stnct then call symput('curstn',stn);run;

        data runsm&stnct;
            array vals(&numobs);
            retain vals;
            set sorigDS end=last;
            vals[_n_]=&curstn;

            if last then do;
                do i=1 to &numobs;
                    keep &curstn;
                    %let inc=0;
                    %let maxinc=%eval(&smlevel/2);

                    %do smcount=1 %to %eval(&smlevel-1) %by 2;
                        %let inc=%eval(&inc+1);
                        %let wtdiv=%eval(&smlevel+1-(&maxinc-1)*%weight);
                        %if %weight ne 0 %then %do;
                            %let wtval=%eval(&weight*%eval(&maxinc-&inc));
                            %let mainwt=%eval(&weight*%maxinc+1);
                            %end;
                        %else %do;
                            %let wtval=1;
                            %let mainwt=1;
                            %end;
                        %if %wtval=0 %then %let wtval=1;

                        %if %smcount=1 %then %do;
                            %let exclude=&inc &numobs;
                            %let curelms=vals(i)*%mainwt+vals(i-1)*%wtval+vals(i+1)*%wtval;
                            %end;
                        %else %do;
                            %let exclude=%exclude &inc %eval(&numobs-&eval(&inc-1));
                            %let curelms=%curelms+vals(i-&inc)*%wtval+vals(i-&inc)*%wtval;
                            %end;
                    %end;

                    %let curstn;
                %end;

                if &smlevel=3 then do;
                    if i in (1 &numobs) then do;

```

```

        &curstn=.;
        output;
        end;
    else do;
        &curstn=(vals{i-1}*(vals{i}*3)+vals{i+1})/5;
        output;
    end;
end;

if &smlevel=5 then do;
    if i in (1 2 &numobs &eval(&numobs-1)) then do;
        &curstn=.;
        output;
        end;
    else do;
        &curstn=(vals{i-2}*(vals{i-1}*1)+(vals{i}*1)+(vals{i+1}*1)+vals{i+2})/5;
        output;
    end;
end;

if &smlevel=13 then do;
    if i in (1 2 3 4 5 6 &numobs &eval(&numobs-1) &eval(&numobs-2)
&eval(&numobs-3) &eval(&numobs-4) &eval(&numobs-5)) then do;
        &curstn=.;
        output;
        end;
    else do;
        &curstn=(vals{i-6}+vals{i-5}+vals{i-4}+vals{i-3}+vals{i-2}+vals{i-1}+vals{i}+vals{i+1}+vals{i+2}+vals{i+3}
+vals{i+4}+vals{i+5}+vals{i+6})/13;
        output;
    end;
end;

end;
run;

%if &stnct = 1 %then %let stnlist = runm&stnct;
%else %let stnlist = &stnlist runm&stnct; /* build up names for a merge statement */
%end; /* stnct do loop */

data stopgap; merge &stnlist; /* dt=_n_ */ run;

data &outDS;
set stopgap;
if _n_=1 then do;
    moncount=1;
    fullcnt=0;
end;
fullcnt=fullcnt+1;
dt=fullcnt;
moncount=moncount+1;
if moncount=31 then do;
    do gap=1 to 4;
        PRIN1=.; PRIN2=.; PRIN3=.; PRIN4=.; PRIN5=.;
        fullcnt=fullcnt+1;
        dt=fullcnt;
        output;
    end;
    moncount=1;
end;
output;
drop gap moncount fullcnt;
retain moncount fullcnt dt;
run;

%end smooth;
/*
%smooth(pcatest2,13,pcatest3,2,PRIN1 PRIN2 PRIN3 PRIN4 PRIN5)
*/

```

P-pcomp2.sas

This is the main program that calls the others and which draws their output together. In here are produced the component score time series plots; here as well the actual PCA is performed. All plots and maps are assembled for final display. There are a number of options that can be set, including how many pages of output, how many component results are displayed, mean temperature plot display, output may be directed to screen, file or printer, whether or not to remove the seasonal trend and to perform running means, whether or not to perform component rotation and which type, whether to confine an analysis to AES data, and, of course, over what time period to perform the analysis.

```

/**** program P-pcomp.sas performs the principal components analysis
for existing data during the time period selected and prepares
the output for mapping.

```

```

This program is called from program P-overall.sas. In there is
specified over which years the program is to be run.

```

```

****/

%macro HUGE;

%let figname=ASM&FIGNO;

%let AESSONLY= YES; /* limit stations to AES stations */
%let minobs = 90; /* minimum # of observations a station must have */
%let smoothck=NO; /* toggle to smooth data (running mean of period SMOOTHLVL) before PCA */
%let smoothlvl=5;
%let normPCA=YES; /* toggle if a full 2-sheet PCA result is desired LEAVE AT YES */
%let onepage=NO; /* toggle YES if only the first page is desired */
%let letters=YES; /* toggle YES to turn on a b c d e f labels on plots (for final pub only) */
%let origtemp=YES; /* toggle to plot original temperature time series for all stations */
%let killhump=NO; /* toggle to remove seasonal trend */
%let limfact=NO; /* toggle to limit factors to one or two */
%let numfact=1; /* limit factors to value shown */
%let fileout=FIL; /* FIL for output to file, SCN to screen, PRT to send to printer */
%let type=PNR; /* PNR for PCA, no rotation
                PKV for PCA, rotation Varimax rotation - try now with 3 scores retained
                PNR for common factor, no rotation
                FRV for common factor, Varimax rotation
                */
%let retain=5; /* number of factors to retain for rotation */
%let rotype=varimax; /* specify rotation type: varimax, equamax, orthomax, parsimax, promax, quartimax */

/**** Sub-Domain stability testing - specify lat/long belt to remove stations beyond ****/
%let SDTEST=YES; /* toggle YES to run sub-domain testing */
%let SDlimit=long>-95; /* state the entire condition to RETAIN points */
%let SDmaplim=long>-100; /* state the map limiting condition (beyond data boundaries) */

options pagesize=60 linesize=80;

%global starttm endtm d2r xmin xmax ymin ymax factor MONTHPT YEARPT MONTHSVL MONTHVL CLYDETAG;

%let d2r=atan(1)/45;
%let idtype = pcpid;

libname pcp 'd:\atkinson\sas\data\'; run;
libname plots 'e:\sastemp\'; run;
libname sql 'd:\atkinson\sas\data\'; run;

data _null_;
    length monthsvl monthvl $9;
    yearpt=year(&starttm);monthpt=month(&starttm);monthpte=month(&endtm);
        if monthpt=1 then monthsvl="Jan";
        if monthpt=2 then monthsvl="Feb";
        if monthpt=3 then monthsvl="Mar";
        if monthpt=4 then monthsvl="Apr";
        if monthpt=5 then monthsvl="May";
        if monthpt=6 then monthsvl="Jun";
        if monthpt=7 then monthsvl="Jul";
        if monthpt=8 then monthsvl="Aug";
        if monthpt=9 then monthsvl="Sep";
        if monthpt=10 then monthsvl="Oct";
        if monthpt=11 then monthsvl="Nov";
        if monthpt=12 then monthsvl="Dec";

        if monthpte=1 then monthvl="Jan";
        if monthpte=2 then monthvl="Feb";
        if monthpte=3 then monthvl="Mar";
        if monthpte=4 then monthvl="Apr";
        if monthpte=5 then monthvl="May";
        if monthpte=6 then monthvl="Jun";
        if monthpte=7 then monthvl="Jul";
        if monthpte=8 then monthvl="Aug";
        if monthpte=9 then monthvl="Sep";
        if monthpte=10 then monthvl="Oct";
        if monthpte=11 then monthvl="Nov";
        if monthpte=12 then monthvl="Dec";

        call symput('monthpt',trim(left(monthpt)));call symput('yearpt',trim(left(yearpt)));
        call symput('monthsvl',trim(left(monthsvl)));call symput('monthvl',trim(left(monthvl)));
run;

Data basetime;
DO date = &starttm TO &endtm;
output;
END;
RETURN; run;

DATA _NULL_; /* determine and assign identifier type */
if year(&starttm) < 1984 OR year(&starttm) = 1987 OR year(&starttm) = 1988 then call symput('idtype','pcpid');
if year(&starttm) < 1987 and year(&starttm) >1983 then call symput('idtype','aesid');
if year(&starttm) > 1988 then call symput('idtype','charid');
run;

DATA baseda;
set pcp.done;
/* format &idtype $7.; */
keep &idtype dbtemp edate;
edate = datepart(date);
if datepart(date) ge &starttm and datepart(date) le &endtm then output;
run;

```

```

if &AESONLY eq YES &THEN &DO:
  /* This empties the DS baseds - it must exist in an empty state */
  DATA baseds;
  set baseds;
  if dbtemp > 1000 then output;
  keep &idtype dbtemp sdate;
run;
&END;

&macro chards;
&put &idtype;
&if &idtype = &str(charid) &then
&do;
  DATA baseds2; /* only executed if CHARID is stn id type */
  set baseds;
  do i = 1 to 7;
    if (substr(&idtype,i,1)=' ' or (substr(&idtype,i,1)='*') then substr(&idtype,i,1)='_';
    if (substr(&idtype,i,1)='?') then substr(&idtype,i,1)='X';
  end;
  thestn = substr(&idtype,1,6);
  drop &idtype;
return;
run;
&end;
&else
&do;
  DATA baseds2;
  set baseds;
  thestn = &idtype;
  drop &idtype;
return;
run;
&end;
&mend chards;

&chards
/* this combines twice daily observations */
proc means data=baseds2 noprint maxdec=1 fw=5 nway;
class thestn sdate;
var dbtemp;
output out=basedsm mean=temp;
RUN;

data _null_; call symput('numstns',0); run;
/* Reset numstns to 0. If stnids has entries this will be reset,
otherwise it must be zero. */

/* output list of stations and retain only those stations
with > a certain number of observations */
data baseex; set basedsm; if temp ne . then output; else return; keep thestn; run;

proc freq data=baseex noprint;
tables thestn /out=basecnt;
run;

data stnids;
set basecnt end=last;
if _n_ = 1 then tally = 0;

retain tally;
currstn=thestn;
keep currstn;

if count ge &minobs then do;
output;
tally=tally+1;
end;
if last then call symput('numstns',tally);
run;

&let obno = 1;
&let count = 1;
&let nowstn = 1;

&macro dsmake;
&put &numstns;
&do &while (&count le &numstns);
  DATA _NULL_;
  set stnids;
  if _n_ = &count then do;
    mstn = left(currstn); /* necessary, otherwise
currstn is right justified
in a wide field and the macro
processor does not interpret
the result properly */
call symput('nowstn',mstn);
end;
run;

/** ORIGINAL VERSION **/
DATA DS&count;
set basedsm;
date = sdate;

```

```

format PCAnovstn 5.1;
PCAnovstn = temp;
keep date PCAnovstn;
if thestn = symget('nowstn') then output;
run;

%let count = 1 %then %let setstore = DS&count;
%else %let setstore = %setstore DS&count; /* build up names for a merge statement */

%let count = %eval(%count + 1);
%end;

%if %numstns=0 %then %let setstore=/* NO PCSP DATASETS */;
/* reel in AES data */
libname aes 'd:\atkinson\aes\data\aes\'; run;
DATA AESPCSPx;
merge basetime
  aes.resbayd(rename=(mnt-resolutT))
  aes.alertd(rename=(mnt-alertT))
  aes.eurekaD(rename=(mnt-eurekaT))
  aes.hallbd(rename=(mnt-hallbT))
  %if %yearpt lt 1976 OR (%yearpt eq 1977 AND %monthpt le 06) %then aes.clyde(rename=(mnt-clyde));
  %if %yearpt lt 1977 AND %yearpt ge 1975 %then aes.arctbd(rename=(mnt-arcticb));
  /* %if (%yearpt eq 1974 and %monthpt ge 01) %then aes.arctbd(rename=(mnt-arcticb)); */
  %if %yearpt le 1977 %then aes.isacsnd(rename=(mnt-isacsnt));
  %if (%yearpt eq 1978 and %monthpt le 06) %then aes.isacsnd(rename=(mnt-isacsnt));
  aes.mouldbd(rename=(mnt-mouldbT))
  %if %yearpt le 1985 %then aes.reaptD(rename=(mnt-reaptT));
  %if (%yearpt eq 1986 AND %monthpt le 03) %then aes.reaptD(rename=(mnt-reaptT));
  %if %yearpt ge 1976 %then aes.pondid(rename=(mnt-pondiT));
  %if %yearpt le 1992 %then aes.sheprd(rename=(mnt-sheprd));
  %if (%yearpt eq 1993 AND %monthpt le 03) %then aes.sheprd(rename=(mnt-sheprd));
  %if %yearpt le 1991 %then aes.pellyd(rename=(mnt-pellyb));
  %if (%yearpt eq 1992 and %monthpt le 05) %then aes.pellyd(rename=(mnt-pellyb));
  %if (%yearpt le 1992 and %yearpt ge 1980) %then aes.holmnd(rename=(mnt-holman));
  %if (%yearpt eq 1993 and %monthpt le 06) %then aes.holmnd(rename=(mnt-holman));
  %if (%yearpt eq 1979 and %monthpt ge 09) %then aes.holmnd(rename=(mnt-holman));
  %if %yearpt le 1992 %then aes.sachsd(rename=(mnt-sachsh));
  %if (%yearpt eq 1993 AND %monthpt le 05) %then aes.sachsd(rename=(mnt-sachsh));
  %if %yearpt eq 1975 %then aes.spencd(rename=(mnt-spencbay));
  %if (%yearpt eq 1974 AND %monthpt ge 06) %then aes.spencd(rename=(mnt-spencbay));
  %if (%yearpt eq 1976 AND %monthpt le 09) %then aes.spencd(rename=(mnt-spencbay));
  %if %yearpt le 1988 OR (%yearpt eq 1989 AND %monthpt le 07) %then aes.cambg(rename=(mnt-cambrig));
  %if (%yearpt ge 1978 AND %yearpt le 1990) %then aes.coper(rename=(mnt-copperm));
  %if (%yearpt ge 1977 AND %yearpt le 1993) AND NOT
    ((%yearpt eq 1980 AND %monthpt le 3) OR
     (%yearpt eq 1981 AND %monthpt eq 8)) %then aes.nanisv(rename=(mnt-nanisiv));
  %if %yearpt le 1990 %then aes.baker(rename=(mnt-bakerlk));
  ;

keep date resolutT alertT eurekaT mouldbT pondiT sheprd pellyb holman isacsnt reapt
sachsh arcticb hallbT spencbay copperm nanisiv bakerlk cambrig clyde;

if date ge %startm and date le %endm then output;
by date;
run;

proc means data=AESPCSPx noprint; output out=asescnt n=; run; /* use stat= to specify all vars */

proc transpose data=asescnt out=asescntt; run;

data _null_;
length keeplist $200;
set asescntt end=last;
retain keeplist;
if coll ge %minobs AND _n_ > 3 then do;
if _n_ = 4 then keeplist=trim(left(_name_));
else keeplist=trim(left(keeplist)||'|' ||trim(left(_name_)));
end;
if last then call symput('keeplist',keeplist);
run;

data AESPCSP;
set AESPCSPx;
keep date &keeplist;
run;

DATA doneaset; /* aes and pcsp stns */
merge basetime AESPCSP %setstore;
by date;
drop date;
run;

DATA doneaset2;
merge basetime AESPCSP %setstore;
by date;
run;

/* Prepare mean temperature dataset */
proc means data=doneaset maxdec=1 noprint; output out=meantemp mean=; run;

proc transpose data=meantemp out=meantt; run;

data meantemp; set meantt; drop coll _name_;
if _n_ < 3 then return;
if substr(_name_,1,2)='PC' then do;
pid=substr(_name_,3);

```

```

        type="P";
    end;
else do;
    pid_name_;
    type="A";
    end;
MnT=round(coll..1);
output; run;

%mend %make;

%make

/** Smoothing option **/
%macro smooth;
    %if %smoothchk="YES" %then %do;
        proc transpose data=doneset out=donetran; run;
        data donetran; set donetran; stn_name_; stno_n_; keep stn stno; run;
        proc univariate data=donetran noprint; output out=count n=n; run;
        proc univariate data=doneset noprint; output out=count2 n=n; run;
        data _null_; set count; call symput('numstns',n); run;
        data _null_; set count2; call symput('numobs',n);
            call symput('stnlist', ' ');run;

    %do stncnt=1 %to &numstns;
        data _null_; set donetran; if _n_=%stncnt then call symput('curstn',stn);run;

        data runm&stncnt;
            array values(&numobs);
            retain values;
            set doneset end=last;
            values(_n_)=&curstn;
            if last then do;
                do i=1 to &numobs;
                    keep &curstn;
                    if %smoothvl=3 then do;
                        if i in (1 &numobs) then do;
                            &curstn=.;
                            output;
                            end;
                        else do;
                            &curstn=(values{i-1}+(values{i}*3)+values{i+1})/5;
                            output;
                            end;
                        end;
                    if %smoothvl=5 then do;
                        if i in (1 2 &numobs %eval(&numobs-1)) then do;
                            &curstn=.;
                            output;
                            end;
                        else do;
                            &curstn=(values{i-2}+(values{i-1}*3)+(values{i}*5)+(values{i+1}*3)+values{i+2})/13;
                            output;
                            end;
                        end;
                    end;
                end;
            end;
        run;

        %if &stncnt = 1 %then %let stnlist = runm&stncnt;
        %else %let stnlist = &stnlist runm&stncnt; /* build up names for a merge statement */

    %mend; /* stncnt do loop */

    data doneset; merge &stnlist;run;

%mend;
%mend smooth;
%smooth

/* seasonal curve removal */
%macro HUMPCHK;
    %if %KILLHUMP=YES %then %do;

        proc univariate data=doneset noprint; output out=count2 n=n; run;

        data _null_; set count2; call symput('numobs',n);run;

        %HUMP(&numobs)

    %mend;
%mend HUMPCHK;
%HUMPCHK

%macro FACTYPE;
%put ***** &TYPE;
    %if &TYPE=PWR %then %str(proc factor data=doneset method=principal score outstat=score rotate=none mineigen=0;run;);
    %if &TYPE=PRV %then %str(proc factor data=doneset method=principal score outstat=score rotate=&rotype n=&retain;run;);
    %if &TYPE=PWR %then %str(proc factor data=doneset method=principal score outstat=score rotate=none mineigen=0
priors=smc;run;);
    %if &TYPE=PRV %then
        %str(proc factor data=doneset method=principal score outstat=score rotate=varimax mineigen=0 priors=smc;run;);
%mend FACTYPE;

```

```

%FACTYPE

proc score data=doneiset score=score out=pcatest; run;

data score2;
  set score;
  if _TYPE_="PATTERN" then output;
  if _TYPE_="EIGENVAL" then do;
    _NAME_="EIGENVAL"; output;
  end;
  format _all_ 6.3 -r;
run;

proc transpose data=score2 out=sql.score3; run;

proc datasets library=sql;
  modify score3;
  rename _name_ protopid;
run; quit;

proc univariate data=sql.score3 noprint; output out=totlstns non; run;

data _null_;
  set totlstns;
  call symput('totlstns',n);
run;

%macro POSN;
  /* extract a lat/long subset from fulstns, convert identifier to character */
data sql.locsub;
  set sql.fulstn;
  keep pid lat long;
  if year = %yearpt then do;
/*
    pid=trim(left(pcspid));
*/
    if year(&starttm) < 1984 or year(&starttm) = 1987 or year(&starttm) = 1988 then pid=trim(left(pcspid));
    else if year(&starttm) < 1987 and year(&starttm) >1983 then pid=trim(left(substr(aesid,4)));
    else if year(&starttm) > 1988 then do;
      pid=upcase(trim(left(substr(stnid,1,6))));
      do i = 1 to 6;
        if (substr(pid,i,1)=' ') or (substr(pid,i,1)='') then substr(pid,i,1)='_';
        if (substr(pid,i,1)='?') then substr(pid,i,1)='X';
      end;
    end;
    output;
  end;
run;

data sql.score4;
  set sql.score3;
  if substr(protopid,1,2)="PC" then do;
    pid=substr(protopid,3);
    type="P";
  end;
  else type="A";
  percent=round(eigenval/&totlstns*100.1);
  drop eigenval protopid;
run;

data percent;
  set sql.score4;
  keep percent;
run;
/* FOR EXPLAINED VARIANCE VALUES OUTPUT ONLY */
proc print data=percent; run;

data sql.score4; set sql.score4; drop percent; run;

proc sort data=percent; by descending percent;run;

data perstrip; set percent; if _n_<6 then output; run;
proc transpose data=perstrip out=percent;run;
proc datasets ;
  modify percent;
  rename _name_ pid col1=factor1 col2=factor2 col3=factor3 col4=factor4 col5=factor5;
run; quit;

proc sql noprint;
  create table sql.PCALocs as
  select a.pid, a.lat, a.long from sql.locsub a, sql.score4 b
  where b.pid = a.pid and a.pid ne "" ;
quit;

proc sql noprint;
  create table sql.AESlocs as
  select b.protopid, a.lat, a.long from sql.AESlocns a, sql.score3 b
  where upcase(substr(a.station,1,3)) = substr(b.protopid,1,3) ;
quit;

proc sort data=sql.PCALocs; by pid; run;
proc sort data=sql.score4; by pid; run;
proc sort data=sql.AESlocs; by protopid; run;
proc sort data=sql.score3; by protopid; run;

data sql.finloc1;
  merge sql.PCALocs (in=a) sql.score4 (in=b);
  by pid;
  if a and b;
run;

```

```

data sql.finloc2;
  merge sql.AESlocs (in=a) sql.score3 (in=b);
  by protopid;
  if a and b;
run;

%let CLYDETAG=N;

data _null_; /* this sets a tag to better space CLYDE's loading value */
  set sql.finloc2;
  if PID="CLYDE" then call symput('CLYDETAG',Y);
run;

proc datasets library=sql;
  modify finloc2;
  rename protopid=pid;
run;
quit;

proc append base=sql.finloc1 new=sql.finloc2 force; run;

proc sort data=meantemp out=meantemp; by pid; run;
proc sort data=sql.finloc1 out=sql.finloc1; by pid; run;

data sql.finloc1;
  merge sql.finloc1 meantemp;
  by pid; run;

proc sort data=sql.finloc1; by pid descending type; run;
proc sort data=sql.finloc1 nodupkey; by pid ; run;

%mend POSM;
%open

%macro PPLOT;

data pctest2;
  set pctest;
  dt=_n_;
  prin1=factor1;
  prin2=factor2;
  prin3=factor3;
  prin4=factor4;
  prin5=factor5;
  run;

data p1 p2 p3 p4 p5;
  keep prin1;
  set pctest2;
  if PRIN1 then output p1;
  if PRIN2 then do;
    prin1=prin2;
    output p2;
  end;
  if PRIN3 then do;
    prin1=prin3;
    output p3;
  end;
  if PRIN4 then do;
    prin1=prin4;
    output p4;
  end;
  if PRIN5 then do;
    prin1=prin5;
    output p5;
  end;

  run;

data xntest2;
  set p1 p2 p3 p4 p5;
run;

proc univariate data=xntest2 noprint;
  var prin1;
  OUTPUT out=xnout max=max min=min;
run;

data _null_;
  set xnout;
  call symput('maxx',ceil(max));
  call symput('minn',floor(min));
  CALL SYMPUT('STDAY',trim(left(DAY(&STARTM))));
  CALL SYMPUT('ENDDAY',trim(left(DAY(&ENDTM))));
  ndays=intck('day',&startm,&endtm);
  if ndays lt 13 then intvl=1;
  else if ndays ge 13 and ndays lt 29 then intvl=2;
  else if ndays ge 29 and ndays lt 40 then intvl=3;
  else if ndays ge 40 and ndays lt 55 then intvl=4;
  else if ndays ge 55 and ndays lt 70 then intvl=5;
  else if ndays ge 70 and ndays lt 85 then intvl=6;
  else if ndays ge 85 and ndays lt 100 then intvl=7;
  else if ndays ge 100 and ndays lt 115 then intvl=8;
  else if ndays ge 115 and ndays lt 130 then intvl=9;
  else if ndays ge 130 and ndays lt 145 then intvl=10;
  else if ndays ge 145 and ndays lt 160 then intvl=11;
  else if ndays ge 160 and ndays lt 175 then intvl=12;
  else if ndays ge 175 and ndays lt 200 then intvl=13;
  else if ndays ge 200 and ndays lt 215 then intvl=14;

```

```

else if ndays ge 215 and ndays lt 230 then intvl=15;
else intvl=30;

outdays=ndays+1;
call symput('numdays',trim(left(outdays)));
call symput('interval',trim(left(intvl)));
numticks=int(ndays/intvl)+1;
call symput('numticks',left(numticks));

call symput('hlinecnt',0);
do hcnt = 0 to outdays;
  currdlay=day(&starttm+hcmt);
  if hcnt = 0 then do;
    oldlay = currdlay;
    hlinecnt=1;
  end;
  if currdlay lt oldlay then do;
    call symput('hline'||left(hlinecnt),'h=2 href='||left(hcnt+1)||' ');
    call symput('hlinecnt',hlinecnt);
    hlinecnt=hlinecnt+1;
  end;
  oldlay=currdlay;
end;

do cnt = 0 to numticks by 1;
  currdlay=day(&starttm+(cnt*intvl));
  currmn=month(&starttm+(cnt*intvl));
  if cnt=0 then oldcurrd=currdlay;
  if currmn=5 then monvl='May';
  if currmn=6 then monvl='Jun';
  if currmn=7 then monvl='Jul';
  if currmn=8 then monvl='Aug';
  if currmn=9 then monvl='Sep';

  if cnt=0 then
    call symput('tck'||trim(left(cnt+1)),
      'tick='||trim(left(cnt+1)||' j=c '||trim(left(currdlay)||' '||
      ' j=c '||trim(monvl)||' '||' j=c '||trim(left(eyearpt)||' '));
  else do;
    if currdlay lt oldcurrd then
      call symput('tck'||trim(left(cnt+1)),
        'tick='||trim(left(cnt+1)||' j=c '||trim(left(currdlay)||' '||
        ' j=c '||trim(monvl)||' '));
    else
      call symput('tck'||trim(left(cnt+1)), 'tick='||trim(left(cnt+1)||' j=c '||trim(left(currdlay)||' '));
    end;
  oldcurrd = currdlay;
  put currdlay;
end;

run;

data _null_;
  do x = 1 to &NUMTICKS;
    z=symget('tck'||left(x));
    put z;
  end;

run;
/*****/

data fig8;
  length function style color $8 text $ 14;
  /* remember without a length statement SAS automatically assigns a length = to the first
  string used. eg without a length assignment, text would be length 1 and would
  cause FACTOR to print as F and ONE as O */
  function='label'; x=10; y=99; text='B'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swissb'; color='black'; size=6; output;
  function='label'; x=10; y=90; text='Component'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
  function='label'; x=10; y=86.5; text='One scores'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

data fig9;
  length function style color $8 text $ 14;
  function='label'; x=10; y=99; text='D'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swissb'; color='black'; size=6; output;
  function='label'; x=10; y=90; text='Component'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
  function='label'; x=10; y=86.5; text='Two scores'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

data figF;
  length function style color $8 text $ 14;
  function='label'; x=10; y=99; text='F'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swissb'; color='black'; size=6; output;
  function='label'; x=10; y=90; text='Component'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
  function='label'; x=10; y=86.5; text='Three scores'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

data figH;
  length function style color $8 text $ 14;
  function='label'; x=10; y=99; text='H'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swissb'; color='black'; size=6; output;
  function='label'; x=10; y=90; text='Component'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
  function='label'; x=10; y=86.5; text='Four scores'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

```

```

    position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

data figJ;
  length function style color $8 text $ 14;
  function='label'; x=10; y=99; text='J'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swissb'; color='black'; size=6; output;
  function='label'; x=10; y=90; text='Component'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
  function='label'; x=10; y=86.5; text='Five scores'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

data figL;
  length function style color $8 text $ 14;
  function='label'; x=10; y=99; text='J'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swissb'; color='black'; size=6; output;
  function='label'; x=10; y=90; text='Component'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
  function='label'; x=10; y=86.5; text='Five scores'; xsys='3'; ysys='3'; hsys='3';
  position='9'; style='swiss'; color='black'; size=3.5; when='a'; output;
run;

options /* reset=all */ gunit=in target=ps600 dev=win hsize=3.5 in vsize = 3.5 in
noprompt goutmode=append ;
symbol1 c=black v=dot h=.05 i=join;
symbol2 c=red v=star h=.1 i=join;
symbol3 c=blue v=square h=.1 i=join;

axis1 color=black
order=(1 to &numdays by &interval)

label=(h=.13 f=swiss c=black j=c 'Date'
h=.09 f=swiss c=black j=c 'Component scores for the dates indicated.'
h=.09 f=swiss c=black j=c 'Analysis performed on daily temperature data.')
value=(f=swiss
/*
  IF &NUMTICKS LT 8 THEN h=.11;
  ELSE IF &NUMTICKS GE 8 AND &NUMTICKS LT 11 THEN h=.09;
  ELSE h=.07;
  DO CNT = 1 %TO %eval(&NUMTICKS);
  @@TCK&CNT
  @END;
)
*/
value=(f=swiss h=.05
  DO CNT = &styr %TO %endyr;
  @@TCK&CNT
  @put @@TCK&CNT ;
  @END;
)
/*
minor=none
;

/*
axis1 color=black
order=(1 to &numdays by &interval)
label=(h=.13 f=swiss c=black j=c 'Date')
value=(f=swiss
  IF &NUMTICKS LT 8 THEN h=.11;
  ELSE IF &NUMTICKS GE 8 AND &NUMTICKS LT 11 THEN h=.09;
  ELSE h=.07;
  DO CNT = 1 %TO %eval(&NUMTICKS);
  @@TCK&CNT
  @END;
)
minor=none
;

/*
axis2 color=black
/*
label=(h=.13 f=swiss c=black j=r 'Component' j=r 'One') */
value=(h=.1 f=swiss)
length=1.5 in
order=(&minn to &maxx by 1)
minor=none
;

axis3 color=black
/*
label=(h=.13 f=swiss c=black j=r 'Component' j=r 'Two') */
value=(h=.1 f=swiss)
length=1.5 in
order=(&minn to &maxx by 1)
minor=none
;

axis4 color=black
/*
label=(h=.13 f=swiss c=black j=r 'Component' j=r 'Three') */
value=(h=.1 f=swiss)
length=1.5 in
order=(&minn to &maxx by 1)
minor=none
;

axis5 color=black
/*
label=(h=.13 f=swiss c=black j=r 'Component' j=r 'Four') */
value=(h=.1 f=swiss)
length=1.5 in
order=(&minn to &maxx by 1)

```

```

        minor=none
        ;

axis6  color=black
/*      label=(h=.13 f=swiss c=black j=r 'Component' j=r 'Five' ) */
value=(h=.1 f=swiss)
length=1.5 in
order=(sminn to smaxx by 1)
minor=none
;

proc gplot data=pcatest2 gout=plots.plots %if %letters=YES %then %str(anno=FigB) ;
  plot prin1*dt / name="PLOT"
    lv=2
    vref=(-1 0 1)
    vaxis=axis2
    haxis=axis1
    %DO x = 1 %to %eval(&HLINECNT);
      %&hline&x
    %END;
  ;
run; quit;

proc gplot data=pcatest2 gout=plots.plots %if %letters=YES %then %str(anno=FigD)::
  plot prin2*dt / name="PLOT1"
    lv=2
    vref=(1 0 -1)
    vaxis=axis3
    haxis=axis1
    %DO x = 1 %to %eval(&HLINECNT);
      %&hline&x
    %END;
  ;
run; quit;

proc gplot data=pcatest2 gout=plots.plots %if %letters=YES %then %str(anno=FigF)::
  plot prin3*dt / name="PLOT2"
    lv=2
    vref=(-1 0 1)
    vaxis=axis4
    haxis=axis1
    %DO x = 1 %to %eval(&HLINECNT);
      %&hline&x
    %END;
  ;
run; quit;

proc gplot data=pcatest2 gout=plots.plots %if %letters=YES %then %str(anno=FigH)::
  plot prin4*dt / name="PLOT3"
    lv=2
    vref=(-1 0 1)
    vaxis=axis5
    haxis=axis1
    %DO x = 1 %to %eval(&HLINECNT);
      %&hline&x
    %END;
  ;
run; quit;

proc gplot data=pcatest2 gout=plots.plots %if %letters=YES %then %str(anno=FigJ)::
  plot prin5*dt / name="PLOT4"
    lv=2
    vref=(-1 0 1)
    vaxis=axis6
    haxis=axis1
    %DO x = 1 %to %eval(&HLINECNT);
      %&hline&x
    %END;
  ;
run; quit;

%if %origtemp=YES %then %plotgen;

%mend PPLOT;

/* used to make scree plot: CARDS doesn't seem to work in a macro */
/*
data rand;
  input ob rand;
cards;
1 13
2 12
3 11
4 10
5 9
;
run;
*/

%macro SCREE;
/** generate a scree plot **/
data scree;
  set percent; run;

```

```

proc transpose data=scre out=scre; run;

data scre; set scre; drop _name_ coll; ob=_n_; actual=coll; run;

data scree; merge scre rand; by ob; run;

options gunit=in target=ps600 dev=win hsize=3.5 in vsize = 3.5 in
  noprompt /* goutmode=append */ colors=(black GrayE9 GrayF9);
title;
symbol1 c=black v=dot h=.05 i=join;
symbol2 c=black v=star h=.1 i=join;

axis1 color=black
  order=(1 to 5)
  label=(h=.13 f=swiss c=black j=c 'Factor')
  value=(f=swiss h=.1)
  minor=none
  ;

axis2 color=black
  label=(h=.13 f=swiss c=black j=r 'Explained' j=r 'Variance' j=r 'in percent')
  value=(h=.1 f=swiss)
  length=1.5 in
  minor=none
  ;

proc gplot data=scree gout=plots.plots  \if &letters=YES \then &str(anno=FigL);:
  plot actual*ob=1
    rand*ob=2 / name='scree' overlay vaxis=axis2 haxis=axis1;
run; quit;
%MEND SCREE;

/***** MAP STATIONS IN PCA *****/
data points;
  set sql.finloc1;
  length name $ 20 province $ 2;

  province='95';
  name=pid;
  x=&d2r*long*-1;
  y=&d2r*lat;
  keep name x y province;
run;

%macro factr;
/* delete all plots except mean temperature plots */
proc catalog cat=plots.plots kill ; run; quit;

\if &ONEPAGE=YES \then \let stoppt=3;
\else \let stoppt=6;
\if &limfact=YES \then \let stoppt=&numfact;

\do fcount = 1 \to &stoppt;
  \if &fcount = 1 \then \do;
    \let factor=factor1;
    \let pltname=FACTSC;
    \end;
  \if &fcount = 2 \then \do;
    \let factor=factor2;
    \let pltname=FACTSC1;
    \end;

  \if &fcount = 3 \then \do;
    \let factor=factor3;
    \let pltname=FACTSC2;
    \end;

  \if &fcount = 4 \then \do;
    \let factor=MtT;
    \let pltname=MEANT;
    \end;

  \if &fcount = 5 \then \do;
    \let factor=factor5;
    \let pltname=FACTSC3;
    \end;

  \if &fcount = 6 \then \do;
    \let factor=factor6;
    \let pltname=FACTSC4;
    \end;

  %raticul
  %basemaps
  %mapit

\end;

%mend factr;

%factr
%scree /* generate scree plot */
%tplot /* time series plot of pca scores */
/*****/

%macro FINDISP;
/** Display 6 plots on one page **/

```

```

%IF &normPCA=YES %then %do;
FILENAME lprun "d:\atkinson\maindoc\figures\Fig3-7.ps"; /* PS600.ps */
options reset=all cback=white noprompt
%if &fileout=PIL %then %str(device=ps600 gsfname=lprun gsfmode=replace);
%if &fileout=SCM %then %str(device=win target=winprt);
%if &fileout=PRT %then %str(device=winprt);
hsize = 7.1 in vsize=10.7 in horigin=.5 in vorigin=.5 in
colors=(grayF9 grayF9 gray89 black);

proc greplay igout=plots.plots /*gout=plots.plots*/ tc=tempcat nofs;

  tdef shade des='Factor Score spatial plots and scores'

    1/ llx=0 lly=67
      ulx=0 uly=100
      urx=50 ury=100
      lrx=50 lry=67

    2/ llx=50 lly=67
      ulx=50 uly=100
      urx=100 ury=100
      lrx=100 lry=67

    3/ llx=0 lly=33
      ulx=0 uly=67
      urx=50 ury=67
      lrx=50 lry=33

    4/ llx=50 lly=33
      ulx=50 uly=67
      urx=100 ury=67
      lrx=100 lry=33

    5/ llx=0 lly=0
      ulx=0 uly=33
      urx=50 ury=33
      lrx=50 lry=0

    6/ llx=50 lly=0
      ulx=50 uly=33
      urx=100 ury=33
      lrx=100 lry=0
  ;

  template shade;

%if &limfact=YES %then %str(treplay 1:factsc 2:plot 3:factsc1 4:plot1 5:factsc2 6:plot2; template shade:);
%else %do;
  %if &numfact=2 %then %str(treplay 1:factsc 2:plot 3:factsc1 4:plot1 5:xx 6:xx; template shade:);
  %if &numfact=1 %then %str(treplay 1:factsc 2:plot 3:xx 4:xx 5:xx 6:xx; template shade:);
%end;

%if &ONEPAGE=YES %then %str(quit:);
%if &ONEPAGE=YES %then %do;
  treplay 1:factsc3
          2:plot1
          3:factsc4
          4:plot4
          5:meanT
          6:scree;

  quit;
%end;
%END;
/*
proc greplay igout=plots.plots gout=work.gseg tc=tempcat nofs;
  tdef shade des='Factor Score spatial plots and scores'

    1/ llx=0 lly=67
      ulx=0 uly=100
      urx=50 ury=100
      lrx=50 lry=67

    2/ llx=50 lly=67
      ulx=50 uly=100
      urx=100 ury=100
      lrx=100 lry=67

    3/ llx=0 lly=33
      ulx=0 uly=67
      urx=50 ury=67
      lrx=50 lry=33

    4/ llx=50 lly=33
      ulx=50 uly=67
      urx=100 ury=67
      lrx=100 lry=33

    5/ llx=0 lly=0
      ulx=0 uly=33
      urx=50 ury=33
      lrx=50 lry=0
  ;

```

```

        6/ llx=50 lly=0
           ulx=50 uly=33
           urx=100 ury=33
           lrx=100 lry=0
;
template shade;
treplay 1:factsc3
        2:plot3
        3:meant64
        4:scree
        5:xx
        6:xx;

treplay 1:clyde
        2:hall
        3:pell
        4:shep
        5:camb
        6:bake;

quit;
*/
%mend FINDISP;
%FINDISP
/*
data _null_;
   ans="&origtemp";
   end;
run;
*/
/* TEST BED - determine relationship btwn order interval and major tick labels */
/* order defines # of major ticks,          TICK labels VISIBLE MAJOR TICKS ONLY*/
/*
%options reset=all;
symbol i=j v=circle h=1;
axis1 order=(1 to 31 by 3) value=(tick=1 j=c 'TICK1' tick=10 j=c 'TICK2') minor=none;
proc gplot data=pcatest2;
   plot prin1*dt /
       haxis=axis1;
run; quit;
*/
/* use for check plots of raw data
data ds; set doneset; dt=_n_; run;
%options reset=all;
symbol i=j v=circle h=1;
proc gplot data=ds;
   plot   pc196*dt
         eurekat*dt sachsh*dt resolutt*dt pondit*dt holman*dt /overlay legend;
run; quit;
*/
%mend HUGE;

```

Appendix C:

SAS code for the topoclimate model to estimate surface air temperatures

m-fileass.sas

```
/******  
/***  
/*** Macro FILEASS specifies preliminary information for the primary stations. ***/  
/*** A window queries the user for the number of primary stations to use ***/  
/*** (although for now 9 should always be used), the year and month of the data ***/  
/*** sets, and the required wind level. All these macro variables are declared ***/  
/*** to be GLOBAL so that any macro can access them. Note that all datafiles ***/  
/*** for the requested time period need to have been dumped into FSL format ***/  
/*** using GETRAOB on the Radiosonde Data of North America CDROM. ***/  
/***  
/*** There are no input parameters other than those requested from the window. ***/  
/*** No output SAS datasets are generated. ***/  
/***  
/***  
/*** Invocation structure is %FILEASS ***/  
/*** User call example: %FILEASS ***/  
/***  
/*** David E. Atkinson, October 1998. ***/  
/***  
/*****  
%Macro FILEASS;  
  %global NUMFILES ST EN START END WINDLEVL P1 P2 P3 P4 P5 P6 P7 P8 P9 STMONTH EDMONTH STYEAR othrpert;  
  %let NUMFILES=9;  
  %let ST=01JUL1988;  
  %let EN=31JUL1988;  
  %let WINDLEVL=900;  
  %let othrpert=0100;  
  %window GETINFO color=gray rows=22 columns=68  
    %2 %2 "Surface Temperature Model Pre-information"  
    %4 %3 "How many primary stations?"  
      %32 NUMFILES 2 attr=underline  
      %37 "max of 9 (shouldn't alter)"  
    %6 %3 "Input starting date (DD/MON/YYYY):"  
      %38 ST 11 attr=underline  
    %8 %3 "Input ending date (DD/MON/YYYY):"  
      %36 EN 11 attr=underline  
/*  
    %5 %3 "Input year:"  
      %15 YEAR 3 attr=underline  
      %19 "(last two digits)"  
    %6 %3 "Input month:"  
      %16 MONTH 3 attr=underline  
      %20 "(two digit number)"  
*/  
    %11 %3 "Input wind level:"  
      %21 WINDLEVL 4 attr=underline  
      %26 "(three digit number in mb)"  
    %13 %3 "Input other part of file name:"  
      %34 OTHERPART 4 attr=underline  
      %39 "(four digit number DDMH)"  
    %15 %5 "Don't change this unless you know what you are doing!"  
    %17 %5 "Press ENTER to continue.";  
%IF %&THORPE=NO %THEN %display GETINFO;  
%let START=%&ST%D;  
%let END=%&EN%D;  
%put %&START %&END;  
%let SMOCLAB=%subetr(%&ST,3,3);  
data _null_;  
  if %&SMOCLAB="MAY" then call symput('STMONTH',05);  
  if %&SMOCLAB="JUN" then call symput('STMONTH',06);  
  if %&SMOCLAB="JUL" then call symput('STMONTH',07);  
  if %&SMOCLAB="AUG" then call symput('STMONTH',08);  
  if %&SMOCLAB="SEP" then call symput('STMONTH',09);  
run;  
%let EMOCLAB=%subetr(%&EN,3,3);  
data _null_;  
  if %&EMOCLAB="MAY" then call symput('EDMONTH',05);
```



```

0 0 -0.5 -0.5 -0.5 -1 -0.5 -0.5 -0.5 0 0
0 0 0 -0.5 -1 -2 -1 -0.5 0 0 0
0 0 0 0 -1 -5 -1 0 0 0 0
0 0 0 0 0 -10 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 10 0 0 0 0 0
0 0 0 0 1 5 1 0 0 0 0
0 0 0 0.5 1 2 1 0.5 0 0 0
0 0 0.5 0.5 0.5 1 0.5 0.5 0.5 0 0
0 0 0.5 0.5 0.5 0.5 0.5 0.5 0 0);
/* low speed, from south val=34 */
filtSL=(0 0 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0 0
0 0 0.5 0.5 0.5 1 0.5 0.5 0.5 0 0
0 0 0 0.5 1 2 1 0.5 0 0 0
0 0 0 0 1 5 1 0 0 0 0
0 0 0 0 0 10 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 -10 0 0 0 0 0
0 0 0 0 -1 -5 -1 0 0 0 0
0 0 0 -0.5 -1 -2 -1 -0.5 0 0 0
0 0 -0.5 -0.5 -0.5 -1 -0.5 -0.5 -0.5 0 0
0 0 -0.5 -0.5 -0.5 -0.5 -0.5 -0.5 0 0);
/* low speed, from west val=66 */
filtWL=(0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
-0.5 -0.5 0 0 0 0 0 0 0 0.5 0.5
-0.5 -0.5 -0.5 0 0 0 0 0 0.5 0.5 0.5
-0.5 -0.5 -1 -1 0 0 0 1 1 0.5 0.5
-0.5 -1 -2 -5 -10 0 10 5 2 1 0.5
-0.5 -0.5 -1 -1 0 0 0 1 1 0.5 0.5
-0.5 -0.5 -0.5 0 0 0 0 0.5 0.5 0.5
-0.5 -0.5 0 0 0 0 0 0 0.5 0.5
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0);
/* low speed, from east val=18 */
filtEL=(0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0
0.5 0.5 0 0 0 0 0 0 0 -0.5 -0.5
0.5 0.5 0.5 0 0 0 0 0 -0.5 -0.5 -0.5
0.5 0.5 1 1 0 0 0 -1 -1 -0.5 -0.5
0.5 1 2 5 10 0 -10 -5 -2 -1 -0.5
0.5 0.5 1 1 0 0 0 -1 -1 -0.5 -0.5
0.5 0.5 0.5 0 0 0 0 0 -0.5 -0.5 -0.5
0.5 0.5 0 0 0 0 0 0 0 -0.5 -0.5
0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0);
finish; /* buildfil */
/** APPLY FILTER **/
start filter;
%if %percent=NO %then %do;
%let DS%mat=%subscr(%target,5,5)mat;
load %DS%mat;
ds=choose(%DS%mat,1,0);
free %DS%mat;
/*
%IMGDUMP(/usr/local/data/modelout/,DS,DS,&rows,&cols,&maxx,&minx,&maxy,&miny) */
SSTEMP=choose(DS,0,&SST); /* make an sea surface temperature mask */
store SSTEMP;
free SSTEMP;
demstamp=shape(ds,&rows);
free ds;
%end;
%if %percent=YES %then %do;
load icewind;
demstamp=icewind;
free icewind;
%end;
filres=j(%rows,&cols,0);
load filtermp;
do row = (1.&filsize) to (nrow(demstamp)-(&filsize-1));
do col=(1.&filsize) to (ncol(demstamp)-(&filsize-1));
if demstamp[row,col]=1 then do;
if filtermp[row,col]=9 then curfil=filTNM;
if filtermp[row,col]=10 then curfil=filTNL;
if filtermp[row,col]=12 then curfil=filTMM;
if filtermp[row,col]=17 then curfil=filTSM;
if filtermp[row,col]=18 then curfil=filTEL;
if filtermp[row,col]=20 then curfil=filTMM;
if filtermp[row,col]=33 then curfil=filTSM;
if filtermp[row,col]=34 then curfil=filTSL;
if filtermp[row,col]=36 then curfil=filTSM;
if filtermp[row,col]=65 then curfil=filTMM;
if filtermp[row,col]=66 then curfil=filTNL;
if filtermp[row,col]=68 then curfil=filTMM;
if filtermp[row,col]=0 then curfil={1};
skipto: filsize=int((nrow(curfil)-1)/2);
fcols=col-filsize;
fcole=col+filsize;
frows=row-filsize;
frowe=row+filsize;
filspace=demstamp(frows:frowe,fcols:fcole);
/** REMOVE FICRD WIND PROBLEM **/
filbin=choose(filspace,1,0);
if nrow(filbin)=1 then do;

```

```

        edge=1; zone=1;
        goto MOVEON;
    end;
    edge = T(filbin[1,1])
    //filbin[2:nrow(filbin),ncol(filbin)]
    //T(filbin[nrow(filbin),ncol(filbin)-1:1])
    //filbin[nrow(filbin)-1:2,1];
    /* extract elements from around edge of dem filter area */
moveon: do posn = 1 to (4*nrow(curfil)-4); /* number of elements in a square matrix edge */
        curval=edge[posn];
        if posn=1 then do;
            zone=1;
            oldval=curval;
        end;
        if posn>1 then do;
            if curval ^= oldval then do;
                zone=zone+1;
                oldval=curval;
            end;
        end;
        /* count number of blocks of 0 and 1.
        If there are more than 3 areas, it means
        two separate land areas are present which will
        spoil the wind effects, eg in a fiord.
        */
        if zone>3 then do;
            curfil=curfil#.9; /* reduce strength of wind effect */
            curfil=curfil[2:ncol(curfil)-1,2:nrow(curfil)-1];
            goto skipto;
        end;
        /* reduce # rows, cols by one on each side */
        /* repeat process until zone LE 3 */
    /** END OF FIORD WIND PROBLEM **/
    x1=filospace#curfil;
    filres[row,col]=sum(x1);
    if filres[row,col] < 0 then filres[row,col]=0;
    end;
    else filres[row,col]=0;
end;
end;
finish; /* filter */
free demstamp;
run buildfil;
run filter;
store filres;
quit;
tmacro FILTER;

```

m-finald.sas

This displays the image using SAS GCONTOUR. The image looks like an IDRISI image plot, but it is not nearly as nice an environment as IDRISI. This was written simply to avoid having to move back and forth between computers to view interim output generated during testing stages.

```

tmacro FINALD;
goptions reset=all;
data trial;
infile '/usr/local/data/modelout/AARES.img';
do y=4rows to 1 by -1;
do x=1 to scols;
input z;
output;
end;
end;
run;
/*
pattern1 value=msolid color=CK0000FF;
pattern2 value=msolid color=CK3000FF;
pattern3 value=msolid color=CK6000E0;
pattern4 value=msolid color=CK9000D0;
pattern5 value=msolid color=CKA000E0;
pattern6 value=msolid color=CKC000C0;
pattern7 value=msolid color=CKE080A0;
pattern8 value=msolid color=CKF07070;
pattern9 value=msolid color=CKF06040;
pattern10 value=msolid color=CKF05020;
pattern11 value=msolid color=CKF04050;
*/
pattern1 value=msolid color=grayf0;
pattern2 value=msolid color=graye0;
pattern3 value=msolid color=grayd0;
pattern4 value=msolid color=grayc5;
pattern5 value=msolid color=grayb0;
pattern6 value=msolid color=graya5;
pattern7 value=msolid color=gray90;
pattern8 value=msolid color=gray85;
pattern9 value=msolid color=gray70;
pattern10 value=msolid color=gray65;
pattern11 value=msolid color=gray50;

```

```

proc gcontour data=trial;
    plot y*x=z /levels=0 1 2 3 4 5 6 7 8 9 10
        pattern
        jcin;
run;

%mend FINALD;

```

m-getdem.sas

This module reads in the external digital elevation model and the ice field location image. These images have to be in IDRIS image formats. They are stored as SAS datasets, and converted later into IML matrices.

```

/**** Macro GETDEM imports the Digital elevation model and places
relevant information about its size into the global environment **/
%macro GETDEM;
    %GLOBAL rows cols SST maxwind;
    %let PATH=/usr/local/data/modelin/;
    %let DEMNAME-AADEM;
    %let ICEFIELD=ICETOPO;
    %let MAXWIND=100;
    %let SST=2;
    %window GETINFO color=gray rows=18 columns=68
    %2 %2 "Surface Temperature Model DEM information"
    %4 %3 "What is the name of the DEM? (no extensions)"
    %5 %3 DEMNAME % attr=underline
    %7 %3 "Provide the file path"
    %8 %5 PATH % attr=underline
    %10 %3 "What is the sea-surface temperature to be used?"
    %52 SST % attr=underline
    %11 %3 "What is the maxwind value?"
    %10 maxwind % attr=underline
    %16 %5 "Press ENTER to continue.";
    %if &ATHOME=NO %then %display GETINFO;

    libname DEM "/usr/local/data/modelin/";
    data DEM.DEMFILE;
        infile "&PATH&DEMNAME..IMG";
        input val;
    run;
    data DEM.ICEFIELD;
        infile "&PATH&ICEFIELD..IMG";
        input val;
        if val > 0 then val=1;
    run;
    data _NULL_;
        format val 5.;
        infile "&PATH&DEMNAME..DOC";
        if _n_ < 4 OR _n_ > 5 then input;
        if _n_ = 4 then do;
            input @15 val;
            call symput('cols',trim(left(val)));
        end;
        if _n_ = 5 then do;
            input @15 val;
            call symput('rows',trim(left(val)));
        end;
    run;
%mend GETDEM;

```

m-griddefn.sas

```

/**** macro GRIDDEFN establishes parameters about the universe and places them in the
global environment **/
%macro GRIDDEFN;
    /* Define grid extents and node separation */
    /* set at GLOBAL so all modules may access these variables */
    %GLOBAL minx maxx miny maxy colinc rowinc colstmap coledmap rowstmap rowedmap
    AL% AL%r EUR% EUR%r ML% ML%r RES% RES%r CAM% CAM%r
    HAL% HAL%r BR% BR%r BAR% BAR%r FRO% FRO%r;
    %let PATH=/usr/local/data/modelin/;
    %let DEMNAME-AADEM;
    /* extract grid extents from idrisi DOC file for mapping grid extents */
    data _NULL_;
        infile "&PATH&DEMNAME..DOC";
        input @1 testchar $6. @;
        put testchar;
        if testchar = "min. X" then do;
            input @15 minx;
            call symput('colstmap',int(minx));
        end;
        if testchar = "max. X" then do;

```

```

        input @15 maxx;
        call symput('coledmap',int(maxx));
        end;
        if testchar = 'min. Y' then do;
            input @15 miny;
            call symput('rowedmap',int(miny));
            end;
        if testchar = 'max. Y' then do;
            input @15 maxy;
            call symput('rowstmap',int(maxy));
            end;

run;
%let minx=&colstmap;
%let maxx=&coledmap;
%let miny=&rowedmap;
%let maxy=&rowstmap;
%let colinc=%eval((&coledmap-&colstmap)/&cols);
%let rowinc=%eval((&rowstmap-&rowedmap)/&rows);
/* end grid extents */
/* Define station locations in terms of grid */
/* remember macro processor %eval only accepts integers */
%let ALTC=%eval((542435-&colstmap)/&colinc);
%let ALTR=%eval((&rowstmap-4256179)/&rowinc);
%let EURC=%eval((283719-&colstmap)/&colinc);
%let EURR=%eval((&rowstmap-3866941)/&rowinc);
%let MLDC=%eval((-525069-&colstmap)/&colinc);
%let MLDR=%eval((&rowstmap-3507485)/&rowinc);
%let RESC=%eval((152737-&colstmap)/&colinc);
%let RESR=%eval((&rowstmap-3277719)/&rowinc);
%let CMRC=%eval((-208361-&colstmap)/&colinc);
%let CMRR=%eval((&rowstmap-2670488)/&rowinc);
%let HALC=%eval((763029-&colstmap)/&colinc);
%let HALR=%eval((&rowstmap-2728698)/&rowinc);
%let BRMC=%eval((-1813027-&colstmap)/&colinc);
%let BRMR=%eval((&rowstmap-3675504)/&rowinc);
%let BARC=%eval((-156531-&colstmap)/&colinc);
%let BARR=%eval((&rowstmap-3267011)/&rowinc);
%let PROC=%eval((1515387-&colstmap)/&colinc);
%let PROX=%eval((&rowstmap-2403340)/&rowinc);
%mend GRIDDEFN;

```

m-ice.sas

This module was a first attempt at applying the ice field temperature estimates. It was dropped in favour of a less complicated approach.

```

%Macro ICE;
Proc IML;
    ICREST=-20;
    START icet;
        /* make icefield temperature estimate map (entire map) */
        load TFIN;
        TICBALL = TFIN + ICREST; /* ICREST is some estimate for ice T */
        store TICBALL;
        free /;
        /* create icefield location binary map */
        load ICEFINAT;
        ICEWIND=CHOOSE(ICEFINAT,0,1);
        store ICEWIND;
        free /;
        /* confine ice temperature estimates to icefield locations */
        load TICBALL, ICEWIND;
        TICPROP = CHOOSE(ICEWIND,TICBALL,0);
        store TICPROP;
        free /;
        /* apply wind filter to the icefields */
        /* %MACRO CALL */
        load ICEFINAT TFIN;
        ICETEMPS=ICEFINAT*TFIN;
        STORE ICETEMPS;
        FREE icetemps;
        TFINHOLE=(ICEFINAT,-100, TFIN);
        store tfinhole;
        free tfinhole;
    run icet;
    /* use the icefield image as a cookie cutter - stamp out the relevant
    zones from TFIN, save them as ICETFIN, make the zones empty in TFIN,
    determine affected border
    pixels in TFIN, modify border pixels in TFIN, add ICETFIN back onto
    TFIN.
    */
quit;
%Mend ICE;

```

apply necessary modifications to ICETFIN.

m-imgdump.sas

```
/** macro IMGDUMP takes the specified matrix and, using values for image total
rows, cols, max x, min x, max y, min y drawn from the global environment,
writes an IDRISI image .img and associated .doc files to the path and file
name specified. Call from within PROC IML.
***/
/* form of call:
%IMGDUMP(/usr/local/data/modelout/,filename,matrix,&rows,&cols,&maxx,&minx,&maxy,&miny)
*/
%Macro IMGDUMP(destpath,imagenam,matrix,row,col,xx,nx,xy,ny);
  result=shape(&matrix,&row*&col);
  filename tresimg "&destpath&imagenam..img";
  file tresimg;
  do i=1 to nrow(result);
    put (result[i,1]) ;
  end;
  closefile tresimg;
  x=max(&matrix);
  n=min(&matrix);
  filename tresdoc "&destpath&imagenam..doc";
  file tresdoc;
  put
    @1 'file title : Final temperature map'/
    @1 'data type : real'/
    @1 'file type : ascii'/
    @1 'columns : &col' /
    @1 'rows : &row' /
    @1 'ref. system : lazea'/
    @1 'ref. units : m'/
    @1 'unit dist. : 1.0000000'/
    @1 'min. X : &nx'/
    @1 'max. X : &xx'/
    @1 'min. Y : &ny'/
    @1 'max. Y : &xy'/
    @1 'pos'n error : unknown'/
    @1 'resolution : unknown'/
    @1 'min. value : ' n'/
    @1 'max. value : ' x'/
    @1 'value units : unknown'/
    @1 'value error : unknown'/
    @1 'flag value : none'/
    @1 'flag def'n : none'/
    @1 'legend cats : 0';
  closefile tresdoc;
%Mend IMGDUMP;
```

m-interpol.sas

```
*****/
/** Macro INTERPOL interpolates point data fed to it.
***/
/** element, dry-bulb temperature, dew point temperature and wind data, in
***/
/** a separate SAS dataset. Upper air data must be in FSL format and are
***/
/** typically generated using the Radio Sonde data of North America CD ROMs.
***/
/** Macro will act on the dataset supplied; the number of ascents desired
***/
/** must be specified when the data are being extracted from the CD-ROM.
***/
/** One parameter required: exponent, to provide the inverse distance weight***/
/**
***/
/** David E. Atkinson, October 1998.
***/
/**
***/
*****/

%Macro INTERPOL(exponent);
/* %INTERPOL(exponent,fields,fname) where fields=# fields to estimate and fname is
the name for the fields, e.g.
for polynomial fields=6 and fnames=P11 P12
for wind fields=2 and fnames=
*/
/* get and store the DEM file into a IML matrix */
/* stuff for memory space problem testing */
libname DEM '/usr/local/data/modelin/';
%$ASTOMAT(DEM.DEMFILE,&rows,&cols)
%$ASTOMAT(DEM.ICEFIELD,&rows,&cols)
proc iml workspace=256000000 symsize=10000000;
  free /;
  start p1;
  param1={&SALTc &SALTr &P11, &SURC &SURR &P12,
&MLDC &MLDR &P13, &RESC &RESR &P14,
&CMC &CMR &P15, &HALC &HALR &P16,
&BRMC &BRMR &P17, &BARC &BARR &P18,
&PROC &PROR
&P19};
  stnlocs=param1[,1:2];
  stnvals=param1[,3];
  stnlocs=choose(stnlocs=0,1,stnlocs);
  param1=stnlocs||stnvals;
  pmat1=j(&rows,&cols,1);
  do row = 1 to &rows;
    do col=1 to &cols;
      dists=((col-param1[,1])##2+(row-param1[,2])##2)##0.5;
      dists=choose(dists=0,.8165,dists);
      weights=1/(dists##&exponent);
    end;
  end;
finish;
run;
```

```

        wtvals=weights@param1[,3];
        pmat1[row,col]=sum(wtvals)/sum(weights);
    end;
end;
TFIN=pmat1;
free pmat1;
store TFIN;
free /;
finish;
run p1;
start p2;
    param2= {&ALTC &ALTR &P21, &EURC &EURR &P22, &MLDC &MLDR &P23, &RESC &RESR &P24,
&CAMC &CAMR &P25, &HALC &HALR &P26, &BRNC &BRWR &P27, &BARC &BARR &P28, &PROC &PROR
&P29};
    stnlocs=param2[1:2];
    stnvals=param2[3];
    stnlocs=choose(stnlocs=0.1,stnlocs);
    param2=stnlocs||stnvals;
    pmat2=j(&rows,&cols,1);
    do row = 1 to &rows;
        do col=1 to &cols;
            dists=((col-param2[1])##2+(row-param2[2])##2)##0.5;
            dists=choose(dists=0..8165,dists);
            weights=1/(dists##6exponent);
            wtvals=weights@param2[,3];
            pmat2[row,col]=sum(wtvals)/sum(weights);
        end;
    end;
    CURPMAT=PMAT2;
    store CURPMAT;
    free /;
    load DDEFINAT CURPMAT;
    DC=DDEFINAT@CURPMAT;
    store DC;
    free /;
    load DC TFIN;
    TFIN=DC*TFIN;
    free DC;
    store TFIN;
    free /;
finish;
run p2;
start p3;
    param3= {&ALTC &ALTR &P31, &EURC &EURR &P32, &MLDC &MLDR &P33, &RESC &RESR &P34,
&CAMC &CAMR &P35, &HALC &HALR &P36, &BRNC &BRWR &P37, &BARC &BARR &P38, &PROC &PROR
&P39};
    stnlocs=param3[1:2];
    stnvals=param3[3];
    stnlocs=choose(stnlocs=0.1,stnlocs);
    param3=stnlocs||stnvals;
    pmat3=j(&rows,&cols,1);
    do row = 1 to &rows;
        do col=1 to &cols;
            dists=((col-param3[1])##2+(row-param3[2])##2)##0.5;
            dists=choose(dists=0..8165,dists);
            weights=1/(dists##6exponent);
            wtvals=weights@param3[,3];
            pmat3[row,col]=sum(wtvals)/sum(weights);
        end;
    end;
    CURPMAT=PMAT3;
    store CURPMAT;
    free /;
    load DDEFINAT;
    DDEKXP=DDEFINAT##2;
    store DDEKXP;
    free /;
    load DDEKXP CURPMAT;
    DC=DDEKXP@CURPMAT;
    store DC;
    free /;
    load DC TFIN;
    TFIN=DC*TFIN;
    free DC;
    store TFIN;
    free /;
finish;
run p3;
start p4;
    param4= {&ALTC &ALTR &P41, &EURC &EURR &P42, &MLDC &MLDR &P43, &RESC &RESR &P44,
&CAMC &CAMR &P45, &HALC &HALR &P46, &BRNC &BRWR &P47, &BARC &BARR &P48,
&PROC &PROR &P49};
    stnlocs=param4[1:2];
    stnvals=param4[3];
    stnlocs=choose(stnlocs=0.1,stnlocs);
    param4=stnlocs||stnvals;
    pmat4=j(&rows,&cols,1);
    do row = 1 to &rows;
        do col=1 to &cols;
            dists=((col-param4[1])##2+(row-param4[2])##2)##0.5;
            dists=choose(dists=0..8165,dists);
            weights=1/(dists##6exponent);

```

```

        wtvals=weights#param4[,3];
        pmat4[row,col]=sum(wtvals)/sum(weights);
    end;
end;
CURPMAT=PMAT4;
store CURPMAT;
free /;
load DEMFINAT;
DEMEXP=DEMFINAT##3;
store DEMEXP;
free /;
load DEMEXP CURPMAT;
DC=DEMEXP#CURPMAT;
store DC;
free /;
load DC TFIN;
TFIN=DC*TFIN;
free DC;
store TFIN;
free /;
finish;
run p4;
start p5;
param5= {&ALTC &ALTR &PS1, &EURC &EURR &PS2,
         &MLDC &MLDR &PS3, &RESC &RESR &PS4,
         &CAMC &CAMR &PS5, &HALC &HALR &PS6,
         &BRWC &BRWR &PS7, &BARC &BARR &PS8,
         &PROC &PROR &PS9};
stnlocs=param5[,1:2];
stnvals=param5[,3];
stnlocs=choose(stnlocs=0,1,stnlocs);
param5=stnlocs||stnvals;
pmat5={&rows,&cols,1};
do row = 1 to &rows;
    do col=1 to &cols;
        dists=((col-param5[,1])##2*(row-param5[,2])##2)##0.5;
        dists=choose(dists=0,..8165,dists);
        weights=1/(dists##exponent);
        wtvals=weights#param5[,3];
        pmat5[row,col]=sum(wtvals)/sum(weights);
    end;
end;
CURPMAT=PMAT5;
store CURPMAT;
free /;
load DEMFINAT;
DEMEXP=DEMFINAT##4;
store DEMEXP;
free /;
load DEMEXP CURPMAT;
DC=DEMEXP#CURPMAT;
store DC;
free /;
load DC TFIN;
TFIN=DC*TFIN;
free DC;
store TFIN;
free /;
finish;
run p5;
start p6;
param6= {&ALTC &ALTR &P61, &EURC &EURR &P62,
         &MLDC &MLDR &P63, &RESC &RESR &P64,
         &CAMC &CAMR &P65, &HALC &HALR &P66,
         &BRWC &BRWR &P67, &BARC &BARR &P68,
         &PROC &PROR &P69};
stnlocs=param6[,1:2];
stnvals=param6[,3];
stnlocs=choose(stnlocs=0,1,stnlocs);
param6=stnlocs||stnvals;
pmat6={&rows,&cols,1};
do row = 1 to &rows;
    do col=1 to &cols;
        dists=((col-param6[,1])##2*(row-param6[,2])##2)##0.5;
        dists=choose(dists=0,..8165,dists);
        weights=1/(dists##exponent);
        wtvals=weights#param6[,3];
        pmat6[row,col]=sum(wtvals)/sum(weights);
    end;
end;
CURPMAT=PMAT6;
store CURPMAT;
free /;
load TFIN;
TemTFIN=TFIN;
store TemTFIN;
free /;
remove TFIN;
load DEMFINAT;
DEMEXP=DEMFINAT##5;
store DEMEXP;
free /;
load DEMEXP CURPMAT;
DC=DEMEXP#CURPMAT;
store DC;

```

```

free /;
remove CURPMAT DEMEXP;
load DC TEMTFIN;
  TFIN=DC*TEMTFIN;
  free DC TEMTFIN;
  store TFIN;
free /;
remove DC TEMTFIN;
finish;
run p6;
/** Apply Icefield temperature estimates **/
start ICET;
  load TFIN ICEFIMAT;
  TFIN2=choose(ICEFIMAT,TFIN-1,TFIN);
  TFIN=TFIN2;
  store TFIN;
  free /;
finish;
run ICET;
quit;
%mend INTERPOL;

```

m-

```

/** Macro MERGE combines results from the temperature with elevation and **/
/** wind effects map **/
%macro MERGE;
proc iaf;
  start merger;
  free /;
  load FILRES;
  RES1F=(FILRES/&maxwind)#&SST;
  store RES1F;
  free RES1F;
  FILRESPC=(1-FILRES/100);
  free FILRES;
  load TFIN;
  RES1TF=TFIN#FILRESPC;
  free TFIN FILRESPC;
  load RES1F;
  RES1=RES1F*RES1TF;
  free RES1F RES1TF;
  remove RES1F;
  store RES1;
  free /;
  load SSTEMP RES1;
  RESULT=choose(SSTEMP>0.2,RES1); /* stamp the sea surface temperature onto the oceans */
  free RES1 SSTEMP;
  store RESULT;
  %DUMP(/usr/local/data/modelout/,AARES,RESULT,&rows,&cols,&maxx,&minx,&maxy,&miny)
  finish;
  run merger;
quit;

```

```
%mend MERGE;
```

m-noinver.sas

```

/** macro NOINVER removes the inversion signal, if present, from UA temperature ascents **/
%macro NOINVER;
%NOINVER2
%IF &HEIGHT > 0 %THEN %DO;
data INVERMOD;
set tdest;
ITOGGLE=0;
do x = 4000 to 0 by -20;
  maint=intercep+elev*x+elev2*x**2+elev3*x**3+elev4*x**4+elev5*x**5;
  fx=elev+2*elev2*x+3*elev3*x**2+4*elev4*x**3+5*elev5*x**4;
  if x = 4000 then oldfx=fx;
  diff=fx-oldfx;
  if x = 4000 then olddiff=diff;
  dd=olddiff-diff;
  if z le 1500 AND (ITOGGLE OR (diff > .0005 AND dd < 0)) then do;
    ITOGGLE=1;
    tval=tval+tinc;
  end;
else do;
  Tinc=tval-oldtval;
  oldtval=tval;
  tval=maint;
end;
z=x;
keep z maint tval fx diff;
output;
end;

```

```

run:
  data raobtd; /* rewrite RAOBTD with the new curve and rerun POLYINT */
  set invermod;
  elev=z;
  temp=tval;
  drop tval z diff maint;
run:
  %POLYINT(YES)
%END;
%mend NOINVER;

%macro NOINVER2;
data INVERSN;
  set tdest;
  ARRAY root(1200,2);
  do i=1 to 1200; do j=1 to 2; root{i,j}=0;end;end;
  ARRAYPOS=1;
  do at = 0 to 1200;
    x=st;
    do iterate=0 to 5;
      tval=intercep+elev*x+elev2*x**2+elev3*x**3+elev4*x**4+elev5*x**5;
      fx=elev+2*elev2*x+3*elev3*x**2+4*elev4*x**3+5*elev5*x**4;
      dfx=2*elev2+6*elev3*x+12*elev4*x**2+20*elev5*x**3;
      if dfx=0 then dfx=.0000001;
      if dfx < .002 AND dfx > -.002 then do;
        xn=x-fx/dfx;
        x=xn;
        if abs(fx)<0.0001 then do;
          x=round(x,1);
          if arraypos=1 then prev=1;
          else prev=arraypos-1;
          if x ne root{prev,1} AND x ne . then do;
            root{ARRAYPOS,1}=x;
            root{ARRAYPOS,2}=round(tval,1);
            ARRAYPOS=ARRAYPOS+1;
          end;
        end;
      end;
    end;
  end;
end;
/**** BEGIN MULTIPLE CONVERGES SECTION ****/
if ARRAYPOS>2 then do;
  /** INSERTION SORT FROM MIKE'S BOOK **/
  do i = 3 to ARRAYPOS-1;
    temp=root{i,1};
    tempval=root{i,2};
    j=i-1;
    do until(j = 1 OR temp ge root{j,1});
      root{j+1,1}=root{j,1};
      root{j+1,2}=root{j,2};
      j=j-1;
    end;
    root{j+1,1}=temp;
    root{j+1,2}=tempval;
  end;
  /** ELIMINATE DUPLICATES **/
  jump=1;
  endlp=arraypos-1;
  do i=1 to endlp;
    do j=jump to arraypos-1;
      if j=arraypos-1 then do;
        arraypos=i+1;
        i=endlp;
      end;
      if root{i,1} ne root{j,1} then do;
        root{i+1,1}=root{j,1};
        root{i+1,2}=root{j,2};
        jump=j; /* new start position for next time */
        j=arraypos; /* breakout of loop */
      end;
    else do;
      if j > 1 then do;
        root{j,1}=0;
        root{j,2}=0;
      end;
    end;
  end;
end;
end;
/**** DONE WITH MULTIPLE CONVERGES SECTION **/
/** GET MAX TEMPERATURE **/
max=root{1,2};
maxpos=1;
if arraypos > 1 then do;
  do i=2 to arraypos-1;
    if root{i,2}>max then do;
      max=root{i,2};
      maxpos=i;
    end;
  end;
else do;
  max=root{1,2};
  maxpos=1;
end;
end;

```

```

x=0;
strength=round(max-(intercep-elev*x-elev2*x**2-elev3*x**3-elev4*x**4-elev5*x**5)..1);
height=root(maxpos,1);
IF HEIGHT < 0 OR HEIGHT > 4000 THEN HEIGHT=0;
call symput('height',HEIGHT);
IF STRENGTH < 0 OR STRENGTH > 30 THEN STRENGTH=0;
do i=1 to ARRAYPOS-1; put root{i,1} root{i,2};end;
keep OB height strength;
output;

run;
%mend NOINVER2;

```

m-polyint.sas

```

/*****
/**** Macro POLYINT fits 5th order polynomial to upper air temperature and ****
/**** dew point data. Supply parameters drybulb and dewpt as YES or NO, which ****
/**** advises the macro whether or not to process the specified dataset. The ****
/**** dataset names are fixed by the macro which reads the upper air data set. ****
/**** RAOBREAD. Specify parameter indiv as YES if each ascent is to have ****
/**** parameter estimates generated or NO if one set of parameter estimates is ****
/**** to be generated for all data. Specify parameter IPLOT as YES if a series ****
/**** of small individual ascent plots are to be generated; turning this on ****
/**** causes extra information to be created. ****
/****
/**** Depending on the values of input parameters drybulb and dewpt, the macro ****
/**** returns two SAS datafiles TDEST and TDWEST, each ****
/**** containing parameter estimates for the ascents for ****
/**** the respective elements. Each file contains one set ****
/**** or many sets of estimates, depending on what the ****
/**** input parameter CHOICE was set to. ****
/****
/**** Note that all SAS datasets, input and output, reside in the WORKING lib. ****
/****
/**** Call is %POLYINT(drybulb, dewpt, indiv,IPLOT), where the parameters ****
/**** can take the values YES or NO (capital letters, no quotes). ****
/****
/**** David E. Atkinson, October 1998. ****
/****
/*****
%macro POLYINT(drybulb, dewpt, indiv,IPLOT);
%IF &drybulb=YES %then %do;
  data tdprep;
    set raobtd;
    elev2=elev**2;
    elev3=elev**3;
    elev4=elev**4;
    elev5=elev**5;
    drop tdew wdir wspd;

  run;
  %if &indiv=YES %then %str(proc sort data=tdprep; by ob temp pressure; run;);
  proc reg data=tdprep outest=tdest noprint;
    model temp=elev elev2 elev3 elev4 elev5;
    %if &INDIV=YES %then %do;
      by ob;
    %end;

  run;quit;
  %if &IPLOT=YES %then %do;
  data dates;
    set raobtd;
    keep ob time day month year;
    if _n_=1 then oldob=0;
    retain oldob;
    if oldob ne ob then do;
      output;
      oldob=ob;
    end;

  run;
  data tdtemp; set tdest; run;
  data tdest;
    merge dates tdtemp ;
    %if &INDIV=YES %then %do;
      by ob;
    %end;

  run;
%end;
%IF &dewpt=YES %then %do;
  data tdewprep;
    set raobtdew;
    elev2=elev**2;
    elev3=elev**3;
    elev4=elev**4;
    elev5=elev**5;
    drop temp wdir wspd;

  run;
  %if &indiv=YES %then %str(proc sort data=tdewprep; by ob temp pressure; run;);
  proc reg data=tdewprep outest=tdewest;
    model tdew=elev elev2 elev3 elev4 elev5;
    %if &INDIV=YES %then %do;

```

```

                                by ob;
                                tend;

run:quit;

end;
%mend POLYINT;
/* SAMPLE CALL
   will analyse the set of dry-bulb observations from the ascent file supplied.
   will not analyse the corresponding set of dew point temperatures, and will
   return 5th order polynomial parameter estimates for each individual ascent
   in the dataset (because choice is set to YES
*/
/*
/*polyint(YES, NO, YES)
*/

```

m-preclean.sas

```

/*****
/****
/**** Macro PRECLEAN empties target files of any information they might have. ****
/**** The option MOD in a SAS output appends data, thus if the program is run ****
/**** more than once extra unwanted data is accumulated in the files. This ****
/**** macro also serves to initialize these file spaces. ****
/****
/**** There are no input parameters; these output datafiles are currently ****
/**** hardwired into the model and cannot be changed. No output SAS datasets ****
/**** are generated. ****
/****
/**** Invocation structure is %PRECLEAN ****
/**** User call example: %PRECLEAN ****
/****
/**** David E. Atkinson, October 1998. ****
/****
/*****
%macro PRECLEAN;
/* clean out the files because the MOD option appends to existing files */
data _null_;
    file "/usr/local/data/modelout/wind.vec";          put;
    file "/usr/local/data/modelout/wzbarlaz.vec";      put;
    file "/usr/local/data/modelout/wybarlaz.vec";      put;
    file "/usr/local/data/modelout/wzbarlaz.dvc";      put;
    file "/usr/local/data/modelout/wybarlaz.dvc";      put;
%DO cnt=1 %to 6; /* currently there are six parameter estimates */
    file "/usr/local/data/modelout/n6cnt.vec";        put;
%END;
run;

%mend PRECLEAN;

```

m-raobread.sas

```

/*****
/****
/**** Macro RAOBREAD reads data from upper air ascents and stores each ****
/**** element, dry-bulb temperature, dew point temperature and wind data, in ****
/**** a separate SAS dataset. Upper air data must be in FSL format and are ****
/**** typically generated using the Radio Sonda data of North America CD ROMs. ****
/**** Macro will act on the dataset supplied; the number of ascents desired ****
/**** must be specified when the data are being extracted from the CD-ROM. ****
/****
/**** There are five input parameters: DS specifies the location of the file ****
/**** to be processed, complete with drive and full path (no quotes); DRYBT, ****
/**** DEWPT and WINDS are specified as YES if a dataset with that element is to ****
/**** be created (a comma to act as place holder if not YES), and LEVEL, which ****
/**** is specified as a number between 1000 and 50 if WINDS is set to YES (level ****
/**** extracts the winds from the level specified). Macro outputs up to three ****
/**** datasets, depending on the input parameters: RAOBTD if DRYBT is set to YES ****
/**** RAOBTDM if DEWPT is set to YES, and RAOBWIND if WINDS is set to YES. ****
/****
/**** Macro also sets the values of three macro variables to YES or NO, depending ****
/**** on which datasets were written: DRYBULB if a dry bulb temperature dataset ****
/**** was created, DEWTEMP if a dew point temperature dataset was created, and ****
/**** WINDDATA if a wind speed dataset was created. These macro variables ****
/**** reside in the background of the session and may be accessed to determine ****
/**** whether or not various element-specific PROCs are executed. ****
/****
/**** Note that all SAS datasets, input and output, reside in the WORKING lib. ****
/****
/**** Invocation structure is %RAOBREAD(ds, drybt, dewt, winds, level) ****
/**** User call example: %RAOBREAD(D:\raobdata\88070100.BRW,YES, .YES,900) ****
/**** This call would request dry bulb temperatures and 900 mb level winds from ****
/**** the FSL datafile 88070100.BRW. ****
/****
/**** David E. Atkinson, October 1998. ****
/****
/*****

```

```

/**
/*****
macro RAOBREAD(ds, drybt, dewt, winds, level);
  %global drybulb dewtemp winddata;
  %let drybulb=NO;
  %let dewtemp=NO;
  %let winddata=NO;
  data
  %if &drybt=YES %then raobtd;
  %if &dewt=YES %then raobtdew;
  %if &winds=YES %then raobwind;
  ;
      stday=%START;
      edday=%END;
      length month $ 4;
      infile "6ds" END=DONE;
      input ID @;
      if _n_ = 1 then ob=0;
      if ID=254 then
          do;
              skip=0;
              input time day month year;input;input;input;
              date=input(trim(left(day))||trim(month)||trim(left(year)),date9.);
              if date LT stday OR date GT edday then skip=1;
              else ob=ob+1;
          end;
          else input pressure elev temp tdew wdir wspd;
              if skip=1 then goto skipit;
              if pressure=. and elev=. and temp=. and tdew=. and wdir=. and wspd=. then goto skipit;
              %if &drybt=YES %then %str(temp=temp/10.);
              %if &dewt=YES %then %str(tdew=tdew/10.);
              format temp tdew 5.1;
              retain time day month year date ob skip;
              keep ob time day month year pressure elev temp tdew wdir wspd;
              %if &drybt=YES %then %do;
                  %str(if temp<3276 and elev < 4500 and (date GE stday OR date LE edday) then output raobtd;);
                  if DONE then call symput('drybulb','YES');
              %end;
              %if &dewt=YES %then %do;
                  %str(if tdew<3276 and elev < 4500 and (date GE stday OR date LE edday) then output raobtdew;);
                  if DONE then call symput('dewtemp','YES');
              %end;
              %if &winds=YES %then %do;
                  %str(if (pressure<=level) AND NOT (wspd > 3000 OR wdir > 3000) and (date GE stday OR date LE edday) then
output raobwind;);
                  if DONE then call symput('winddata','YES');
              %end;
              skipit: ob=ob;
              run;
/*
      data transfer;
          drybulb=symget('drybulb');
          dewtemp=symget('dewtemp');
          winddata=symget('winddata');
      run;
*/
%mend RAOBREAD;

```

m-getdem.sas

```

/*****
/****
/**** Macro GETDEM gets a SAS dataset and converts it into a matrix. User
/**** specifies size. Makes single column digital elevation model currently
/**** stored as an IDRISI
/**** GIS image file and converts it to a matrix for use in PROC IML. This
/**** must be called from within a PROC IML shell.
/**** David E. Atkinson, October 1998.
/****
/*****
macro SASTOMAT(SASName,nrows,ncols);
  proc iml;
    start CONVERT;
      use &SASName;
      read all into newmat;
      %substr(&SASName,5,5)mat=shape(newmat,&nrows,&ncols);
      STORE %substr(&SASName,5,5)mat;
      close &SASName;
    finish;
    run CONVERT;
    free /;
  quit;
%mend SASTOMAT;

/* note the original should be (&SASName,1,5) ; I have
changed it to 5,5 for testing 5,5 used to support

```

m-sastomw.sas

```

This is a modified version of m-sastomat.sas SAS dataset to IML matrix converter for the winds images.
%macro SASTOMW(SASname,nrows,ncols);
  start CONVERT;
    use &SASname;
    read all into newmat;
    %substr(&SASname,1,5)mat=shape(newmat,&nrows,&ncols);
    STORE %substr(&SASname,1,5)mat;
    close &SASname;
  finish;
  run CONVERT;
/* free /; */
%mend SASTOMW;

```

m-tget.sas

```

/*****
/****
/**** Macro TGET extracts necessary temperature data from UA ascent (RAOBREAD),
/**** fits a fifth-order polynomial (POLYINT) and places the parameter estimates
/**** into the global macro variable environment for future access.
/****
/**** No parameters are required; output is a set of parameter estimates for
/**** each station, labelled T11, T12, etc.
/****
/**** Note that all SAS datasets, input and output, reside in the WORKING lib.
/****
/**** Call is %TGET
/****
/**** David E. Atkinson, October 1998.
/****
/****
/*****/
%macro CLEAN;
  %let HEIGHT=0;
  PROC DATASETS library=work;
    delete invermod raobtd tdest;
  run;
%mend CLEAN;
%macro TGET;
  %CLEAN
  %let number=&numfiles;
  %do count=1 %to &number;
    %GLOBAL p1%count p2%count p3%count p4%count p5%count p6%count;
    %let f1=%STYEAR&STMONTH&othrpart..YLT; /* AES Alert */
    %let f2=%STYEAR&STMONTH&othrpart..YEU; /* AES Eureka */
    %let f3=%STYEAR&STMONTH&othrpart..YMD; /* AES Mould Bay */
    %let f4=%STYEAR&STMONTH&othrpart..YRB; /* AES Resolute Bay */
    %let f5=%STYEAR&STMONTH&othrpart..YCB; /* AES Cambridge Bay */
    %let f6=%STYEAR&STMONTH&othrpart..YUX; /* AES Hall Beach */
    %let f7=%STYEAR&STMONTH&othrpart..BTI; /* Barter Island, AK */
    %let f8=%STYEAR&STMONTH&othrpart..BRW; /* Barrow, AK */
    %let f9=%STYEAR&STMONTH&othrpart..YVW; /* Probrisher Bay */
    %RAOBREAD(/usr/local/data/modelin/&f%count,YES)
    proc sort data=raobtd out=raobtd; by elev temp; run;
    %POLYINT(YES)
    libname DA "/usr/local/data/modelin/";
    proc datasets library=work; copy out=DA; select tdest; run;quit;
    %%INVER
    data _null_;
      set tdest;
      call symput('P1' || trim(left(%count)), INTERCEP);
      call symput('P2' || trim(left(%count)), elev);
      call symput('P3' || trim(left(%count)), elev2);
      call symput('P4' || trim(left(%count)), elev3);
      call symput('P5' || trim(left(%count)), elev4);
      call symput('P6' || trim(left(%count)), elev5);
    run;
  /*
  data intplot;
    set DA.tdest;
    do z=0 to 4000;
      temp=INTERCEP+elev*z+elev2*z**2+elev3*z**3+elev4*z**4+elev5*z**5;
      tempnew=%P1%count + %P2%count*z + %P3%count*z**2 +
        %P4%count*z**3 + %P5%count*z**4 + %P6%count*z**5;
      output;
    end;
  run;
  %options noprompt;
  title1 f=swissb c=green h=1.5 "Ascent profile for %substr(%f%count,10)";
  title2 h=1 f=swiss "Mean for month %substr(%f%count,3,2) in 19%substr(%f%count,1,2)";
  title3 h=1 f=swiss "Check on curve with inversion signal removed";
  symbol1 i=join value=none w=4 color=black;
  symbol2 i=join value=none w=4 color=red;

```

```

axis1 order=(0 to 5000 by 500) label=(angle=90 f=swiss 'Elevation in metres')
value=(f=swiss h=.75);
axis2 order=(-25 to 15 by 5) label=(f=swiss 'Temperature') value=(f=swiss h=.75);
proc gplot data=inp1plot;
plot z*temp=1 z*tempnew=2 /overlay
vaxis=axis1
haxis=axis2
lhref=2
href=0
;
run; quit;
*/
%end;
%mend TGET;

```

m-wget.sas

```

/*****
/****
/**** Macro WGET retrieves wind data from the UA profiles, converts to radians. ****
/**** takes the means and writes the x and y component means to IDRISI GIS ****
/**** format point vector files. These files are later used for interpolation. ****
/****
/**** There are no input parameters other than those requested from the window. ****
/**** No output SAS datasets are generated; four ascii text files are generated. ****
/****
/**** Invocation structure is %WGET ****
/**** User call example: %WGET ****
/****
/**** David E. Atkinson, October 1998. ****
/****
/*****/
%macro WGET;
%let number=&numfiles;
%do count=1 %to &number;
%RAOSREAD(/usr/local/data/modelin/%%&count...YES.&windlevl)
/*
data winds;
set raob&windlevl;
if wdir > 1000 OR wspd > 1000 then return;
keep ob elev wdir wspd;
output;
input ob elev wdir wspd;
run;
*/
data windmean;
set raobwind;
xcomp=round(sin(wdir/360)*2*.14159)*wspd,.0001;
ycomp=round(cos(wdir/360)*2*.14159)*wspd,.0001;
run;
proc summary data=windmean;
var xcomp ycomp;
output out=windstat mean=xmean ymean;
run;
data windstat; set windstat; xmean=round(xmean,.001); ymean=round(ymean,.001); run;
/* removes rounding errors */
data _null_;
set windstat;
loc=&count;
done=&number;
file '/usr/local/data/modelout/wxbarlaz.vec' mod;
put xmean * 1.00*;
if loc=1 then put '542435.4375 4256179.5'; /* Alert */
else if loc=2 then put '283719.25 3866941.'; /* Eureka */
else if loc=3 then put '-525069.3125 3507485.75'; /* Mould */
else if loc=4 then put '152737.203125 3277719.75'; /* Rea */
else if loc=5 then put '-208361.109375 2670488.5'; /* Cambridge */
else if loc=6 then put '763029.375 2728698.75'; /* Hall */
else if loc=7 then put '-1813027.625 3675504.'; /* Barrow */
else if loc=8 then put '-1565531.375 3267011.'; /* Barter Island */
else if loc=9 then put '1515387.75 2403340.75'; /* Probieher */
if loc=done then do;
put '0 0';
file '/usr/local/data/modelout/wxbarlaz.dvc';
put @1 'file title' @13 ':' @15 'Upper air data: coefficient for the wind xcomponent mean'/
@1 'id type' @13 ':' @15 'real'/
@1 'file type' @13 ':' @15 'ascii'/
@1 'object type' @13 ':' @15 'point'/
@1 'ref. system' @13 ':' @15 'larea'/
@1 'ref. units' @13 ':' @15 'm'/
@1 'unit dist.' @13 ':' @15 '1.00000000'/
@1 'min. X' @13 ':' @15 '-1900000' /
@1 'max. X' @13 ':' @15 '1550000' /
@1 'min. Y' @13 ':' @15 '2350000' /
@1 'max. Y' @13 ':' @15 '4300000' /
@1 'pos'n error' @13 ':' @15 'unknown'/
@1 'resolution' @13 ':' @15 'unknown';

```

```

end;
file "/usr/local/data/modelout/wybarlaz.vec" mod;
put ymean * 1.00";
if loc=1 then put "542435.4375 4256179.5"; /* Alert */
else if loc=2 then put "283719.25 3866941."; /* Eureka */
else if loc=3 then put "-525069.3125 3507485.75"; /* Mould */
else if loc=4 then put "152737.203125 3277719.75"; /* Res */
else if loc=5 then put "-208361.109375 2670488.5"; /* Cambridge */
else if loc=6 then put "763029.375 2728698.75"; /* Hall */
else if loc=7 then put "-1813027.625 3675504."; /* Barrow */
else if loc=8 then put "-1565531.375 3267011."; /* Barter Island */
else if loc=9 then put "1515387.75 2403340.75"; /* Probisher */
if loc=done then do;
put "0 0";
file "/usr/local/data/modelout/wybarlaz.dvc";
put @1 'file title' @13 ':' @15 'Upper air data: coefficient for the wind y component mean'/
@1 'id type' @13 ':' @15 'real'/
@1 'file type' @13 ':' @15 'ascii'/
@1 'object type' @13 ':' @15 'point'/
@1 'ref. system' @13 ':' @15 'laza'/
@1 'ref. units' @13 ':' @15 'm'/
@1 'unit dist.' @13 ':' @15 '1.0000000'/
@1 'min. X' @13 ':' @15 '-1900000' /
@1 'max. X' @13 ':' @15 '1550000' /
@1 'min. Y' @13 ':' @15 '2350000' /
@1 'max. Y' @13 ':' @15 '4300000' /
@1 'pos'n error' @13 ':' @15 'unknown'/
@1 'resolution' @13 ':' @15 'unknown';
end;

run;
tend;
tmend WGET;

```

m-windy.sas

This reads in processed wind images that have been output in IDRISI format.

```

macro WINDXY;
/* Get x and y vector data point */
data xidrimpt;
infile "/usr/local/data/modelout/wxbarlaz.vec";
INPUT val;
if (MOD(_n_,2)=0 or _n_ > 18) THEN RETURN;
output;

run;
data yidrimpt;
infile "/usr/local/data/modelout/wybarlaz.vec";
INPUT val;
if (MOD(_n_,2)=0 or _n_ > 18) THEN RETURN;
output;

run;
/* END get x and y vector point data */
tmend WINDXY;

```

m-winterpl.sas

```

/*****
/*****
/*****
/* macro WINTERPL */
/* Main program: WIND EFFECT IMAGE */
/* Call is %WINTERPL(exponent) */
/*
/*
/*
/* 1. Read in raw point or image data. */
/* 2. If point data run interpolation routine; */
/* interpolate x-component and y-component */
/* mean fields. */
/* 3. Use x and y mean field interpolation */
/* results to generate resultant wind direction */
/* and velocity interpolated over entire image. */
/* 4. Create wind speed and direction */
/* classification image. */
/* 5. Separate out land areas by filter type zones; */
/* create masks (give buffer zone equal to one */
/* half filter width plus 1). */
/* 6. Apply each filter to its zone. Filter will */
/* only act where pixel = 1; masked out land */
/* pixels will be coded to -999, ocean to 0. */
/* 7. Recombine results into one image. This */
/* image describes influence of average winds */
/* for a given period of time over entire region.*/
/*
/*
/*
/* David B. Atkinson October 1998. */
/*
/*****
/*****

```

```

/*****
macro WINTERPL(exponent);
  %WINDXY /* create x and y vector fields as SAs files*/
proc ial;
  /* INTERPOLATE xmean vector and y mean vector fields */
  %SASTOWN(xidrimpt,&numfiles,1)
  start xinterp;
    stnlocs={%ALTc %ALTr, %EURc %EURR, %MLDc %MLDR,
             %RESc %RESR, %CAMc %CAMR, %HALc %HALR,
             %BRMc %BRMR, %BARc %BARR, %PROC %PROR};
    stationx=stnlocs||XIDRIMAT;
    /* to force edge locns to non zero coordinates (not sure its needed)
    stnlocs=stationx[,1:2];
    strvals=stationx[,3];
    stnlocs=choose(stnlocs=0,1,stnlocs); */
    /* stationx=stnlocs||strvals; */
    xmat=j(&rows,&cols,1); /* create grid, initialize to 1 */
    do row = 1 to &rows;
      do col=1 to &cols;
        dists=((col-stationx[,1])#2+(row-stationx[,2])#2)#0.5;
        dists=choose(dists=0,.8165,dists); /* .8165 gives wt=1.5 */
        weights=1/(dists##exponent);
        wtvals=weights#stationx[,3];
        xmat[row,col]=round(sum(wtvals)/sum(weights),1);
      end;
    end;
  store xmat;
  free xmat;
  finish;
  %SASTOWN(yidrimpt,&numfiles,1)
  start yinterp;
    stationy=stnlocs||YIDRIMAT;
    ymat=j(&rows,&cols,1); /* create grid, initialize to 1 */
    do row = 1 to &rows;
      do col=1 to &cols;
        dists=((col-stationy[,1])#2+(row-stationy[,2])#2);
        dists=choose(dists=0,.8165,dists);
        weights=1/(dists##exponent);
        wtvals=weights#stationy[,3];
        ymat[row,col]=sum(wtvals)/sum(weights);
      end;
    end;
  store ymat;
  free ymat;
  finish;
  /* END INTERPOLATE */
start analys;
load xmat;
x1=(xmat>0);
x3=(xmat<0);
x5=(xmat=0);
store x1 x3 x5;
free /;
load ymat;
x2=(ymat>0);
x4=(ymat<0);
x6=(ymat=0);
store x2 x4 x6;
free /;
load x1 x2;
amask=x1#x2; /* x > 0 and y > 0 */
store amask;
free amask x1;
load x3;
bmask=x3#x2; /* x < 0 and y > 0 */
store bmask;
free bmask x2;
load x4;
cmask=x3#x4; /* x < 0 and y < 0 */
store cmask;
free cmask x3;
load x1;
dmask=x1#x4; /* x > 0 and y < 0 */
store dmask;
free /;
load x2 x5;
emask=x5#x2; /* x = 0 and y > 0 */
store emask;
free emask x2;
load x4;
fmask=x5#x4; /* x = 0 and y < 0 */
store fmask;
free /;
load x1 x6;
gmask=x6#x1; /* y = 0 and x > 0 */
store gmask;
free gmask x1;
load x3;
hmask=x6#x3; /* y = 0 and x < 0 */
store hmask;
free /;
remove x1 x2 x3 x4 x5 x6;
load xmat ymat;
combdir=((atan(xmat/ymat))#(180/3.14159));
free xmat ymat;

```

```

load amask;
adir=combdir#amask; /* element-wise multiplication is # */
store adir;
free adir amask;
load bmask;
bdir=(combdir*360)#bmask;
store bdir;
free bmask bdir;
lowdir=combdir*180;
free combdir;
show storage;
load cmask;
cdir=(lowdir)#cmask;
store cdir; free cmask cdir; load dmask;
ddir=(lowdir)#dmask;
store ddir; free dmask ddir lowdir; load emask;
edir=360#emask;
store edir; free emask edir; load fmask;
fdir=180#fmask;
store fdir; free fdir fmask; load gmask;
gdir=90#gmask;
store gdir; free gdir gmask; load hmask;
hdir=270#hmask;
free hmask;
show storage;
load gdir;
sumdbg=hdir+gdir;
remove hdir gdir;
store sumdbg; free hdir gdir sumdbg; load fdir edir;
sumdef=fdir+edir;
remove fdir edir;
store sumdef; free fdir edir sumdef; load ddir cdir;
sumddc=ddir+cdir;
store sumddc; free ddir cdir sumddc; load bdir adir;
sumdba=bdir+adir;
store sumdba; free /; load sumdba sumddc;
sumd1=sumdba+sumddc;
remove sumdba sumddc;
store sumd1; free /; load sumdef sumdbg;
sumd2=sumdef+sumdbg;
remove sumdef sumdbg;
store sumd2;
free /;
show storage;
load sumd1 sumd2;
sumdir=sumd1+sumd2;
store sumdir;
remove sumd1 sumd2;
free /;
show storage;
PRINT "END OF DIR START OF SPD";
load xmat adir;
aspd1=(abs(xmat/(sin((adir/360)#2#3.14159))));
remove adir;
free adir xmat;
load amask;
aspd=aspd1#amask;
remove amask;
store aspd;
free /;
show storage;
load bdir xmat;
bspd1=(abs(xmat/(sin((bdir/360)#2#3.14159))));
remove bdir;
free bdir xmat;
load bmask;
bspd=bspd1#bmask;
remove bmask;
store bspd;
free /;
show storage;
load cdir xmat;
cpsd1=(abs(xmat/(sin((cdir/360)#2#3.14159))));
remove cdir;
free cdir xmat;
load cmask;
cpsd=cspd1#cmask;
/* CSPD AND DSPD may require YMAT not XMAT */
remove cmask;
store cpsd;
free /;
show storage;
load ddir xmat;
dspd1=(abs(xmat/(sin((ddir/360)#2#3.14159))));
remove ddir;
free ddir xmat;
load dmask;
dspd=dspd1#dmask;
remove dmask;
store dspd;
free /;
show storage;
load ymat emask;
espd=abs(ymat)#emask;
remove emask;

```

```

store espd;
free espd emask;
load fmask;
  fspd=abs(ymat)#fmask;
remove fmask;
store fspd;
free /;
show storage;
load xmat gmask;
  gspd=abs(xmat)#gmask;
remove gmask;
store gspd;
free gspd gmask;
load hmask;
  hspd=abs(xmat)#hmask;
remove hmask;
store hspd;
free /;
show storage;
load aspd bspd;
  sum1=aspd+bspd;
remove aspd bspd;
store sum1;
free /;
load cspd dspd;
  sum2=cspd+dspd;
remove cspd dspd;
store sum2;
free /;
load espd fspd;
  sum3=espd+fspd;
remove espd fspd;
store sum3;
free /;
load gspd hspd;
  sum4=gspd+hspd;
remove gspd hspd;
store sum4;
free /;
show storage;
load sum1 sum2;
  sum12=sum1+sum2;
free sum1 sum2;
remove sum1 sum2;
store sum12;
load sum3 sum4;
  sum14=sum3+sum4;
free sum3 sum4;
remove sum3 sum4;
load sum12;
  sumspd=sum12+sum14;
remove sum12;
store sumspd;
free /;
show storage;
remove xmat ymat;
/* speed category reclass */
load sumspd;
  cala=choose(sumspd<1,0,1);
store cala;
free cala;
  medspd=choose((8<=sumspd)&(sumspd<15),1,0);
store medspd;
free medspd;
  lowspd=choose(sumspd<8,2,0);
store lowspd;
free lowspd;
  highspd=choose(sumspd>=15,4,0);
store highspd;
free /;
remove sumspd;
Print "Begin direction category reclass";
/* direction category reclass */
load sumdir;
  north1=choose(((sumdir>=316)&(sumdir<=360)),8,0);
store north1;
free north1;
  north2=choose(((sumdir>=0)&(sumdir<=45)),8,0); /* July 15,99 changed sumdir>=0; was sumdir>=0.001 */
store north2;
free north2;
  east=choose((sumdir>45)&(sumdir<=135),16,0);
store east;
free east;
  south=choose((sumdir>135)&(sumdir<=225),32,0);
store south;
free south;
  west=choose((sumdir>225)&(sumdir<=315),64,0);
store west;
free /;
load north1 north2;
  north=north1+north2;
store north;
free /;
remove north1 north2;
remove sumdir;

```

```

Print "begin filter classification":
/* filter classification */
load medspd lowspd;
  spdml=medspd+lowspd;
  free medspd lowspd;
load highspd;
  spd=spdml+highspd;
store spd;
print "got past spd=medspd+lowspd+highspd";
free /;
remove medspd lowspd highspd;
load north east;
  ne=north+east;
store ne;
free /;
remove north east;
load south west;
  sw=south+west;
store sw;
free /;
remove south west;
load ne sw;
  new=ne+sw;
store new;
remove ne sw;
free /;
load new spd;
  totl=new+spd;
store totl;
remove new spd;
free new spd;
load cala;
  filtermp=cala@totl;
remove cala totl;
store filtermp;
$IMGDUMP /usr/local/data/modelout/.windres.filtermp,&rows,&cols,&maxx,&minx,&maxy,&miny)
free /;
/* filtermp=cala@(medspd+lowspd+highspd+north+east+south+west);
*/
/** APPLY FILTER **/
/*
  create 12 resultant speed/direction categories: twelve filters to choose from
generate masks to apply each filter
recombine results
  finimg=combination of (findir and finspd) to get twelve groups;
generate filter for each group.
*/
finish; /* analysis */
/*
start exprt:
  finmat1=shape(sumdir,0,1);
  maxx=max(finmat1);
  minn=min(finmat1);
  finmat2=shape(sumsdpd,0,1);
  maxxpd=max(finmat2);
  minspd=min(finmat2);
  create opendir from finmat1[colname='pixels'];
  append from finmat1;
  close opendir;
  filename dir '/usr/local/data/modelout/testdir.img';
  file dir;
  do i=1 to nrow(finmat1);
    put (finmat1{i,1}) ;
  end;
  closefile dir;
  create opspd from finmat2[colname='pixels'];
  append from finmat2;
  close opspd;
  filename spd '/usr/local/data/modelout/testspd.img';
  file spd;
  do i=1 to nrow(finmat2);
    put (finmat2{i,1}) ;
  end;
  closefile spd;
finish;
*/
run xinterp;
run yinterp;
run analys;
/* run exprt: */
quit;

%mend WINTERPL;

```