34

THE ESTIMATION OF MISSING VALUES IN HYDROLOGICAL RECORDS USING THE EM ALGORITHM AND REGRESSION METHODS

by

TONDANI MAKHUVHA

Thesis

Submitted in fulfilment of the requirements for the degree of

MASTER OF SCIENCE

In the department of

Mathematical Statistics

University of Cape Town

SUPERVISORS: Prof W. Zucchini

Dr R.S. Sparks

September 1988

ACKNOWLEDGEMENTS

I am extremely grateful to my supervisors, Prof Walter Zucchini, who suggested the topic, and Dr Ross Sparks. They both encouraged, guided and assisted me throughout the research.

I owe special thanks to Mrs Tib Cousins, the TEX-pert, who was always willing to help when I encountered problems with TEX.

Thanks also go to my parents, Thetshelesani and Mashudu, and my uncle, Ratshilumela, for their support and encouragement.

TABLE OF CONTENTS

	Page
1. INTRODUCTION	1-1
2. REGRESSION METHODS - VARIABLE SELECTION	
2.1 Introduction	2-1
2.2 The model	2-4
2.3 Selection criteria	2-7
2.3.1 The C_p criterion	2-7
2.3.2 The A_p criterion	2-8
2.3.3 The square multiple correlation coefficient	2-9
2.3.4 The adjusted R_p^2	2-10
2.3.5 The residual mean square	2-10
2.3.6 The standardized residual sum of squares	2-11
2.3.7 The prediction sum of squares	2-11
2.3.8 Average predictive variance	2-12
2.3.9 The T_p criterion	2-12
2.3.10 The S_p criterion	2-13
2.4 Selection procedures	2-13
2.4.1 All possible control stations at a point	2-14
2.4.2 All possible control stations for several values	2-14
2.4.3 Forward selection	2-14
2.4.4 Backward elimination	2-14
2.4.5 Stepwise regression procedure	2-15
2.5 Selecting control records for individual missing values	2-15
2.6 Selecting control records for several missing values	2-17
2.7 Forward selection	2-19

2.8 Recommendation 2-20

3. THE EM ALGORITHM

3.1 Introduction 3-1

3.2 General description of the EM algorithm 3-3

3.3 Definition of the EM algorithm 3-4

3.4 The E step and the M step of EM 3-7

3.5 The general EM algorithm 3-9

3.6 Theory of the EM algorithm 3-9

3.6.1 Summary of the convergence of the EM 3-11

3.7 Estimation of missing values using the EM 3-12

3.8 Algorithms 3-13

3.8.1 *Method 1*: Condition on real records only 3-13

3.8.2 *Method 2*: Condition on real and estimated records 3-16

4. SIMULATION STUDY 4-1

4.1 Correlation structure 4-2

4.2 Generating annual rainfall totals 4-3

4.3 Discarding (hiding) observations from artificial data 4-4

4.4 Disaggregation of annual rainfall data to monthly rainfall data 4-4

5. COMPARISON OF THE SELECTED PROCEDURES - ANNUAL DATA

5.1 Sum of square error due to prediction (SSEP) 5-2

5.2 Preservation of the standard deviation 5-4

5.2.1 Regression methods 5-4

5.2.2 EM algorithm 5-6

5.3 Computation 5-19

5.3.1 Regression methods 5-19

5.3.2 EM algorithm 5-20

5.4 Conclusion 5-22

6. MONTHLY DATA - THE EM ALGORITHM

6.1 Introduction 6-1

6.2 EM algorithm on monthly rainfall totals 6-2

6.3 Disaggregation of the annual rainfall totals to monthly rainfall totals 6-3

6.4 Daily rainfall records 6-4

6.5 Comparison 6-4

6.5.1 Preservation of the mean 6-5

6.5.2 Sum of square error due to prediction 6-6

6.5.3 Preservation of the standard deviation 6-6

6.5.4 Computation 6-11

7. APPLICATION OF THE METHODS 7-1

8. SUMMARY AND CONCLUSIONS 8-1

REFERENCES

APPENDICES

A. EXAMPLE: THE EM ALGORITHM A-1

A1. *Method 1:* Condition on real records A-3

A2. *Method 2:* Condition on real and estimated records A-14

B. PROOF FOR SECTION 6.3.1

B1. Simple linear regression B-1

B2. Multiple regression B-3

C. PROGRAMS C-1

CHAPTER 1

INTRODUCTION

It is to be expected that any data records which are collected over a long period of time will contain gaps, and that the number of gaps in the records increases with the length of the records. This is certainly the case with rainfall records in South Africa where practically all records contain gaps.

Several circumstances contribute to the occurance of gaps, for example loss of records, temporary absence of observers, breakdown of measuring devices or simply the cessation of measurement at a particular rainfall station. Whatever the reason for their occurance, gaps in rainfall records are problematic in a number of respects. Hydrologists and engineers often require complete records for the purpose of planning and design. For example, relatively few streamflow records in South Africa are sufficiently long for accurate reservoir design and so rainfall records, which are generally longer, are often used to estimate past streamflow. For this it is necessary, or at least very convenient, to have complete rather than partial rainfall records. The same is true for the estimation of crop yield estimated using growth models by Agricultural Engineers. Other applications in which the occurance of gaps in rainfall records inconvenient include the estimation of drought risk and severity and the estimation of the frequency and severity of storms.

The terminology which was used by Zucchini and Sparks (1984), is adhered to. We are concerned with two types of rainfall stations, namely, the *target* station and the *control* station. The target station is the dependent variable whose missing records are to be estimated. The control stations are the independent variables which are chosen from the stations neighbouring the target station under consideration. Both the target station and the control stations records are of different lengths, and have gaps. The control stations which are used for the estimation of the target station are chosen in such a way that they are more correlated to the target station and/or have longer records. Since control stations also have gaps in the data, the selected stations should have concurrent records sufficient enough to be used to estimate the target station.

There are several methods that can be used to estimate the missing values in rainfall

records. However, in the absence of literature on the subject many practitioners have resorted to relatively ad hoc procedures. The simplest methods in common use include the replacement of missing values by

- the untransformed concurrent value at some neighbouring station.
- the average amount at the target station, i.e. not making use of information from neighbouring stations.
- the average of a small number, usually 3 or 4, neighbouring stations.

These procedures are in fact special cases of the linear regression methods in which the missing values at a target station are estimated as a linear combination of the concurrent values at one, none or several control stations.

Zucchini and Sparks (1984) considered the problem from the point of view of variable selection. (In this application the variables are the control stations.) More recently Adamson (1987) applied regression methods to estimate missing values at 2 500 rainfall stations in South Africa.

The objective of this thesis is to review existing methods for estimating missing values in rainfall records and to propose a number of new procedures. Two classes of methods are considered. The first is based on the theory of variable selection in regression. Here the emphasis is on finding efficient methods to identify the set of control stations which are likely to yield the best regression estimates of the missing values in the target station. The second class of methods is based on the EM algorithm, proposed by Dempster, Laird and Rubin (1977). The emphasis here is to estimate the missing values directly without first making a detailed selection of control stations. All "relevant" stations are included. This method has not previously been applied in the context of estimating missing rainfall values.

To compare and validate the methods we used simulated data from a multivariate normal distribution. Simulation is the most convenient way to assess the performance of the various methods which are discussed. This has several advantages over using "real" data. Firstly it is possible to compute the accuracy of the methods directly. This can be achieved by generating complete records, "hiding" some of the values, estimating "missing" values and then comparing the estimates with the corresponding "true" generated values. A second advantage is that it is easy, by

simulation, to vary some of the factors which are likely to be important in determining the performance of the different methods. Such factors include the correlations between observations at different stations, the proportion of missing values and the length of records at each station.

The thesis is layed out as follows: In Chapter 2 we discuss regression methods applied to non-seasonal data, that is annual data. These methods can however, be applied for the estimation of monthly gaps by treating each month of the year separately. Chapter 3 gives the theoretical description of the EM algorithm. This can be applied for annual, monthly, or even weekly data. We look at two variations of this method, the second variation being the modification of the "standard" method.

The simulation study is described in Chapter 4. This includes the method used in generating artificial rainfall values and the formation of the correlation structure used. Chapter 5 gives the results obtained from the simulated data and the comparison of the methods applied for annual data.

Monthly data is discussed in Chapter 6 which also contains a brief discussion on daily data. In this chapter we consider whether it would be preferable to treat each month separately, i.e. construct a separate data matrix for each of the twelve monthly series, or alternatively to group the months into a year thereby reduce the number of data matrices from twelve to one. Chapter 7 gives results obtained on the application of real rainfall data. A brief summary and the conclusions are given in Chapter 8.

There are 3 Appendices. Appendix A gives a step-by-step application of the two variations of the EM algorithm discussed in the text. The purpose of giving this is to illustrate in detail using a simple example how the algorithms are implemented in practice. Appendix B gives a proof that regression methods lead to a systematic downward bias in the variance of estimated missing values. Finally, Appendix C contains listings of the FORTRAN programs which were developed to implement the methods discussed in this thesis.

CHAPTER 2

REGRESSION METHODS - SELECTION OF VARIABLES

2.1 INTRODUCTION

The most important issue to be considered when applying regression analysis for estimating missing rainfall data, is the selection of control stations. This problem, is made difficult because, as a rule, some of the control stations which are available for the estimation of missing values at the target station are of different lengths, and themselves have gaps. Most of the statistical literature about the selection of variables only deals with cases where the samples are complete, but in our case, the samples are far from complete. For example, the target station may have data for 50 years available, whereas some of the control stations have data for only 15 years available. This leads to complications such as: the control station, which is highly correlated to the target station might be very short, whereas the other available control station, which is less correlated to the target station is longer. If the shorter record is utilized for the estimation of the regression coefficient, then a higher standard error of estimation than that of the longer station, is then obtained. The question then arises as to whether to base estimation on the record with higher correlation coefficient or alternatively on the one with the more reliable regression coefficients. This type of question does not arise if the control stations are complete.

Suppose that in Table 2.1, *Control*1 is highly correlated to *Target* and *Control*2 is less correlated to *Target*. The correlation is calculated by using the concurrent records between *Target* and *Control*1, and *Target* and *Control*2. If *Control*1 and *Control*2 are both used for the estimation of missing *Target* values, then only cases number 1, 4, 11, and 12 will be used for the calculation of the regression coefficient. It is therefore clear that the inclusion of additional control stations, while theoretically increasing the multiple correlation coefficient, can severely reduce the degrees of freedom. If *Control*1 is used to estimate *Target*, then only 7 observations will be utilized, whereas if *Control*2 is used, then 10 observations would be utilized. By using *Control*1 for estimation, then the standard error for the regression coefficient will be higher than when using *Control*2.

TABLE 2.1: *Example*

This example illustrates the need for control station selection.

Suppose a data set consists of 12 cases on 3 variables (stations) as shown below:

Case No.	*Target* Station	*Control*1	*Control*2
1	1	1	1
2	1	0	1
3	0	1	1
4	1	1	1
5	1	0	1
6	1	1	0
7	0	0	1
8	0	0	1
9	1	1	0
10	1	0	1
11	1	1	1
12	1	1	1

where 1 represents that a point is observed,

0 represents that a point is missing.

It is therefore important that the selection of control stations be performed in such a way as to take account of these various and conflicting effects.

In this chapter, we will review three methods (procedures) of selection of control stations. To make a decision on which method to use, it is necessary to evaluate the criterion on which it is based, and also the computational effort involved. In section 2.3 we will briefly review a number of criteria which could be used to select control stations. More detailed accounts of these (except for the T_p criterion, which is new) can be found in Sparks (1984), Draper and Smith (1981), Thompson (1978a) and Thompson (1978b).

Although each criterion focusses on a particular aspect of the regression model many of them are closely related and it is not always clear which criterion should be used

in a particular application. The criteria can be classified in a number of ways. For example, some criteria, such as the A_p and T_p, are based on individual points rather than on sets of points. In effect they require one to select a set of control stations for *every* missing value in the target record. Clearly this is computationally very expensive. In contrast, criteria such as the C_p, $\bar{R}_p^2$, MSE_p and J_p are suitable for selecting control stations for the set of all missing values of the target record.

A second way to classify the selection criteria is in terms of whether the predictor variables should be regarded as *fixed* or *random*. Thus the C_p and A_p criteria are suitable for applications where the levels of the predictor variables are *fixed*, for example by experimental design. In contrast the S_p criterion is appropriate where the levels of the predictor variables follow some *multivariate distribution*. Finally, in the case of the J_p and T_p criteria, the past values of the predictor variables are regarded as *fixed* and the future values as *random*. It is not necessary to take this classification literally; for example, one may use the J_p criterion even if the past observations could be regarded as realisation of *random* variables rather than quantities *fixed* by experimental design.

In section 2.4 we outline five procedures for going about selecting the control stations which have small values of a given criterion. The simplest, but most expensive way to do this is to compute the criterion for all possible subsets of the control stations. This guarantees that we find that subset which minimizes the selection criterion. Less expensive methods such as the forward selection, the backward elimination and the stepwise regression methods can also be used but one then has no guarantee that the best subset of control stations has been found. It is also the case that in our particular application these methods do not save as much computing effort as they do in conventional applications. The reason for this is that we have missing values in the control records. This leads to the problem that the selected subset of control stations may be unsuitable to estimate a particular missing value because one or more of the control stations have concurrent missing values. In such cases it is then necessary to eliminate some of the control stations and begin the selection procedure again. In contrast if one examines all subsets, one can then arrange them in order from best to worst and then use the best suitable subset to estimate

each missing value. (In practice it is only necessary to keep a record of the best few subsets.) Thus it is not always the case that forward selection leads to less computation in our particular application.

Sections 2.5, 2.6 and 2.7 give outlines of the algorithms to carry out three of the selection procedures, namely selecting control records for individual missing values, selecting control records for several missing values and forward selection.

2.2 THE MODEL

Let

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

denotes the $(n \times 1)$ vector of data points in the target station where some of the data points are missing. And let

$$X = (X_1, X_2, \ldots, X_k) = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ x_{21} & x_{22} & \cdots & x_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{2k} & \cdots & x_{nk} \end{pmatrix}$$

to be the $(n \times k)$ matrix of concurrent data points in k control stations where some of the x_{ij}'s are missing.

We assume that there are $n \geq (k+1)$ observations of k control stations such that the ith observed value of the target station is determined by

$$y_i = \sum_{j=1}^{k} x_{ij}\beta_j + e_i, \tag{2.1}$$

where

$$\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}$$

denotes the k vector of parameters, and

$$e = \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix}$$

denotes the $(n \times 1)$ vector of residuals.

It is assumed that the control stations $X = (X_1, X_2, \ldots, X_k)$ include all the relevant stations. We allow for the possibility that some of the observations in *both* the target station vector y and the control stations matrix X are missing. The vector β is calculated by using those observations which are common to all the stations (concurrent records), that is, the vector of the target station y and matrix of control stations X.

The residuals, e_i, $i = 1, 2, \ldots, n$, are assumed to be identically and independently distributed, with mean zero and unknown variance σ^2, where σ^2 is a parameter which does not depend on the x_{ij}. It is usually assumed that e_i follows a normal distribution. It may be necessary to transform the stations to achieve normality, e.g. by using the *log* transforms. The assumption on e_i can be summarized by writing

$$e_i \sim N(0, \sigma^2).$$

In matrix notation, equation (2.1) is summarized by writing the n-vector of observations, y, as

$$y = X\beta + e. \tag{2.2}$$

We now consider the question of selecting a subset of control stations. Equation (2.2) can be written as:

$$y = X_p\beta_p + X_r\beta_r + e \tag{2.3}$$

where the matrix of control stations X has been partitioned into X_p which denotes an $(n \times p)$ matrix of control stations to be included in the subset and X_r which denotes an $(n \times r)$ matrix of control stations to be excluded from the regression model, $k = p + r$;

and where

p denotes the number of control stations which are retained in the model,

and

r denotes the number of control stations which are deleted from the regression model.

The vector of estimates β is partitioned accordingly.

After partitioning equation (2.2), we can now write the subset model as:

$$y = X_p\beta_p + e, \tag{2.4}$$

which can then be partitioned as follows:

$$\begin{pmatrix} y^* \\ y^- \end{pmatrix} = \begin{pmatrix} X_p^*\beta_p \\ X_p^-\beta_p \end{pmatrix} + \begin{pmatrix} e^* \\ e^- \end{pmatrix} \tag{2.5}$$

where

y^* and X_p^* contain the concurrent records from *both* the target station vector y and the matrix of control stations X_p respectively,

and y^- and X_p^- contain all those records which have gaps either in y or one of the control stations in X_p.

Let n^* denotes the number of concurrent records from *both* the vector y and the matrix X_p. Then y^* is of dimension $(n^* \times 1)$, X_p^* is of dimension $(n^* \times p)$, y^- is of dimension $(n - n^*) \times 1$ and X_p^- is of dimension $(n - n^*) \times p$.

An unusual feature of our particular application is that n^* changes when we change the subset. This arises from the fact that each control station may be of different length and have different missing values.

After partitioning equation (2.4), we can now write the subset model containing only the complete records as:

$$y^* = X_p^*\beta_p + e^* \tag{2.6}$$

Let $\hat{\beta}$, with components $\tilde{\beta}_p$ and $\tilde{\beta}_r$, denotes the least squares estimates for the full model and let $\tilde{\beta}_p$ denotes the subset least squares estimates of β_p. That is

$$\hat{\beta} = (X^{*t}X^*)^{-1}X^{*t}y^* \tag{2.7}$$

where y^* and X^* contain only the concurrent records from *both* the target station vector y and the matrix of control stations X respectively. The least squares estimates of the regression parameters of the subset model is then given by:

$$\tilde{\beta}_p = (X_p^{*t} X_p^*)^{-1} X_p^{*t} y^* \tag{2.8}$$

2.3 SELECTION CRITERIA

The purpose of the analysis is a major factor that will influence which criterion is "best" for selecting subsets. For example, in this chapter, the average predictive variance criterion is used to achieve the "best" subset. This criterion is only suitable for non-seasonal data. Since the criterion is derived under the assumption that the regression model relating to the target and control stations is fixed, monthly data can therefore be considered for a fixed month of the year. Regression coefficients cannot be assumed to be the same for different months. Every month of the year has to be treated separately, that is, has different regression coefficients. We will concern ourselves with the annual data only. Literature about monthly data is also available (Zucchini and Sparks, 1984).

Some of the selection criteria which are in common use will be briefly reviewed.

2.3.1 The C_p Criterion

This criterion is used for the case where the control stations $X_1, X_2, \ldots, X_k$ are assumed to be *fixed*. This is the most commonly used criterion.

Mallows has suggested that the Γ_p be used as a criterion and this is defined as:

$$\Gamma_{p+1} = \frac{1}{\sigma^2}\left\{\sum_{i=1}^{n^*} \mathrm{var}(\hat{y}_{pi}^*) + \sum_{i=1}^{n^*} [\mathrm{E}(\hat{y}_{pi}^* - X_i^* \beta)^2]\right\}$$

$$= \frac{1}{\sigma^2}\left\{\mathrm{E}(\hat{Y}_p^* - X^*\beta)^t \mathrm{E}(\hat{Y}_p^* - X^*\beta)\right\} + (p+1) \tag{2.9}$$

where $\hat{Y}_p^* = X_p^* \tilde{\beta}_p$; with $\tilde{\beta}_p$ as defined in section 2.1.

It can be proved (Thompson M.L., 1978b) that (2.9) can be expressed as the following parameter of the p-station regression being considered:

$$\Gamma_{p+1} = \frac{\mathrm{E}(\mathrm{RSS}_p)}{\sigma^2} - n^* + 2(p+1) \tag{2.10}$$

where RSS_p, the residual sum of squares from a model which contains p stations, is defined as:

$$\mathrm{RSS}_p = y^{*t}y^* - \tilde{\beta}_p^t X_p^{*t} y^*$$

and $(p+1)$ is the number of parameters in the model with β_0 (intercept term) included.

If $\mathrm{E}(\mathrm{RSS}_p)$ and σ^2 are replaced by unbiased estimates, then we get an appropriate estimate of Γ_{p+1} which is defined as:

$$C_{p+1} = (\mathrm{RSS}_p/s^{*2}) - (n^* - 2(p+1)) \tag{2.11}$$

where s^{*2}, the residual mean square from the complete regression, which is an unbiased estimate of the error variance σ^2, is defined by:

$$s^{*2} = \frac{1}{(n^* - k + 2)} y^{*t}(I - X^*(X^{*t}X^*)^{-1}X^{*t})y^*$$

Mallows has shown that regressions with small bias have C_{p+1} approximately equal to $p+1$. It is therefore suggested that, for subset selection, subsets with small C_{p+1}, and C_{p+1} close to $p+1$, be considered.

2.3.2 The A_p Criterion

This criterion, like the C_p criterion, can be applied to the case where $X_1, X_2, \ldots, X_k$ are *fixed*. With this criterion, a subset of control stations is selected for each individual predictor set. In effect this implies that one would select a group of control stations for *each* missing value in the target record. Clearly this criterion is computationally more expensive that the C_p criterion where a single set of control stations is selected for *all* the missing values.

When trying to find a set of control stations which would minimize $\mathrm{MSEP}(\hat{y}_p)$, it is only necessary to find the difference between $\mathrm{MSEP}(\hat{y}_p)$ and $\mathrm{MSEP}(\hat{y}_k)$. This difference is given by

$$\Delta_p = (z\beta)^2 - z(X^{*t}X^*)^{-1}z^t\sigma^2$$

where

$$z = x - x_p(X_p^{*t}X_p^*)^{-1}X_p^{*t}X^*$$

and x is the vector of control stations' values which correspond to the missing target station value which we wish to estimate. Note that the selected model must not contain control stations for which a component of x is missing. To ensure this it may be necessary to eliminate some of the control stations for a given value of x, i.e. to redefine X^* in such a way that only feasible control stations can be selected.

If Δ_p is negative, then this would indicate that the p-station subset is preferable to the full k station subset.

An asymptotically unbiased estimate of Δ_p is

$$A_p = (z\hat{\beta})^2 - 2z(X^{*t}X^*)^{-1}z^t s^2 \tag{2.12}$$

The subset which produces the largest A_p value is considered the "best" for predicting the future response value at the given data point.

2.3.3 The Square Multiple Correlation Coefficient

This criterion has been used widely in the past, and is defined as:

$$R_p^2 = 1 - \frac{\text{RSS}_p}{\text{CTSS}} \tag{2.13}$$

where RSS_p is as defined in equation (2.10)

CTSS, the corrected total sum of squares, is given by: $y^t y - n\bar{y}^2$.

R_p^2 takes on values between zero and one inclusive. The closer R_p^2 is to one, the higher the proportion of the variability in y which is explained by X_p, the subset of the control stations under consideration. Generally, by increasing the number of variables in the subset, R_p^2 also increases. It is therefore likely that the selected subset will be that of the full model and for that reason we have to find a suitable subset with high R_p^2. (There may be cases where a smaller subset is selected because we are dealing with incomplete control station records.)

The relationship between R_p^2 and C_{p+1} is discussed in Hocking (1976).

2.3.4 Adjusted R_p^2

This is a criterion which is closely related to R_p^2 and is defined by:

$$\begin{aligned}\bar{R}_p^2 &= 1 - \frac{(\mathrm{RSS}_p)/(n^* - p)}{(\mathrm{CTSS})/(n^* - 1)} \\ &= 1 - (1 - R_p^2)\left(\frac{n^* - 1}{n^* - p}\right)\end{aligned} \tag{2.14}$$

The adjustment has been done according to the degrees of freedom involved in RSS_p and CTSS. We note that if n^* is small compared to p, then $\bar{R}_p^2$ can be negative. Unlike the criterion considered in section 2.3.3, $\bar{R}_p^2$ does not continue to increase as the number of stations in the subset model are increased.

The relationship between $\bar{R}_p^2$ and C_p criteria is discussed by Kennard (1971). When $\bar{R}_p^2 \geq \bar{R}_k^2$ then the subset models considered are estimated to have zero (or negligible) bias. That is $\bar{R}_p^2 \geq \bar{R}_k^2$ is equivalent to $C_{p+1} \leq p + 1$.

For estimation, the p-station subset which has the maximum $\bar{R}_p^2$ is then considered to be the "best" for estimating the target station.

2.3.5 The Residual Mean Square

This criterion is defined as

$$\mathrm{MSE}_p = \mathrm{RSS}_p/(n^* - p - 1) \tag{2.15}$$

where MSE_p is the residual mean square for the p-station equation.

Its minimization for the subset model, can be a useful selection criterion.

The condition

$$\mathrm{MSE}_p \leq \mathrm{MSE}_k$$

is equivalent to $\bar{R}_p^2 \geq \bar{R}_k^2$ or $C_{p+1} \leq p + 1$.

2.3.6 The Standardized Residual Sum of Squares

The definition of this criterion is given by:

$$\begin{aligned} \mathrm{RSS}_p^* &= e_p^t \left\{ \mathrm{Diag}(I - X_p^*(X_p^{*t}X_p^*)^{-1}X_p^{*t}) \right\}^{-1} e_p \\ &= e_p^t D_p^{-1} e_p \end{aligned} \tag{2.16}$$

where

$$\begin{aligned} e_p &= y^* - \hat{y}_p^*, \\ \hat{y}_p^* &= X_p^*(X_p^{*t}X_p^*)^{-1}X_p^{*t}y^*, \end{aligned}$$

and

$$D_p = \mathrm{Diag}(I - X_p^*(X_p^{*t}X_p^*)^{-1}X_p^{*t}).$$

The subset model which minimizes $\mathrm{E}(\mathrm{RSS}_p^*)$ is considered the "best".

2.3.7 The Prediction Sum of Squares

The prediction sum of squares criterion is defined as:

$$\mathrm{PRESS}_p = \sum_{i=1}^{n^*} (y_i^* - \hat{y}_{p(i)}^*)^2$$

where $\hat{y}_{p(i)}^* = x_p^* \tilde{\beta}_p$ with the ith observation excluded when computing $\tilde{\beta}_p$.

If we let e_p denote the vector of residuals for the p-station equation, and D_p as defined in equation (2.16), then it can be shown that

$$\mathrm{PRESS}_p = e_p^t D_p^{-2} e_p \tag{2.17}$$

As a criterion for determining a subset model, PRESS_p is evaluated for all possible subsets and a selection based on minimum values of PRESS_p is made.

PRESS_p is closely related to RSS_p^*. It can be seen from equations (2.16) and (2.17) that they are both weighted sum of squares of the residuals and it would therefore not be easy to compare them with the other criteria. The "best" model will have a comparatively small PRESS but not involve too may stations.

2.3.8 Average Predictive Variance

This criterion can be applied to the case where the past observations are considered to be *fixed* and the future ones *random*. This is a group criterion in the sense that it uses the average of the data. An estimate of the average predictive variance is given by:

$$J_p = (n^* + p + 1)\text{MSE}_p/n^* \tag{2.18}$$

It is clear that J_p arises by computing the average predictive variance over the current data for a particular subset and then σ^2 is estimated by MSE_p.

Since this criterion uses the subset model in predicting the response variable, it ignores the bias term. When using J_p, special emphasis is placed on the observed data. The subsets with small values of J_p are selected.

2.3.9 The T_p Criterion

This criterion, like the J_p criterion, can be applied to the case where the past observations are considered to be *fixed* and the future ones *random*. The T_p criterion is for predicting at a point, where the x values at which one wants to predict are known, that is, it is the point version of the J_p criterion. In this criterion, a subset of control stations is chosen for each predictor set.

This criterion is defined as:

$$T_p = (1 + x_p(X_p^{*t}X_p^*)^{-1}x_p^t)\text{MSE}_p \tag{2.19}$$

where σ^2 is estimated by MSE_p.

As when using J_p, special emphasis is placed on the observed data. The subset with small value of T_p is selected and this subset is used for predicting the future response value at the given data point. This is useful because the value of x_p is known for each missing y.

2.3.10 The S_p Criterion

This criterion is used when the control stations are *random*. It is taken as the most suitable for selection of stations in multivariate regression analysis when the assumption that the target station y and the k control stations $X_1, X_2, \ldots, X_k$ are $(k+1)$ dimensional normally distributed, is true. The criterion used is that which minimizes the expected mean square error of prediction (MSEP) where

$$\text{MSEP}(\hat{y}_p) = \text{E}_y(y - \hat{y}_p)^2 \tag{2.20}$$

in which $\hat{y}_p$ is the predicted value of y corresponding to a set of p control stations in section 2.1.

The expected value of the MSEP is then calculated over all regression samples and predictor sets of the p stations. This is estimated by E_p which is defined as:

$$\text{E}_p = \frac{\text{RSS}_p}{n^*(n^* - p)} \left(1 + n^* + \left\{ \frac{p(n^*+1)}{n^* - p - 2} \right\}\right)$$

A criterion which minimizes the $\text{MSEP}(\hat{y}_p)$ is then

$$S_p = \text{RSS}_p/(n^* - p)(n^* - p - 2) \tag{2.21}$$

The subset of control stations which gives a minimal S_p over all p, is then considered the "best" regression model for estimating the target station.

2.4 SELECTION PROCEDURES

There are various procedures which are used to identify the stations to be included in the regression equation. These procedures can be applied when the control stations are assumed to be *random* or *fixed*. A brief explanation of five of these commonly used procedures is given below, although only the first three procedures - which are recommended for the type of data with which we are dealing - will be discussed in detail.

2.4.1 All possible control stations at a point (Section 2.5)

This requires the fitting of every possible regression equation that can be obtained by selecting $0, 1, \ldots, p$ of the control stations. It selects the "best" model for *each* missing data point individually using an exhaustive search. The results is that, for different missing data points, different sets of the control stations may be selected for use in patching the same target station.

2.4.2 All possible control stations for several values (Section 2.6)

This selects the set of control stations which, when compared with the other sets, is reckoned to perform "best" on average. The average referred to here, is taken over the observed data points and not the estimated missing values. By this compromise, the computing cost is greatly reduced.

2.4.3 Forward Selection (Section 2.7)

This is a method whereby, at each step, a single station is added to the current regression model until the "best" model is achieved. The process is performed for each missing data point individually. The aim is to reduce computing cost, although it cannot be guaranteed that the "best" models will be selected. This procedure is recommended if there is a high degree of multicollinearity; that is, if there is strong intercorrelation, which means that the best subset is expected to contain only a few control stations (Mantel, 1970).

2.4.4 Backward Elimination

This procedure is similar to the Forward selection procedure except that it starts with the full model and then eliminates the least important control stations at each stage. Mantel (1970) recommends the use of this procedure if, by applying the all subsets procedure, it is found that the best subset contains most of the control stations. On the other hand, Beale (1970) points out that when using this procedure, a station may be eliminated at an early stage when it is in fact an important station. This may happen because some of the stations might have high "nonsense correlation".

2.4.5 Stepwise Regression Procedure

This procedure is an extension of the Forward Selection procedure. At each stage, a station may be deleted or included. It starts the same as in Forward Selection. At each stage, after the current equation has at least two stations in, we consider four alternatives: add a station, delete a station, exchange two stations, or stop.

2.5 SELECTING CONTROL RECORDS FOR INDIVIDUAL MISSING VALUES

This procedure requires the fitting of every possible regression equation for each missing value individually. This is based on the belief that the best model for one missing data point might not be the best for the other missing data points. The criterion which is used to choose the equation that has the "best" predictors, is the estimated predictive variance, i.e. the variance of the estimator of the missing value. This criterion is computed for every possible subset of the set of control stations and a comparison is performed after each computation.

The total number of control stations is k, and since each $X_i, i = 1, \ldots, k$ can either be or not be in the equation, therefore there are $2^k - 1$ possible subsets on which we can base our prediction. The other subset is obtained when none of the control stations are used and the sample average of the target station is used to estimate the missing value. Altogether, there are 2^k possible subsets.

The following steps describe how the predictive variance is estimated for a subset of p control stations where $1 \leq p \leq k$. Suppose that, before the subset under discussion can be considered, the control stations are re-ordered in such a way that the subset consists of the first p records. Suppose that y_ℓ is the ℓth observation in the target station which we wish to estimate.

STEP 1

Check that this subset is able to estimate y_ℓ, i.e. that none of the data points $x_{\ell 1}, x_{\ell 2}, \ldots, x_{\ell p}$ are missing. If any of these data points are missing, then reject the subset. It is no longer necessary to continue with further computation.

STEP 2

Check for concurrent records from *both* the vector y and the matrix X_p where

$$X_p = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{pmatrix}$$

and y still the same. Eliminate from *both,* all the rows which have one or more missing observations in *either.* For example, if y_j is missing, then remove the jth row from *both* y and X_p. Similarly if x_{st} is missing, then remove the sth row from *both* the y vector and the matrix X_p, and so on. Let the remaining data points in the vector y be vector y^* and the remaining data points in the matrix X_p be matrix X_p^*. If y^* has n^* observations, then X_p^* is an $(n^* \times p)$ matrix.

STEP 3

Compute

$$\mathrm{V} = [1 + x_\ell \, \mathrm{A} \, x_\ell^t] \, \hat{\sigma}^2$$

where

$$\hat{\sigma}^2 = y^{*t} y^* - y^{*t} X_p^* \tilde{\beta}_p,$$
$$\mathrm{A} = (X_p^{*t} X_p^*)^{-1},$$
$$\tilde{\beta}_p = \mathrm{A} X_p^{*t} y^*,$$

and

$$x_\ell = (x_{\ell 1}, x_{\ell 2}, \dots, x_{\ell p})$$

If we suppose that there are $k = 4$ control stations labelled $1, 2, 3$ and 4, then for $p = 1$ we will have the subsets $\{1\}$, $\{2\}$, $\{3\}$, $\{4\}$, for $p = 2$ the subsets $\{1,2\}$, $\{1,3\}$, $\{1,4\}$, $\{2,3\}$, $\{2,4\}$, $\{3,4\}$, for $p = 3$ the subsets $\{1,2,3\}$, $\{1,2,4\}$, $\{2,3,4\}$, $\{1,3,4\}$ and for $p = 4$, the subset $\{1,2,3,4\}$ is obtained. It is clear that this procedure should at least find, for each value of $p = 1, 2, \dots, k$, that p regressor subset, from among all the $\binom{k}{p}$ possible p control station models, which has the smallest predictive variance.

If the chosen subset is an empty subset, then the variance of the "*degenerate model*" corresponding to it is computed as:

$$V = \frac{1}{n^*} \sum_{i=1}^{n^*} y_i^{*2} - \bar{y}^{*2}$$

where

$$\bar{y}^{*2} = \frac{1}{n^*} \sum_{i=1}^{n^*} y_i^*$$

After all the subsets have been obtained, the subset with the smallest estimated predictive variance is then selected and used to estimate the missing observation as follows:

$$\hat{y}_\ell = x_\ell A^{*t} y^*$$

where $\hat{y}_\ell$ will be the new estimated missing value and x_ℓ, A, X_p^* and y^* are vectors and matrices computed in *Steps 2* and *3* for the selected subset. If the selected subset was empty, then the missing observation is estimated by the mean, that is, $\hat{y}_\ell = \bar{y}^*$. The standard deviation of y_ℓ is simply the square root of the selection criterion for the selected model.

2.6 SELECTING CONTROL RECORDS FOR SEVERAL MISSING VALUES

When using the procedure described in section 2.5, one needs to fit 2^k regression models for every missing data point. Therefore, if there are m missing data points, then $m2^k$ estimates of the predictive variance are computed. In the procedure discussed here, the number of estimates is reduced from $m2^k$ to only 2^k. This reduction of the number of estimates is a compromise, in that we find the model which would have best predicted the observed target station data points and assume that this model will be good for predicting the missing data points.

It can occur that the single best model is not capable of estimating all the missing data points in the target station, because one or more of the selected control stations may also be missing some of the relevant data points. It is therefore advisable not to lose any of the computed predicted variances and their relevant subsets. We keep

track of which model is second best, third best, and so on. After a hierarchy has been formed, the missing data point is then estimated by using the highest member of the hierarchy which has all the relevant data points or observed values.

The following steps show how the predictive variance, for a particular subset of p control stations is computed. We will again, for convenience, suppose that the k control stations have been re-ordered so that the p records under consideration appear first in the matrix X.

STEP 1

Construct the vector y^* and the matrix X_p^* and define n^* as in *Step* 2 of section 2.5.

STEP 2

Compute

$$V = (n^* + p)\hat{\sigma}^2/n^*$$

where

$$\hat{\sigma}^2 = y^{*t}y^* - y^{*t}X_p^*\tilde{\beta}_p,$$
$$\tilde{\beta}_p = AX_p^{*t}y^*,$$
$$A = (X_p^{*t}X_p^*)^{-1}$$

For the "*degenerate model*", the predictive variance V, is computed as in Section 2.5, in which this procedure is performed for each of the $\binom{k}{p}$ subsets of size p, for each $p = 1, 2, \ldots, k$. After the 2^k estimated predictive variance are computed, they are then sorted in ascending order. The subset which led to the smallest V is used to estimate as many missing data points as it is capable, if not, then the second best model is used, and so on.

The estimator of a missing value, y_ℓ, is then given by

$$\hat{y}_\ell = x_\ell A X_p^{*t} y^*,$$

where A, X_p^* and y^* are the vectors and matrices computed in *Step 2* for the subset being used for estimation and x_ℓ remains as defined in *Step 3* of section 2.5. An estimator of the standard deviation of y_ℓ is given by

$$(1 + x_\ell A x_\ell^t)^{1/2}\hat{\sigma},$$

where A and $\hat{\sigma}$ are computed as in *Step 2*, for the subset being used for estimation. Similarly, for the "*degenerate model*", $\hat{y}_t$ and its standard deviation are estimated as in section 2.5.

2.7 FORWARD SELECTION

This procedure uses a computational algorithm which limits possible models to a relatively small number. We only consider one of the stepwise procedures, namely *Forward Selection*, because it is expected that the best subset will contain only a few control stations due to the high degree of multicollinearity (Mantel (1970), cited in Zucchini and Sparks (1984)). By using this selection procedure, the number of computed predictive variances which was found to be $m2^k$ in section 2.5 and 2^k in section 2.6, is then reduced to only a maximum of $m(k(k+1)/2+1)$. This procedure examines only a few subsets of each size. One control station is added to the current regression model at each step until a stopping rule is met. The same criterion as applied in the previous procedures, is used as a stopping rule. The steps which follows show how the criterion is applied in forward selection:

STEP 1

Compute the predictive variance for the "*degenerate model*" and let it be V_0, i.e.

$$V_0 = \frac{1}{n^*}\sum_{i=1}^{n^*} y_i^{*2} - \bar{y}^{*2}.$$

STEP 2

For each of the subsets which contain exactly one control station, compute the predictive variance V. Denote the smallest of the computed predictive variances by V_1 and the corresponding control station by G_1.

STEP 3

If $V_{p-1} < V_p$, then the algorithm terminates and the set of control records corresponding to V_{p-1} is used for filling in the missing data point. Otherwise proceed to *Step 4*.

STEP 4

If $p = k$ then the algorithm terminates and the full set of control stations is selected. Otherwise increase p by 1 and proceed to *Step 5*.

STEP 5

Compute the predictive variance, V, for each of the subsets of size p which contain $G_1, G_2, \ldots, G_{p-1}$; there will be $k-p+1$ such subsets. Denote the smallest value of V obtained by V_p and the new corresponding control station by G_p and go to *Step 3*.

Note that this algorithm can terminate before $p = k$. In practice a value of more than 5, even if k is large, is seldom reached by p. Therefore the number of predictive variances which need to be estimated is generally less than $5(k-3)+1$ (Zucchini and Sparks, 1984).

2.8 RECOMMENDATION

The J_p and the S_p criteria are appropriate in situations where the predictor variables are *random* (as opposed to *fixed* as would be the case if we were in a position to specify their values, as one can do, for example, in an experimental situation). Technically these two criteria differ in that they estimate different expectations. The latter estimates an expectation taken over both the *past* and *future* observations, whereas the J_p estimates the expectation taken over future observations of y at *fixed* target values, with the *past* as *random*. This, as well as the fact that the J_p criterion does not require any assumption of normality of the data, made it the (slightly) more attractive option for our application. However the main results and conclusion of this study are unlikely to differ substantially if the S_p criterion were used instead.

Note that the J_p method does not necessarily lead to a model which can fill all the gaps in the target record. The criterion simply selects that subset of control stations which is estimated to perform best on average. However, it may be the case that for some particular gaps there are concurrent gaps in one or more of the selected control stations. In such cases, one has to consider selecting a set of control stations for each individual remaining gaps. The T_p criterion, which is a point version of

the J_p criterion, has been developed for this purpose (Section 2.3.9).

CHAPTER 3

THE EM ALGORITHM

3.1 INTRODUCTION

In chapter 2, we discussed selection of control stations in regression analysis as one of the methods of estimation in incomplete data problems. In this chapter, we discuss a method known as the EM algorithm, which is a very general iterative method for maximum likelihood estimation in incomplete data problems. Although this method has been proposed as early as the late 1950's (Hartley, 1958), the term EM was introduced by Dempster, Laird and Rubin (1977) (henceforth DLR). The work of DLR has exposed the full generality of the algorithm by proving general results about its behaviour and providing a wide range of examples.

The EM algorithm comprises the following steps:

1. missing values are replaced by estimated values,
2. parameters are estimated,
3. missing values are re-estimated assuming that new parameter estimates are correct,
4. parameters are re-estimated and so forth, iterating until convergence.

DLR applied their theory to a similar but somewhat simpler case to the one which we are considering. In effect they demonstrated the use of the EM algorithm for the case where there are missing observations in one of the variables. In our context, this would cover the situation where there are missing values at the target station but no missing values at the control stations. Since practically all control stations also have missing values in practice, it is necessary for us to extend their results to cover this case.

In applications in the literature of the EM algorithm the focus of attention is on estimating the parameters of the model when some observations are missing. In our application however the focus of interest is in the missing values themselves rather than the model parameters. So for example we have based our convergence criteria on the estimated missing values rather than on the successive parameter estimates.

We also point out that the methodology covered in this chapter is suitable for estimating all the missing values for a *set* of stations, rather than at a given target station. Each station in the set cycles between being a target station and a control station. The two algorithms presented here automatically estimate all the missing values in the set, they cannot be used to estimate only those at a given station without estimating all the missing values at the other stations in the set.

We note also that in order to apply the EM algorithm the data has to meet certain requirements. In particular, the complete (non-missing) records must overlap sufficiently, that is there must be sufficient concurrent records. Essentially there must be enough overlap for one to be able to regress the observations of the target station on those of (all) the control stations. If this requirement is not met it is necessary to eliminate some of the control stations to ensure that it is.

We will give details pertaining to the case where the observations are normally distributed. In theory it would be possible to apply the EM algorithm to other distributions but substantial theoretical work would have to be carried out to implement this. Fortunately the application with which we are dealing, annual rainfall totals or monthly rainfall totals, can be reasonably modelled using either the normal distribution or alternatively the log-normal. In the case of the latter one simply works with the logs of the observations. However it may be necessary in some cases to apply a different transformation to achieve approximate normality.

This chapter contains an outline of the definition and theory of the EM algorithm. For completeness we have included the case of the general EM algorithm, i.e. the case covering non-exponential families.

We give two methods to implement the algorithm for our application. The first (Method 1 of section 3.8) is based on the "standard steps" of the EM algorithm. The second (Method 2 of section 3.8) is a variation which has not been previously considered. It has been developed in order to simplify the algorithm and thus to make it more efficient.

Our experiments have shown that although Method 2 requires more iterations to converge, it is substantially more efficient than Method 1 in terms of computing effort.

A step by step application of the two algorithms is given in Appendix A. The purpose of giving this is to illustrate precisely how the algorithms are implemented in practice and also to provide a numerical check for users who might wish to prepare software to apply them.

3.2 GENERAL DESCRIPTION OF THE EM ALGORITHM

The EM algorithm is a method which iteratively computes the maximum likelihood estimates when some observations are missing, i.e. when dealing with incomplete data. Let Z be a matrix of n observations on k stations, where $k \geq 2$ and $n \geq k+2$. Suppose that we have the complete data set Z, where Z is matrix valued which contains two or more rainfall stations. We assume that the data is generated by a model described by a density function $f(Z|\phi)$, indexed by unknown parameter ϕ. Given the model and parameter vector ϕ, $f(Z|\phi)$ is a function of Z, that is, of the observations.

Definition 3.1: The likelihood function $L(\phi|Z)$ is any function of ϕ which is proportional to $f(Z|\phi)$ when given the data value Z.

It should be noted that one regards the likelihood function as a function of the parameter ϕ for given Z, whereas the density function $f(Z|\phi)$ is regarded as a function of Z for fixed ϕ.

It is usually more convenient to work with the log-likelihood function than with the likelihood function. We denote the log-likelihood function by

$$\ell(\phi, Z) = ln\ \mathrm{L}(\phi|Z).$$

Let $Z = (Z_{obs}, Z_{mis})$ where Z_{obs} denotes the observed values of Z and Z_{mis} denotes the missing values of Z. Write

$$Z_{obs} = (z_{obs,1}, z_{obs,2}, \ldots, z_{obs,n})$$

where $z_{obs,i}$ represents the set of stations having observation at $i, i = 1, 2, \ldots, n$.

Let $f(Z|\phi) \equiv f(Z_{obs}, Z_{mis}|\phi)$ denote the density function of the joint distribution of Z_{obs} and Z_{mis}. To obtain the marginal probability density of Z_{obs}, the missing data Z_{mis} is integrated out. That is:

$$f(Z_{obs}|\phi) = \int f(Z_{obs}, Z_{mis}|\phi) dZ_{mis} \tag{3.1}$$

The likelihood function of ϕ based on Z_{obs} is defined to be any function of ϕ proportional to $f(Z_{obs}|\phi)$:

$$L(\phi|Z_{obs}) \propto f(Z_{obs}|\phi)$$

In situations where values are missing at random, $L(\phi|Z_{obs})$ is called the true likelihood of ϕ based on the observed data Z_{obs}.

By making use of the complete data specification $f(Z|\phi)$, the EM algorithm is used to estimate the parameter ϕ which maximizes $f(Z_{obs}|\phi)$. In other words, we try to maximize the likelihood function

$$L(\phi|Z_{obs}) = \int f(Z_{obs}, Z_{mis}|\phi) dZ_{mis} \tag{3.2}$$

with respect to ϕ.

3.3 DEFINITION OF THE EM ALGORITHM

The EM algorithm has a useful and simple interpretation when the complete data Z has a distribution from the *regular exponential family* defined by

$$f(Z|\phi) = \frac{b(Z)\exp(\phi t(Z)^t)}{a(\phi)} \tag{3.3}$$

where

ϕ denotes a $(1 \times r)$ vector of parameters,

$t(Z)$ denotes a $(1 \times r)$ vector of complete data sufficient statistics,

and

a and b are functions of ϕ and Z respectively.

The parameterization of ϕ in (3.3) is unique up to an arbitrary non-singular $(r \times r)$ linear transformation as is the corresponding choice of $t(Z)$.

In this chapter, we restrict our attention to only one class of the exponential type of distribution, namely, the *Multivariate Normal* distribution. We say that a distribution is *Multivariate Normal* if its density function is given by:

$$f(z|\mu,\Sigma) = (2\pi)^{-k/2}|\Sigma|^{-1/2}\exp[-\frac{1}{2}(z-\mu)^t\Sigma^{-1}(z-\mu)] \tag{3.5}$$

where

$$z^t = (z_1 \quad z_2 \quad \dots \quad z_k), \tag{3.5}$$

$$\mu = (\mu_1 \quad \mu_2 \quad \dots \quad \mu_k), \tag{3.6}$$

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \dots & \cdot & \dots & \sigma_{1k} \\ \vdots & & \vdots & & \vdots \\ \cdot & \dots & \sigma_{ij} & \dots & \cdot \\ \vdots & & \vdots & & \vdots \\ \sigma_{k1} & \dots & \cdot & \dots & \sigma_k^2 \end{pmatrix} \tag{3.7}$$

where σ_{ij} is the covariance of the ith and jth component of Z.

Suppose we are dealing with more than one set of observations, that is, we have a matrix of n sets of observations such that

$$Z = \begin{pmatrix} z_{11} & z_{12} & \dots & z_{1k} \\ z_{21} & z_{22} & \dots & z_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & \dots & z_{nk} \end{pmatrix} \tag{3.8}$$

The likelihood of the observations (3.8) is

$$L(\mu,\Sigma|Z) = (2\pi)^{-nk/2}|\Sigma|^{-n/2}\exp\left[-1/2\sum_{i=1}^{n}(z_i-\mu)^t\Sigma^{-1}(z_i-\mu)\right] \tag{3.9}$$

Using (3.9) we can find the sufficient statistics for the parameters.

$$
\begin{aligned}
&L(\mu, \Sigma | Z) \\
&\quad = (2\pi)^{-nk/2} |\Sigma|^{-n/2} \exp[-\frac{1}{2} n \mathrm{tr}(\mu^t \mu \Sigma^{-1})] \\
&\qquad \exp\left[-\frac{1}{2}\sum_{i=1}^{n} \mathrm{tr}\left(z_i \quad z_i z_i^t\right)\begin{pmatrix} -2\Sigma^{-1} \\ \Sigma^{-1} \end{pmatrix}\right] \\
&\quad = (2\pi)^{-nk/2} |\Sigma|^{-n/2} \exp[-\frac{1}{2} n \mathrm{tr}(\mu^t \mu \Sigma^{-1})] \\
&\qquad \exp\left[-\frac{1}{2}\sum_{i=1}^{n}\left(\sum_{j=1}^{k}\sum_{\ell=1}^{k}[z_{ij}\mu_\ell \sigma_{j\ell} + z_{ij} z_{i\ell} \sigma_{j\ell}]\right)\right] \\
&\quad = (2\pi)^{-nk/2} |\Sigma|^{-n/2} \exp\left[-\frac{1}{2} n \mathrm{tr}(\mu^t \mu \Sigma^{-1})\right] \\
&\qquad \exp\left[-\frac{1}{2}\sum_{i=1}^{n}(1_n \otimes z_i)^t \begin{pmatrix} \mu_1 \underline{\sigma}_1 \\ \vdots \\ \mu_k \underline{\sigma}_k \end{pmatrix} - \frac{1}{2}\sum_{i=1}^{n}(z_i \otimes z_i)^t \underline{\sigma}_c\right]
\end{aligned}
$$

Therefore

$$t(Z) = \begin{pmatrix} \sum_{i=1}^{n} 1_n \otimes z_i \\ \sum_{i=1}^{n} z_i \otimes z_i \end{pmatrix} \tag{3.10}$$

$$\phi = \begin{pmatrix} \mu_1 \underline{\sigma}_1 \\ \vdots \\ \mu_k \underline{\sigma}_k \\ -\frac{1}{2}\underline{\sigma}_1 \\ \vdots \\ -\frac{1}{2}\underline{\sigma}_k \end{pmatrix} \tag{3.11}$$

where

ϕ is a vector of parameters, and

$t(Z)$ is the sufficient statistic for ϕ since it does not depend on any parameter.

Because the statistic $t(Z)$ is sufficient for the parameter ϕ, it therefore has all the relevant information contained in Z for inference about the parameter.

3.4 THE E STEP AND THE M STEP OF EM

Each iteration of the EM algorithm involves two steps which are called the expectation step (**E step**) and the maximization step (**M step**). Here follows the steps which may be applied if equation (3.9) satisfies the conditions of it being a class of the exponential type of distribution.

Suppose that $\phi^{(p)}$ denotes the current value of ϕ after p cycles of the algorithm. The next cycle involves the following two steps:

E step: At the $(p+1)$st cycle, the E step is the computation of the conditional expectation of the complete data sufficient statistics given:

(i) the observed data $Z_{obs} = (z_{obs,1}, \cdots, z_{obs,n})$, and

(ii) the estimated value of the parameter from the pth cycle.

That is we compute

$$t^{(p)} = E[t(Z)|Z_{obs}, \phi^{(p)}] \tag{3.12}$$

where the superscript (p) denotes the pth cycle.

M step: At the $(p+1)$st cycle, the M step is the maximization of the complete data likelihood function in which the complete data sufficient statistics $t(Z)$ has been replaced by its conditional expectation obtained in the E step. We set the derivatives of the complete data likelihood function to zero and determine $\phi^{(p+1)}$, i.e. as the solution of the equation

$$E(t(Z)|\phi) = t^{(p)} \tag{3.13}$$

which defines the maximum likelihood estimator of ϕ under the assumption that (3.9) is a class of the exponential type of distribution.

We now show how the E and M steps of the EM algorithm are obtained under the assumption that the distribution is *multivariate normal.* Note that what we want to find is the value ϕ^* of ϕ which maximizes the log-likelihood function of the incomplete data function. That is

$$L(\phi|Z_{obs}) = f(Z_{obs}|\phi) \tag{3.14}$$

where $f(Z_{obs}|\phi)$ is as defined by (3.1)

DLR have shown how the E and M steps were found for any class of the exponential family. Here, since we are concerned with the *Multivariate Normal* distribution, to define our expectation step, we find the expected value of the sufficient statistic (3.10).

Therefore if, at the pth iteration (cycle), $\phi^{(p)}$ denotes the current estimates of the parameters, then the E step of the algorithm consists of calculating:

$$\mathrm{E}\left(\sum_{i=1}^{n} z_{ij}|Z_{obs},\phi^{(p)}\right) = \sum_{i=1}^{n} z_{ij}^{(p)}, \qquad j = 1,2,\ldots,k. \tag{3.15}$$

$$\mathrm{E}\left(\sum_{i=1}^{n} z_{ij}z_{i\ell}|Z_{obs},\phi^{(p)}\right) = \sum_{i=1}^{n} z_{ij}^{(p)}z_{i\ell}^{(p)} + c_{j\ell i}^{(p)}, \quad j,\ell = 1,\ldots,k, \tag{3.16}$$

where:

$$\begin{aligned} z_{ij}^{(p)} &= z_{ij} && \text{if } z_{ij} \text{ is observed} \\ &= \mathrm{E}(z_{ij}|z_{obs,i},\phi^{(p)}) && \text{if } z_{ij} \text{ is missing} \end{aligned}$$

and

$$\begin{aligned} c_{j\ell i}^{(p)} &= 0 && \text{if } z_{ij} \text{ or } z_{i\ell} \text{ are observed} \\ &= \mathrm{Cov}(z_{ij},z_{i\ell}|z_{obs,i},\phi^{(p)}) && \text{if } z_{ij} \text{ and } z_{i\ell} \text{ are missing} \end{aligned}$$

Missing values z_{ij} are therefore replaced by the conditional mean of z_{ij} given the set of values $z_{obs,i}$ observed for that observation.

Similarly, the maximization step (M step) is found from equation (3.10). The new estimates $\phi^{(p+1)}$ of the parameters are estimated as follows:

$$\mu_j^{(p+1)} = \frac{1}{n}\sum_{i=1}^{n} z_{ij}^{(p)}, \quad j = 1,2,\ldots,k. \tag{3.17}$$

$$\begin{aligned} \sigma_{j\ell}^{(p+1)} &= 1/nE\left(\sum_{i=1}^{n} z_{ij}z_{i\ell}|Z_{obs}\right) - \mu_j^{(p+1)}\mu_\ell^{(p+1)} \\ &= \frac{1}{n}\sum_{i=1}^{n}\left[(z_{ij}^{(p)} - \mu_j^{(p+1)})(z_{i\ell}^{(p)} - \mu_\ell^{(p+1)}) + c_{j\ell i}^{(p)}\right], \qquad j,\ell = 1,2,\ldots,k. \end{aligned} \tag{3.18}$$

3.5 THE GENERAL EM ALGORITHM (not necessarily Exponential family)

Suppose that we do not know under which distribution family the complete data specification falls. Then the EM iteration $\phi^{(p)} \rightarrow \phi^{(p+1)}$ is defined in the following way:

Let $\phi^{(p)}$ be the current estimate of the parameter ϕ. Then

E step: Find the expected log-likelihood function if ϕ were $\phi^{(p)}$:

$$Q(\phi|\phi^{(p)}) = \int \ell(\phi|Z) f(Z_{mis}|Z_{obs}, \phi = \phi^{(p)}) dZ_{mis}. \tag{3.19}$$

M step: Determine $\phi^{(p+1)}$ by maximizing this expected log-likelihood function:

$$Q(\phi^{(p+1)}|\phi^{(p)}) \geq Q(\phi|\phi^{(p)}) \qquad \text{for all } \phi. \tag{3.20}$$

3.6 THEORY OF THE EM ALGORITHM

The distribution of the complete data Z can be factored as follows:

$$f(Z|\phi) = f(Z_{obs}, Z_{mis}|\phi) = f(Z_{obs}|\phi) f(Z_{mis}|Z_{obs}, \phi) \tag{3.21}$$

where

$f(Z_{obs}|\phi)$ is as defined in equation (3.1),

$f(Z_{mis}|Z_{obs}, \phi)$ is the density of the missing data given the observed data.

The log-likelihood function of the complete data is then defined as:

$$\begin{aligned} \ell(\phi|Z) &= \ell(\phi|Z_{obs}, Z_{mis}) \\ &= \ell(\phi|Z_{obs}) + ln\ f(Z_{mis}|Z_{obs}, \phi) \end{aligned} \tag{3.22}$$

By maximizing the incomplete data log-likelihood $\ell(\phi|Z_{obs})$ with respect to ϕ for fixed Z_{obs}, we wish to estimate ϕ. We write

$$\ell(\phi|Z_{obs}) = \ell(\phi|Z) - ln\ f(Z_{mis}|Z_{obs}, \phi) \tag{3.23}$$

where

$\ell(\phi|Z_{obs})$ is the log – likelihood of the observed data to be maximized,

$\ell(\phi|Z)$ is the log – likelihood of the complete data,

and

$ln\ f(Z_{mis}|Z_{obs},\phi)$ is the log – likelihood function of the missing part of the complete data.

The expectation of both sides of equation (3.23) over the distribution of the missing data Z_{mis}, given the observed data Z_{obs}, and a current estimate of ϕ say $\phi^{(p)}$ is

$$\ell(\phi|Z_{obs}) = Q(\phi|\phi^{(p)}) - H(\phi|\phi^{(p)}) \tag{3.24}$$

where

$Q(\phi|\phi^{(p)})$ is as defined by equation (3.19),

and

$$H(\phi|\phi^{(p)}) = \int [ln\ f(Z_{mis}|Z_{obs},\phi)] f(Z_{mis}|Z_{obs},\phi^{(p)}) dZ_{mis} \tag{3.25}$$

Consider a sequence of iterates $\phi^{(0)},\phi^{(1)},\ldots,$ where $\phi^{(p+1)} = M(\phi^{(p)})$ for some function M(.). At successive iterates, the difference in values of $\ell(\phi|Z_{obs})$ is given by

$$\begin{aligned}\ell(\phi^{(p+1)}|Z_{obs}) - \ell(\phi^{(p)}|Z_{obs}) &= [Q(\phi^{(p+1)}|\phi^{(p)}) - Q(\phi^{(p)}|\phi^{(p)})] \\ &\quad - [H(\phi^{(p+1)}|\phi^{(p)}) - H(\phi^{(p)}|\phi^{(p)})]\end{aligned} \tag{3.26}$$

In the EM algorithm, $\phi^{(p+1)}$ is chosen so as to maximize $Q(\phi|\phi^{(p)})$ with respect to ϕ. In general, a Generalized EM (GEM) algorithm chooses $\phi^{(p+1)}$ so that $Q(\phi^{(p)}|\phi^{(p)}) < Q(\phi^{(p+1)}|\phi^{(p)})$. The following results regarding the EM algorithm convergence, are in the papers by Dempster, Laird and Rubin (1977), Wu (1983) and Boyles (1982).

1. Every GEM algorithm increases $\ell(\phi|Z_{obs})$ at each iteration, that is

$$\ell(\phi^{(p+1)}|Z_{obs}) \geq \ell(\phi^{(p)}|Z_{obs})$$

with equality if and only if

$$Q(\phi^{(p+1)}|\phi^{(p)}) = Q(\phi^{(p)}|\phi^{(p)}).$$

2. If for some ϕ^* in the parameter space of ϕ, $\ell(\phi^*|Z_{obs}) \geq \ell(\phi|Z_{obs})$ for all ϕ, then for every GEM algorithm,

$$\ell(\mathrm{M}(\phi^*)|Z_{obs}) = \ell(\phi^*|Z_{obs})$$

$$Q(\mathrm{M}(\phi^*)|\phi^*) = Q(\phi^*|\phi^*)$$

and

$$f(Z_{mis}|Z_{obs}, \mathrm{M}(\phi^*)) = f(Z_{mis}|Z_{obs}, \phi^*)$$

almost everywhere.

3. If for some ϕ^* in the parameter space of ϕ, $\ell(\phi^*|Z_{obs}) > \ell(\phi|Z_{obs})$ for all ϕ, then for every GEM algorithm,

$$\mathrm{M}(\phi^*) = \phi^*$$

4. If $\ell(\phi|Z_{obs})$ is bounded, $\ell(\phi^{(p)}|Z_{obs})$ converges to some ℓ^*.

5. If $f(Z|\phi)$ is a *general exponential family* and $\ell(\phi|Z_{obs})$ is bounded, then $\ell(\phi^{(p)}|Z_{obs})$ converges to a stationary value ℓ^*.

3.6.1 Summary on the convergence of the EM

Result (1.) implies that $\ell(\phi|Z_{obs})$ is non-decreasing on each iteration of a GEM algorithm, and is strictly increasing on any iteration where

$$Q(\phi^{(p+1)}|\phi^{(p)}, Z_{obs}) > Q(\phi^{(p)}|\phi^{(p)}, Z_{obs})$$

Wu (1983) has argued that it is difficult to guarantee that, by using the EM algorithm, we always converge to the global maximum. An example given by Murray (1977) illustrates the possibility of ϕ converging to a stationary value which is not a global maximum.

An example from Boyles (1982) illustrates the fact that $\phi^{(p)}$ does not converge to a stationary point ϕ^* as claimed by Dempster *et al* (1977), but it converges to

the circle of unit radius. A unique maximizer ϕ^* of $L(\phi|Z_{obs})$ can certainly be obtained only if $\ell(\phi|Z_{obs})$ is unimodal.

Note that the rate of convergence of the EM algorithm is closely related to the following quantities.

The greater the proportion of missing information, the slower the rate of convergence. Dempster *et al* (1977) have shown that if the iterates $\phi^{(p)}$ converge to ϕ^*, then for $\phi^{(p)}$ near ϕ^*,

$$|\phi^{(p+1)} - \phi^*| = \lambda|\phi^{(p)} - \phi^*|$$

where λ is the ratio of the missing information to the complete information for scalar ϕ.

3.7 ESTIMATION OF MISSING VALUES USING THE EM

There are two methods which can be used for the estimation of missing observations in incomplete data problems by using the EM algorithm.

Method 1: Algorithm (3.8.1)

In this method, estimation is performed by conditioning on the real or original records only. For example: missing data point y_ℓ can only be estimated by making use of those $x_{\ell j}$'s which are real values from the control stations, corresponding to y_ℓ. That is all the stations with the ℓth value missing are ignored. In this method, the same control stations are utilized for the estimation of a particular missing observation.

Method 2: Algorithm (3.8.2)

This method estimates the missing data point, say y_ℓ, by using all the records, i.e. estimated and real records. In this method all the observations are utilized after the initial estimation stage.

In spite of the fact that Method 2 is not the accepted method of estimation in the sense that its use has never been proposed in literature, for all the examples that we have looked at, it converged to the same answer. It also proved to be substantially faster, and therefore less expensive than Method 1 even though Method 2 requires

more iterations. We therefore recommend the use of Method 2, but if uncertain that it gives the same answer as Method 1, then the estimates obtained from Method 2 can be used as the initial estimate for the slow method (Method 1).

Section 3.8 describes these algorithms. Appendix A gives a step by step worked example of application to illustrate the steps required to actually implement the algorithms, and also to provide a numerical check for users who might prepare software to apply these algorithms.

3.8 ALGORITHMS

Suppose we are considering a rainfall stations matrix Z of dimension $n \times (k+1)$. Partition the Z matrix into a vector of observations in the target station, y, of dimension $(n \times 1)$ and a matrix of observations in the control stations, X, of dimension $(n \times k)$. Note that any station in the Z matrix can be regarded as the target station, depending on which station's missing values we are currently estimating.

3.8.1 *Method 1*: CONDITION ON REAL RECORDS ONLY

Suppose we wish to estimate the missing value y_ℓ :

CYCLE 0

STEP 1

Construct the matrix X_p of dimension $(n \times p)$ from the $(n \times k)$ matrix X by eliminating from it all the columns (stations) which contain a missing observation in the ℓth row. As an example, suppose $x_{\ell 3}$ is missing, then the 3rd column is eliminated. Let the number of columns (stations) remaining after elimination be p, then X_p is an $(n \times p)$ matrix of observations on the control stations.

STEP 2

Construct the vector y^* from y and the matrix X_p^*, from the $n \times p$ matrix X_p, by eliminating from *both* all the rows which contain one or more missing observations in *either*. For example, because y_ℓ is one of the missing observations then the ℓth row in *both* y and X_p is eliminated. Suppose that y^* ends up with n^* entries,

then X_p^* is an $n^* \times p$ matrix. The vector y^* and matrix X_p^* should now contain no missing observations. Check that there is sufficient data to regress y^* on X_p^*. If there is not, then some of the control stations will have to be removed and one must begin again.

STEP 3

Calculate the least squares estimates of the regression parameters using the target station vector y^* and the matrix of control stations X_p^*. That is find:

$$\hat{\beta}^{(0)} = (X_p^{*t} X_p^*)^{-1} X_p^{*t} y^*$$

and

$$\hat{\beta}_0^{(0)} = \bar{y}^* - \bar{X}_p^* \hat{\beta}^{(0)}$$

where

$$\bar{y}^* = \frac{1}{n^*} \sum_{i=1}^{n^*} y_i^*,$$

and

$$\bar{X}_p^* = \frac{1}{n^*} \sum_{i=1}^{n^*} x_{ij}^*, \qquad j = 1, 2, \ldots, p$$

and where the superscript (0) represents the initial estimation cycle.

Estimate the missing record y_ℓ using the regression model:

$$y_\ell^{(0)} = \hat{\beta}_0^{(0)} + \sum_{j=1}^{p} x_{\ell j} \hat{\beta}_j^{(0)}$$

where $x_{\ell j}$, $(j = 1, 2, \ldots, p)$ are the observed values from the control stations matrix X_p.

Notes

The EM algorithm requires the estimation of all the missing observations in the data. Therefore, before we continue with the next step, the previous steps are repeated until all the missing observations (for all the $k+1$ stations) are estimated. The missing observations which were estimated in the present cycle are not utilized for the estimation of other missing values.

Create a new "data" matrix, say $Z^{(0)}$ containing estimates obtained in place of missing values, where the superscript (b) represents the current cycle.

CYCLE b

STEP 1

Suppose that the new data matrix created in the previous cycle is $Z^{(b-1)}$, then after the partitioning of $Z^{(b-1)}$, re-construct the matrix $X_p^{(b-1)}$ from the $(n \times k)$ matrix of control stations $X^{(b-1)}$ by eliminating from it all the columns which originally contained a missing observation in the ℓth row. $X_p^{(b-1)}$ is similar to X_p except that the missing observations in X_p have been replaced by the estimated values.

STEP 2

Calculate the least squares estimates:

$$\hat{\beta}^{(b)} = (X_p^{(b-1)t} X_p^{(b-1)})^{-1} X_p^{(b-1)t} y^{(b-1)}$$

and

$$\hat{\beta}_{(0)}^{(b)} = \bar{y}^{(b-1)} - \bar{X}_p^{(b-1)} \hat{\beta}^{(b)}$$

where

$$\bar{y}^{(b-1)} = \frac{1}{n} \sum_{i=1}^{n} y_i^{(b-1)},$$

and

$$\bar{X}_p^{(b-1)} = \frac{1}{n} \sum_{i=1}^{n} x_{ij}^{(b-1)}, \qquad j = 1, 2, \ldots, p$$

Re-estimate the missing record y_ℓ:

$$y_\ell^{(b)} = \hat{\beta}_0^{(b)} + \sum_{j=1}^{p} x_{\ell j} \hat{\beta}_j^{(b)}.$$

Check for convergence by using the following criterion:

$$\text{Crit}_\ell = \left[\frac{y_\ell^{(b)} - y_\ell^{(b-1)}}{y_\ell^{(b)}} \right]^2$$

where b represents the current cycle (iteration),

$b-1$ represents the previous cycle (iteration).

If $\text{Crit}_\ell \leq \text{F}$ then y_ℓ is considered the required estimate of the missing value and this value is no longer re-estimated.

STEP 3

Repeat steps 1 and 2 until all the estimates of the missing values have converged.

3.8.2 *Method 2:* CONDITION ON REAL AND ESTIMATED RECORDS

Suppose again that we wish to estimate the missing value y_ℓ.

CYCLE 0

STEP 1

Construct the vector y^* from y and the matrix X^* from the $(n \times k)$ matrix X, by eliminating from *both* all the rows which contain one or more missing observations in *either*. For example, because y_ℓ is one of the missing observations then the ℓth row in *both* y and X is eliminated. Suppose that y^* ends up with n^* entries, then X^* is an $(n^* \times k)$ matrix. The vector y^* and matrix X^* should now contain no missing observations. Check that there is sufficient data to regress y^* on X^*. If there is not then some of the control stations will have to be removed and one must begin again.

STEP 2

Calculate the least squares estimates of the regression parameters using the target station vector y^* and the matrix of control stations X^*. That is find:

$$\hat{\beta}^{(0)} = (X^{*t}X^*)^{-1}X^{*t}y^*$$

and

$$\hat{\beta}_0^{(0)} = \bar{y}^* - \bar{X}^*\hat{\beta}^{(0)}$$

where

$$\bar{y}^* = \frac{1}{n^*}\sum_{i=1}^{n^*} y_i^*,$$

and

$$\bar{X}^* = \frac{1}{n^*}\sum_{i=1}^{n^*} x_{ij}^*, \qquad j = 1,2,\ldots,k$$

and where the superscript (0) represents the initial estimation cycle.

Estimate the missing record y_ℓ using the regression model:

$$y_\ell^{(0)} = \hat{\beta}_0^{(0)} + \sum_{i=1}^{k} x_{\ell j}\,\hat{\beta}_j^{(0)}$$

where $x_{\ell j}$, $(j = 1,2,\ldots,k)$ are the observed values from the control stations matrix X.

STEP 3

After all the missing values in matrix Z have been estimated, create a new "data" matrix, say $Z^{(0)}$ containing estimates obtained in place of missing values.

CYCLE b

STEP 1

Suppose that the new data matrix created in the previous cycle is $Z^{(b-1)}$, then partition $Z^{(b-1)}$ into a matrix of control stations $X^{(b-1)}$ and a vector of the target station $y^{(b-1)}$.

STEP 2

Calculate the least squares estimates by using the new target station vector $y^{(b-1)}$ and the matrix of control stations $X^{(b-1)}$, where the superscript $(b-1)$ represents the previous cycle. That is find:

$$\hat{\beta}^{(b)} = (X^{(b-1)t}X^{(b-1)})^{-1}X^{(b-1)t}y^{(b-1)}$$

and

$$\hat{\beta}_{(0)}^{(b)} = \bar{y}^{(b-1)} - \bar{X}^{(b-1)}\hat{\beta}^{(b)}$$

where

$$\bar{y}^{(b-1)} = \frac{1}{n}\sum_{i=1}^{n} y_i^{(b-1)},$$

and

$$\bar{X}^{(b-1)} = \frac{1}{n}\sum_{i=1}^{n} x_{ij}^{(b-1)}, \qquad j = 1, 2, \ldots, k$$

Re-estimate the missing record y_ℓ :

$$y_\ell^{(b)} = \hat{\beta}_0^{(b)} + \sum_{j=1}^{k} x_{\ell j}^{(b-1)} \hat{\beta}_j^{(b)}$$

Let

$$\text{Conv}_\ell = \left| \frac{y_\ell^{(b)} - y_\ell^{(b-1)}}{y_\ell^{(b)}} \right|$$

where b represents the current cycle (iteration);

$b-1$ represents the previous cycle (iteration)

STEP 3

If all the missing values in the current station have been estimated, then check for convergence by using the following criterion:

$$\text{Crit} = \sum_{j=1}^{k} \sum_{\ell=1}^{n-n_j^*} \text{Conv}_{\ell j}$$

where n_j^* is the number of observed values in the current station.

STEP 4

If Crit $\leq$ F then y_ℓ is considered the required estimate of the missing value and the re-estimation discontinue, otherwise repeat steps 1, 2 and 3.

A detailed example of application is given in Appendix A.

Notes:

Other convergence criteria might be used, e.g.:

1. $\max_i (y_i^{(b)} - y_i^{(b-1)})^2$.
2. $[\hat{\beta}^{(b-1)} - \hat{\beta}^{(b)}]^t [\hat{\beta}^{(b-1)} - \hat{\beta}^{(b)}]$.

CHAPTER 4

SIMULATION STUDY

The most convenient way to assess the performance of the various methods of estimating missing values described in the previous chapters is by simulation. This has several advantages over using "real" data. Firstly it is possible to compute the accuracy of the methods directly. This can be achieved by generating complete records, "hiding" some of the values, estimating "missing" values and then comparing the estimates with the corresponding "true" generated values. A second advantage is that it is easy, by simulation, to vary some of the factors which are likely to be important in determining the performance of the different methods. Such factors include the correlations between observations at different stations, the proportion of missing values and the length of records at each station.

This chapter gives details of the methods used to generate artificial rainfall sequences. In particular we discuss

- The correlation structure used to generate data for neighbouring stations.
- The generating of artificial rainfall values with a given correlation structure.
- The discarding (hiding) of observations from the artificial data to produce incomplete records.
- The disaggregation of annual rainfall data to generate monthly rainfall data.

The first decision to be made is that of the positioning of the artificial rainfall stations. Computationally it would be simpler to place the stations on a regular grid, but this leads to a number of problems. Real rainfall stations do not occur on a regular grid, and it is possible that generated values on such a grid would lead to results which are atypical of what one would expect in practice. This is particularly the case if we assume that the correlation of rainfall totals at different stations depends essentially on the distance between the stations. On a regular grid there will then always be a number of stations which will be (exactly) equally correlated with target station. This leads to some unexpected and undesirable effects, such as non-convergence of some of the algorithms.

A preferable design is to randomly place the stations on a surface. We used uni-

formly distributed stations on a square.

4.1 CORRELATION STRUCTURE

For the purpose of carrying out comparison between methods we assumed that the correlations between the rainfall totals at two stations depend only on the distance between the stations. This assumption is not unreasonable in practice for relatively homogeneous regions. However the results reported here do not depend on this asssumption being met; it simply provides a convenient way to generate rainfall totals which have realistic cross-correlation structures.

The particular function used to relate the correlation, ρ_{ij}, between two rainfall stations i and j which are a distance d_{ij} apart is

$$\rho_{ij} = \max \begin{cases} \alpha e^{-\beta d_{ij}} \\ \gamma \end{cases} \tag{4.1}$$

Based on the findings of Welding M.C. and Havenga C.M. (1974) and using kilometres as units of distance, we selected $\gamma = 0.4$ and α which varied from 0.5 to 0.8. The value of β was then computed using equation (4.1).

The positions of the stations and correlation matrix of the rainfall totals were then computed as follows:

- *Position of stations*

The position of the k stations was assigned by generating independently and uniformly distributed coordinates (x_i, y_i), $i = 1, 2, \ldots, k$, over a 70 km by 70 km square.

- *Computation of the variance-covariance matrix*

Since the methods considered in this thesis are location and scale invariant it would be sufficient, for the purposes of assessing the performance of the methods, to generate "rainfall totals" which have mean zero and variance one. However we felt that it would be preferable to generate values which also had more typical means and variances. This made it easier to check the algorithms and to obtain a quick impression on the performance of the methods.

We used

$$\mu = 1000 \text{ mm} \qquad \text{and} \qquad \sigma^2 = 200^2 \text{ mm}^2.$$

Expression (4.1) was used to compute the correlations ρ_{ij} with

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

The ijth entry of the variance-covariance matrix Σ is then given by

$$\sigma_{ij} = \sigma^2 \rho_{ij}, \qquad i, j = 1, 2, \ldots, k.$$

4.2 GENERATING ANNUAL RAINFALL TOTALS

To generate annual rainfall totals, we started by generating data points which we assumed to be normally distributed with mean 0 and variance 1. That is, we assumed that the vectors $Y_1, Y_2, \ldots, Y_k$ were independently and identically normally distributed with mean zero and variance one. This assumption can be summarised by writing

$$Y_i \sim N(0, 1), \qquad i = 1, 2, \ldots, k$$

where k is the number of stations.

To generate multivariate normal data with mean vector μ and variance-covariance matrix Σ, since Σ is a positive definite symmetric matrix, we made use of the Cholesky decomposition method. We factored Σ into triangular matrices V and V^t such that $\Sigma = VV^t$ (see Acton, 1970 for details). Then set

$$Z = VY + \mu.$$

It then follows that $Z \sim N(\mu, \Sigma)$ since

$$\mathrm{E}(Z) = \mu$$

and

$$\begin{aligned} \mathrm{Var}(Z) &= \mathrm{E}(Z - \mu)^t (Z - \mu) \\ &= \mathrm{E}(VYY^tV^t) \\ &= VV^t \\ &= \Sigma \end{aligned}$$

4.3 DISCARDING (HIDING) OBSERVATIONS FROM ARTIFICIAL DATA

There are many ways in which observations can be discarded from the data. The aim is to randomly hide some of the observations from the data in order to form an incomplete data matrix. One way of doing this is to randomly generate a set of data, say U, which is assumed to be independently and uniformly distributed between the values 0 and 1. The size of U should be the same as the size of the data set, Z, from which we wish to discard some of the observations. Depending on the required percentage of missing values, we discarded from Z all those observations which corresponded to those elements of U which were less or equal to the required percentage of missing values.

For example, if the required percentage of missing values is 25%, then z_{ij} would be discarded if $u_{ij} \leq 0.25$, where u_{ij} is a data point from U.

4.4 DISAGGREGATION OF ANNUAL RAINFALL DATA TO MONTHLY RAINFALL DATA

Since we consider methods to estimate missing monthly values as well as missing annual values, it was necessary for us to devise some method of generating monthly data. For this purpose we selected a method of disaggregating the annual totals which were generated as described above. We note that the precise seasonal structure generated is not important for the purpose of making the type of comparisons which we carry out in this study. Any reasonable seasonal cycles would do, so long as these exhibit the cross-correlation (between stations) which are evident in monthly rainfall totals at neighbouring stations.

-Shape of monthly rainfall totals:

We assumed that the seasonal structure is defined by a *cosine* function. In particular, we calculated

$$p_i = \frac{\cos(\frac{2\pi}{12}i) + 1.2}{\sum_{i=1}^{12} \cos(\frac{2\pi}{12}i) + 14.4}, \qquad i = 1, 2, \ldots, 12$$

$$\sigma_i^2 = p_i(1 - p_i)$$

where 1.2 is added to give a reasonable seasonal structure and 14.4 is a multiple of 1.2.

- *Generating proportional disaggregation:*

Let year to year variation be defined by

$$0.25\sigma_i[U(0,1) - 1/2]$$

where $U(0,1)$ is independently and uniformly distributed between the values zero and one.

For each year, we generated proportional disaggregation which was defined by

$$\hat{p}_i = p_i + 0.25\sigma_i[U(0,1) - 1/2] \qquad i = 1,2,\ldots,11$$

To accommodate the variation from station to station, to each $\hat{p}_i$, $i = 2,\ldots,11$ we added

$$\frac{1}{6}\sigma_i[U(0,1) - 1/2].$$

If a $\hat{p}_i < 0$ was encountered, then all the $\hat{p}_i$ which were previously generated were ignored and simulation started again.

To find the 12th proportion, we summed up the 11 proportions and subtracted the sum from 1. That is

$$\hat{p}_{12} = 1 - \sum_{i=1}^{11} \hat{p}_i$$

If $\sum_{i=1}^{11} \hat{p}_i \geq 1$ then the process of generating the proportional disaggregation was repeated. Otherwise the annual data points were then disaggregated according to the proportions obtained.

CHAPTER 5

COMPARISON OF THE SELECTED PROCEDURES - ANNUAL DATA

In chapters 2 and 3 we looked at two approaches which can be applied for the estimation of missing rainfall records, namely, selection of stations in regression analysis and the EM algorithm. This chapter is concerned with comparisons between these classes of methods. To carry out such comparisons it is first necessary to decide on the criteria by which the methods will be judged.

Clearly one would favour methods which, on average, yield the most accurate estimates of the missing values; in fact the variable selection methods described in chapter 2 are specifically designed to try and achieve this goal. Here the term "most accurate" is used to refer to mean square error of prediction. However there are other requirements which we wish to meet. Firstly there are some statistical requirements, namely, that the estimated values should not introduce systematic bias in the estimates of the parameters of importance of the record (cf Zucchini and Hiemstra, 1984). In particular the mean and variance of the completed record should not be systematically distorted by the estimates. The methods considered in this thesis do not introduce a systematic bias in the mean. However, as is shown in Appendix B the regression methods introduce a systematic downward bias to the variance of completed record, the properties of the EM algorithm in this respect are not known. Secondly one has to consider computational complexity and, more importantly, computational expense. The methods discussed in this thesis require considerable computation. If they are to be applied on a routine basis to a large number of records then the computing expense can be substantial.

In this chapter we report on the results of the simulation studies described in chapter 4 and the different methods will be assessed with respect to the criteria outlined in the preceeding paragraph.

In order to make the comparisons, we generated 30 sets of artificial data which we assumed to be multivariate normally distributed. Each of the data sets comprised 10 stations (1 target station and 9 control stations) each having 100 observations. The highest correlation coefficient between the rainfall stations used was $\rho = 0.8$ and the lowest correlation coefficient was $\rho = 0.4$. From each complete data set,

we temporarily discarded (or hid) some of the values which were assumed to be missing. "Missing" values were then estimated by applying the methods which were discussed in chapters 2 and 3.

Bear in mind that these data points were discarded in such a way that the remaining data points were sufficient to fit the regression model, when applying the estimation methods.

In section 5.1, we compare the methods in terms of the sum of square error due to prediction (SSEP). The smaller the value of SSEP is, the more accurate the estimated values are. We again use the standard deviation as a means of comparison, the higher the reduction of the standard deviation, the less accurate the estimated values are. The preservation of the standard deviation is discussed in section 5.2. Section 5.3 compares the computational expense of the methods of estimation.

5.1 SUM OF SQUARE ERROR DUE TO PREDICTION (SSEP)

Sum of square error due to prediction is used as a measure of accuracy. SSEP is calculated by finding the sum of square differences between the estimated values and the corresponding "true" generated values. That is for each "missing" value in a station, we find

$$\text{SSEP} = \sqrt{\frac{\sum_i (y_{observed,i} - y_{patched,i})^2}{n_{mis}}}$$

where $y_{patched,i}$ is the estimated value for the "missing" ith value,

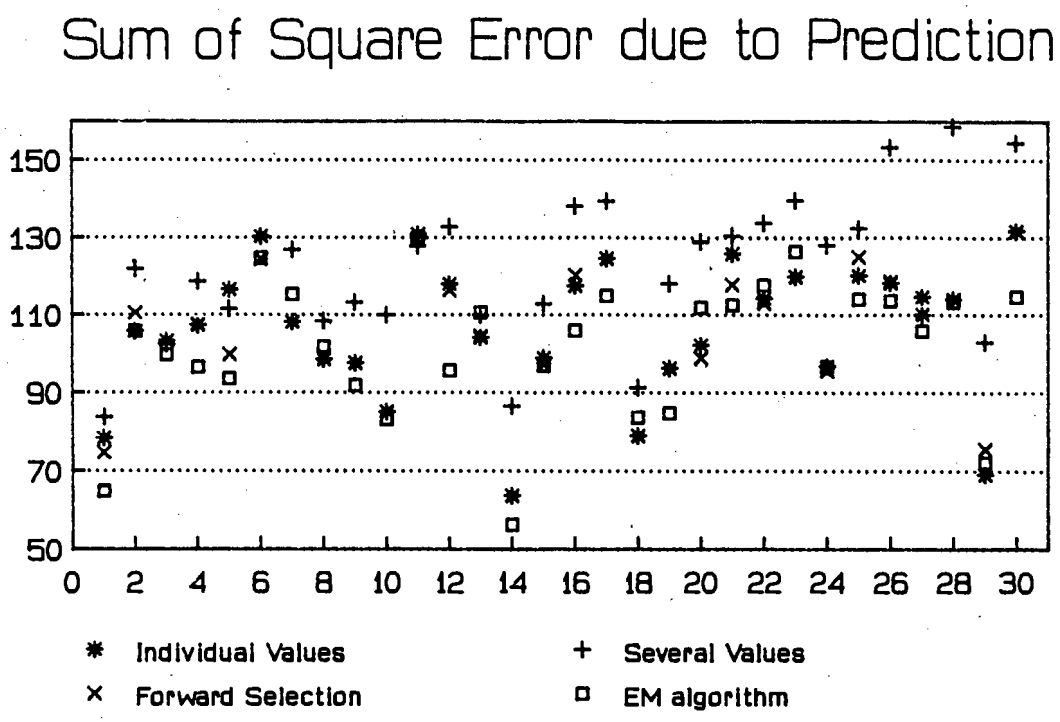
$y_{observed,i}$ is the "true" generated value corresponding to $y_{patched,i}$,

and n_{mis} is the number of "missing" values in that particular station which is being estimated.

The method which gives the smallest value of SSEP is regarded as more accurate than the other methods.

Figure 5.1 shows the points of the SSEP obtained by using the three different procedures of regression analysis and the EM algorithm. The further away the point is from the $x-$ axis , the less accurate the method of estimation is and vice versa.

FIGURE 5.1: *Sum of square error due to prediction - EM algorithm and the three regression methods*



By looking at Figure 5.1, it is clear that on average, the EM algorithm produced the smallest sum of square error due to prediction. This implies that the EM algorithm's estimated values were more accurate than the estimated values obtained by using the other methods. Forward selection procedure was the second most accurate method while the Selection of Control Records for several missing values procedure was the least accurate method of estimating missing records.

The table below which shows the averages of the SSEP's, confirms that the EM algorithm was the best among all the other methods considered.

	Selection of Control Records for Individual Missing Values	Selection of Control Records for Several Missing Values	Forward Selection	EM Algorithm
Average SSEP	106.3	121.9	105.5	101.8

5.2 PRESERVATION OF THE STANDARD DEVIATION

The comparison of the methods in terms of the preservation of the standard deviations was also carried out using the thirty generated data sets which were discussed. For convenience we take the natural logarithms of the standard deviations for the estimated data and subtract them from the natural logs of the standard deviations for the "true" data. Negative differences imply that the standard deviation is under-estimated, while positive differences imply over-estimation. These differences of the natural logs are then represented by graphs. For the purpose of comparing the methods, only one station has been estimated. This was done to avoid the computational expense involved when estimating missing values using the regression methods.

5.2.1 REGRESSION METHODS

As has already been indicated and as is proved in Appendix B, regression methods lead to a systematic downward bias of the variances. The extent of the bias depends on the correlation coefficients and the percentage of missing values from the data. Artificially generated data were used to confirm that by applying the regression methods, the standard deviation is under-estimated, and to determine the how much the bias varies with these two factors.

FIGURE 5.2: *Log differences of standard deviations - EM algorithm and the three regression methods*

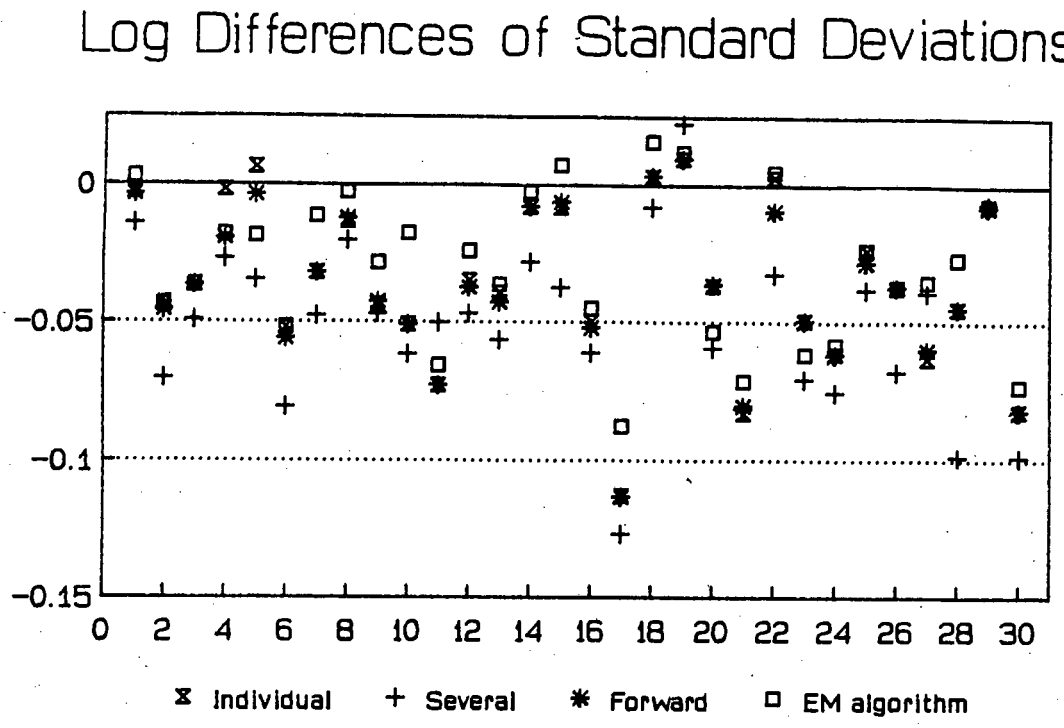


Figure 5.2 is a graph of points which represent the *log* differences of the standard deviations. The horizontal axis represents the different data sets, whereas the vertical axis represents the differences of the logs.

a. Selecting Control Records for Individual Missing Values

Most of the points representing this method in Figure 5.2 are below the horizontal line, it can therefore be seen from the points that most of the data sets underestimated the standard deviation. The highest reduction of the standard deviation of the "true" data is about 11.3%. A few of the standard deviations have slight reductions.

b. Selecting Control Records for Several Missing Values

In this method, the highest reduction of the standard deviation is about 12.7% of the "true" data's standard deviation.

c. Forward Selection

Most of the points for this method are below the horizontal line which implies the under-estimation of the standard deviation. As in the first procedure, the highest reduction of the "true" data's standard deviation is 11.3%.

5.2.2 EM ALGORITHM

Before we compare the regression methods with the EM algorithm, we will first demonstrate how much of the standard deviations are preserved when using the EM algorithm and the effects of the reduction of the standard deviation. This will be done by using data which were assumed to be both bivariate normally distributed and multivariate normally distributed. The digression is necessary to ascertain in a simpler context whether the EM algorithm leads to the type of bias which occurs using regression methods, and because it is difficult to determine theoretically whether we should expect such a bias using the EM algorithm.

BIVARIATE NORMAL DISTRIBUTED DATA

These data are generated in such a way that the different data sets have different correlation coefficients between the two stations. We randomly generated bivariate normally distributed variables say Z_1 and Z_2, with mean 0 and variance 1. Let

$$X = Z_1$$

and

$$Y = \rho Z_1 + \sqrt{1-\rho^2}\, Z_2$$

where ρ takes on the values 0.05, 0.10, 0.15, ..., 0.95.

We then "hid" different percentages of the generated values and estimated them by applying the EM algorithm. The ratios of the "true" data's standard deviation and the standard deviations of the estimated data are then calculated. The closer the ratio is to 1, the less the reduction of the "true" data's standard deviation.

FIGURE 5.3: *Standard deviation Ratios - Bivariate normally distrubuted data*

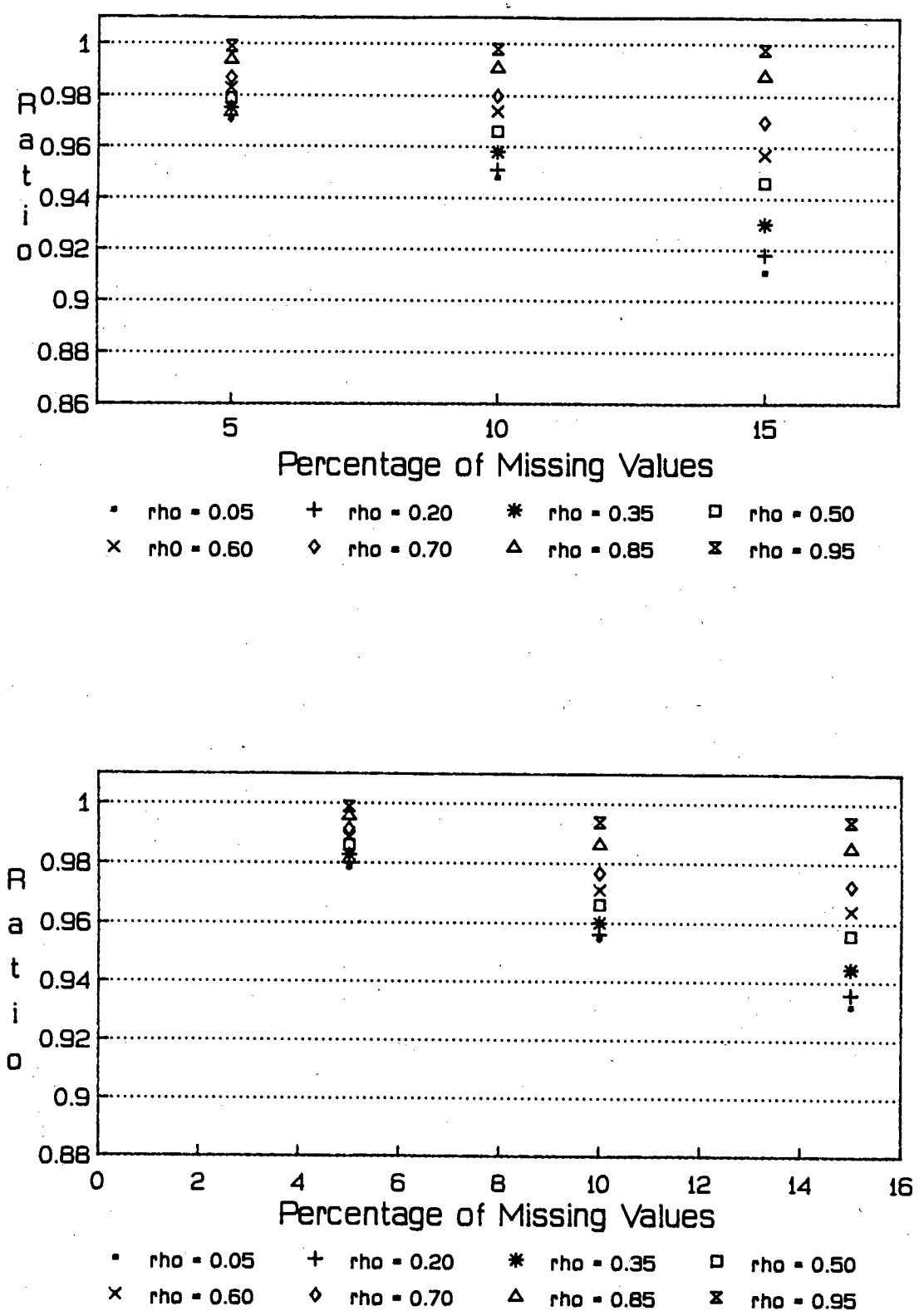


Figure 5.3 shows the graphs which were drawn by using the ratios of the standard deviations obtained from the two generated stations. It is concluded from these graphs that the preservation of the standard deviation depends on the correlation coefficient and/or the percentage of the missing records in the data. As it can be seen from the graphs, the lower the correlation coefficient, the higher the reduction of the standard deviation, and vice versa. The higher the percentage of missing records in the data, the higher the reduction of the standard deviation, and vice versa.

MULTIVARIATE NORMAL DISTRIBUTED DATA

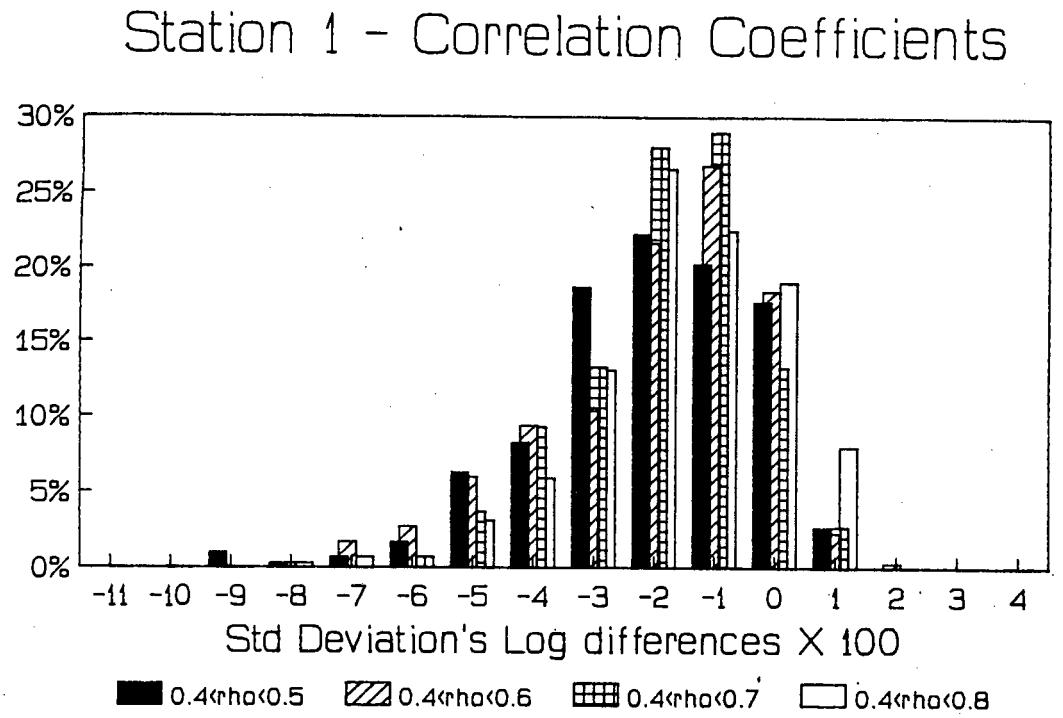
We then generated sets of data which were assumed to be multivariate normally distributed. The data were generated in such a way that the lowest correlation coefficient in each of them is 0.4 with varying highest correlation coefficients from 0.5 to 0.8. (For the algorithms of how to generate the data, see Chapter 4). Each data set was a 100×10 matrix and all 10 stations were estimated and standard deviations were calculated thereafter.

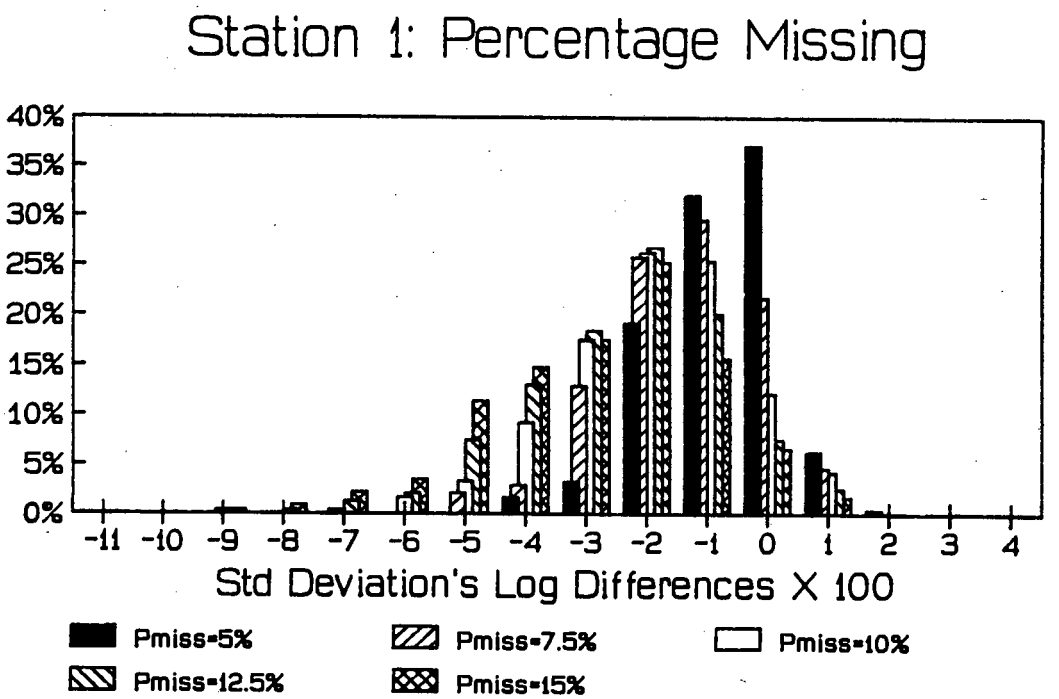
Figure 5.4 shows the histograms which were drawn by using the *log* differences of the standard deviations. Different histograms were drawn for the correlation coefficients and percentages of missing values. From these graphs, it can be seen that by reducing the correlation coefficient, the standard deviation is also reduced. Similarly, by increasing the percentage of missing values, the standard deviation is reduced.

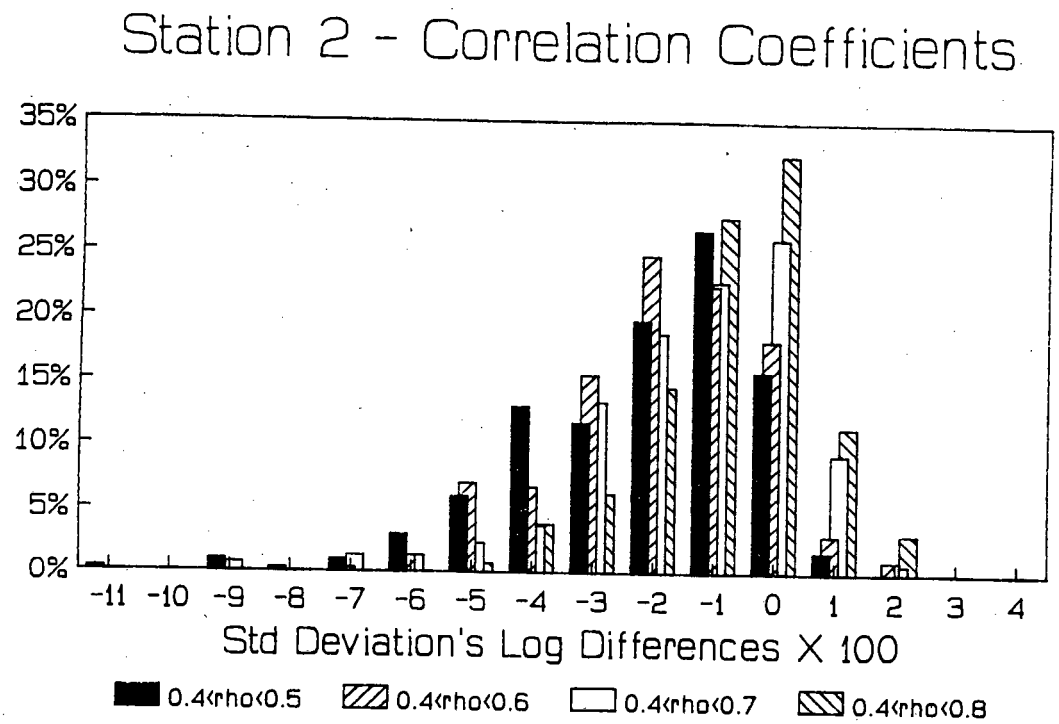
EM algorithm

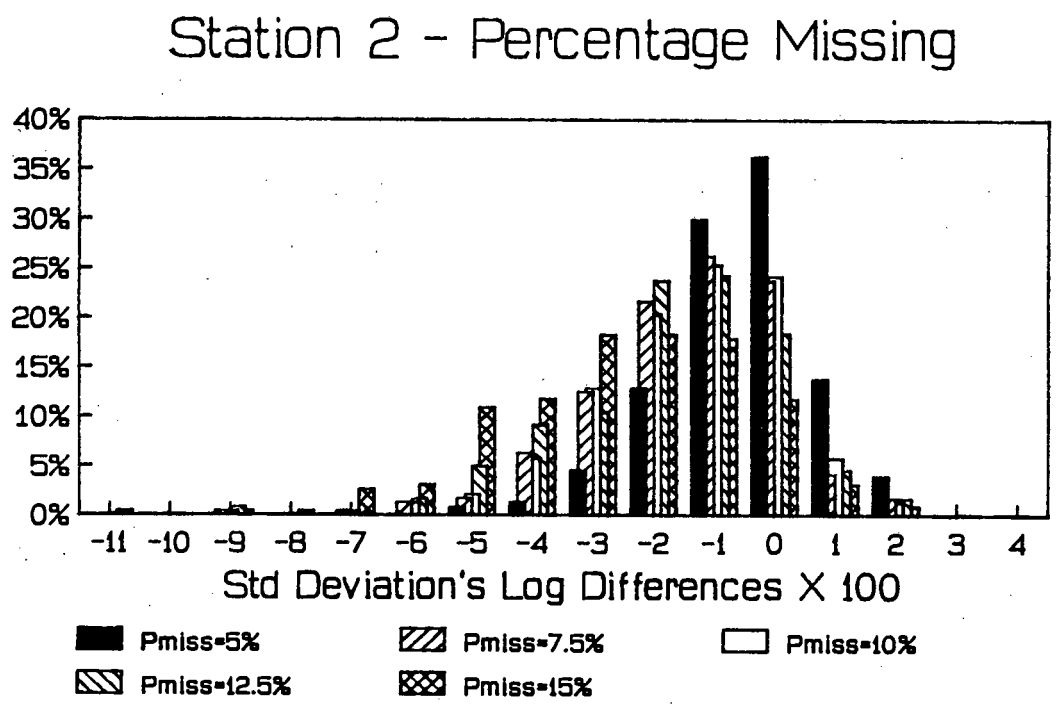
To compare the EM algorithm with the regression methods, we make use of Figure 5.2. It can be seen from this graph that very few of the points representing this method are above the horizontal line, which then implies that most of the standard deviations are under-estimated. The highest reduction of the standard deviation is 8.8%. Compared to the standard deviations which were obtained when using the regression methods, it can be seen from the graph that most of the points representing this method are above those points obtained when using any other method. We can therefore conclude that this method performed better than all the regression methods although it also tends to introduce a downward bias.

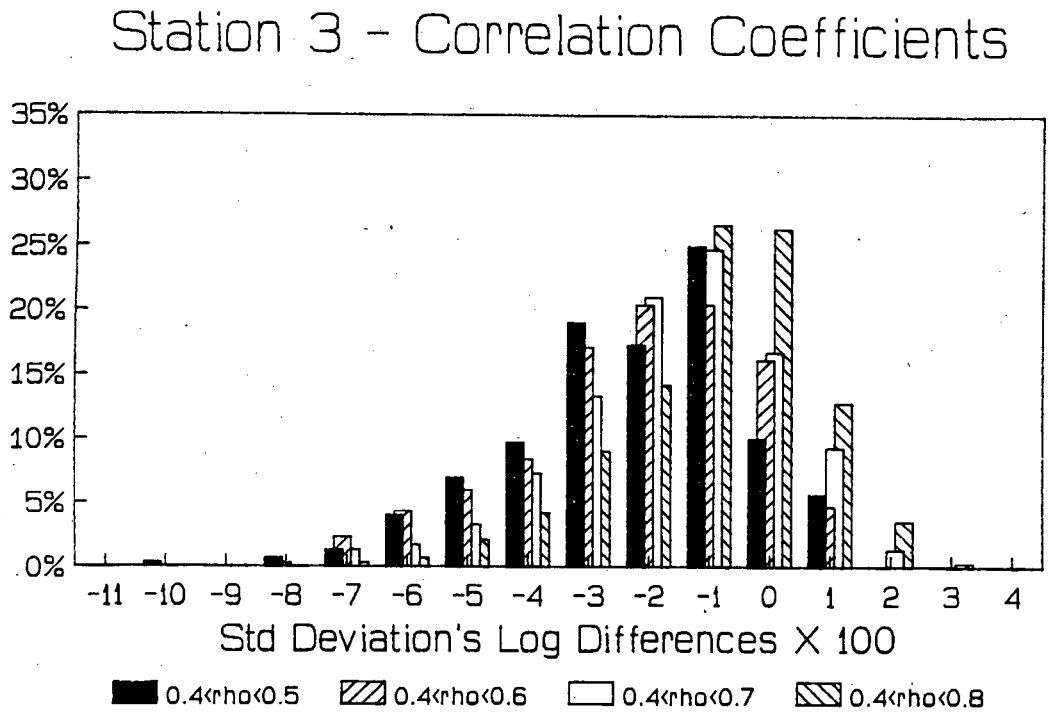
FIGURE 5.4: *Log differences of annual standard deviations - Multivariate normally distributed data*

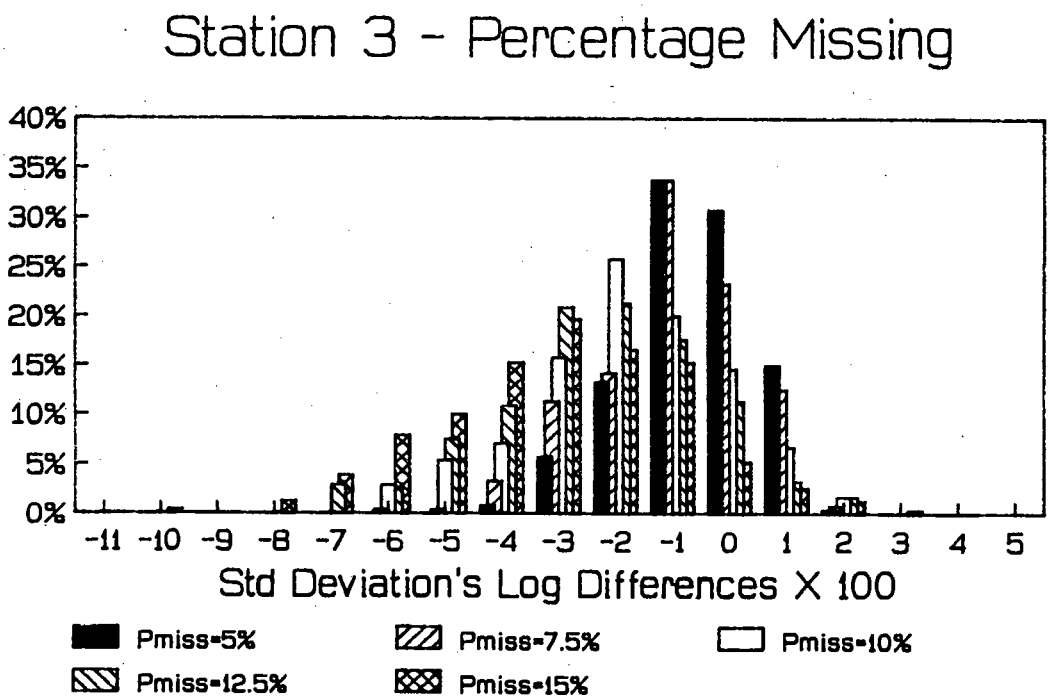


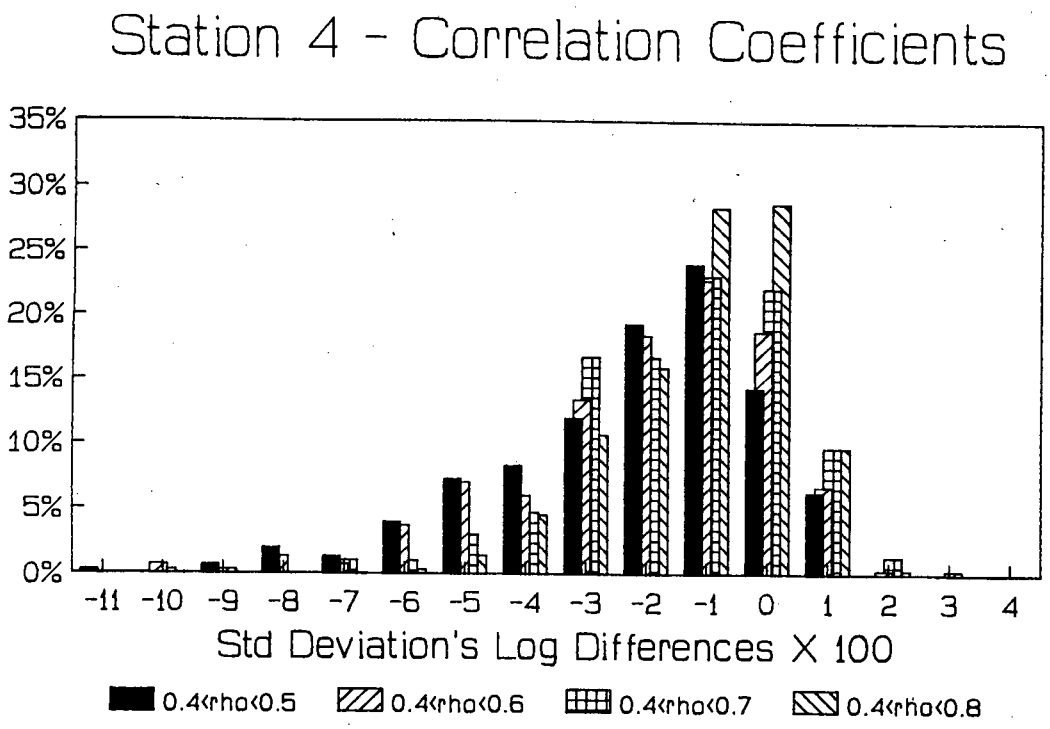


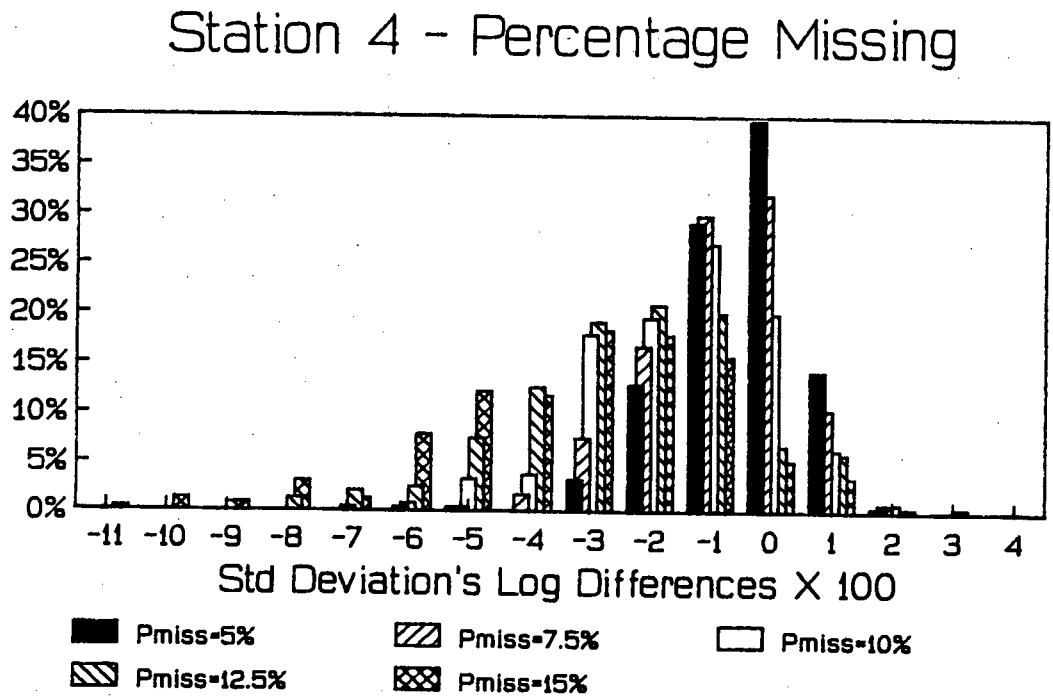

Station 2 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8

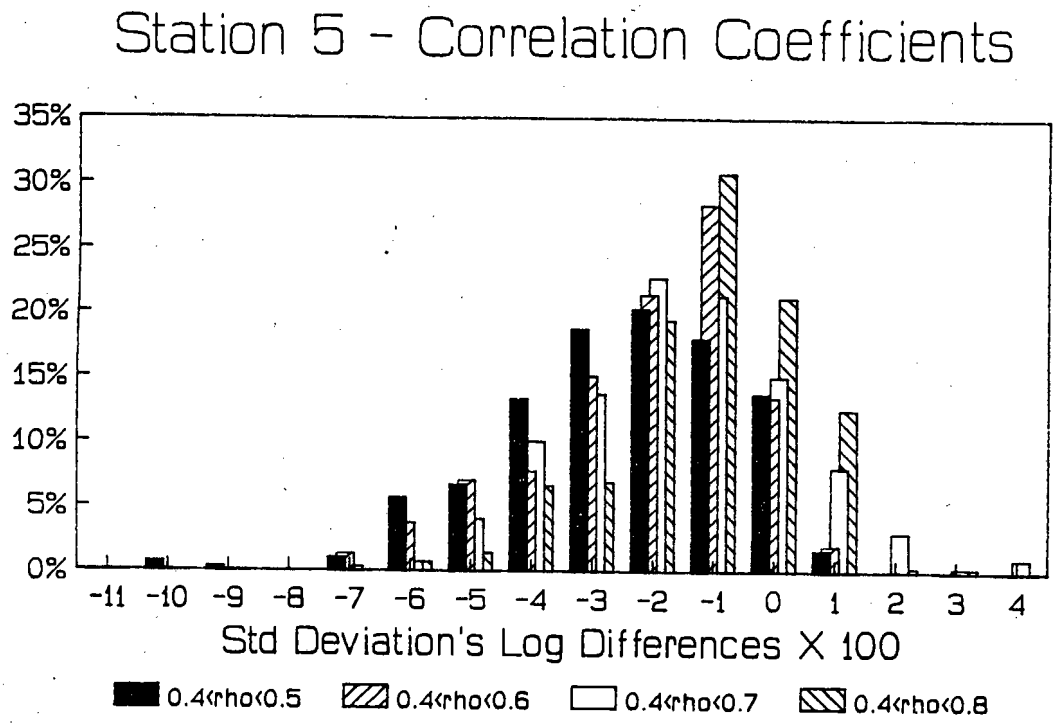

Station 2 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

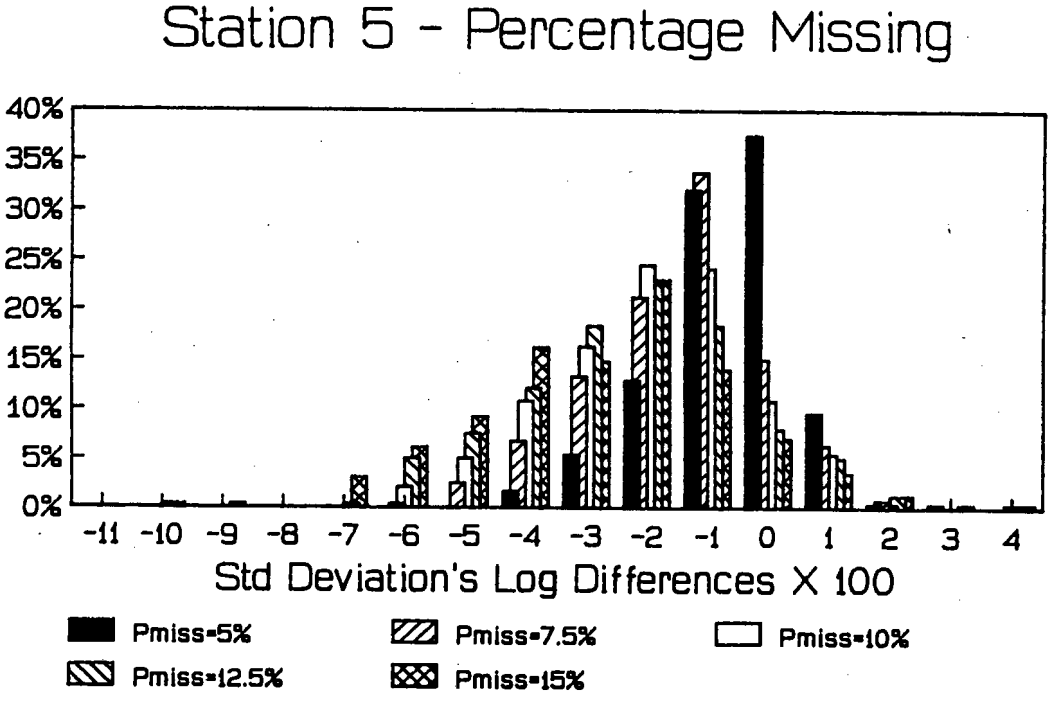

Station 3 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8

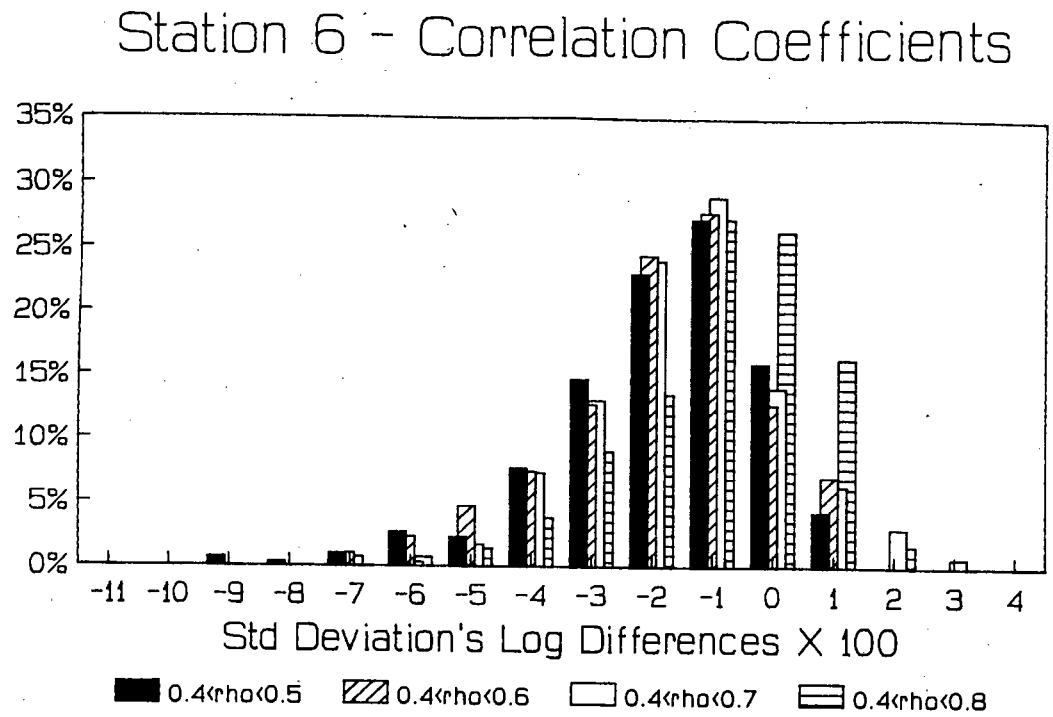

Station 3 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

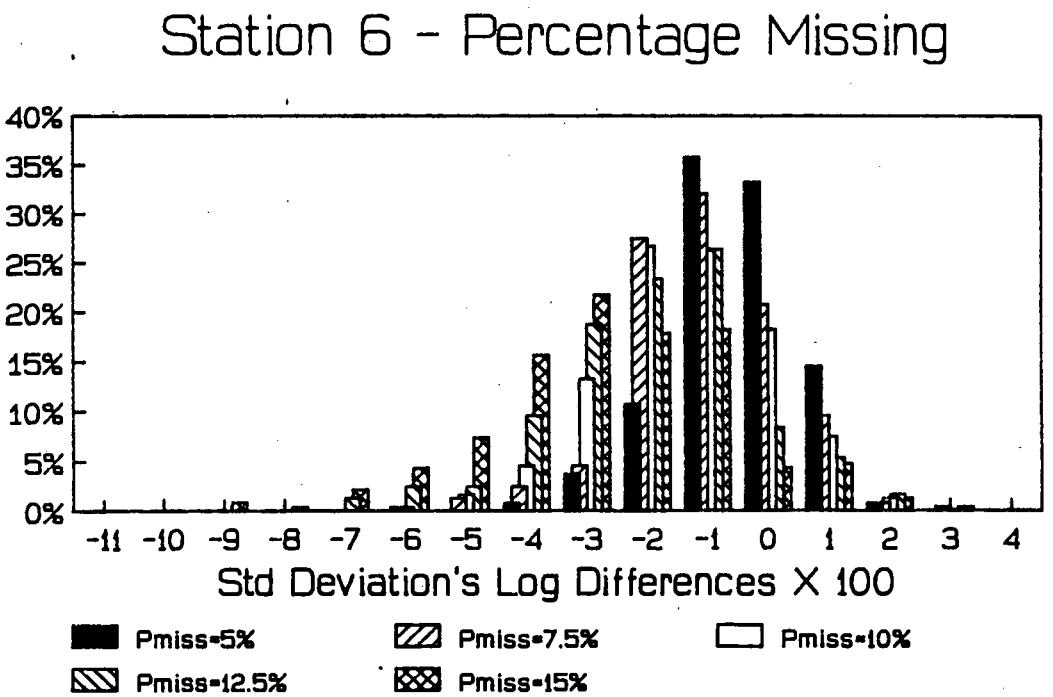

Station 4 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8

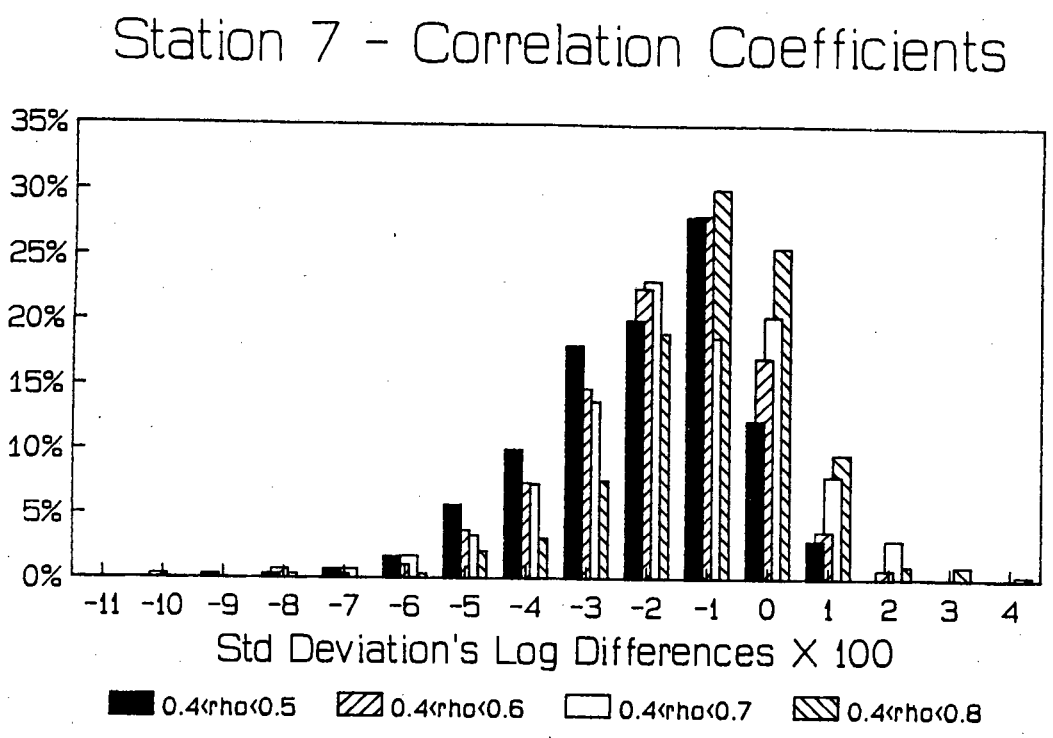

Station 4 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

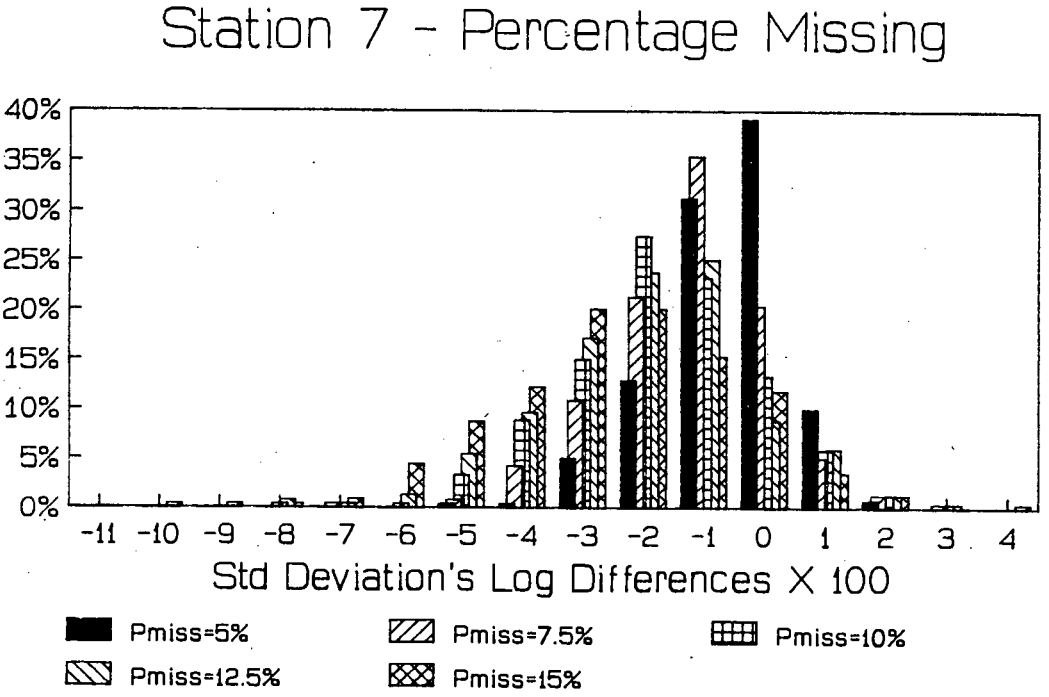

Station 5 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8

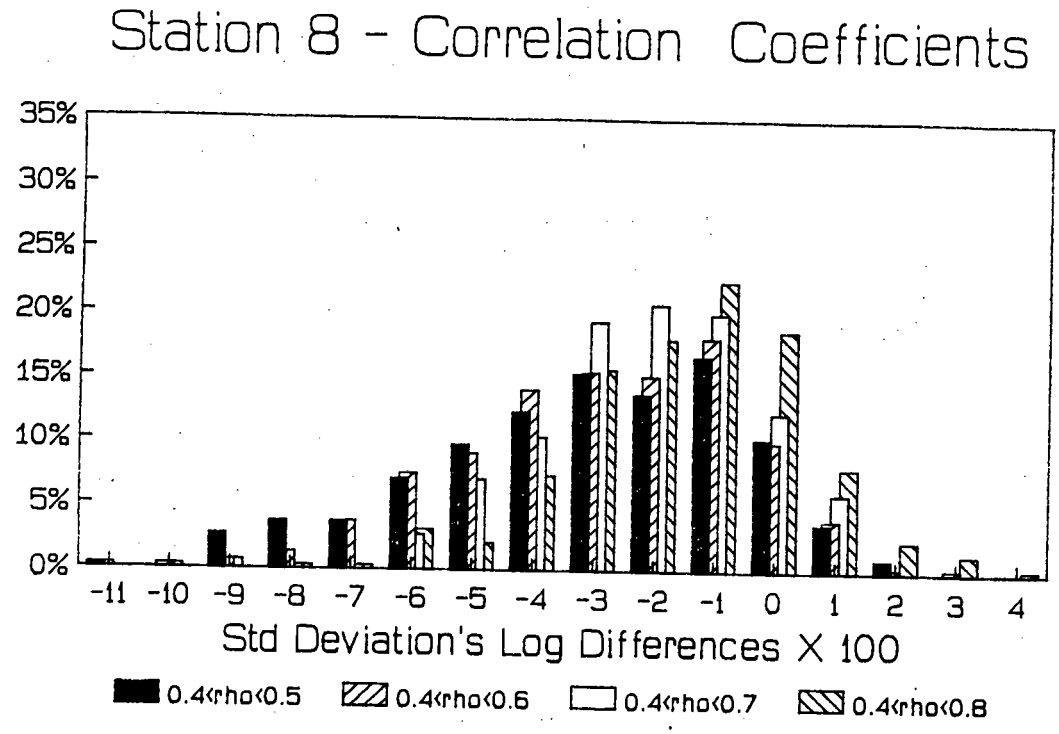

Station 5 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

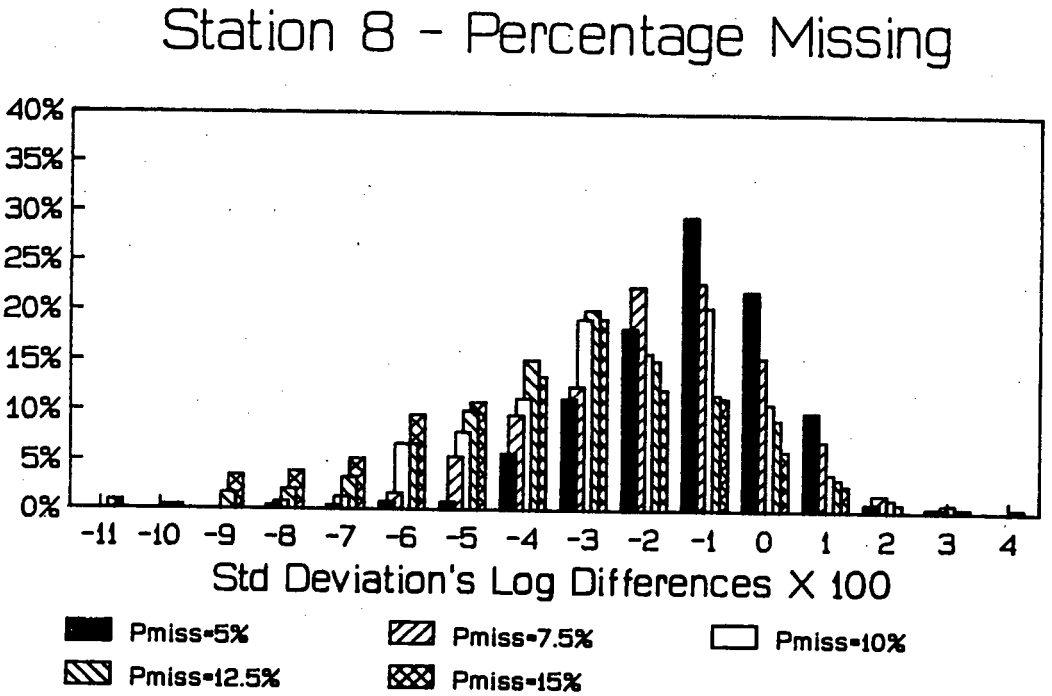

Station 6 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8

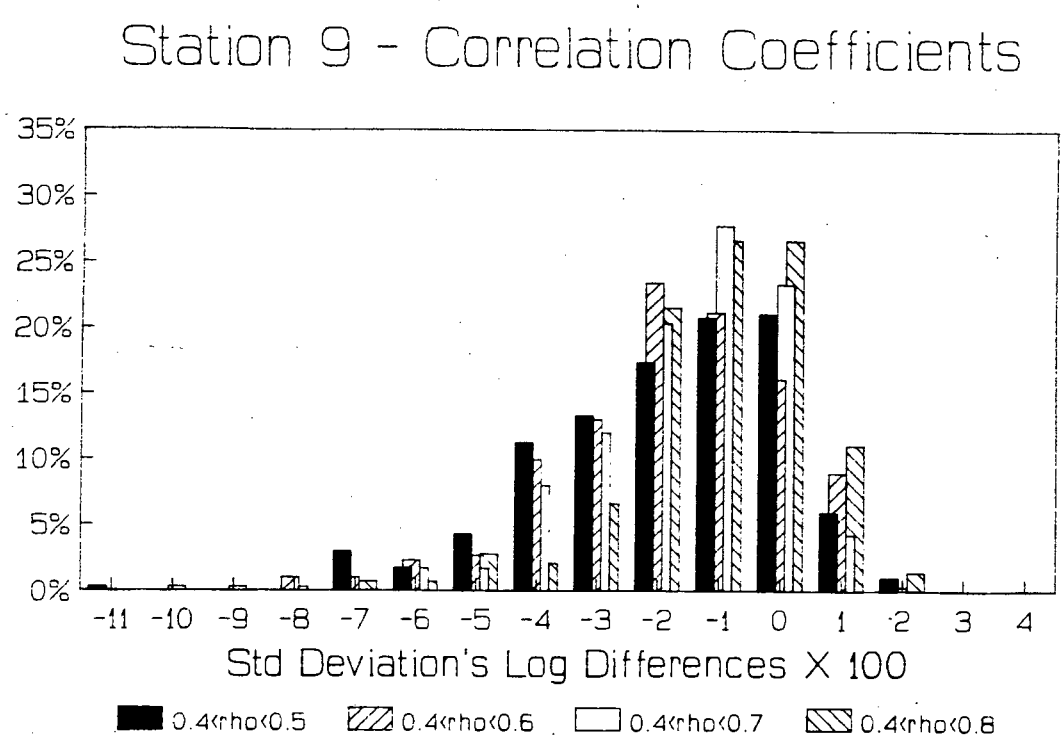

Station 6 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

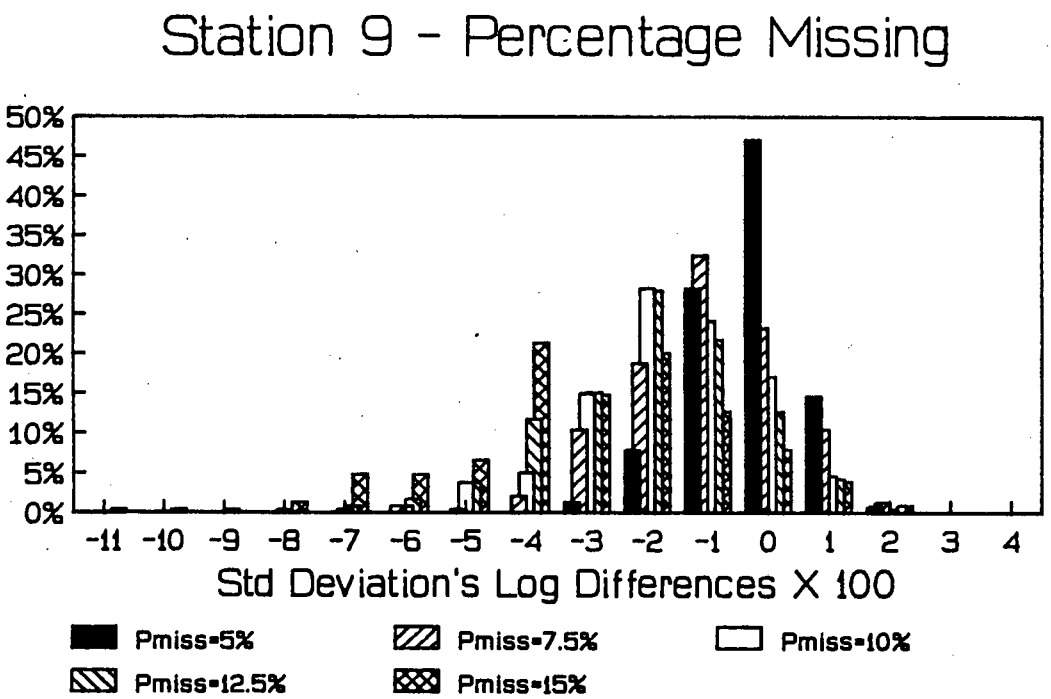

Station 7 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8

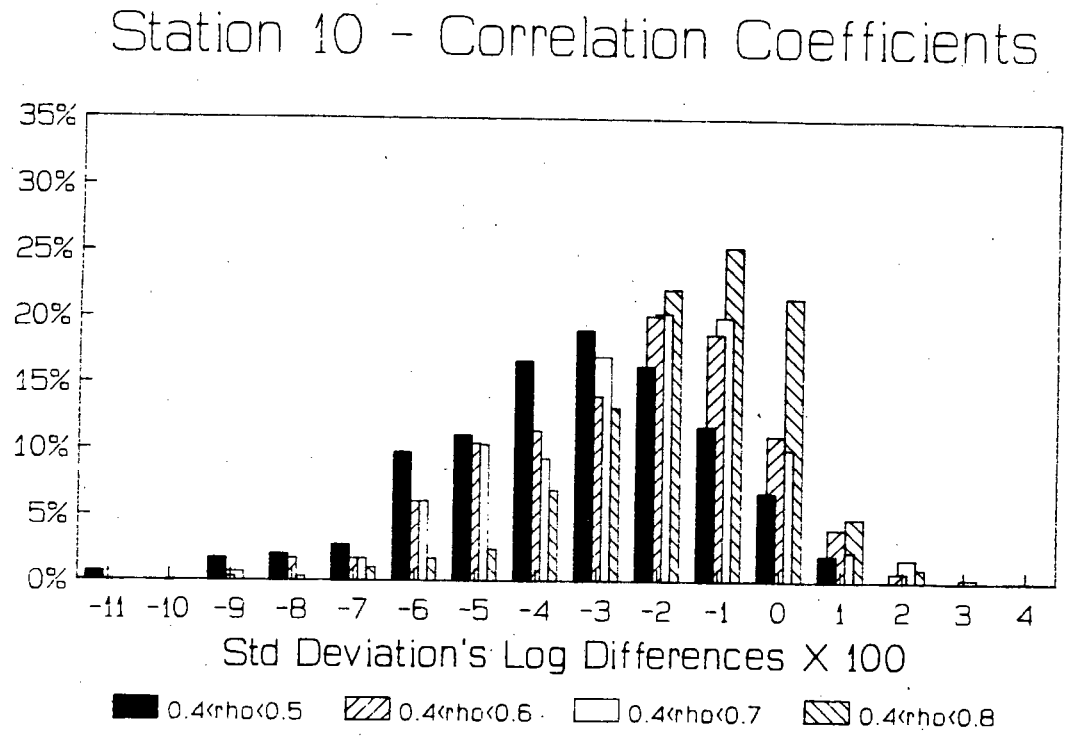

Station 7 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

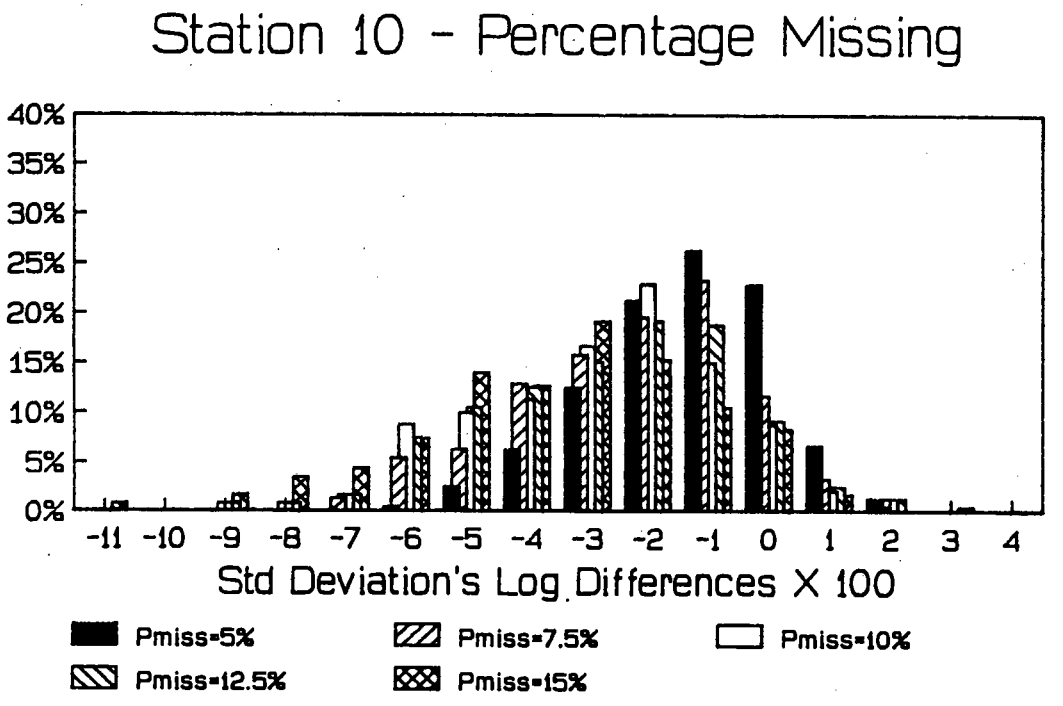

Station 8 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8


Station 8 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%


Station 9 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8


Station 9 - Percentage Missing
50%
45%
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%


Station 10 - Correlation Coefficients
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
0.4<rho<0.5
0.4<rho<0.6
0.4<rho<0.7
0.4<rho<0.8


Station 10 - Percentage Missing
40%
35%
30%
25%
20%
15%
10%
5%
0%
-11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Pmiss=5%
Pmiss=7.5%
Pmiss=10%
Pmiss=12.5%
Pmiss=15%

5.3 COMPUTATION

Figure 5.5 represents the CPU time (University UNIVAC SPERRY computer) used when applying the three selection of stations procedures and the most expensive method of the EM algorithm (Method 2, Section 3.8). Figure 5.6 represents the CPU time used by applying the two methods of the EM algorithm. Since computing time depends on the computer being used, the purpose of these comparisons is to indicate the relative times required by each method.

5.3.1 REGRESSION METHODS

a. Selecting Control Records for Individual Missing Values

When comparing the four different types of points in Figure 5.5, it can be seen that this method is most expensive. It is also clear that, by increasing the number of missing obsevations in the target station, the CPU time also increases. This is understandable since this procedure requires the fitting of every possible regression equation for each missing observation individually. This procedure's computation gets very demanding when the number of control stations is large, and the computer storage becomes a problem. On average, it required about 8 minutes:28 seconds to estimate 20% missing observations from only one target station.

b. Selecting Control Records for Several Missing Values

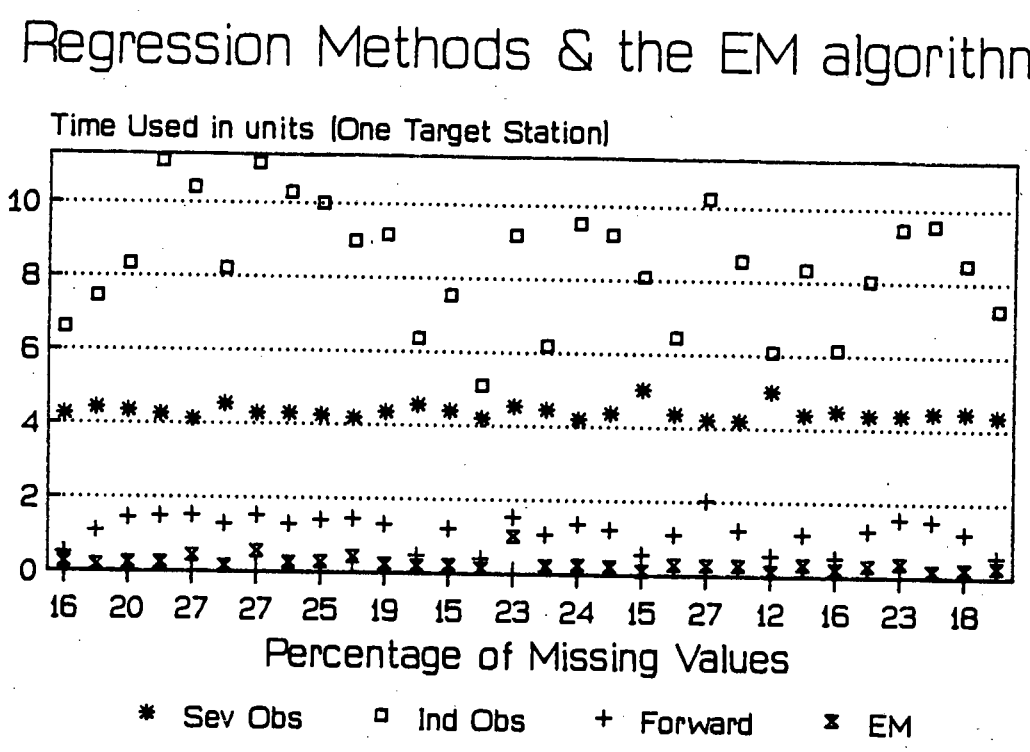
This is the second most expensive method of estimating missing values. As it can be seen from Figure 5.5, the computation time seems to be constant, that is, there is not much difference in the CPU time used in estimating 14% missing observations or used in estimating 27% missing observations. On average, it required about 4 minutes:37 seconds - which is about half of the time spent when using method 5.3.1(a) above - to estimate 20% missing observations from only one target station.

c. Forward Selection

Compared to the other two regression procedures, it is clear from the graph that this method is the cheapest to use. All the points representing the CPU time used by applying this method are below all the other points obtained by applying the other two selection of stations procedures. On average, it required about 1 minute:24 seconds to estimate about 20% missing values from a single target station. The CPU

time is reduced by about $\frac{1}{7}$th of the CPU time spent by the method considered under 5.3.1(a). As when using procedure 5.3.1(a), the more missing observations are there in the target station, the longer it takes to estimate them.

FIGURE 5.5: *Computation Time - Regression Methods and the EM algorithm*



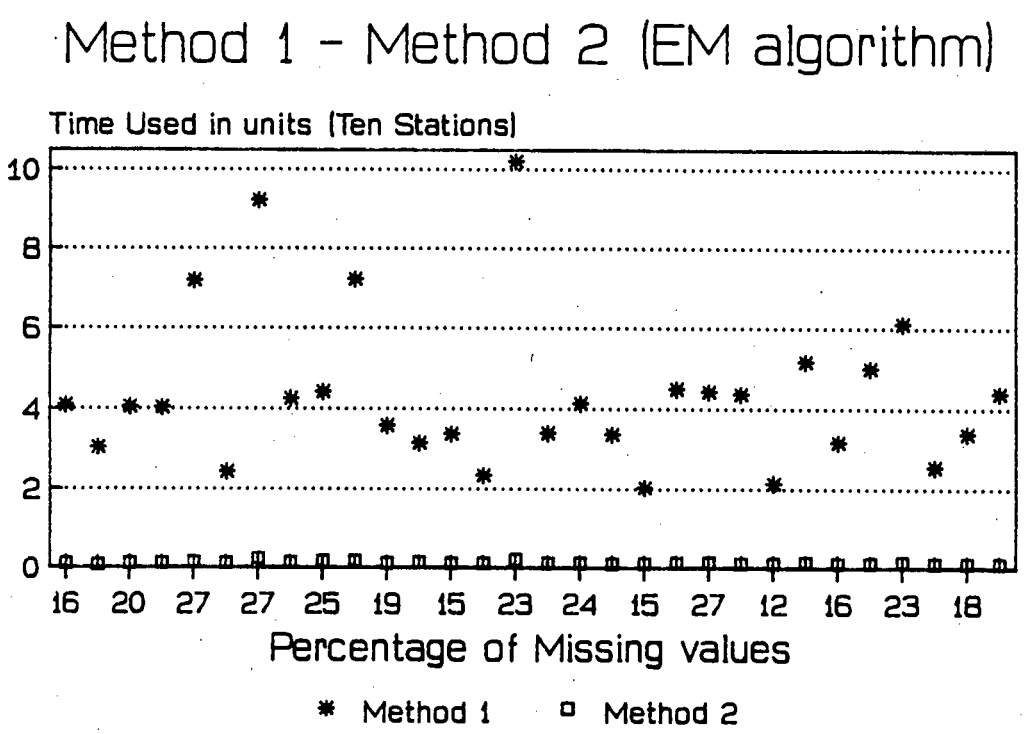
5.3.2 EM ALGORITHM

When applying the EM algorithm, the CPU time depends on how many of the observations are missing in the whole data matrix. This is because the EM algorithm requires the estimation of not only the target station, but also the control stations. This is performed until all the required observations are estimated. In general, the higher the percentage missing and/or the lower the correlation coefficient, then the longer it takes to estimate the missing observations. Recall that there were two algorithms which were considered (cf section 3.8):

Method 1: Condition on real records only

From Figure 5.5, it is clear that this method is substantially less expensive than the regression procedures. This comparison is performed by regarding only one target station. On average, it only took about 26 seconds to estimate about 20% missing observations from one target station. This method is about 5 times faster than *Forward Selection*, and it is about 20 times faster than the procedure considered under 5.3.1(a).

FIGURE 5.6: *Computation Time - EM algorithm Methods*



Method 2: Condition on real and estimated records

It can be seen from Figure 5.6 that this second method of the EM algorithm is faster compared to the first method of the EM algorithm. Thus this method is the fastest of all the discussed methods of estimating missing values. On average, it is 21 times faster than the first method of the EM algorithm which makes it extremely fast compared to the regression methods. The ratio of computation time depends also on the data's size and the number of missing observations encountered in the

data.

5.4 CONCLUSION

Taking all the properties discussed above into consideration, it is clear that the EM algorithm's method 2 is the best method for estimating missing values. Although both methods of the EM algorithm give the same results and therefore more accurate than the regression methods, the computation of the second method of the EM algorithm is less expensive than that of the first method of the EM algorithm. Forward selection was found to be the best regression method while the Selection of control records for several missing values, although not most expensive, is least accurate.

CHAPTER 6

MONTHLY DATA - THE EM ALGORITHM

6.1 INTRODUCTION

So far we have only considered the estimation of missing *annual* rainfall data. A year was regarded as missing if either:

1. No values were recorded for that year (whole year missing), or
2. Only some data was available for the year, in which case the whole year was then treated as missing (partial year missing).

Often only a few observations in the year are missing, and one wishes to estimate monthly, weekly or even daily rainfall records. In this chapter, we look at a number of options which can be applied to rainfall data which contain missing records. We will restrict our attention to the case where some, but not all, the *monthly totals* of a year are missing. We consider the question as to whether it would be preferable to treat each month separately, or alternatively, to estimate the total for the whole year and disaggregate. We will only discuss in detail the estimation of *monthly rainfall data* by applying the EM algorithm. The estimation of the same type of data by applying the regression methods has been discussed by Zucchini and Sparks (1984), Linhart and Zucchini (1986) and Adamson (1987).

In the case of partial year missing rainfall data, there are two options that can be used when applying the EM algorithm as a method of estimation. Firstly, the whole year record for the year can be treated as missing. Here one would estimate the total for whole year and thereafter subtract the sum of the available monthly records from the estimated annual value. The calculated difference of the two values is then disaggregated so that the values for individual monthly records are obtained. This method of estimation can lead to negative monthly rainfall values because it might happen that the estimated annual value is less than the sum of the available monthly totals. The second (alternative) option involves the estimation of the missing monthly records by applying the EM algorithm directly to monthly totals.

These two alternatives procedures are also applicable if the whole year's record

is missing. One can either estimate the missing monthly totals individually or alternatively estimate a single annual total and then disaggregate this to obtain estimates of the monthly totals.

When applying the EM algorithm to monthly totals the regression model is fitted without the intercept term and the data is not standardized. The reason for this is to avoid obtaining negative estimates. In arid and semi-arid regions, monthly rainfall totals of zero are frequent. Unless the regression line is forced through the origin, some monthly rainfall totals will inevitably be estimated as negative. In the case of multicollinearity (station totals correlated), this problem of negative rainfall totals is however unavoidable. Multicollinearity leads to negative regression coefficients even though the totals between each control and target stations are positively correlated. The consequence of having some negative estimates of regression coefficients is that it can lead to negative estimated values of rainfall totals. For example, suppose one particular control station has a negative regression coefficient and it only rained at this station (not at the others), then the estimate is negative. One way out of this problem is to omit the station which leads to negative regression coefficients.

In section 6.2, we give the algorithm of how to apply the EM algorithm to monthly totals. In section 6.3 we describe a method which can be used to disaggregate annual totals to monthly totals. Section 6.4 explains how to modify annual or monthly rainfall data to obtain daily rainfall values. Section 6.5 gives the comparison of the two methods in terms of computational time, sum of squared errors due to prediction, preservation of the standard deviation and preservation of the mean. Although from chapter 5 we know that the mean was not systematically biased, this might not be the case when using monthly data since the regression model is fitted without the intercept term.

6.2 EM ALGORITHM ON MONTHLY RAINFALL TOTALS

In this section we explain how the EM algorithm, which was discussed in the previous chapters, can be modified so that it can be applied to monthly totals.

When trying to estimate missing monthly rainfall totals by applying regression theory, the pronounced seasonality does not allow us to simply assume that the coefficients in the regression equation relating to target and control totals remain

constant over the year. Thus, separate regressions, i.e. one for each month, should be estimated. Regression models for this type of data are fitted with zero intercepts. This is done to avoid two difficulties associated with regression models whose intercept terms are constants, namely if the intercept is positive, then the estimated values can never be zero even though it did not rain at any of the control stations during the relevant period; on the other hand, if the estimated value is negative, then one might end up with negative estimates for missing rainfall totals.

Since we do not fit the intercept term, the data is then not standardized as it was when using annual data.

To overcome the above-mentioned problems, the alternative is to use annual rainfall data which will be discussed in section 6.3, although it does have the disadvantages mentioned above.

The EM algorithm applied to monthly data is similar to the one applied to annual data with the difference that data is not first standardized and there is no intercept term fitted to regression models.

Instead of having one data matrix as for annual rainfall totals, here one looks at twelve data matrices for 12 different months which are then estimated separately. In other words, the EM algorithm (see section 3.8) is performed 12 times whereas it was only performed once for the annual rainfall data.

6.3 DISAGGREGATION OF THE ANNUAL RAINFALL TOTALS TO MONTHLY RAINFALL TOTALS

This method entails the estimation of annual rainfall totals and the disaggregation of that data into monthly rainfall totals. In this case, it means that if some of the monthly totals are missing in a year, then that year is regarded as missing and the partially available annual data are initially not utilized.

There is one major problem which is often encountered in the case where some of the monthly totals in a year are available while the others are not. Since the missing monthly totals are obtained by the disaggregation of annual data after estimation, this can lead to negative monthly rainfall values being obtained.

To implement this method one begins by treating the whole year as missing and

then applying the algorithm discussed in section 3.8 to estimate the total for the year. From this total one then subtracts the sum of the monthly totals for those months of the year when there was in fact data available. This remaining rainfall is then disaggregated over the relevant months in proportion most appropriate to the corresponding monthly totals recorded at the most appropriate control station, i.e. the station whose annual rainfall totals are most correlated with those of the target station. If some of the values from the selected control station corresponding to the estimated target station were originally missing, then disaggregation is performed by considering the second most highly correlated control station and so forth.

The only time when one can guarantee not getting negative monthly rainfall estimates using this method, is when the whole year is missing. For this reason we would only recommend the use of the method in this case.

6.4 DAILY RAINFALL RECORDS

Estimating daily rainfall data is often difficult because of a high probability of zero rainfall on individual days, i.e. the distribution of daily rainfall values is in part continuous and in part discrete. We suggest that if daily data is required, this should be obtained by disaggregation only. Disaggregating annual or monthly data into daily data can be done by using the proportion of daily data from most appropriate control station, i.e. whose rainfall totals are most correlated with that of the target station. We recommend the use of the disaggregation method since estimating individual daily values would take-up vast amounts of storage and computing time.

6.5 COMPARISON

As we have seen from section 6.1, there are four options that can be used when estimating missing rainfall data, namely: estimating missing annual rainfall totals to get annual rainfall totals, estimating missing monthly rainfall totals and aggregate to obtain annual rainfall totals, estimating missing annual rainfall totals and disaggregate to obtain monthly rainfall totals, and estimating missing monthly rainfall totals to obtain monthly rainfall totals. We will refer to these options as option 1, option 2, option 3 and option 4 respectively.

In this section, we compare the four options in terms of the sum of square error due to prediction (SSEP), the preservation of the mean, the preservation of the standard deviation and the computational complexity and expense.

TABLE 6.1: *Computed averages of the "actual" means, estimated means and SSEP's for the annual rainfall totals*

	ST1	ST2	ST3	ST4	ST5	ST6	ST7	ST8	ST9	ST10
A-MEAN	997	1001	1000	995	1001	1000	1004	1003	1002	996
					EM to Annual Rainfall Totals					
E-MEAN	998	1002	1000	996	1001	1001	1004	1003	1003	996
SSEP	134	102	114	104	99	103	97	99	110	102
					EM to Monthly Rainfall Totals and aggregate					
E-MEAN	998	1001	1001	996	1001	1001	1004	1003	1003	996
SSEP	9	6	8	7	6	7	6	7	7	7

A-MEAN represents the "actual" mean

E-MEAN represents the estimated mean

6.5.1 Preservation of the Mean

We have already mentioned in chapter 5 that the EM algorithm method discussed in chapter 3 and the regression methods discussed in chapter 2 do not introduce systematic bias on the mean. This conclusion was based on the methods being applied to annual data to obtain annual rainfall totals. From Table 6.1, it can be seen that if the annual data was estimated by applying the EM algorithm to monthly totals and aggregated to get annual rainfall totals, the mean was still preserved. From Table 6.2, we can see that, although the mean was preserved when monthly rainfall totals were estimated by applying the EM algorithm to monthly totals, it was under-estimated when the method was applied to annual totals and then disaggregated.

TABLE 6.2: *Computed averages of the "actual" means, estimated means and SSEP's for monthly rainfall totals*

	ST1	ST2	ST3	ST4	ST5	ST6	ST7	ST8	ST9	ST10
A-MEAN	84	84	84	84	84	84	84	84	84	84
	EM to Annual Rainfall Totals and disaggregate									
E-MEAN	80	77	81	79	80	80	78	79	81	79
SSEP	84	84	46	84	65	84	84	65	65	83
	EM to Monthly totals									
E-MEAN	84	84	84	84	84	84	84	84	84	84
SSEP	2	2	1	1	2	2	3	1	0	0

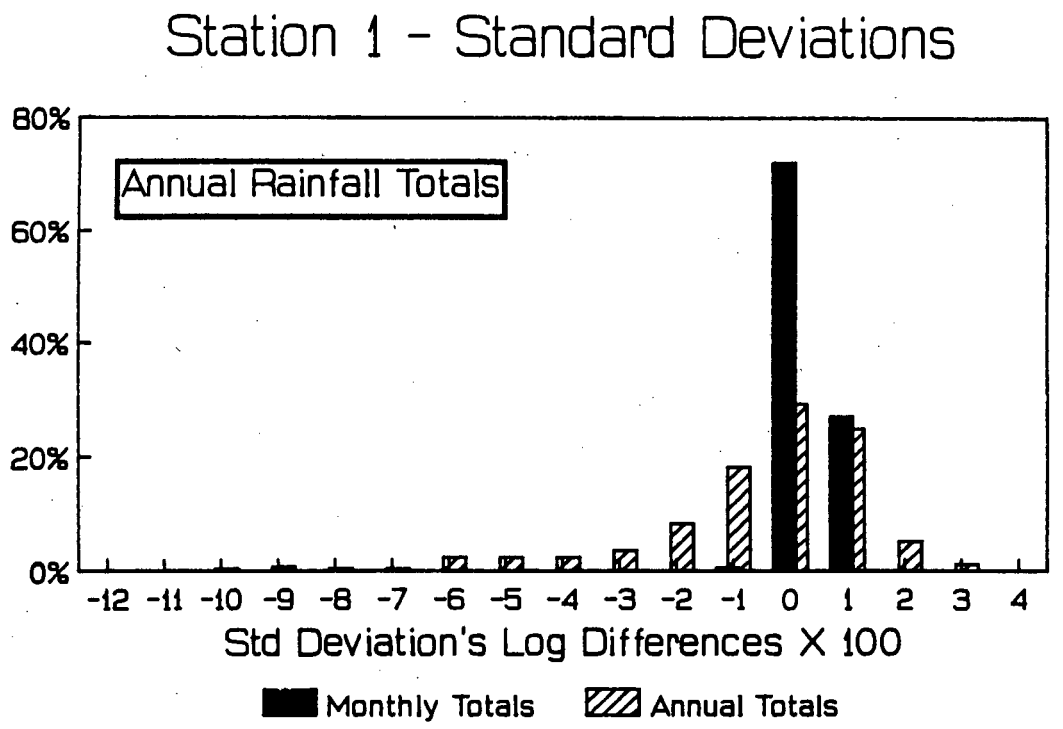
6.5.2 Sum of Square Error due to Prediction

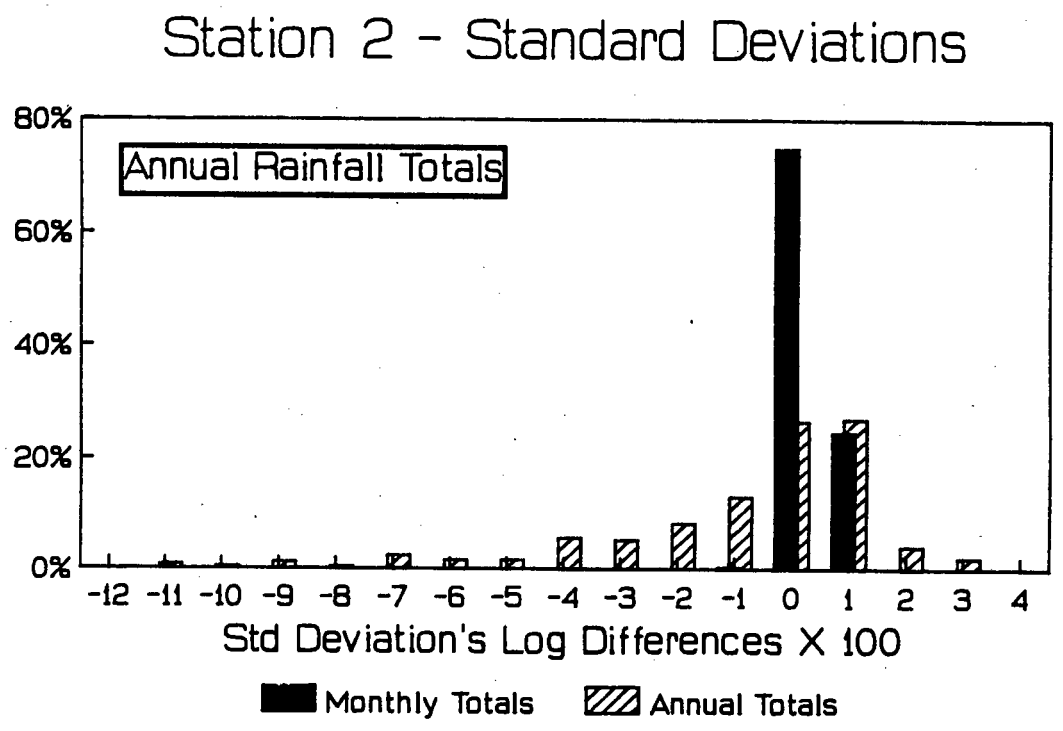
Comparing the 4 options in Tables 6.1 and 6.2, it is clear that when the EM algorithm was applied to monthly data, either to obtain monthly rainfall totals or annual rainfall totals, the values of the sum of square error due to prediction were much smaller than when using the other two options. We mentioned in chapter 5 that, the smaller the SSEP, the more accurate the method of estimation is. It then means that estimating monthly data gave the most accurate estimates of missing rainfall values although there is a drawback in that it cannot be guaranteed that all the estimated monthly rainfall totals are positive.

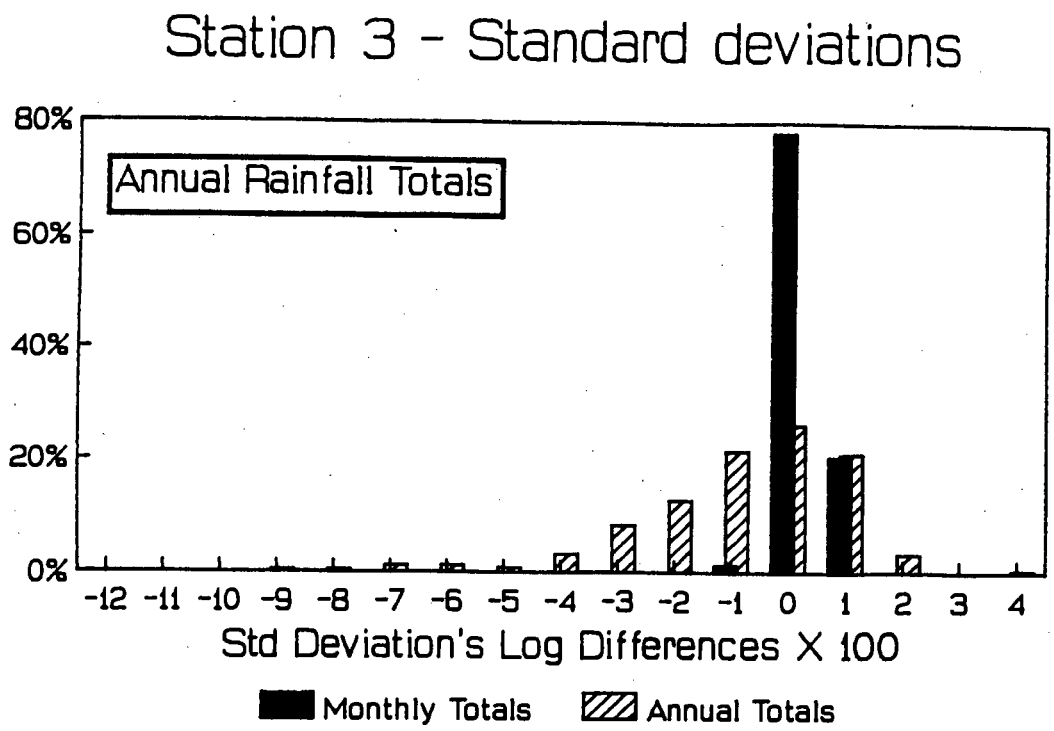
6.5.3 Preservation of the Standard Deviation

Figure 6.1 shows the graphs of the log differences of the standard deviations obtained when applying the EM algorithm to annual rainfall totals to get annual rainfall totals and when the disaggregated annual rainfall totals were estimated by applying the EM algorithm to monthly rainfall totals. It is clear from these graphs that applying the EM algorithm to monthly totals did not lead to high reduction of the standard deviation although at the same time it did not preserve the standard deviation. We have already mentioned that the higher the correlation coefficient,

FIGURE 6.1: *Log differences of annual standard deviations - Annual rainfall totals and aggregated monthly rainfall totals*






Station 3 - Standard deviations
80%
60%
40%
20%
0%
Annual Rainfall Totals
-12 -11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Monthly Totals
Annual Totals

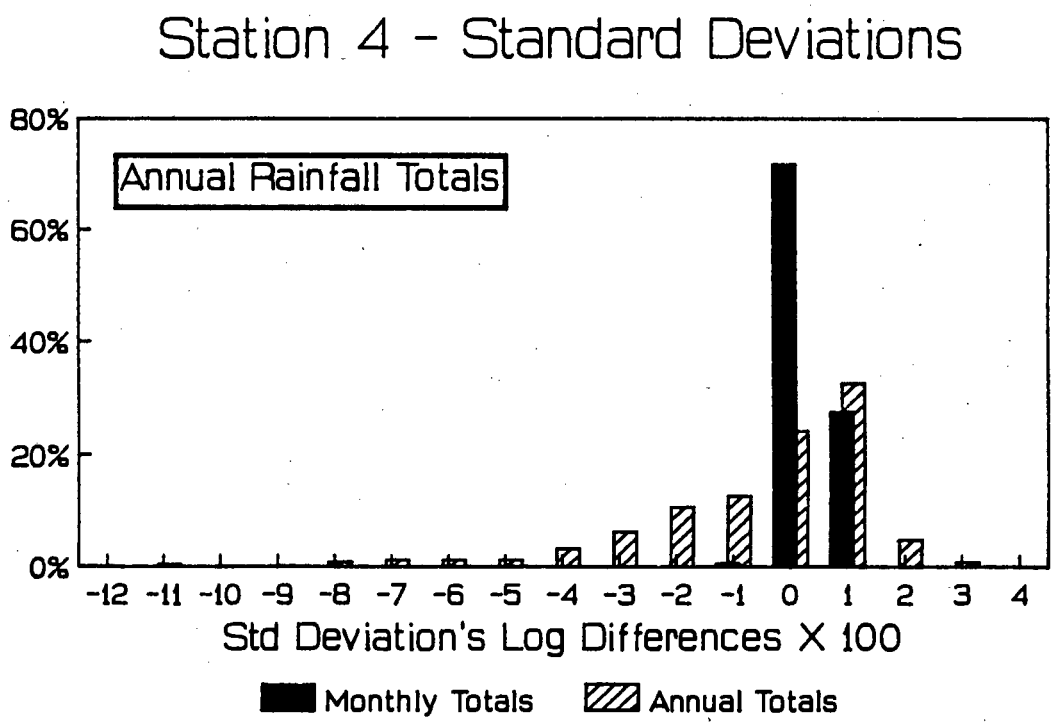
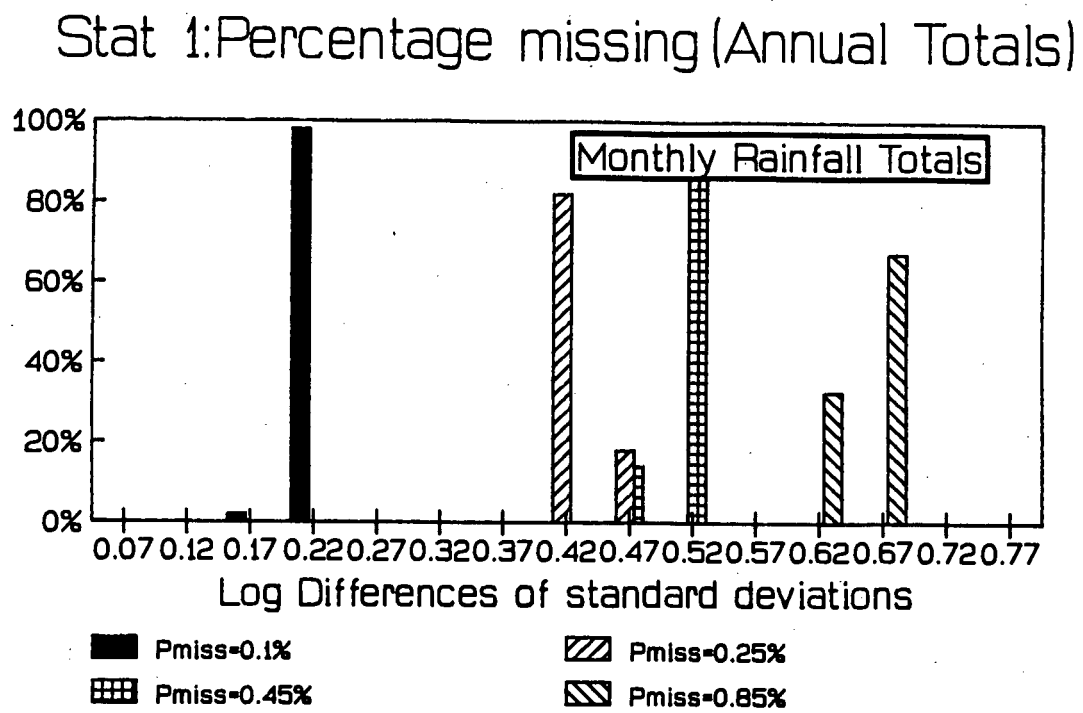
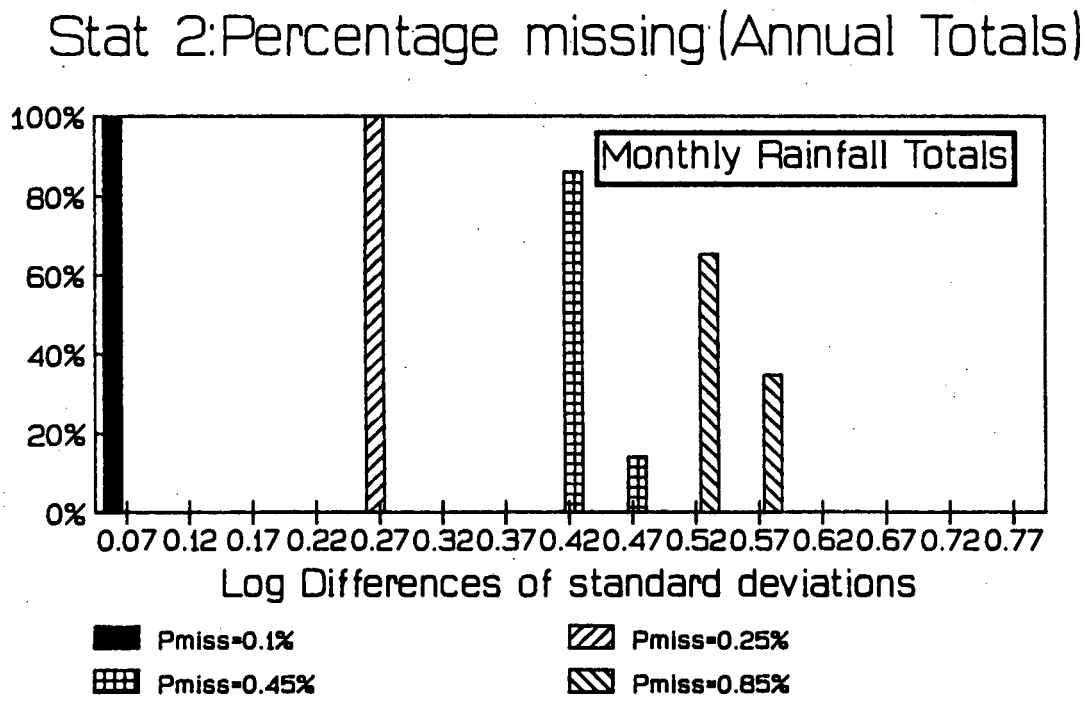
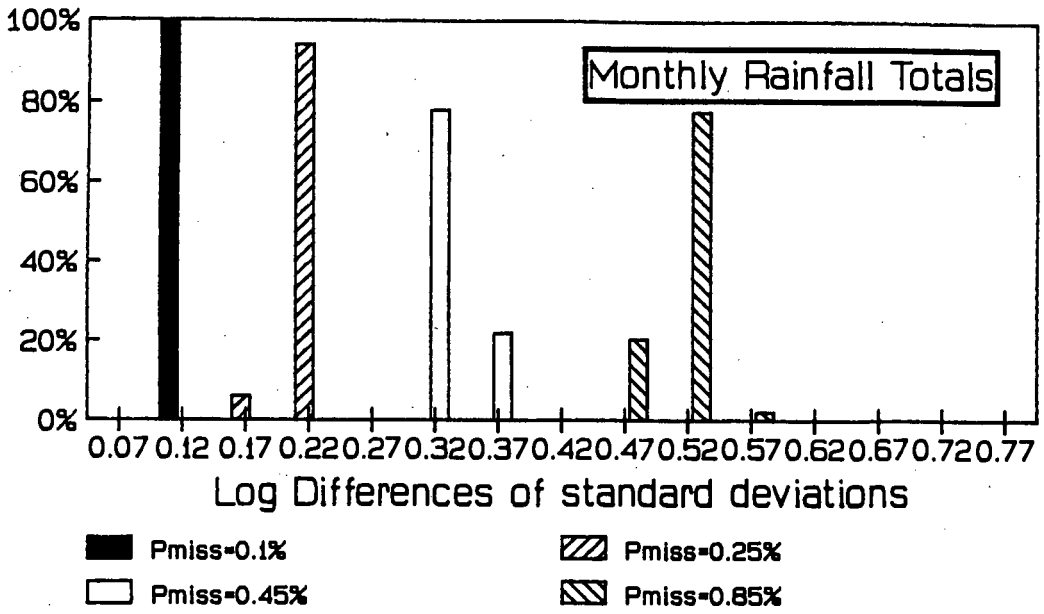

Station 4 - Standard Deviations
80%
60%
40%
20%
0%
Annual Rainfall Totals
-12 -11 -10 -9 -8 -7 -6 -5 -4 -3 -2 -1 0 1 2 3 4
Std Deviation's Log Differences X 100
Monthly Totals
Annual Totals

FIGURE 6.2: *Log differences of monthly standard deviations - Disaggregated annual rainfall totals*

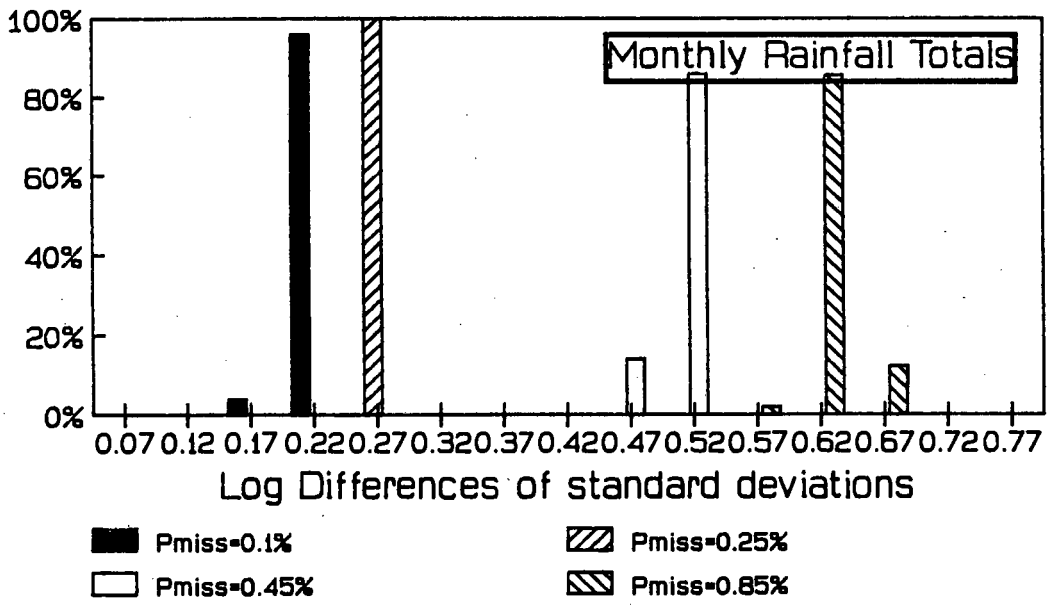




Stat 3:Percentage missing (Annual Totals)



Stat 4:Percentage missing (Annual Totals)



the less reduction of the standard deviation, similarly, the more missing values are there in the data, the higher reduction of the standard deviation and vice versa. This also applied when the EM algorithm was applied to monthly rainfall totals. From Chapter 5, we have seen that when applying the EM algorithm to annual rainfall totals, we introduce a downward bias to the standard deviation (see Chapter 5).

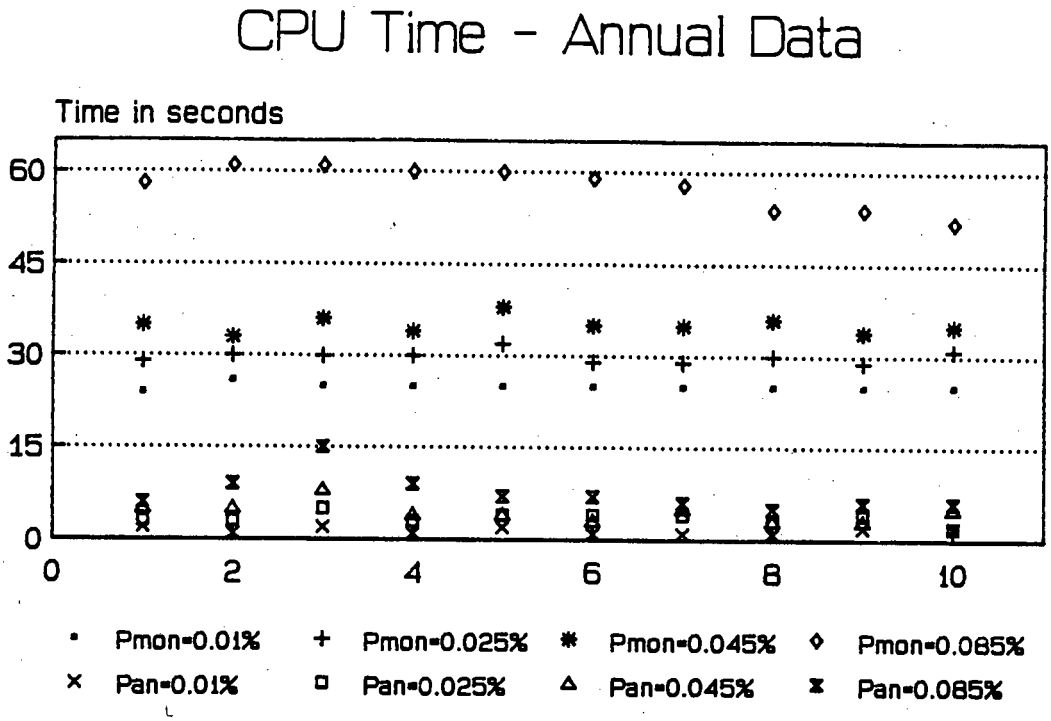
Figure 6.2 shows the graphs obtained when the EM algorithm was applied to annual rainfall totals data which was then disaggregated to obtain monthly rainfall totals. It can be seen from these graphs that, increasing the percentage of missing values in the data, increased the over-estimation of the standard deviation. From chapter 5 we saw that the EM algorithm introduced a downward bias to the standard deviation which increased when the percentage of missing values increased. We must bear in mind that, disaggregation of annual rainfall totals leads to negative monthly rainfall totals being obtained. The standard deviations obtained by applying the EM algorithm to monthly rainfall totals to get monthly totals gave the same shape of histograms as those obtained by using the disaggregated monthly rainfall totals (Figure 6.1).

6.5.4 Computation

Figure 6.3 represents the CPU time used by applying the 1st and 2nd options of estimating missing values. It is clear that applying the EM algorithm to monthly totals was more expensive than when the algorithm was applied to annual rainfall totals. For both options, the higher the percentage of missing values in the data or the lower the correlation coeffiecient, the more expensive the estimation becomes. On average, estimating disaggregated monthly data and then aggregating it was about 8 times more expensive than estimating annual data.

Figure 6.4 represents the CPU time used by applying the 3rd and the 4th options of estimating missing values. It is again clear that the EM algorithm was more expensive when it was applied to monthly rainfall totals than when it was applied to annual rainfall totals. On average, estimating monthly data was about 4 times more expensive than estimating annual data and disaggregating.

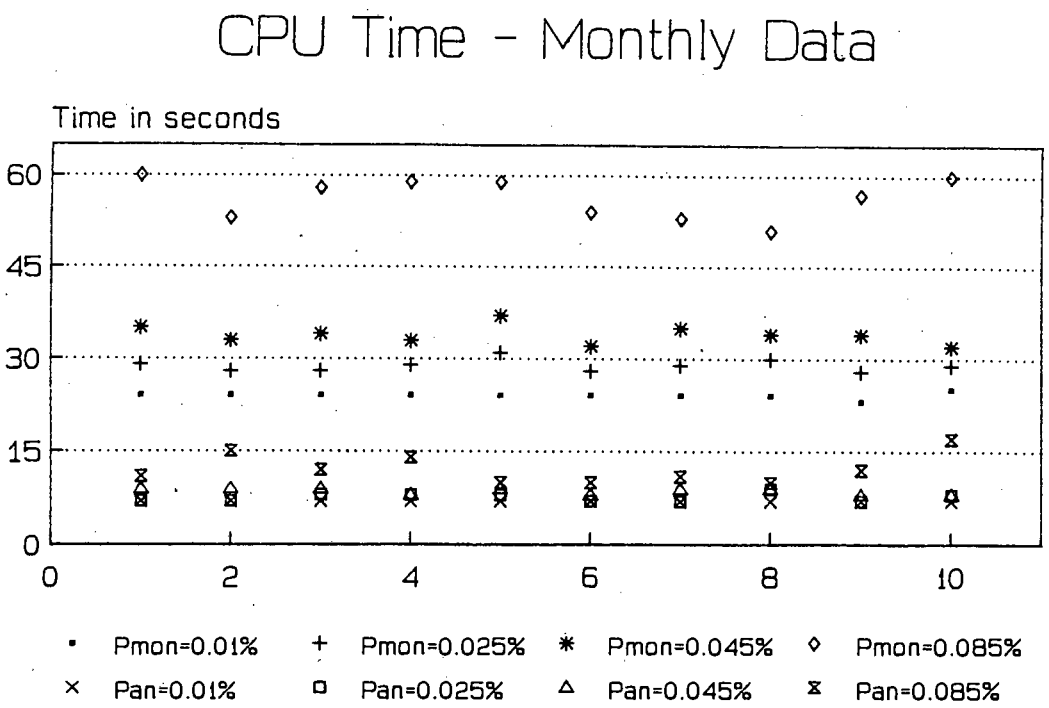
FIGURE 6.3: *Computation Time - Annual Rainfall Totals and Aggregated Monthly Rainfall Totals*



Note

All the results were obtained for whole month missing only. In other words, the partly available monthly data was not utilized when the estimation of monthly data was performed.

FIGURE 6.4: *Computation Time - Monthly Rainfall Totals and Disaggregated Annual Rainfall Totals*



CHAPTER 7

APPLICATION OF THE METHODS

In chapters 2 and 3 we discussed several methods of estimating missing records, namely, selecting control stations for individual missing values, selecting control stations for several missing values, forward selection and two variations of the EM algorithm. The performance of these methods was assessed by using simulation which was discussed in chapter 4. The comparisons themselves were performed in chapter 5. These were performed in terms of the sum of squared error due to prediction, preservation of the mean, preservation of the standard deviation and the computational complexity and expense.

In this chapter, as an illustration of the methods discussed in the previous chapters, we give an example of real rainfall data with computed estimates of the missing rainfall records. The intention is to show that, although our comparison was based of estimates which were obtained from using simulated data, the methods we have looked at can be applied to real data.

Table 7.1 gives the annual rainfall totals for the rainfall *sector number 239* for six different stations, namely *station 97*, *station 138*, *station 482*, *station 566*, *station 577* and *station 605*. Some of the stations contain missing annual rainfall records.

Table 7.2 gives the estimated values obtained when the five methods of estimation have been applied. From this table as expected, it can be seen that the two EM algorithm methods gave the same estimates for all the missing annual totals. Forward Selection and selecting control stations for individual missing values yield similar estimates of the missing values except for *station 138* where Forward selection selected the "degenerate model". Selecting control stations for several values has few estimates which are common to those obtained from the other two regression methods.

These results indicate that at least for this data set, each of the methods yield credible estimates and that these do not differ substantially from each other. For the purpose of illustration the methodology proposed yields reasonable estimates of monthly values, we applied the EM algorithm to estimate the missing monthly

TABLE 7.1: *Annual Rainfall totals*

Year	0239-97	0239-138	0239-482	0239-566	0239-577	0239-605
1947	10858	9858	11007	10900	*-999*	*-999*
1948	8998	8906	9045	8454	*-999*	10209
1949	8473	7375	7851	8707	*-999*	10631
1950	9450	8798	7401	10549	8865	9395
1951	8145	7428	6886	7214	8641	9319
1952	8971	6899	7570	7160	7218	10874
1953	*-999*	7761	7707	8818	9829	9681
1954	9084	8464	7649	8003	8577	10691
1955	*-999*	10300	9890	9978	10956	11368
1956	11239	10692	9225	9656	9337	10872
1957	12615	8895	12599	9693	11324	12586
1958	9693	7591	8395	8155	9498	10909
1959	10887	9426	9553	*-999*	*-999*	10532
1960	10637	8052	7578	9283	9211	9955
1961	9671	7734	8475	10041	10868	11045
1962	9862	8436	7606	7759	7406	9056
1963	13834	8341	9174	9180	8378	9067
1964	10194	7905	9114	9071	8586	9745
1965	11786	7375	7575	9619	8049	7743
1966	9203	9473	7831	8178	8202	7938
1967	8493	9165	9784	11397	11685	10003
1968	9364	6147	7370	7226	7638	8750
1969	9377	8703	9278	9546	10086	9577
1970	9402	9103	7228	9323	9278	8620
1971	9300	7567	8285	9522	9520	11101
1972	11263	*-999*	7927	7950	8327	8290
1973	10296	*-999*	9214	7511	9203	11201
1974	11535	*-999*	10928	*-999*	9357	*-999*

TABLE 7.2: *Estimates of the missing values using different methods*

Station	Year	Individual Missing Values	Several Missing Values	Forward Selection	EM Algorithm "Standard"	EM Algorithm "Variation"
0239-97	1953	8612	9622	8612	8568	8568
	1955	10077	10746	10077	9696	9696
0239-138	1972	7816	7787	8250	7828	7828
	1973	8697	7555	8250	7854	7854
	1974	9438	9438	8250	8982	8982
0239-566	1959	9784	9676	9784	9614	9614
	1974	9037	9037	9037	8721	8721
0239-577	1947	11433	10653	11433	11379	11379
	1948	9514	8821	9514	9497	9497
	1949	9303	9011	9303	9432	9432
	1959	9715	9745	9715	9832	9832
0239-605	1947	11550	11471	11550	11389	11389
	1974	11279	11424	11279	11160	11160

(Units mm/10)

values. (Recall that the EM algorithm is computationally the least expensive method to implement.)

Tables 7.3, 7.4, 7.5, 7.6 and 7.7 are tables of the monthly totals for stations 97, 138 482, 566, 577 and 605 respectively. The values which are in italics and have negative signs are the estimates of those monthly totals which were missing.

TABLE 7.3 : *Station no. 239-97: Monthly Rainfall Totals*

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1947	1762	1383	1714	553	133	813	150	56	335	1018	1574	1367
1948	966	1918	2110	796	97	10	50	26	255	836	733	1201
1949	1415	1106	979	424	135	10	91	314	562	675	1148	1614
1950	1148	1324	1708	509	282	18	272	690	202	458	1034	1805
1951	1701	917	873	414	39	30	0	867	524	1089	439	1252
1952	1569	1377	803	654	260	163	223	438	194	635	1316	1339
1953	1191	2498	549	491	91	221	0	722	*-862*	*-1157*	1173	1801
1954	1087	1415	907	346	632	125	359	86	622	1386	1242	877
1955	2185	2456	1186	672	284	114	36	74	*-489*	*-731*	1196	2054
1956	262	3045	1374	426	358	39	28	518	356	1005	1484	2344
1957	2366	1483	1207	757	74	41	181	498	2116	1515	1260	1117
1958	1357	1285	722	1450	115	155	15	97	325	444	1599	2129
1959	1452	1517	735	476	2159	0	105	543	312	1094	1117	1377
1960	651	1694	1186	1250	193	8	94	200	615	969	1596	2181
1961	830	1142	1567	1108	235	164	195	180	825	556	1506	1363
1962	2055	1355	890	570	70	2	80	130	1385	510	1065	1750
1963	2840	2282	2780	675	360	0	60	288	838	1450	726	1535
1964	2152	981	1701	350	150	130	0	337	557	895	1074	1867
1965	1938	904	1330	1245	111	41	0	361	819	1336	1519	2182
1966	1597	1486	901	866	429	66	375	512	475	523	821	1152
1967	1372	857	628	530	199	0	20	75	1280	621	1457	1454
1968	1548	1633	712	539	241	349	85	918	615	449	1277	998
1969	883	777	3008	421	177	65	43	68	422	1057	1152	1304
1970	904	479	1104	403	272	40	392	282	335	810	2024	2357
1971	1913	1260	927	854	171	283	185	249	161	1582	884	831
1972	1603	2821	540	276	66	26	77	15	232	1811	1448	2348
1973	2556	606	1842	322	20	222	0	773	348	1225	1153	1229
1974	1082	2375	835	665	2390	1	235	27	155	780	1365	1625

TABLE 7.4: *Station no. 239-138: Monthly Rainfall Totals*

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1947	1003	1632	1407	377	117	1524	178	36	369	821	1661	733
1948	1435	1413	1506	633	86	0	0	5	74	867	1295	1592
1949	1237	956	1142	483	79	0	198	112	426	753	1012	977
1950	1405	1204	1155	229	318	0	409	673	149	595	843	1818
1951	1756	953	948	186	46	33	0	953	406	727	227	1193
1952	1469	648	709	778	76	147	147	191	107	208	1291	1128
1953	1920	655	706	375	118	357	0	547	390	878	661	1154
1954	1504	1769	1225	262	486	152	125	20	518	844	982	577
1955	2110	2645	1218	1208	208	50	27	46	419	515	840	1014
1956	206	2155	1577	516	379	80	0	262	295	767	1561	2894
1957	1448	590	1649	365	88	0	115	455	981	1088	962	1154
1958	1300	1223	531	989	0	0	0	0	344	379	1234	1591
1959	1052	1188	601	284	3108	35	71	101	378	563	1161	884
1960	762	780	1842	696	104	0	50	89	562	434	1034	1699
1961	1227	1191	1296	964	144	60	0	0	536	243	1120	953
1962	1548	1411	1217	552	0	0	0	778	103	507	1680	640
1963	1110	265	2400	592	0	155	1030	90	0	821	1045	833
1964	1749	736	666	695	82	665	118	11	560	856	565	1202
1965	1581	1012	60	176	0	1004	113	420	237	902	1190	680
1966	2942	1215	385	370	305	125	0	308	358	662	1716	1087
1967	1500	1464	1857	1029	60	265	475	0	65	591	1158	70
1968	1165	602	1155	306	28	0	0	422	458	354	746	911
1969	644	1580	2286	273	747	0	365	0	384	1029	528	867
1970	1255	840	352	119	262	480	20	1010	1100	1035	755	1875
1971	465	1363	876	335	740	8	530	760	250	724	645	871
1972	1970	*-1308*	1179	187	269	105	292	97	115	712	1125	*-778*
1973	1565	1695	1229	*-467*	140	0	130	535	995	245	2273	2766
1974	*-2322*	*-1625*	*-1471*	*-480*	*-1382*	*-189*	*-272*	*-250*	*-186*	*-464*	*-925*	*-1199*

TABLE 7.5: *Station no. 239-482: Monthly Rainfall Totals*

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1947	1674	1303	1199	768	169	1054	189	101	374	853	1934	1389
1948	1501	1017	1555	1402	181	5	5	29	339	662	1284	1065
1949	1203	1263	970	544	262	20	84	36	457	419	1259	1334
1950	891	853	1148	618	237	8	234	652	113	436	789	1422
1951	1604	623	523	394	20	74	0	1008	317	830	256	1237
1952	1748	1164	824	477	161	60	77	328	138	656	852	1085
1953	1186	2099	320	296	101	30	0	830	340	778	519	1208
1954	1311	1718	436	411	455	64	45	0	588	1160	978	483
1955	1923	2257	1294	362	692	234	8	99	400	510	996	1115
1956	271	1807	1228	545	207	33	70	169	304	612	2091	1888
1957	2184	722	980	602	38	47	160	370	1591	3765	1103	1037
1958	919	1145	602	1389	46	16	14	42	519	382	1488	1833
1959	1301	1550	508	549	2138	6	69	226	170	847	1069	1120
1960	393	1149	1158	990	30	35	67	91	435	417	1172	1641
1961	1798	585	1084	1212	168	64	286	179	590	349	1133	1027
1962	1324	1049	715	653	40	0	0	424	73	572	1461	1295
1963	1331	1001	2240	584	12	249	911	116	67	1457	796	410
1964	1977	527	1203	752	109	395	130	18	763	1000	1047	1193
1965	1130	696	178	153	251	735	236	589	359	744	1258	1246
1966	1931	1144	118	390	315	69	11	364	211	740	1485	1053
1967	1858	1523	1806	645	159	82	132	15	72	842	1339	1311
1968	1350	770	1122	538	28	5	0	606	468	526	746	1211
1969	764	1123	1360	512	581	119	132	287	623	1368	650	1759
1970	1503	621	500	96	232	0	0	0	0	1867	893	1516
1971	1021	1000	648	683	1124	8	398	704	321	615	740	1023
1972	1246	1117	1173	359	511	197	189	129	63	778	1678	487
1973	1872	1446	844	949	45	1	163	500	784	443	970	1197
1974	3189	1871	1579	888	170	159	259	81	96	337	1120	1179

TABLE 7.6: *Station no. 239-566: Monthly Rainfall Totals*

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1947	1373	1428	1971	667	137	1090	246	124	375	779	1714	996
1948	1375	1181	1297	580	191	0	0	88	229	1296	1145	1072
1949	1201	1417	1215	432	127	0	79	150	580	793	1582	1131
1950	1955	1892	1215	368	510	0	384	682	158	622	696	2067
1951	1375	970	892	84	96	64	0	1150	472	943	255	913
1952	1584	982	987	549	15	178	135	140	284	412	794	1100
1953	893	1656	475	358	193	193	0	1095	642	932	924	1457
1954	1022	730	698	584	585	196	132	43	805	1529	1183	496
1955	2384	1762	901	660	234	252	61	102	573	803	1033	1213
1956	216	1993	2060	496	320	119	0	323	350	682	1094	2003
1957	1804	999	1414	869	121	0	135	431	1201	1023	923	773
1958	992	1563	254	1103	0	145	99	43	569	407	1399	1581
1959	1126	1118	287	330	*-2337*	0	46	285	386	566	875	815
1960	598	929	1405	848	94	18	41	243	539	768	1687	2113
1961	1806	568	1378	1957	200	203	74	33	1150	343	1280	1049
1962	1237	829	1077	206	84	0	0	864	159	919	1568	816
1963	2315	628	1683	516	48	74	944	58	61	592	842	1419
1964	1870	800	736	821	51	496	46	145	759	1304	711	1332
1965	937	686	569	278	597	1279	275	618	462	1728	1338	852
1966	2263	1354	249	303	537	134	46	371	529	635	1202	555
1967	1599	2075	2158	1500	61	279	389	53	105	965	1331	882
1968	1384	1007	1407	175	71	0	0	817	537	599	764	465
1969	343	1193	1611	579	650	109	295	322	567	1457	923	1497
1970	808	783	401	506	363	590	33	810	1186	1408	1035	1400
1971	637	645	1410	959	1033	0	531	892	551	737	890	1237
1972	1440	1896	796	211	415	315	117	310	97	593	904	856
1973	939	1169	1060	595	58	0	119	660	708	376	1084	743
1974	*-1834*	1159	1295	465	612	130	279	284	155	757	1095	1206

TABLE 7.7: *Station no. 239-577: Monthly Rainfall Totals*

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1947	*-1528*	*-1274*	*-1323*	*-546*	*- 83*	*-710*	*-104*	*-145*	*-248*	*-1246*	*-1818*	*-1365*
1948	*-1360*	*-1061*	*-1798*	*-760*	*-161*	*- 3*	*-116*	*-146*	*-284*	*-972*	*-1261*	*-1511*
1949	*-1245*	*-1857*	*-1365*	*-398*	*-110*	*- 11*	*-135*	*-340*	*-557*	*-828*	*-1786*	1531
1950	1413	728	1453	628	234	31	427	483	163	479	818	2008
1951	2043	718	815	393	56	45	0	1093	422	1151	526	1379
1952	2143	1476	1042	546	221	109	6	22	102	304	594	653
1953	1248	2296	895	492	92	61	4	1063	659	992	529	1498
1954	1923	1080	740	193	635	6	70	100	615	1685	930	600
1955	2090	2735	1804	525	82	159	9	131	403	784	1056	1178
1956	251	1419	1401	438	224	106	30	205	374	1066	1708	2115
1957	1536	897	1110	1187	124	114	154	354	1771	1317	1390	1370
1958	999	1609	626	1345	33	45	2	139	424	436	1567	2273
1959	1650	1339	722	446	*-1168*	2	72	253	*-154*	1003	1267	1472
1960	474	1189	967	857	10	50	92	167	380	832	1293	2900
1961	1859	1289	1831	1547	244	58	63	232	725	587	1365	1068
1962	1499	1026	815	437	2	0	0	231	116	665	1627	988
1963	1193	952	2269	686	6	91	795	91	155	757	989	394
1964	1938	653	849	494	37	310	183	23	817	1228	774	1280
1965	1398	721	214	120	222	615	77	556	626	549	1531	1420
1966	2016	950	218	493	244	30	65	194	266	1005	1781	940
1967	2058	2351	1567	1129	227	49	30	15	132	1430	1564	1133
1968	1408	815	1251	246	18	2	0	470	661	730	855	1182
1969	603	1091	1412	463	500	105	91	254	1145	1568	1105	1749
1970	1633	742	680	120	336	77	33	802	835	974	1265	1781
1971	1236	661	1256	713	826	0	316	428	438	1014	1266	1366
1972	1255	1266	986	424	486	223	270	191	186	999	1127	914
1973	1217	1155	1023	656	35	0	152	475	911	717	1807	1055
1974	2668	1066	1406	635	215	30	205	111	240	364	1016	1401

TABLE 7.8: *Station no. 239-605: Monthly Rainfall Totals*

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1947	*-1817*	1266	1245	565	61	418	87	134	307	1366	1569	1334
1948	1357	1211	1883	1015	232	15	422	115	310	882	977	1790
1949	1374	1764	1462	422	132	10	68	323	611	770	2071	1624
1950	1598	751	1654	836	255	18	361	629	181	530	1013	1569
1951	2131	847	746	366	41	26	166	1168	557	1175	611	1485
1952	2336	1594	1073	495	857	89	69	345	246	699	1447	1624
1953	1619	1867	1127	470	121	110	23	882	354	1062	626	1420
1954	2474	1581	807	283	742	28	111	82	677	1861	1192	853
1955	2202	2579	1762	551	118	251	6	155	476	834	1125	1309
1956	247	1851	1420	626	551	74	25	185	437	1037	1854	2565
1957	1609	887	1204	1440	80	84	221	446	1962	1497	1649	1507
1958	1402	1772	744	1434	44	41	18	130	552	557	1586	2629
1959	1619	1315	532	619	1789	8	84	209	175	1401	1454	1327
1960	477	1662	812	1043	25	48	172	146	573	1094	1241	2662
1961	2200	1067	1720	1395	213	71	131	231	830	480	1520	1187
1962	1358	1164	1138	471	20	637	366	294	144	940	1404	1120
1963	1321	870	2482	809	91	120	876	64	64	772	1129	469
1964	2029	663	1000	1094	49	391	181	25	968	1162	644	1539
1965	1141	607	152	140	330	683	105	532	689	590	1345	1429
1966	2029	910	148	553	254	31	38	320	187	978	1603	887
1967	2038	1762	1427	806	168	54	79	15	115	830	1666	1043
1968	1319	614	1440	321	36	1279	0	524	597	627	834	1159
1969	533	986	1349	413	499	82	67	308	1072	1498	986	1784
1970	1346	648	625	156	343	96	48	864	834	877	1181	1602
1971	1866	667	1157	853	839	11	387	791	470	1541	1126	1393
1972	1384	1394	789	381	464	152	118	165	171	969	1325	978
1973	2177	1273	861	1149	36	716	142	565	1254	707	1274	1047
1974	3895	1169	1290	937	177	121	216	132	290	*-478*	1044	1411

CHAPTER 8

SUMMARY AND CONCLUSIONS

Two classes of methods of estimating missing data have been discussed. The first one was based on the theory of variable selection in regression analysis. The emphasis in this class was on finding efficient methods to identify the set of control stations which were likely to yield the best regression estimates of the missing values in the target station. We discussed three of the methods in this class, namely, selecting control records for individual missing values, selecting control records for several missing values and forward selection procedure.

The second class of methods was based on the EM algorithm, which was proposed by Dempster, Laird and Rubin (1977). The emphasis was to estimate the missing values directly without first making a detailed selection of control stations; all "relevant" stations are included. This method has not previously been applied in the context of estimating missing rainfall values.

To assess the performance of the two classes of methods, we simulated artificial multivariate normally distributed data. Factors which were likely to be important in determining the performance of the different approaches were allowed to vary. Such factors included the correlation between observations at different stations, the proportion of missing values and the length of records at each station. Some of the simulated values were "hidden" and were then assumed to be missing. The "missing" values were then estimated by applying the different methods of estimation and comparison was performed based on the sum of square error due to prediction, preservation of the mean, preservation of the standard deviation and computational complexity and expense.

All methods which were discussed preserved the mean except for the estimation of monthly rainfall totals which was performed by applying the EM algorithm to annual rainfall totals and disaggregating the totals. Even in this case the bias was slight. The standard deviations for all the methods showed a downward bias although the EM algorithm performed somewhat better than the regression methods in this respect. The values obtained when calculating the sum of square error due to prediction were small when the estimation was performed by applying the EM

algorithm to annual rainfall totals, and were even smaller when monthly rainfall totals were utilized for the estimation of the missing values. Computationally, the regression methods were more expensive than the EM algorithm.

Estimating monthly data has a drawback in that it is always possible to get negative estimates for the monthly rainfall totals. On the other hand utilization of annual data and disaggregation to obtain monthly rainfall totals has the same disadvantage in that, if some of the monthly totals are available, then disaggregation of the estimated annual rainfall totals can also lead to negative estimates.

From the properties which were used for the comparison of these methods, we can conclude that, of the methods investigated, the EM algorithm is the most efficient method that can be applied for the estimation of missing records. In terms of accuracy it performs at least as well as methods based on the selection of control stations.

REFERENCES

ACTON, F.S. (1970). *Numerical Methods that work,* Harper & Row Publishers, Inc., New York.

ADAMSON, P.T. (1987). South African rainfall database. *Technical Report TR 133.* Published by Department of Water Affairs, Private Bag X313, Pretoria.

ALLEN, D.M. (1971). Mean square error of prediction as a criterion for selecting variables. *Technometrics,* **13,** 469-481.

BEALE, E.M.L. (1970). Note on procedures for variable selection in multiple regression. *Technometrics,* **12,** 909-914.

BOYLES, R.A. (1983). On the convergence of the EM algorithm. *Journal of the Royal Statistical Society,* **B45,** 45-50.

DEMPSTER, A.P., LAIRD, N.M. and RUBIN, D.B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society,* **B39,** 1-38.

DRAPER, N. and SMITH, H. (1981). *Applied regression analysis,* John Wiley & Sons, New York.

FURNIVAL G.M. and WILSON R.W. Jr (1974). Regression by leaps and bounds. *Technometrics,* **16,** 499-511.

GORMAN, J.W. and TOMAN, R.J. (1966). Selection of variables for fitting equations to data. *Technometrics,* **8,** 27-51.

HARTLEY, H.O. (1958). Maximum likelihood estimation from incomplete data. *Biometrics,* **14,** 174-194.

HOCKING, R.R. (1972). Criteria for selection of a subset regression: Which one should be used? *Technometrics,* **14,** 967-970.

HOCKING, R.R. (1976). The analysis and selection of variables in linear regression. *Biometrics,* **32,** 1-49.

HOCKING, R.R. (1985). *The analysis of linear models,* Brooks/Cole Publishing Company, Monterey, California.

KENNARD, R.W. (1971). A note on the C_p statistics. *Technometrics,* **13,** 899-900.

LINHART H. and ZUCCHINI W. (1986). *Model Selection,* John Wiley & Sons, New York.

LITTLE, R.J.A. and RUBIN, D.B. (1987). *Statistical analysis with missing data,* John Wiley & Sons, New York.

MANTEL, N. (1970). Why stepdown procedures in variable selection. *Technometrics,* **12,** 621-625.

MURRAY, G.D. (1977). Contribution to discussion of paper by A.P. Dempster. N.M. Laird and D.B. Rubin. *Journal of the Royal Statistical Society,* **B39,** 27-28.

SEBER, G.A.F. (1977). *Linear regression analysis,* John Wiley & Sons, New York.

SPARKS, R.S. (1984). Selection of variables in multivariate and generalised multivariate regression models. Unpublished Ph.D. thesis, University of Natal.

SUNDBERG, R. (1974). Maximum likelihood theory for incomplete data from an exponential family. *Scand. Journal of Statistics,* **1,** 49-58.

THOMPSON, M.L. (1978a). Selection of variables in multiple regression: Part I. A review and evaluation. *International Stat. Review,* **46,** 1-19.

THOMPSON, M.L. (1978b). Selection of variables in multiple regression: Part II. Chosen procedures, computations and examples. *International Stat. Review,* **46,** 129-146.

WEISBERG, S. (1985). *Applied Linear Regression,* John Wiley & Sons, New York.

WELDING, M.C. and HAVENGA, C.M. (1974). The statistical classification of rainfall stations in the Republic of South Africa. *Agrochemophysica,* **6,** 5-24.

WU, C.F.J. (1983). On the convergence properties of the EM algorithm. *The Annals of Statistics,* **11,** 95-103.

ZUCCHINI, W. and ADAMSON, P.T. (1984). The occurrence and severity of droughts in South Africa. *WRC Report No. 91/1/84,* Water Research Commission, Pretoria.

ZUCCHINI, W. and HIEMSTRA, L.A.V. (1984). Augmenting hydrological records. *WRC Report No. 91/3/84,* Water Research Commission, Pretoria.

ZUCCHINI, W. and SPARKS, R.S. (1984). Estimating the missing values in rainfall records. *WRC Report No. 91/3/84*, Water Research Commission, Pretoria.

APPENDIX A

EXAMPLE: THE EM ALGORITHM

In this appendix we give a worked example of the application of the EM algorithms described in section 3.8. The purpose is to illustrate the methodology step by step, and to serve as a check for software which users might prepare to implement the methods.

Suppose a data set Z, consists of $n = 10$ observations on $k + 1 = 4$ stations as given below:

Case No.	Z_1	Z_2	Z_3	Z_4
1	103	80	96	120
2	101	83	86	108
3	**-999**	**-999**	80	98
4	61	94	75	65
5	92	121	104	104
6	80	83	86	74
7	119	91	104	**-999**
8	91	70	77	102
9	116	115	97	116
10	126	87	94	97

where $-\mathbf{999}$ represents a missing value.

Two methods of estimation using the EM algorithm will be used for the estimation of the missing values from the above data matrix. In the first method **(Method 1)**, the regression model is fitted by conditioning on real records only, whilst in the second method **(Method 2)**, all the records, that is, real and estimated records, are utilized.

Note that all the regression models are fitted with intercept terms.

Let

$$y = \beta_0 + X\beta + e$$

where

y is a (10×1) vector of the *target station*,

β_0 is the intercept term,

X is a (10×3) matrix of *control stations*

β is a (3×1) vector of parameter estimates

e is a (10×1) vector of residuals.

Assume that the data is *Multivariate Normally* distributed, in which case the computational formulae required to implement the algorithm are:

EXPECTATION:

$$\mathrm{E}(y|x^t, \hat{\beta}_0, \hat{\beta}, \sigma_{yy.x}) = \hat{\beta}_0 + x^t\hat{\beta}$$

where

$$\begin{aligned}
\hat{\beta}_0 &= \hat{\mu}_y - \hat{\mu}_x\beta \\
\beta &= \Sigma_{xx}^{-1}\sigma_{yx} \\
&= (X^tX)^{-1}X^ty \\
\sigma_{yy.x} &= \sigma_{yy} - \sigma_{yx}^t\Sigma_{xx}^{-1}\sigma_{yx}
\end{aligned}$$

MAXIMUM LIKELIHOOD:

$$\begin{aligned}
\hat{\mu}_x^t &= \frac{1}{n}1_n^tX \\
\hat{\mu}_y &= \frac{1}{n}1_n^ty \\
\hat{\Sigma}_{xx} &= \frac{1}{n}X^t[I_n - 1_n1_n^t/n]X \\
\hat{\sigma}_{yy.x} &= \frac{1}{n}y^t[I_n - 1_n1_n^t/n]y
\end{aligned}$$

where $1_n^t = (1 \quad 1 \quad 1 \quad \ldots \quad 1)$ is a (1×10) vector of ones.

A1. *Method 1*: CONDITION ON REAL RECORDS

CRITERION USED:

$$\text{Crit} = \left[\frac{y_\ell^{(b)} - y_\ell^{(b-1)}}{y_\ell^{(b)}}\right]^2$$

where b represents the current cycle

$b-1$ represents the previous cycle

ℓ represents the current missing value.

If Crit $< 10^{-4}$, then the estimate of y_ℓ is obtained.

CYCLE 0

Step 1(1)

We begin by estimating all the missing values from the first column, Z_1 of Z. Partition Z into a vector of the *target station* $y_{(1)}$ which contains the first column of Z, and a matrix of *control stations* $X_{(1)}$ which contains the 2^{nd}, 3^{rd} and the 4^{th} columns of Z. That is, let

$$y_{(1)} = Z_1 \text{ and } X_{(1)} = (Z_2, Z_3, Z_4)$$

where the subscript (1) represents that missing values from the first column of Z are being estimated.

To estimate y_3 we have to eliminate the first column from $X_{(1)}$ because x_{31} is missing. The number of columns remaining is $p = 2$ and therefore $X_{2(1)}$ is a (10×2) dimensional matrix. $X_{2(1)}$ contains the 2^{nd} and the 3^{rd} columns (*control stations*) of X.

Step 2(1)

Eliminate from both $y_{(1)}$ and $X_{2(1)}$ the 3^{rd} row and the 7^{th} row because y_3 and x_{72} are missing. The number of remaining rows is

$$n^* = n - \text{ number of eliminated rows} = 8$$

and therefore the (8×2) matrix

$$X^*_{2(1)} = \begin{pmatrix} 96 & 120 \\ 86 & 108 \\ 75 & 65 \\ 104 & 104 \\ 86 & 74 \\ 77 & 102 \\ 97 & 116 \\ 94 & 97 \end{pmatrix} \qquad \text{and} \qquad y^*_{(1)} = \begin{pmatrix} 103 \\ 101 \\ 61 \\ 92 \\ 80 \\ 91 \\ 116 \\ 126 \end{pmatrix}$$

Let

$$\tilde{X}^*_{2(1)} = X^*_{2(1)} - \bar{X}^*_{(2(1)}, \qquad \text{where} \qquad \bar{X}^*_{2(1)} = \frac{1}{8}\sum_{i=1}^{8} x^*_{ij}, \quad j = 1,2$$

and $\tilde{y}^*_{(1)} = y^*_{(1)} - \bar{y}^*_{(1)}$, where $\bar{y}^*_{(1)} = \frac{1}{8}\sum_{i=1}^{8} y^*_i$

represent the standardized $X^*_{2(1)}$ matrix and $y^*_{(1)}$ vector respectively:

$$\tilde{X}^*_{2(1)} = \begin{pmatrix} 6.625 & 21.75 \\ -3.375 & 9.75 \\ -14.375 & -33.25 \\ 14.625 & 5.75 \\ -3.375 & -24.25 \\ -12.375 & 3.75 \\ 7.625 & 17.75 \\ 4.625 & -1.25 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}^*_{(1)} = \begin{pmatrix} 6.75 \\ 4.75 \\ -35.25 \\ -4.25 \\ -16.25 \\ -5.25 \\ 19.75 \\ 29.75 \end{pmatrix}$$

Step 3(1)

Calculate the least squares estimates:

$$\hat{\beta}^{(0)} = (\tilde{X}^{*t}_{2(1)}\tilde{X}^*_{2(1)})^{-1}\tilde{X}^{*t}_{2(1)}\tilde{y}^*_{(1)}$$

$$= \begin{pmatrix} 0.516547 \\ 0.607695 \end{pmatrix}$$

$$\hat{\mu}_y = 96.25$$

$$\hat{\mu}_x = (89.375 \quad 98.25)$$

$$\hat{\beta}^{(0)}_0 = 96.25 - (89.375 \quad 98.25)\begin{pmatrix} 0.516547 \\ 0.607695 \end{pmatrix}$$

$$= -9.62247$$

Use the fitted regression model to estimate y_3 :

$$y^{(0)}_{3(1)} = \hat{\beta}^{(0)}_0 + \sum_{j=1}^{2} x_{3j}\hat{\beta}^{(0)}_j$$

$$= -9.62247 + 80(0.516547) + 98(0.607695)$$

$$= 91.2554$$

Create a new "data" column vector $Z_1^{(0)}$ similar to Z_1 except that it contains the estimate $y_{3(1)}^{(0)}$ in place of the missing value.

Step 1(2)

Consider estimating missing values from Z_2. Let the vector of the *target station*: $y_{(2)} = Z_2$, and the matrix of *control stations*: $X_{(2)} = (Z_1, Z_3, Z_4)$, where the subscript (2) represents that the missing values from the 2^{nd} column of Z are being estimated.

To estimate y_3 we must eliminate the first column from $X_{(2)}$ because x_{31} is missing. Therefore $p = k - 1 = 2$ and X_2 is a (10×2) dimensional matrix of *control stations*.

Step 2(2)

Eliminate from both $y_{(2)}$ and $X_{2(2)}$ the 3^{rd} row and the 7^{th} row because y_3 and x_{72} are missing. The number of the remaining rows is $n^* = 8$ and therefore the (8×2) matrix

$$X^*_{2(2)} = \begin{pmatrix} 96 & 120 \\ 86 & 108 \\ 75 & 65 \\ 104 & 104 \\ 86 & 74 \\ 77 & 102 \\ 97 & 116 \\ 94 & 97 \end{pmatrix} \quad \text{and} \quad y^*_{(2)} = \begin{pmatrix} 80 \\ 83 \\ 94 \\ 121 \\ 83 \\ 70 \\ 115 \\ 87 \end{pmatrix}$$

Let

$$\tilde{X}^*_{2(2)} = X^*_{2(2)} - \bar{X}^*_{(2(2)}, \qquad \text{where} \qquad \bar{X}^*_{2(2)} = \frac{1}{8}\sum_{i=1}^{8} x^*_{ij}, \quad j = 1, 2$$

and $\tilde{y}^*_{(1)} = y^*_{(2)} - \bar{y}^*_{(2)}$, where $\bar{y}^*_{(2)} = \frac{1}{8}\sum_{i=1}^{8} y^*_i$

represent the standardized $X^*_{2(2)}$ matrix and $y^*_{(2)}$ vector respectively:

$$\tilde{X}^*_{2(2)} = \begin{pmatrix} 6.625 & 21.75 \\ -3.375 & 9.75 \\ -14.375 & -33.25 \\ 14.625 & 5.75 \\ -3.375 & -24.25 \\ -12.375 & 3.75 \\ 7.625 & 17.75 \\ 4.625 & -1.25 \end{pmatrix} \quad \text{and} \quad \tilde{y}^*_{(2)} = \begin{pmatrix} -11.625 \\ -8.625 \\ 2.375 \\ 29.375 \\ -8.625 \\ -21.625 \\ 23.375 \\ -4.625 \end{pmatrix}$$

Step 3(2)

Calculate the least squares estimates:

$$\begin{aligned}
\hat{\beta}^{(0)} &= (\tilde{X}^{*t}_{2(2)}\tilde{X}^{*}_{2(2)})^{-1}\tilde{X}^{*t}_{2(2)}\tilde{y}^{*}_{(2)} \\
&= \begin{pmatrix} 1.55841 \\ -0.382628 \end{pmatrix} \\
\hat{\mu}_y &= 91.625 \\
\hat{\mu}_x &= (\,89.375 \quad 98.25\,) \\
\hat{\beta}^{(0)}_0 &= 91.625 - (\,89.375 \quad 98.25\,)\begin{pmatrix} 1.55841 \\ -0.382628 \end{pmatrix} \\
&= -10.065
\end{aligned}$$

Use the fitted regression model to estimate y_3 :

$$\begin{aligned}
y^{(0)}_{3(2)} &= \hat{\beta}^{(0)}_0 + \sum_{j=1}^{2} x_{3j}\hat{\beta}^{(0)}_j \\
&= -10.06505 + 80(1.558414) + 98(-0.382628) \\
&= 77.1105
\end{aligned}$$

Create a new "data" column vector $Z^{(0)}_2$ similar to Z_2 except that it contains the estimate $y^{(0)}_{3(2)}$ in place of the missing value.

Step 1(3)

Consider estimating missing values from Z_3. Because vector Z_3 does not contain any missing values, then let $Z^{(0)}_3 = Z_3$.

Step 1(4)

Consider estimating missing values from Z_4.

Vector of the *target station*: $y_{(4)} = Z_4$, and the matrix of control stations: $X_{(4)} = (Z_1, Z_2, Z_3)$,

where the subscript (4) represents that the missing values from the 4^{th} column of Z are being estimated.

To estimate y_7 all the values in the *control stations* corresponding to y_7 (that is, values in the 7^{th} row), are available, therefore none of the control stations are eliminated and $p = k = 3$ and $X_{3(4)} = X_{(4)}$ is a (10×3) dimensional matrix.

Step 2(4)

Eliminate from both $y_{(4)}$ and $X_{3(4)}$ the 3^{rd} row and the 7^{th} row because x_{31}, x_{32} and y_7 are missing. The number of the remaining rows is $n^* = 8$ and therefore the (8×3) matrix

$$X^*_{3(4)} = \begin{pmatrix} 103 & 80 & 96 \\ 101 & 83 & 86 \\ 61 & 94 & 75 \\ 92 & 121 & 104 \\ 80 & 83 & 86 \\ 91 & 70 & 77 \\ 116 & 115 & 97 \\ 126 & 87 & 94 \end{pmatrix} \quad \text{and} \quad y_{(4)} = \begin{pmatrix} 120 \\ 108 \\ 65 \\ 104 \\ 74 \\ 102 \\ 116 \\ 97 \end{pmatrix}$$

Let

$$\tilde{X}^*_{3(4)} = X^*_{3(4)} - \bar{X}^*_{(3(4)}, \qquad \text{where} \qquad \bar{X}^*_{3(4)} = \frac{1}{8}\sum_{i=1}^{8} x^*_{ij}, \quad j = 1,2,3$$

and $\tilde{y}^*_{(4)} = y^*_{(4)} - \bar{y}^*_{(4)}$, where $\bar{y}^*_{(4)} = \frac{1}{8}\sum_{i=1}^{8} y^*_i$

represent the standardized $X^*_{3(4)}$ matrix and $y^*_{(4)}$ vector respectively:

$$\tilde{X}^*_{3(4)} = \begin{pmatrix} 6.75 & -11.625 & 6.625 \\ 4.75 & -8.625 & -3.375 \\ -35.25 & 2.375 & -14.375 \\ -4.25 & 29.375 & 14.625 \\ -16.25 & -8.625 & -3.375 \\ -5.25 & -21.625 & -12.375 \\ 19.75 & 23.375 & 7.625 \\ 29.75 & -4.625 & 4.625 \end{pmatrix} \quad \text{and} \quad \tilde{y}^*_{(4)} = \begin{pmatrix} 21.75 \\ 9.75 \\ -33.25 \\ 5.75 \\ -24.25 \\ 3.75 \\ 17.75 \\ -1.25 \end{pmatrix}$$

Step 3(4)

Calculate the least squares estimates:

$$\hat{\beta}^{(0)} = (\tilde{X}^{*t}_{3(4)}\tilde{X}^*_{3(4)})^{-1}\tilde{X}^{*t}_{3(4)}\tilde{y}^*_{(4)}$$

$$= \begin{pmatrix} 0.460277 \\ -0.227307 \\ 0.853942 \end{pmatrix}$$

$$\hat{\mu}_y = 98.25$$

$$\hat{\mu}_x = (96.25 \quad 91.625 \quad 89.375)$$

$$\hat{\beta}^{(0)}_0 = 98.25 - (96.25 \quad 91.625 \quad 89.375)\begin{pmatrix} 0.460277 \\ -0.227307 \\ 0.853942 \end{pmatrix}$$

$$= -1.5458$$

Use the fitted regression model to estimate y_7 :

$$y_{7(4)}^{(0)} = \hat{\beta}_0^{(0)} + \sum_{j=1}^{3} x_{7j}\hat{\beta}_j^{(0)}$$
$$= -1.5458 + 119(0.460277) + 91(-0.227307) + 104(0.853942)$$
$$= 121.352$$

Create a new "data" column vector $Z_4^{(0)}$ similar to Z_4 except that it contains the estimate of $y_{7(4)}^{(0)}$ in place of the missing value.

CYCLE 1

The new "data" matrix $Z^{(0)}$ contains new estimates in place of the missing values:

$$Z^{(0)} = \begin{pmatrix} 103 & 80 & 96 & 120 \\ 101 & 83 & 86 & 108 \\ \mathbf{91.255} & \mathbf{77.111} & 80 & 98 \\ 61 & 94 & 75 & 65 \\ 92 & 121 & 104 & 104 \\ 80 & 83 & 86 & 74 \\ 119 & 91 & 104 & \mathbf{121.352} \\ 91 & 70 & 77 & 102 \\ 116 & 115 & 97 & 116 \\ 126 & 87 & 94 & 97 \end{pmatrix}$$

Step 1(1)

Let

$$y_{(1)}^{(0)} = Z_1^{(0)} \qquad \text{and} \qquad X_{(1)}^{(0)} = (Z_2^{(0)}, Z_3^{(0)}, Z_4^{(0)}).$$

Required to re-estimate: y_3

Eliminate the first column from the new *control stations* matrix $X_{(1)}^{(0)}$ because x_{31} was originally missing. Therefore $p = 2$ and $X_{2(1)}^{(0)}$ is a (10×2) dimensional matrix.

Let

$$\tilde{X}_{2(1)}^{(0)} = X_{2(1)}^{(0)} - \bar{X}_{2(1)}^{(0)}, \qquad \text{where} \qquad \bar{X}_{2(1)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(0)}, \qquad j = 1, 2$$

and $\tilde{y}_{(1)}^{(0)} = y_{(1)}^{(0)} - \bar{y}_{(1)}^{(0)}$, where $\bar{y}_{(1)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(0)}$

represent the standardized $X_{2(1)}^{(0)}$ matrix and $y_{(1)}^{(0)}$ vector respectively:

$$\tilde{X}^{(0)}_{2(1)} = \begin{pmatrix} 6.1 & 19.4648 \\ -3.9 & 7.4648 \\ -9.9 & -2.5352 \\ -14.9 & -35.5352 \\ 14.1 & 3.4648 \\ -3.9 & -26.5352 \\ 14.1 & 20.8168 \\ -12.9 & 1.4648 \\ 7.1 & 15.4648 \\ 4.1 & -3.5352 \end{pmatrix} \quad \text{and} \quad \tilde{y}^{(0)}_{(1)} = \begin{pmatrix} 4.9745 \\ 2.9745 \\ -6.7705 \\ -37.0255 \\ -6.0255 \\ -18.0255 \\ 20.9745 \\ -7.0255 \\ 17.9745 \\ 27.9745 \end{pmatrix}$$

Step 2(1)

Calculate the least squares estimates:

$$\begin{aligned} \hat{\beta}^{(1)} &= (\tilde{X}^{(0)t}_{2(1)} \tilde{X}^{(0)}_{2(1)})^{-1} \tilde{X}^{(0)t}_{2(1)} \tilde{y}^{(0)}_{(1)} \\ &= \begin{pmatrix} 0.529065 \\ 0.610745 \end{pmatrix} \\ \hat{\mu}_y &= 98.0255 \\ \hat{\mu}_x &= (89.9 \quad 100.535) \\ \hat{\beta}^{(1)}_0 &= 98.0255 - (89.9 \quad 100.535) \begin{pmatrix} 0.529065 \\ 0.610745 \end{pmatrix} \\ &= -10.9388 \end{aligned}$$

Use the fitted regression model to re-estimate y_3 :

$$\begin{aligned} y^{(1)}_{3(1)} &= \hat{\beta}^{(1)}_0 + \sum_{j=1}^{2} x_{3j} \hat{\beta}^{(1)}_j \\ &= -10.9388 + 80(0.529065) + 98(0.610745) \\ &= 91.239 \end{aligned}$$

Create a new "data" column vector $Z^{(1)}_1$ which contains the re-estimated value $y^{(1)}_{3(1)}$ in place of the originally missing value.

Step 3(1)

Check for convergence:

$$\begin{aligned} \text{Crit}_{3(1)} &= \left[\frac{91.239 - 91.255}{91.239} \right]^2 \\ &= 10^{-7}(0.3) \\ &< 10^{-4} \end{aligned}$$

Since $\text{Crit}_{3(1)}$ is very close to zero, therefore the required estimate of $y_{3(1)}$ is obtained, and this value will not be re-estimated again.

Step 1(2)

Let

$$y_{(2)}^{(0)} = Z_2^{(0)} \qquad \text{and} \qquad X_{(2)}^{(0)} = (Z_1^{(0)}, Z_3^{(0)}, Z_4^{(0)}).$$

Required to re-estimate: y_3

Eliminate the first column from the new *control stations* matrix $X_{(2)}^{(0)}$ because x_{31} was originally missing. Therefore $p = 2$ and $X_{2(2)}^{(0)}$ is a (10×2) dimensional matrix.

Let

$$\tilde{X}_{2(2)}^{(0)} = X_{2(2)}^{(0)} - \bar{X}_{2(2)}^{(0)}, \qquad \text{where} \qquad \bar{X}_{2(2)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(0)}, \qquad j = 1, 2$$

and $\quad \tilde{y}_{(2)}^{(0)} = y_{(2)}^{(0)} - \bar{y}_{(2)}^{(0)}, \qquad \text{where} \qquad \bar{y}_{(2)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(0)}$

represent the standardized $X_{2(2)}^{(0)}$ matrix and $y_{(2)}^{(0)}$ vector respectively:

$$\tilde{X}_{2(2)}^{(0)} = \begin{pmatrix} 6.1 & 19.4648 \\ -3.9 & 7.4648 \\ -9.9 & -2.5352 \\ -14.9 & -35.5352 \\ 14.1 & 3.4648 \\ -3.9 & -26.5352 \\ 14.1 & 20.8168 \\ -12.9 & 1.4648 \\ 7.1 & 15.4648 \\ 4.1 & -3.5352 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(2)}^{(0)} = \begin{pmatrix} -10.1111 \\ -7.1111 \\ -13.0001 \\ 3.8889 \\ 30.8889 \\ -7.1111 \\ 0.8889 \\ -20.1111 \\ 24.8889 \\ -3.1111 \end{pmatrix}$$

Step 2(2) Calculate the least squares estimates:

$$\begin{aligned} \hat{\beta}^{(1)} &= (\tilde{X}_{2(2)}^{(0)t}\tilde{X}_{2(2)}^{(0)})^{-1}\tilde{X}_{2(2)}^{(0)t}\tilde{y}_{(2)}^{(0)} \\ &= \begin{pmatrix} 1.40071 \\ -0.42109 \end{pmatrix} \\ \hat{\mu}_y &= 90.1111 \\ \hat{\mu}_x &= (89.9 \quad 100.535) \\ \hat{\beta}_0^{(1)} &= 90.1111 - (89.9 \quad 100.535)\begin{pmatrix} 1.40071 \\ -0.42109 \end{pmatrix} \\ &= 6.52131 \end{aligned}$$

Use the fitted regression model to re-estimate y_3 :

$$\begin{aligned} y_{3(2)}^{(1)} &= \hat{\beta}_0^{(1)} + \sum_{j=1}^{2} x_{3j}\hat{\beta}_j^{(1)} \\ &= 6.52131 + 80(1.40071) + 98(-0.42109) \\ &= 77.312 \end{aligned}$$

Create a new "data" column vector $Z_2^{(1)}$ which contains the re-estimated value $y_{3(2)}^{(1)}$ in place of the originally missing value.

Step 3(2)

Check for convergence:

$$\begin{aligned} \text{Crit}_{3(2)} &= \left[\frac{77.312 - 77.111}{77.312}\right]^2 \\ &= 10^{-5}(0.6759) \\ &< 10^{-4} \end{aligned}$$

Since $\text{Crit}_{3(2)}$ is close to zero, therefore the required estimate of $y_{3(2)}$ is obtained, and this value will not be re-estimated again.

Step 1(3)

Let $Z_3^{(1)} = Z_3$.

Step 1(4)

Let

$$y_{(4)}^{(0)} = Z_4^{(0)} \text{ and } X_{(4)}^{(0)} = (Z_1^{(0)}, Z_2^{(0)}, Z_3^{(0)}).$$

Required to re-estimate: y_7

Let $X_{3(4)}^{(0)} = X_{(4)}^{(0)}$ and let

$$\tilde{X}_{3(4)}^{(0)} = X_{3(4)}^{(0)} - \bar{X}_{3(4)}^{(0)}, \qquad \text{where} \qquad \bar{X}_{3(4)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(0)}, \qquad j = 1,2,3$$

and $\tilde{y}_{(4)}^{(0)} = y_{(4)}^{(0)} - \bar{y}_{(4)}^{(0)}$, where $\bar{y}_{(4)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(0)}$

represent the standardized $X_{3(4)}^{(0)}$ matrix and $y_{(4)}^{(0)}$ vector respectively:

$$\tilde{X}_{3(4)}^{(0)} = \begin{pmatrix} 4.9745 & -10.1111 & 6.1 \\ 2.9745 & -7.1111 & -3.9 \\ -6.7705 & -13.0001 & -9.9 \\ -37.0255 & 3.8889 & -14.9 \\ -6.0255 & 30.8889 & 14.1 \\ -18.0255 & -7.1111 & -3.9 \\ 20.9745 & 0.8889 & 14.1 \\ -7.0255 & -20.1111 & -12.9 \\ 17.9745 & 24.8889 & 7.1 \\ 27.9745 & -3.111 & 4.1 \end{pmatrix} \quad \text{and} \quad \tilde{y}_{(4)}^{(0)} = \begin{pmatrix} 19.4648 \\ 7.4648 \\ -2.5352 \\ -35.5352 \\ 3.4648 \\ -26.5352 \\ 20.8168 \\ 1.4648 \\ 15.4648 \\ -3.5352 \end{pmatrix}$$

Step 2(4)

Calculate the least squares estimates:

$$\begin{aligned} \hat{\beta}^{(1)} &= (\tilde{X}_{3(4)}^{(0)t}\tilde{X}_{3(4)}^{(0)})^{-1}\tilde{X}_{3(4)}^{(0)t}\tilde{y}_{(4)}^{(0)} \\ &= \begin{pmatrix} 0.475807 \\ -0.236362 \\ 0.777542 \end{pmatrix} \\ \hat{\mu}_y &= 100.535 \\ \hat{\mu}_x &= (98.0255 \quad 90.1111 \quad 89.9) \\ \hat{\beta}_0^{(1)} &= 100.535 - (98.0255 \quad 90.1111 \quad 89.9)\begin{pmatrix} 0.475807 \\ -0.236362 \\ 0.777542 \end{pmatrix} \\ &= 5.29179 \end{aligned}$$

Use the fitted regression model to re-estimate y_7 :

$$\begin{aligned} y_{7(4)}^{(1)} &= \hat{\beta}_0^{(1)} + \sum_{j=1}^{3} x_{7j}\hat{\beta}_j^{(1)} \\ &= 5.29179 + 119(0.475807) + 91(-0.236362) + 104(0.777542) \\ &= 121.268 \end{aligned}$$

Creat a new "data" column vector $Z_4^{(1)}$ which contains the re-estimated value of $y_{7(4)}^{(1)}$ in place of the originally missing value.

Step 3(4)

Check for convergence:

$$\begin{aligned} \text{Crit}_{7(4)} &= \left[\frac{121.268 - 121.352}{121.268}\right]^2 \\ &= 10^{-6}(0.479) \\ &< 10^{-4} \end{aligned}$$

Since $\text{Crit}_{7(4)}$ is very close to zero, the required estimate of $y_{7(4)}$ is obtained. This value will not be re-estimated again.

RESULTS

All the missing values are now estimated. Let $Z = Z^{(1)}$. Note that these estimates have been obtained after only *two* cycles or iterations.

THE MATRIX WITH COMPLETE DATA:

$$Z = \begin{pmatrix} 103 & 80 & 96 & 120 \\ 101 & 83 & 86 & 108 \\ \mathbf{91.239} & \mathbf{77.312} & 80 & 98 \\ 61 & 94 & 75 & 65 \\ 92 & 121 & 104 & 104 \\ 80 & 83 & 86 & 74 \\ 119 & 91 & 104 & \mathbf{121.268} \\ 91 & 70 & 77 & 102 \\ 116 & 115 & 97 & 116 \\ 126 & 87 & 94 & 97 \end{pmatrix}$$

A2. *Method 2*: CONDITION ON REAL AND ESTIMATED RECORDS

CRITERION USED:

$$\text{Crit} = \sum_{j=1}^{k} \sum_{\ell=1}^{n-n^*_{(j)}} \left| \frac{y^{(b)}_{(j)\ell} - y^{(b-1)}_{(j)\ell}}{y^{(b)}_{(j)\ell}} \right|$$

where $n^*_{(j)}$ is the number of observed values in the current station.

CYCLE 0

Step 1(1)

Consider estimating all the missing values from the first column, Z_1 of Z. Partition Z into a vector of the *target station* $y_{(1)}$ which contains the first column of Z, and a matrix of *control stations* $X_{(1)}$ which contains the 2^{nd}, 3^{rd} and the 4^{th} columns of Z. That is, let

$$y_{(1)} = Z_1 \text{ and } X_{(1)} = (Z_2, Z_3, Z_4)$$

where the subscript (1) represents that missing values from the first column of Z are being estimated.

To estimate y_3 we first eliminate from *both* $y_{(1)}$ and $X_{(1)}$ the 3^{rd} row and the 7^{th} row because y_3, x_{31} and x_{72} are missing. The number of remaining rows is

$$n^* = n - \text{ number of eliminated rows} = 8$$

and therefore the (8×3) matrix

$$X^*_{(1)} = \begin{pmatrix} 80 & 96 & 120 \\ 83 & 86 & 108 \\ 94 & 75 & 65 \\ 121 & 104 & 104 \\ 83 & 86 & 74 \\ 70 & 77 & 102 \\ 115 & 97 & 116 \\ 87 & 94 & 97 \end{pmatrix} \quad \text{and} \quad y^*_{(1)} = \begin{pmatrix} 103 \\ 101 \\ 61 \\ 92 \\ 80 \\ 91 \\ 116 \\ 126 \end{pmatrix}$$

Let

$$\tilde{X}^*_{(1)} = X^*_{(1)} - \bar{X}^*_{(1)}, \qquad \text{where} \qquad \bar{X}^*_{(1)} = \frac{1}{8}\sum_{i=1}^{8} x^*_{ij}, \quad j = 1,2,3$$

and $$\tilde{y}^*_{(1)} = y^*_{(1)} - \bar{y}^*_{(1)}, \qquad \text{where} \qquad \bar{y}^*_{(1)} = \frac{1}{8}\sum_{i=1}^{8} y^*_i$$

represent the standardized $X^*_{(1)}$ matrix and $y^*_{(1)}$ vector respectively:

$$\tilde{X}^*_{(1)} = \begin{pmatrix} -11.625 & 6.625 & 21.75 \\ -8.625 & -3.375 & 9.75 \\ 2.375 & -14.375 & -33.25 \\ 29.375 & 14.625 & 5.75 \\ -8.625 & -3.375 & -24.25 \\ -21.625 & -12.375 & 3.75 \\ 23.375 & 7.625 & 17.75 \\ -4.625 & 4.625 & -1.25 \end{pmatrix} \quad \text{and} \quad \tilde{y}^*_{(1)} = \begin{pmatrix} 6.75 \\ 4.75 \\ -35.25 \\ -4.25 \\ -16.25 \\ -5.25 \\ 19.75 \\ 29.75 \end{pmatrix}$$

Step 2(1)

Calculate the least squares estimates; that is:

$$\hat{\beta}^{(0)} = (\tilde{X}^{*t}_{(1)}\tilde{X}^*_{(1)})^{-1}\tilde{X}^{*t}_{(1)}\tilde{y}^*_{(1)}$$

$$= \begin{pmatrix} -0.329375 \\ 1.02985 \\ 0.481667 \end{pmatrix}$$

$$\hat{\mu}_y = 96.25$$

$$\hat{\mu}_x = (91.625 \quad 89.375 \quad 98.25)$$

$$\hat{\beta}^{(0)}_0 = 96.25 - (91.625 \quad 89.375 \quad 98.25)\begin{pmatrix} -0.329375 \\ 1.02985 \\ 0.481667 \end{pmatrix}$$

$$= -12.9376$$

Use the fitted regression model to estimate y_3 :

$$y^{(0)}_{3(1)} = \hat{\beta}^{(0)}_0 + \sum_{j=1}^{3} u_j x_{3j}\hat{\beta}^{(0)}_j$$

where $u_j = 1$ if x_{3j} is observed

$= 0$ if x_{3j} is missing

$$= -12.9376 + 0(-0.329375) + 80(1.02985) + 98(0.481667)$$

$$= 116.654$$

Create a new "data" column vector $Z^{(0)}_1$ similar to Z_1 except that it contains the estimate $y^{(0)}_{3(1)}$ in place of the missing value.

Step 1(2)

Consider estimating missing values from Z_2. Let vector of the *target station*: $y_{(2)} = Z_2$, and the matrix of *control stations*: $X_{(2)} = (Z_1, Z_3, Z_4)$,

where the subscript (2) represents that the missing values from the second column of Z are being estimated.

To estimate y_3 eliminate from *both* $y_{(2)}$ and $X_{(2)}$ the 3^{rd} row and the 7^{th} row because y_3 , x_{31} and x_{72} are missing. The number of the remaining rows is $n^* = 8$ and therefore the (8×2) matrix

$$X^*_{(2)} = \begin{pmatrix} 103 & 96 & 120 \\ 101 & 86 & 108 \\ 61 & 75 & 65 \\ 92 & 104 & 104 \\ 80 & 86 & 74 \\ 91 & 77 & 102 \\ 116 & 97 & 116 \\ 126 & 94 & 97 \end{pmatrix} \quad \text{and} \quad y^*_{(2)} = \begin{pmatrix} 80 \\ 83 \\ 94 \\ 121 \\ 83 \\ 70 \\ 115 \\ 87 \end{pmatrix}$$

Let

$$\tilde{X}^*_{(2)} = X^*_{(2)} - \bar{X}^*_{(2)}, \qquad \text{where} \qquad \bar{X}^*_{(2)} = \frac{1}{8}\sum_{i=1}^{8} x^*_{ij}, \quad j = 1,2,3$$

and $$\tilde{y}^*_{(2)} = y^*_{(2)} - \bar{y}^*_{(2)}, \qquad \text{where} \qquad \bar{y}^*_{(2)} = \frac{1}{8}\sum_{i=1}^{8} y^*_i$$

represent the standardized $X^*_{(2)}$ matrix and $y^*_{(2)}$ vector respectively:

$$\tilde{X}^*_{(2)} = \begin{pmatrix} 6.75 & 6.625 & 21.75 \\ 4.75 & -3.375 & 9.75 \\ -35.25 & -14.375 & -33.25 \\ -4.25 & 14.625 & 5.75 \\ -16.25 & -3.375 & -24.25 \\ -5.25 & -12.375 & 3.75 \\ 19.75 & 7.625 & 17.75 \\ 29.75 & 4.625 & -1.25 \end{pmatrix} \quad \text{and} \quad \tilde{y}^*_{(2)} = \begin{pmatrix} -11.625 \\ -8.625 \\ 2.375 \\ 29.375 \\ -8.625 \\ -21.625 \\ 23.375 \\ -4.625 \end{pmatrix}$$

Step 2(2)

Calculate the least squares estimates:

$$\hat{\beta}^{(0)} = (\tilde{X}^{*t}_{(2)}\tilde{X}^{*}_{(2)})^{-1}\tilde{X}^{*t}_{(2)}\tilde{y}^{*}_{(2)}$$

$$= \begin{pmatrix} -0.287716 \\ 1.70703 \\ -0.207784 \end{pmatrix}$$

$$\hat{\mu}_y = 91.625$$

$$\hat{\mu}_x = (96.25 \quad 89.375 \quad 98.25)$$

$$\hat{\beta}^{(0)}_0 = 91.625 - (96.25 \quad 89.375 \quad 98.25)\begin{pmatrix} -0.287716 \\ 1.70703 \\ -0.207784 \end{pmatrix}$$

$$= -12.8335$$

Use the fitted regression model to estimate y_3 :

$$y^{(0)}_{3(2)} = \hat{\beta}^{(0)}_0 + \sum_{j=1}^{3} u_j x_{3j}\hat{\beta}^{(0)}_j \qquad \text{where } u_j = 1 \text{ if } x_{3j} \text{ is observed}$$

$$= 0 \text{ if } x_{3j} \text{ is missing}$$

$$= -12.8335 + 0(-0.287716) + 80(1.70703) + 98(-0.207784)$$

$$= 103.366$$

Create a new "data" column vector $Z^{(0)}_2$ similar to Z_2 except that it contains the estimate $y^{(0)}_{3(2)}$ in place of the missing value.

Step 1(3)

Consider estimating missing values from Z_3. Because vector Z_3 does not contain any missing values, then let $Z^{(0)}_3 = Z_3$.

Step 1(4)

Consider estimating missing values from Z_4. Let the vector of the target station: $y_{(4)} = Z_4$, and the matrix of control stations: $X_{(4)} = (Z_1, Z_2, Z_3)$, where the subscript (4) represents that the missing values from the 4^{th} column of Z are being estimated.

To estimate y_7 eliminate from *both* $y_{(4)}$ and $X_{(4)}$ the 3^{rd} row and the 7^{th} row because x_{31}, x_{32} and y_7 are missing. The number of the remaining rows is

$n^* = 8$ and therefore the (8×3) matrix

$$X^*_{(4)} = \begin{pmatrix} 103 & 80 & 96 \\ 101 & 83 & 86 \\ 61 & 94 & 75 \\ 92 & 121 & 104 \\ 80 & 83 & 86 \\ 91 & 70 & 77 \\ 116 & 115 & 97 \\ 126 & 87 & 94 \end{pmatrix} \quad \text{and} \quad y_{(4)} = \begin{pmatrix} 120 \\ 108 \\ 65 \\ 104 \\ 74 \\ 102 \\ 116 \\ 97 \end{pmatrix}$$

Let

$$\tilde{X}^*_{(4)} = X^*_{(4)} - \bar{X}^*_{(4)}, \qquad \text{where} \qquad \bar{X}^*_{(4)} = \frac{1}{8}\sum_{i=1}^{8} x^*_{ij}, \quad j = 1,2,3$$

$$\text{and} \qquad \tilde{y}^*_{(4)} = y^*_{(4)} - \bar{y}^*_{(4)}, \qquad \text{where} \qquad \bar{y}^*_{(4)} = \frac{1}{8}\sum_{i=1}^{8} y^*_i$$

represent the standardized $X^*_{(4)}$ matrix and $y^*_{(4)}$ vector respectively:

$$\tilde{X}^*_{(4)} = \begin{pmatrix} 6.75 & -11.625 & 6.625 \\ 4.75 & -8.625 & -3.375 \\ -35.25 & 2.375 & -14.375 \\ -4.25 & 29.375 & 14.625 \\ -16.25 & -8.625 & -3.375 \\ -5.25 & -21.625 & -12.375 \\ 19.75 & 23.375 & 7.625 \\ 29.75 & -4.625 & 4.625 \end{pmatrix} \quad \text{and} \quad \tilde{y}^*_{(4)} = \begin{pmatrix} 21.75 \\ 9.75 \\ -33.25 \\ 5.75 \\ -24.25 \\ 3.75 \\ 17.75 \\ -1.25 \end{pmatrix}$$

Step 2(4)

Calculate the least squares estimates:

$$\hat{\beta}^{(0)} = (\tilde{X}^{*t}_{(4)}\tilde{X}^*_{(4)})^{-1}\tilde{X}^{*t}_{(4)}\tilde{y}^*_{(4)}$$

$$= \begin{pmatrix} 0.460277 \\ -0.227307 \\ 0.853942 \end{pmatrix}$$

$$\hat{\mu}_y = 98.25$$

$$\hat{\mu}_x = (96.25 \quad 91.625 \quad 89.375)$$

$$\hat{\beta}^{(0)}_0 = 98.25 - (96.25 \quad 91.625 \quad 89.375)\begin{pmatrix} 0.460277 \\ -0.227307 \\ 0.853942 \end{pmatrix}$$

$$= -1.5458$$

Use the fitted regression model to estimate y_7 :

$$y_{7(4)}^{(0)} = \hat{\beta}_0^{(0)} + \sum_{j=1}^{3} x_{7j}\hat{\beta}_j^{(0)} \qquad \text{where } u_j = 1 \text{ if } x_{3j} \text{ is observed}$$
$$= 0 \text{ if } x_{3j} \text{ is missing}$$

$$= -1.5458 + 119(0.460277) + 91(-0.227307) + 104(0.853942)$$
$$= 121.352$$

Create a new "data" column vector $Z_4^{(0)}$ similar to Z_4 except that it contains the estimate of $y_{7(4)}^{(0)}$ in place of the missing value.

CYCLE 1

The new "data" matrix $Z^{(0)}$ contains new estimates in place of the missing values:

$$Z^{(0)} = \begin{pmatrix} 103 & 80 & 96 & 120 \\ 101 & 83 & 86 & 108 \\ \mathbf{116.654} & \mathbf{103.366} & 80 & 98 \\ 61 & 94 & 75 & 65 \\ 92 & 121 & 104 & 104 \\ 80 & 83 & 86 & 74 \\ 119 & 91 & 104 & \mathbf{121.352} \\ 91 & 70 & 77 & 102 \\ 116 & 115 & 97 & 116 \\ 126 & 87 & 94 & 97 \end{pmatrix}$$

Step 1(1)

Let

$$y_{(1)}^{(0)} = Z_1^{(0)} \qquad \text{and} \qquad X_{(1)}^{(0)} = (Z_2^{(0)}, Z_3^{(0)}, Z_4^{(0)}).$$

Required to re-estimate: y_3

Let

$$\tilde{X}_{(1)}^{(0)} = X_{(1)}^{(0)} - \bar{X}_{(1)}^{(0)}, \qquad \text{where} \qquad \bar{X}_{(1)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(0)}, \quad j = 1,2,3$$

and $\tilde{y}_{(1)}^{(0)} = y_{(1)}^{(0)} - \bar{y}_{(1)}^{(0)}$, where $\bar{y}_{(1)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(0)}$

represent the standardized $X_{(1)}^{(0)}$ matrix and $y_{(1)}^{(0)}$ vector respectively.

$$\tilde{X}_{(1)}^{(0)} = \begin{pmatrix} -12.7366 & 6.1 & 19.4648 \\ -9.7366 & -3.9 & 7.4648 \\ 10.6294 & -9.9 & -2.5352 \\ 1.2634 & -14.9 & -35.5352 \\ 28.2634 & 14.1 & 3.4648 \\ -9.7366 & -3.9 & -26.5352 \\ -1.7366 & 14.1 & 20.8168 \\ -22.7366 & -12.9 & 1.4648 \\ 22.2634 & 7.1 & 15.4648 \\ -5.7366 & 4.1 & -3.5352 \end{pmatrix} \quad \text{and} \quad \tilde{y}_{(1)}^{(0)} = \begin{pmatrix} 2.4346 \\ 0.4346 \\ 16.0886 \\ -39.5654 \\ -8.5654 \\ -20.5654 \\ 18.4346 \\ -9.5654 \\ 15.4346 \\ 25.4346 \end{pmatrix}$$

Step 2(1)

Calculate the least squares estimates:

$$\hat{\beta}^{(1)} = (X_{(1)}^{(0)t} X_{(1)}^{(0)})^{-1} X_{(1)}^{(0)t} y_{(1)}^{(0)}$$

$$= \begin{pmatrix} 0.0834068 \\ 0.0547696 \\ 0.761131 \end{pmatrix}$$

$$\hat{\mu}_y = 100.565$$

$$\hat{\mu}_x = (92.7366 \quad 89.9 \quad 100.535)$$

$$\hat{\beta}_0^{(1)} = 100.565 - (92.7366 \quad 89.9 \quad 100.535) \begin{pmatrix} 0.0834068 \\ 0.0547696 \\ 0.761131 \end{pmatrix}$$

$$= 11.3863$$

Use the fitted regression model to re-estimate y_3 :

$$y_{3(1)}^{(1)} = \hat{\beta}_0^{(1)} + \sum_{j=1}^{3} x_{3j} \hat{\beta}_j^{(1)}$$

$$= 11.3863 + \mathbf{103.366}(0.0834068) + 80(0.0547696) + 98(0.7366)$$

$$= 98.98$$

Create a new "data" column vector $Z_1^{(1)}$ which contains the re-estimated value $y_{3(1)}^{(1)}$ in place of the originally missing value.

Step 3(1)

Calculate:

$$\text{Conv}_1 = \left| \frac{98.98 - 116.654}{98.98} \right|$$

$$= 0.1785613$$

Step 1(2)

Let

$$y_{(2)}^{(0)} = Z_2^{(0)} \qquad \text{and} \qquad X_{(2)}^{(0)} = (Z_1^{(0)}, Z_3^{(0)}, Z_4^{(0)}).$$

Required to re-estimate: y_3

Let

$$\tilde{X}_{(2)}^{(0)} = X_{(2)}^{(0)} - \bar{X}_{(2)}^{(0)}, \qquad \text{where} \qquad \bar{X}_{(2)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(0)}, \quad j = 1,2,3$$

and $\tilde{y}_{(2)}^{(0)} = y_{(2)}^{(0)} - \bar{y}_{(2)}^{(0)}$, where $\bar{y}_{(2)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(0)}$

represent the standardized $X_{(2)}^{(0)}$ matrix and $y_{(2)}^{(0)}$ vector respectively.

$$\tilde{X}_{(2)}^{(0)} = \begin{pmatrix} 2.4346 & 6.1 & 19.4648 \\ 0.4346 & -3.9 & 7.4648 \\ 16.0886 & -9.9 & -2.5352 \\ -39.5654 & -14.9 & -35.5352 \\ -8.5654 & 14.1 & 3.4648 \\ -20.5654 & -3.9 & -26.5352 \\ 18.4346 & 14.1 & 20.8168 \\ -9.5654 & -12.9 & 1.4648 \\ 15.4354 & 7.1 & 15.4648 \\ 25.4346 & 4.1 & -3.5352 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(2)}^{(0)} = \begin{pmatrix} -12.7366 \\ -9.7366 \\ 10.6294 \\ 1.2634 \\ 28.2634 \\ -9.7366 \\ -1.7366 \\ -22.7366 \\ 22.2634 \\ -5.7366 \end{pmatrix}$$

Step 2(2)

Calculate the least squares estimates:

$$\hat{\beta}^{(1)} = (X_{(2)}^{(0)t} X_{(2)}^{(0)})^{-1} X_{(2)}^{(0)t} y_{(2)}^{(0)}$$

$$= \begin{pmatrix} 0.0841815 \\ 0.98473 \\ -0.352731 \end{pmatrix}$$

$$\hat{\mu}_y = 92.7366$$

$$\hat{\mu}_x = (100.565 \quad 89.9 \quad 100.535)$$

$$\hat{\beta}_0^{(1)} = 92.736617 - (100.565 \quad 89.9 \quad 100.535) \begin{pmatrix} 0.0841815 \\ 0.98473 \\ -0.352731 \end{pmatrix}$$

$$= 31.2056$$

Use the fitted regression model to re-estimate y_3 :

$$y_{3(2)}^{(1)} = \hat{\beta}_0^{(1)} + \sum_{j=1}^{3} x_{3j}\hat{\beta}_j^{(1)}$$

$$= 31.2056 + \mathbf{116.654}(0.0841815) + 80(0.98473) + 98(-0.352731)$$

$$= 85.236$$

Create a new "data" column vector $Z_2^{(1)}$ which contains the re-estimated value $y_{3(2)}^{(1)}$ in place of the originally missing value.

Step 3(2)

Calculate:

$$\text{Conv}_2 = \left| \frac{85.236 - 103.366}{85.236} \right| = 0.2127035$$

Step 1(3)

Let $Z_3^{(1)} = Z_3$.

Step 1(4)

Let

$$y_{(4)}^{(0)} = Z_4^{(0)} \qquad \text{and} \qquad X_{(4)}^{(0)} = (Z_1^{(0)}, Z_2^{(0)}, Z_3^{(0)}).$$

Required to re-estimate: y_7

Let

$$\tilde{X}_{(4)}^{(0)} = X_{(4)}^{(0)} - \bar{X}_{(4)}^{(0)}, \qquad \text{where} \qquad \bar{X}_{(4)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(0)}, \quad j = 1,2,3$$

and $$\tilde{y}_{(4)}^{(0)} = y_{(4)}^{(0)} - \bar{y}_{(4)}^{(0)}, \qquad \text{where} \qquad \bar{y}_{(4)}^{(0)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(0)}$$

represent the standardized $X_{(4)}^{(0)}$ matrix and $y_{(4)}^{(0)}$ vector respectively.

$$\tilde{X}_{(4)}^{(0)} = \begin{pmatrix} 2.4346 & 12.7366 & 6.1 \\ 0.4346 & -9.7366 & -3.9 \\ 16.0886 & 10.6294 & -9.9 \\ -39.5654 & 1.2634 & -14.9 \\ -8.5654 & 28.2634 & 14.1 \\ -20.5654 & -9.7366 & -3.9 \\ 18.4346 & -1.7366 & 14.1 \\ -9.5654 & -22.7366 & -12.9 \\ 15.4346 & 22.2634 & 7.1 \\ 25.4346 & -5.7366 & 4.1 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(4)}^{(0)} = \begin{pmatrix} 19.4648 \\ 7.4648 \\ -2.5352 \\ -35.5352 \\ 3.4648 \\ -26.5352 \\ 20.8168 \\ 1.4648 \\ 15.4648 \\ -3.5352 \end{pmatrix}$$

Step 2(4)

Calculate the least squares estimates:

$$\hat{\beta}^{(1)} = (X_{(4)}^{(0)t} X_{(4)}^{(0)})^{-1} X_{(4)}^{(0)t} y_{(4)}^{(0)}$$

$$= \begin{pmatrix} 0.471262 \\ -0.216387 \\ 0.825874 \end{pmatrix}$$

$$\hat{\mu}_y = 100.535$$

$$\hat{\mu}_x = (100.565 \quad 92.7366 \quad 89.9)$$

$$\hat{\beta}_0^{(1)} = 100.535 - (100.565 \quad 92.7366 \quad 89.9) \begin{pmatrix} 0.471262 \\ -0.216387 \\ 0.825874 \end{pmatrix}$$

$$= -1.03656$$

Use the fitted regression model to re-estimate y_7 :

$$y_{7(4)}^{(1)} = \hat{\beta}_0^{(1)} + \sum_{j=1}^{3} x_{7j} \hat{\beta}_j^{(1)}$$

$$= -1.03656 + 119(0.471262) + 91(-0.216387) + 104(0.825874)$$

$$= 121.243$$

Step 3(4)

Calculate:

$$\text{Conv}_4 = \left| \frac{121.243 - 121.352}{121.243} \right|$$

$$= 0.00089902$$

Step 4

Check for convergence:

$$\text{Crit} = 0.1785613 + 0.2127035 + 0.00089902$$

$$= 0.3921638$$

$$> 10^{-2}$$

Crit is greater than 10^{-2} therefore continue with the next cycle.

CYCLE 2

The new "data" matrix $Z^{(1)}$ contains new estimates in place of the missing values:

$$Z^{(1)} = \begin{pmatrix} 103 & 80 & 96 & 120 \\ 101 & 83 & 86 & 108 \\ \mathbf{98.980} & \mathbf{85.236} & 80 & 98 \\ 61 & 94 & 75 & 65 \\ 92 & 121 & 104 & 104 \\ 80 & 83 & 86 & 74 \\ 119 & 91 & 104 & \mathbf{121.243} \\ 91 & 70 & 77 & 102 \\ 116 & 115 & 97 & 116 \\ 126 & 87 & 94 & 97 \end{pmatrix}$$

Step 1(1)

Let

$$y_{(1)}^{(1)} = Z_1^{(1)} \qquad \text{and} \qquad X_{(1)}^{(1)} = (Z_2^{(1)}, Z_3^{(1)}, Z_4^{(1)}).$$

Required to re-estimate: y_3

Let

$$\tilde{X}_{(1)}^{(1)} = X_{(1)}^{(1)} - \bar{X}_{(1)}^{(1)}, \qquad \text{where} \qquad \bar{X}_{(1)}^{(1)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(1)}, \quad j = 1,2,3$$

and $$\tilde{y}_{(1)}^{(1)} = y_{(1)}^{(1)} - \bar{y}_{(1)}^{(1)}, \qquad \text{where} \qquad \bar{y}_{(1)}^{(1)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(1)}$$

represent the standardized $X_{(1)}^{(1)}$ matrix and $y_{(1)}^{(1)}$ vector respectively.

$$\tilde{X}_{(1)}^{(1)} = \begin{pmatrix} -10.9236 & 6.1 & 19.4757 \\ -\ 7.9236 & -\ 3.9 & 7.4757 \\ -\ 5.6876 & -\ 9.9 & -\ 2.5243 \\ 3.0764 & -14.9 & -35.5243 \\ 30.0764 & 14.1 & 3.4757 \\ -\ 7.9236 & -\ 3.9 & -26.5243 \\ 0.0764 & 14.1 & 20.7187 \\ -20.9236 & -12.9 & 1.4757 \\ 24.0764 & 7.1 & 15.4757 \\ -\ 3.9236 & 4.1 & -\ 3.5243 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(1)}^{(1)} = \begin{pmatrix} 4.202 \\ 2.202 \\ 0.182 \\ -37.798 \\ -\ 6.798 \\ -18.798 \\ 20.202 \\ -\ 7.798 \\ 17.202 \\ 27.202 \end{pmatrix}$$

Step 2(1)

Calculate the least squares estimates:

$$\hat{\beta}^{(2)} = (X_{(1)}^{(1)^t} X_{(1)}^{(1)})^{-1} X_{(1)}^{(1)^t} y_{(1)}^{(1)}$$

$$= \begin{pmatrix} -0.250909 \\ 0.730681 \\ 0.553834 \end{pmatrix}$$

$$\hat{\mu}_y = 98.798$$

$$\hat{\mu}_x = (90.9236 \quad 89.9 \quad 100.524)$$

$$\hat{\beta}_0^{(2)} = 98.798 - (90.9236 \quad 89.9 \quad 100.524) \begin{pmatrix} -0.250909 \\ 0.730681 \\ 0.553834 \end{pmatrix}$$

$$= 0.249555$$

Use the fitted regression model to re-estimate y_3 :

$$y_{3(1)}^{(2)} = \hat{\beta}_0^{(2)} + \sum_{j=1}^{3} x_{3j}\hat{\beta}_j^{(2)}$$

$$= 0.249555 + \mathbf{85.236}(-0.250909) + 80(0.730681) + 98(0.553834)$$

$$= 91.593$$

Create a new "data" column vector $Z_1^{(2)}$ which contains the re-estimated value $y_{3(1)}^{(2)}$ in place of the originally missing value.

Step 3(1)

Calculate:

$$\text{Conv}_1 = \left| \frac{91.593 - 98.98}{91.593} \right|$$

$$= 0.0806502$$

Step 1(2)

Let

$$y_{(2)}^{(1)} = Z_2^{(0)} \qquad \text{and} \qquad X_{(2)}^{(1)} = (Z_1^{(1)}, Z_3^{(1)}, Z_4^{(1)}).$$

Required to re-estimate: y_3

Let

$$\tilde{X}_{(2)}^{(1)} = X_{(2)}^{(1)} - \bar{X}_{(2)}^{(1)}, \qquad \text{where} \qquad \bar{X}_{(2)}^{(1)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(1)}, \quad j = 1, 2, 3$$

and $\tilde{y}_{(2)}^{(1)} = y_{(2)}^{(1)} - \bar{y}_{(2)}^{(1)}$, where $\bar{y}_{(2)}^{(1)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(1)}$

represent the standardized $X_{(2)}^{(1)}$ matrix and $y_{(2)}^{(1)}$ vector respectively.

$$\tilde{X}_{(2)}^{(1)} = \begin{pmatrix} 4.202 & 6.1 & 19.4757 \\ 2.202 & -3.9 & 7.4757 \\ 0.182 & -9.9 & -2.5243 \\ -37.798 & -14.9 & -35.5243 \\ -6.798 & 14.1 & 3.4757 \\ -18.798 & -3.9 & -26.5243 \\ 20.202 & 14.1 & 20.7187 \\ -7.798 & -12.9 & 1.4757 \\ 17.202 & 7.1 & 15.4757 \\ 27.202 & 4.1 & -3.5243 \end{pmatrix} \quad \text{and} \quad \tilde{y}_{(2)}^{(1)} = \begin{pmatrix} -10.9236 \\ -7.9236 \\ -5.6876 \\ 3.0764 \\ 30.0764 \\ -7.9236 \\ 0.0764 \\ -20.9236 \\ 24.0764 \\ -3.9236 \end{pmatrix}$$

Step 2(2)

Calculate the least squares estimates:

$$\hat{\beta}^{(2)} = (X_{(2)}^{(1)t} X_{(2)}^{(1)})^{-1} X_{(2)}^{(1)t} y_{(2)}^{(1)}$$

$$= \begin{pmatrix} -0.250013 \\ 1.37715 \\ -0.217914 \end{pmatrix}$$

$$\hat{\mu}_y = 90.9236$$

$$\hat{\mu}_x = (98.798 \quad 89.9 \quad 100.524)$$

$$\hat{\beta}_0^{(2)} = 90.9236 - (98.798 \quad 89.9 \quad 100.524) \begin{pmatrix} -0.250013 \\ 1.37715 \\ -0.217914 \end{pmatrix}$$

$$= 13.7239$$

Use the fitted regression model to re-estimate y_3 :

$$y_{3(2)}^{(2)} = \hat{\beta}_0^{(2)} + \sum_{j=1}^{3} x_{3j} \hat{\beta}_j^{(2)}$$

$$= 13.7242 + \mathbf{98.98}(-0.250013) + 80(1.37715) + 98(-0.217914)$$

$$= 77.794$$

Create a new "data" column vector $Z_2^{(2)}$ which contains the re-estimated value $y_{3(2)}^{(2)}$ in place of the originally missing value.

Step 3(2)

Calculate:

$$\text{Conv}_2 = \left| \frac{77.794 - 85.236}{77.794} \right|$$

$$= 0.0956629$$

Step 1(3)

Let $Z_3^{(2)} = Z_3$.

Step 1(4)

Let

$$y_{(4)}^{(1)} = Z_4^{(1)} \qquad \text{and} \qquad X_{(4)}^{(1)} = (Z_1^{(1)}, Z_2^{(1)}, Z_3^{(1)}).$$

Required to re-estimate: y_7

Let

$$\tilde{X}_{(4)}^{(1)} = X_{(4)}^{(1)} - \bar{X}_{(4)}^{(1)}, \qquad \text{where} \qquad \bar{X}_{(4)}^{(1)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(1)}, \quad j = 1,2,3$$

$$\text{and} \qquad \tilde{y}_{(4)}^{(1)} = y_{(4)}^{(1)} - \bar{y}_{(4)}^{(1)}, \qquad \text{where} \qquad \bar{y}_{(4)}^{(1)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(1)}$$

represent the standardized $X_{(4)}^{(1)}$ matrix and $y_{(4)}^{(1)}$ vector respectively.

$$\tilde{X}_{(4)}^{(1)} = \begin{pmatrix} 4.202 & -10.9236 & 6.1 \\ 2.202 & -7.9236 & -3.9 \\ 0.182 & -5.6876 & -9.9 \\ -37.798 & 3.0764 & -14.9 \\ -6.798 & 30.0764 & 14.1 \\ -18.798 & -7.9236 & -3.9 \\ 20.202 & 0.0764 & 14.1 \\ -7.798 & -20.9236 & -12.9 \\ 17.202 & 24.0764 & 7.1 \\ 27.202 & -3.9236 & 4.1 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(4)}^{(1)} = \begin{pmatrix} 19.4757 \\ 7.4757 \\ -2.5243 \\ -35.5243 \\ 3.4757 \\ -26.5243 \\ 20.7187 \\ 1.4757 \\ 15.4757 \\ -3.5243 \end{pmatrix}$$

Step 2(4)

Calculate the least squares estimates:

$$\hat{\beta}^{(2)} = (X_{(4)}^{(1)t} X_{(4)}^{(1)})^{(-1)} X_{(4)}^{(1)t} y_{(4)}^{(1)}$$

$$= \begin{pmatrix} 0.500462 \\ -0.19762 \\ 0.732124 \end{pmatrix}$$

$$\hat{\mu}_y = 100.524$$

$$\hat{\mu}_x = (98.798 \quad 90.9236 \quad 89.9)$$

$$\hat{\beta}_0^{(2)} = 100.524 - (98.798 \quad 90.9236 \quad 89.9) \begin{pmatrix} 0.500462 \\ -0.19762 \\ 0.732124 \end{pmatrix}$$

$$= 3.23005$$

Use the fitted regression model to re-estimate y_7 :

$$\begin{aligned} y_{7(4)}^{(2)} &= \hat{\beta}_0^{(2)} + \sum_{j=1}^{3} x_{7j}\hat{\beta}_j^{(2)} \\ &= 3.23005 + 119(0.500462) + 91(-0.19762) + 104(0.732124) \\ &= 120.942 \end{aligned}$$

Step 3(4)

Calculate:

$$\begin{aligned} \text{Conv}_4 &= \left| \frac{120.942 - 121.243}{120.942} \right| \\ &= 0.0024887 \end{aligned}$$

Step 4

Check for convergence:

$$\begin{aligned} \text{Crit} &= 0.0806502 + 0.0956629 + 0.0024887 \\ &= 0.1788018 \\ &> 10^{-2} \end{aligned}$$

Since Crit is not small enough, continue with the next cycle.

CYCLE 3

The new "data" matrix $Z^{(2)}$ contains new estimates in place of the missing values:

$$Z^{(2)} = \begin{pmatrix} 103 & 80 & 96 & 120 \\ 101 & 83 & 86 & 108 \\ \mathbf{91.593} & \mathbf{77.794} & 80 & 98 \\ 61 & 94 & 75 & 65 \\ 92 & 121 & 104 & 104 \\ 80 & 83 & 86 & 74 \\ 119 & 91 & 104 & \mathbf{120.942} \\ 91 & 70 & 77 & 102 \\ 116 & 115 & 97 & 116 \\ 126 & 87 & 94 & 97 \end{pmatrix}$$

Step 1(1)

Let

$$y_{(1)}^{(2)} = Z_1^{(2)} \qquad \text{and} \qquad X_{(1)}^{(2)} = (Z_2^{(2)}, Z_3^{(2)}, Z_4^{(2)}).$$

Required to re-estimate: y_3

Let

$$\tilde{X}_{(1)}^{(2)} = X_{(1)}^{(2)} - \bar{X}_{(1)}^{(2)}, \qquad \text{where} \qquad \bar{X}_{(1)}^{(2)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(2)}, \quad j = 1,2,3$$

and $$\tilde{y}_{(1)}^{(2)} = y_{(1)}^{(2)} - \bar{y}_{(1)}^{(2)}, \qquad \text{where} \qquad \bar{y}_{(1)}^{(2)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(2)}$$

represent the standardized $X_{(1)}^{(2)}$ matrix and $y_{(1)}^{(2)}$ vector respectively.

$$\tilde{X}_{(1)}^{(2)} = \begin{pmatrix} -10.1794 & 6.1 & 19.5058 \\ -7.1794 & -3.9 & 7.5058 \\ -12.3854 & -9.9 & -2.4942 \\ 3.8206 & -14.9 & -35.4942 \\ 30.8206 & 14.1 & 3.5058 \\ -7.1794 & -3.9 & -26.4942 \\ 0.8206 & 14.1 & 20.4478 \\ -20.1794 & -12.9 & 1.5058 \\ 24.8206 & 7.1 & 15.5058 \\ -3.1794 & 4.1 & -3.4942 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(1)}^{(2)} = \begin{pmatrix} 4.9407 \\ 2.9407 \\ -6.4663 \\ -37.0593 \\ -6.0593 \\ -18.0593 \\ 20.9407 \\ -7.0593 \\ 17.9407 \\ 27.9407 \end{pmatrix}$$

Step 2(1)

Calculate the least squares estimates:

$$\hat{\beta}^{(3)} = (X_{(1)}^{(2)^t} X_{(1)}^{(2)})^{-1} X_{(1)}^{(2)^t} y_{(1)}^{(2)}$$

$$= \begin{pmatrix} -0.300026 \\ 0.942566 \\ 0.488113 \end{pmatrix}$$

$$\hat{\mu}_y = 98.0593$$

$$\hat{\mu}_x = (90.1794 \quad 89.9 \quad 100.494)$$

$$\hat{\beta}_0^{(3)} = 98.0593 - (90.1794 \quad 89.9 \quad 100.494) \begin{pmatrix} -0.300026 \\ 0.942566 \\ 0.488113 \end{pmatrix}$$

$$= -8.6738$$

Use the fitted regression model to re-estimate y_3 :

$$y_{3(1)}^{(3)} = \hat{\beta}_0^{(3)} + \sum_{j=1}^{3} x_{3j}\hat{\beta}_j^{(3)}$$

$$= -8.6738 + \mathbf{77.794}(-0.300026) + 80(0.942566) + 98(0.488113)$$

$$= 91.226$$

Create a new "data" column vector $Z_1^{(3)}$ which contains the re-estimated value $y_{3(1)}^{(3)}$ in place of the originally missing value.

Step 3(1)

Check for convergence:

$$\text{Conv}_1 = \left| \frac{91.226 - 91.593}{91.226} \right| = 0.0040229$$

Step 1(2)

Let

$$y_{(2)}^{(2)} = Z_2^{(2)} \qquad \text{and} \qquad X_{(2)}^{(2)} = (Z_1^{(2)}, Z_3^{(2)}, Z_4^{(2)}).$$

Required to re-estimate: y_3

Let

$$\tilde{X}_{(2)}^{(2)} = X_{(2)}^{(2)} - \bar{X}_{(2)}^{(2)}, \qquad \text{where} \qquad \bar{X}_{(2)}^{(2)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(2)}, \quad j = 1,2,3$$

$$\text{and} \qquad \tilde{y}_{(2)}^{(2)} = y_{(2)}^{(2)} - \bar{y}_{(2)}^{(2)}, \qquad \text{where} \qquad \bar{y}_{(2)}^{(2)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(2)}$$

represent the standardized $X_{(2)}^{(2)}$ matrix and $y_{(2)}^{(2)}$ vector respectively.

$$\tilde{X}_{(2)}^{(2)} = \begin{pmatrix} 4.9407 & 6.1 & 19.5058 \\ 2.9407 & -3.9 & 7.5058 \\ -6.4663 & -9.9 & -2.4942 \\ -37.0593 & -14.9 & -35.4942 \\ -6.0593 & 14.1 & 3.5058 \\ -18.0593 & -3.9 & -26.4942 \\ 20.9407 & 14.1 & 20.4478 \\ -7.0593 & -12.9 & 1.5058 \\ 17.9407 & 7.1 & 15.5058 \\ 27.9407 & 4.1 & -3.4942 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(2)}^{(2)} = \begin{pmatrix} -10.1794 \\ -7.1794 \\ -12.3854 \\ 3.8206 \\ 30.8206 \\ -7.1794 \\ 0.8206 \\ -20.1794 \\ 24.8206 \\ -3.1794 \end{pmatrix}$$

Step 2(2)

Calculate the least squares estimates:

$$\hat{\beta}^{(3)} = (X_{(2)}^{(2)t} X_{(2)}^{(2)})^{-1} X_{(2)}^{(2)t} y_{(2)}^{(2)}$$

$$= \begin{pmatrix} -0.299159 \\ 1.54325 \\ -0.232441 \end{pmatrix}$$

$$\hat{\mu}_y = 90.1794$$

$$\hat{\mu}_x = (98.0593 \quad 89.9 \quad 100.494)$$

$$\hat{\beta}_0^{(3)} = 90.1794 - (98.0593 \quad 89.9 \quad 100.494) \begin{pmatrix} -0.299159 \\ 1.54325 \\ -0.232441 \end{pmatrix}$$

$$= 4.13541$$

Use the fitted regression model to re-estimate y_3 :

$$y_{3(2)}^{(3)} = \hat{\beta}_0^{(3)} + \sum_{j=1}^{3} x_{3j} \hat{\beta}_j^{(3)}$$

$$= 4.13541 + \mathbf{91.593}(-0.299159) + 80(1.54325) + 98(-0.232441)$$

$$= 77.415$$

Create a new "data" column vector $Z_2^{(3)}$ which contains the re-estimated value $y_{3(2)}^{(3)}$ in place of the originally missing value.

Step 3(2)

Calculate:

$$\text{Conv}_2 = \left| \frac{77.415 - 77.794}{77.415} \right|$$

$$= 0.0048956$$

Step 1(3)

Let $Z_3^{(3)} = Z_3$.

Step 1(4)

Let

$$y_{(4)}^{(2)} = Z_4^{(2)} \qquad \text{and} \qquad X_{(4)}^{(2)} = (Z_1^{(2)}, Z_2^{(2)}, Z_3^{(2)}).$$

Required to re-estimate: y_7

Let

$$\tilde{X}_{(4)}^{(2)} = X_{(4)}^{(2)} - \bar{X}_{(4)}^{(2)}, \qquad \text{where} \qquad \bar{X}_{(4)}^{(2)} = \frac{1}{10}\sum_{i=1}^{10} x_{ij}^{(2)}, \quad j = 1,2,3$$

and $\tilde{y}_{(4)}^{(2)} = y_{(4)}^{(2)} - \bar{y}_{(4)}^{(2)}$, where $\bar{y}_{(4)}^{(2)} = \frac{1}{10}\sum_{i=1}^{10} y_i^{(2)}$

represent the standardized $X_{(4)}^{(2)}$ matrix and $y_{(4)}^{(2)}$ vector respectively.

$$\tilde{X}_{(4)}^{(2)} = \begin{pmatrix} 4.9407 & -10.1794 & 6.1 \\ 2.9407 & -7.1794 & -3.9 \\ -6.4663 & -12.3854 & -9.9 \\ -37.0593 & 3.8206 & -14.9 \\ -6.0593 & 30.8206 & 14.9 \\ -18.0593 & -7.1794 & -3.9 \\ 20.9407 & 0.8206 & 14.1 \\ -7.0593 & -20.1794 & -12.9 \\ 17.9407 & 24.8206 & 7.1 \\ 27.9407 & -3.1794 & 4.1 \end{pmatrix} \qquad \text{and} \qquad \tilde{y}_{(4)}^{(2)} = \begin{pmatrix} 19.5058 \\ 7.5058 \\ -2.4942 \\ -35.4942 \\ 3.5058 \\ -26.4942 \\ 20.4478 \\ 1.5058 \\ 15.5058 \\ -3.4942 \end{pmatrix}$$

Step 2(4)

Calculate the least squares estimates:

$$\hat{\beta}^{(3)} = (X_{(4)}^{(2)t} X_{(4)}^{(2)})^{-1} X_{(4)}^{(2)t} y_{(4)}^{(2)}$$

$$= \begin{pmatrix} 0.479215 \\ -0.228865 \\ 0.76085 \end{pmatrix}$$

$$\hat{\mu}_y = 100.494$$

$$\hat{\mu}_x = (98.0593 \quad 90.1794 \quad 89.9)$$

$$\hat{\beta}_0^{(3)} = 100.494 - (98.0593 \quad 90.1794 \quad 89.9) \begin{pmatrix} 0.479215 \\ -0.228865 \\ 0.76085 \end{pmatrix}$$

$$= 5.74118$$

Use the fitted regression model to re-estimate y_7 :

$$y_{7(4)}^{(3)} = \hat{\beta}_0^{(3)} + \sum_{j=1}^{3} x_{7j}\hat{\beta}_j^{(3)}$$

$$= 5.74118 + 119(0.479215) + 91(-0.228865) + 104(0.76085)$$

$$= 121.069$$

Step 3(4)

Calculate:

$$\text{Conv}_4 = \left| \frac{121.069 - 120.942}{121.069} \right|$$
$$= 0.00104898$$

Step 4

Check for convergence:

$$\text{Crit} = 0.0040229 + 0.0048956 + 0.00104498$$
$$= 0.0099634$$
$$< 10^{-2}$$

Since Crit is less than 10^{-2}, therefore the required estimates for the missing values are obtained, and these values are not re-estimated again.

RESULTS

All the missing values are now estimated. Let $Z = Z^{(3)}$. Note that these estimates have been obtained after *four* cycles or iterations.

THE MATRIX WITH COMPLETE DATA:

$$Z = \begin{pmatrix} 103 & 80 & 96 & 120 \\ 101 & 83 & 86 & 108 \\ \mathbf{91.226} & \mathbf{77.415} & 80 & 98 \\ 61 & 94 & 75 & 65 \\ 92 & 121 & 104 & 104 \\ 80 & 83 & 86 & 74 \\ 119 & 91 & 104 & \mathbf{121.069} \\ 91 & 70 & 77 & 102 \\ 116 & 115 & 97 & 116 \\ 126 & 87 & 94 & 97 \end{pmatrix}$$

The estimated values are the same as those obtained using **Method 1**.* Although this algorithm required more cycles, note that algorithmically it is simpler than the previous algorithm, and, in fact, it is computationally less expensive.

* to the accuracy required by the criterion.

APPENDIX B

PROOF FOR SECTION 6.3.1

In this appendix we give a proof for the statement in section 6.3.1, namely that regression methods lead to a systematic downward bias in the estimates of the standard deviation of the completed record. We consider the simple (2 variables) regression case first and then the multivariate case.

B1. SIMPLE LINEAR REGRESSION

Consider complete observations

$$y_i = \alpha + \beta x_i + e_i \qquad i = 1, 2, \ldots, n_1$$

with y missing for $(x_{n_1+1}, x_{n_2+1}, \ldots, x_n)$

$$\hat{\beta}_{n_1} = \frac{\sum_{i=1}^{n_1}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n_1}(x_i - \bar{x})^2}$$

and

$$\hat{\alpha}_{n_1} = \bar{y}_{n_1} - \bar{x}_{n_1}\hat{\beta}_{n_1}$$

where

$$\bar{y}_{n_1} = \frac{1}{n_1}\sum_{i=1}^{n_1} y_i \qquad \text{and} \qquad \bar{x}_{n_1} = \frac{1}{n_1}\sum_{i=1}^{n_1} x_i$$

Patch the missing values with

$$\hat{y}_i = \hat{\alpha}_{n_1} + \hat{\beta}_{n_1} x_i \qquad i = n_1 + 1, n_1 + 2, \ldots, n$$
$$= \bar{y}_{n_1} + (x_i - \bar{x}_{n_1})\hat{\beta}_{n_1}$$

Examine the sample variance of

$$y_1, y_2, \cdots, y_{n_1}, \hat{y}_{n_1+1}, \hat{y}_{n_1+2}, \cdots, \hat{y}_n$$

$$S^2_{y(patched)} = \frac{1}{n}\left(\sum_{i=1}^{n_1}(y_i - \bar{y})^2 + \sum_{i=n_1+1}^{n}(\hat{y}_i - \bar{y})^2\right)$$

where:

$$\bar{y} = \frac{1}{n}\left(\sum_{i=1}^{n_1} y_i + \sum_{i=n_1+1}^{n} \hat{y}_i\right)$$

$$= \frac{n_1}{n}\bar{y}_{n_1} + \left(\frac{n-n_1}{n}\right)\left(\bar{y}_{n_1} + (\bar{x}_{n-n_1} - \bar{x}_{n_1})\hat{\beta}_{n_1}\right)$$

$$= \bar{y}_{n_1} + (\frac{n-n_1}{n})(\bar{x}_{n-n_1} - \bar{x}_{n_1})\hat{\beta}_{n_1}$$

$$S^2_{y(patched)} = \frac{1}{n}\left(\sum_{i=1}^{n_1}(y_i - \bar{y}_{n_1})^2 + \frac{n_1(n-n_1)^2}{n^2}\{(\bar{x}_{n-n_1} - \bar{x}_{n_1})\hat{\beta}_{n_1}\}^2\right.$$

$$\left. + \left(\frac{n_1}{n}\right)^{2(n-n_1)}\{(\bar{x}_{n-n_1} - \bar{x}_{n_1})\hat{\beta}_{n_1}\}^2 + \sum_{i=n_1+1}\{(x_i - \bar{x}_{n-n_1})\hat{\beta}_{n_1}\}^2\right)$$

$$= \frac{1}{n}\{n_1 S^2_{y_{n_1}} + \frac{n_1(n-n_1)}{n}\{(\bar{x}_{n-n_1} - \bar{x}_{n_1})\hat{\beta}_{n_1}\}^2 + C_{xx(n-n_1)}\hat{\beta}_{n_1}\}$$

$$= \frac{S^2_{y_{n_1}}}{n}\left\{n_1 + \frac{n_1^2(n-n_1)}{n}\frac{(\bar{x}_{n-n_1} - \bar{x}_{n_1})}{C_{xx(n_1)}}\hat{\rho}^2 + n_1\frac{C_{xx(n-n_1)}}{C_{xx(n_1)}}\hat{\rho}^2\right\}$$

If sample means and variances for $x_1, x_2, \ldots, x_{n_1}$ and $x_{n_1+1}, x_{n_1+2}, \ldots, x_n$ are the same, that is,

$$\bar{x}_{n-n_1} = \bar{x}_{n_1},$$

$$\frac{C_{xx(n-n_1)}}{n-n_1} = \frac{C_{xx(n_1)}}{n} \quad ,$$

and

$$\hat{\rho}^2 = 1$$

then

$$S^2_{y(patched)} = \frac{S^2_{y_{n_1}}}{n}\{n_1 + 0 + (n-n_1)\}$$
$$= S^2_{y_{n_1}}$$

If however $\hat{\rho}^2 = 0$ then

$$S^2_{y(patched)} = S^2_{y_{n_1}}\frac{n_1}{n}.$$

In such cases the percentage reduction in the patched data's variance is given by $\frac{n_1}{n}100\%$.

B2. MULTIPLE REGRESSION

Consider complete observations given by

$$y_i = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \ldots + \beta_k x_{ik} + e_i \qquad i = 1, 2, \ldots, n_1$$

Data matrix:

$$\begin{pmatrix} y_1 & x_{1,1} & x_{1,2} & \cdots & x_{1,k} \\ y_2 & x_{2,1} & x_{2,2} & \cdots & x_{2,k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{n_1} & x_{n_1,1} & x_{n_1,2} & \cdots & x_{n_1,k} \\ - & x_{n_1+1,1} & x_{n_1+1,2} & \cdots & x_{n_1+1,k} \\ - & x_{n_1+2,1} & x_{n_1+2,2} & \cdots & x_{n_1+2,k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ - & x_{n,1} & x_{n,2} & \cdots & x_{n,k} \end{pmatrix}$$

where " $-$ " indicates missing observations.

Let

$$y_{n_1} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n_1} \end{pmatrix}$$

$$X_1 = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,k} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n_1,1} & x_{n_1,2} & \cdots & x_{n_1,k} \end{pmatrix}$$

and

$$X_2 = \begin{pmatrix} x_{n_1+1,1} & x_{n_1+1,2} & \cdots & x_{n_1+1,k} \\ x_{n_1+2,1} & x_{n_1+2,2} & \cdots & x_{n_1+2,k} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,3} \end{pmatrix}$$

$$\hat{\beta}_{n_1} = (X_1^t X_1)^{-1} X_1^t y_{n_1}$$

and

$$\hat{\alpha}_{n_1} = \bar{y}_{n_1} - \bar{x}_{n_1}^t \hat{\beta}_{n_1}$$

Patch the missing values with

$$\hat{y}_i = \hat{\alpha}_{n_1} + x_i^t \hat{\beta}_{n_1} \qquad i = n_1 + 1, n_1 + 2, \ldots, n$$

The sample variance of the y's including patched values

$$y_1, y_2, \cdots, y_{n_1}, \hat{y}_{n_1+1}, \hat{y}_{n_1+2}, \cdots, \hat{y}_n$$

is

$$S^2_{y(patched)} = \frac{1}{n}\left(\sum_{i=1}^{n_1}(y_i - \bar{y})^2 + \sum_{i=n_1+1}^{n}(\hat{y}_i - \bar{y})^2\right)$$

where:

$$\bar{y} = \frac{1}{n}\left(\sum_{i=1}^{n_1} y_i + \sum_{i=n_1+1}^{n}\hat{y}_i\right)$$

$$= \bar{y}_{n_1} + \frac{n-n_1}{n}(\bar{y}_{n_1} + (\bar{x}_{n-n_1} - \bar{x}_{n_1})^t\hat{\beta}_{n_1})$$

$$= \bar{y}_{n_1} + \frac{n-n_1}{n}(\bar{x}_{n-n_1} - \bar{x}_{n_1})^t\hat{\beta}_{n_1}$$

$$S^2_{y(patched)} = \frac{1}{n}\left\{n_1 S^2_{y_{n_1}} + \frac{n_1(n-n_1)}{n}\{(\bar{x}_{n-n_1} - \bar{x}_{n_1})^t\hat{\beta}_{n_1}\}^2 + \hat{\beta}^t_{n_1}\tilde{X}^t_2\tilde{X}_2\hat{\beta}_{n_1}\right\}$$

where

$$\tilde{X}_2 = (I_{n-n_1} - 1\,1^t/(n-n_1))X_2$$

If

$$\tilde{X}^t_2\tilde{X}_2 = \frac{n-n_1}{n_1}\tilde{X}^t_1\tilde{X}_1,$$
$$\bar{x}_{n-n_1} = \bar{x}_{n_1}$$

and

$$\frac{\hat{\beta}^t_{n-1}X^t_1X_1\hat{\beta}_{n_1}}{S^2_{y_{n_1}}} = R^2_{n_1} = 1$$

then

$$S^2_{y(patched)} = S^2_{y_{n_1}}$$

If the data were uncorrelated, i.e.

$$\hat{\rho} = 0 \qquad \text{or} \qquad \hat{\beta}_{n_1} = 0$$

then

$$S^2_{y(patched)} = \frac{n_1}{n}S^2_{y_{n_1}}$$

APPENDIX C

PROGRAMS

In this appendix, we give a listing of the FORTRAN programs which were developed to implement the methods discussed in this thesis.

PROGRAM 1

```
CC*****************************************************************
CC******                                                          **
CC******  A PROGRAM TO  ESTIMATE  MISSING  RAINFALL  DATA  BY    **
CC******  MAKING  USE OF ONE OF THE  SELECTION OF  VARIABLES     **
CC******  PROCEDURE - SELECTING CONTROL RECORDS FOR INDIVIDUAL   **
CC******  MISSING VALUES.                                        **
CC******                                                          **
CC*****************************************************************
CC*****************************************************************
CC                                                                **
CC   THIS PROGRAM IS USED TO ESTIMATE DATA MATRICES WHICH         **
CC   CONTAIN MISSING OBSERVATIONS IN ALMOST ALL THE RAINFALL      **
CC   STATIONS.                                                    **
CC                                                                **
CC   THE STATIONS ARE READ AS ONE BIG MATRIX WHICH CONSISTS       **
CC   OF A COLUMN OF THE TARGET STATION - WHICH, IN THIS           **
CC   PROGRAM, SHOULD ALWAYS BE THE FIRST COLUMN, AND THE          **
CC   REMAINING COLUMNS BEING THE MATRIX OF CONTROL STATIONS.      **
CC                                                                **
CC   EACH ROW OF DATA REPRESENTS AN OBSERVATION (ANNUAL           **
CC   RAINFALL TOTAL). MISSING RAINFALL DATA POINTS ARE            **
CC   REPRESENTED BY A "-999".                                     **
CC                                                                **
CC   THE DATA IS STORED IN A MATRIX CALLED THE Z-MATRIX,          **
CC   AND THAT IS PARTITIONED INTO THE:                            **
CC              Y-VECTOR = A VECTOR OF THE TARGET STATION         **
CC              X-MATRIX = A MATRIX OF THE CONTROL STATIONS.      **
CC                                                                **
CC   THE MAXIMUM DIMENSIONS OF THE MATRICES ARE:                  **
CC              TARGET STATION          :  1                      **
CC              CONTROL STATIONS        :  25                     **
CC              OBSERVATIONS            :  100                    **
CC                                                                **
CC   NOTE THAT SOME OF THE ROUTINES WHICH ARE IN THIS             **
CC   PROGRAM WERE COPIED FROM THE PROGRAMS WRITTEN BY             **
CC   DR ROSS. S. SPARKS.                                          **
CC                                                                **
CC*****************************************************************
CC*****************************************************************
```

```
CC******* VARIABLES DECLARATIONS *******
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER NOBS, NSTAT, IV, DV
CC
**********************************************************************
**                                                                  **
**  IV     =  NUMBER OF CONTROL STATIONS.                           **
**  DV     =  NUMBER OF TARGET STATIONS.                            **
**  NSTAT  =  NUMBER OF ALL THE STATIONS - TARGET + CONTROL.        **
**  NOBS   =  NUMBER OF ALL THE OBSERVATIONS.                       **
**                                                                  **
**********************************************************************
CC
CC******* PARAMETER STATEMENTS *******
CC
      PARAMETER (NOBS   =   50)
      PARAMETER (NSTAT  =    5)
      PARAMETER (IV     =    4)
      PARAMETER (DV     =    1)
CC
CC******* REAL VARIABLES *******
CC
      REAL Z(NOBS,NSTAT), ZOR(NOBS,NSTAT)
CC
**********************************************************************
**  Z    =  MATRIX OF ALL THE STATIONS.                             **
**  ZOR  =  MATRIX OF THE ORIGINAL DATA MATRIX Z.                   **
**                                                                  **
**********************************************************************
CC
      REAL Y(NOBS,DV), YOR(NOBS,DV), YST(NOBS,DV)
CC
**********************************************************************
**  Y    =  VECTOR OF THE TARGET STATION                            **
**  YOR  =  VECTOR OF CONCURRENT OBSERVATIONS FROM THE              **
**          TARGET STATION                                          **
**  YST  =  VECTOR OF STANDARDIZED YOR MATRIX                       **
**                                                                  **
**********************************************************************
CC
      REAL X(NOBS,IV), XOR(NOBS,IV), XPST(NOBS,NSTAT)
CC
**********************************************************************
**  X     =  MATRIX OF CONTROL STATIONS.                            **
**  XOR   =  MATRIX OF CONCURRENT OBSERVATIONS                      **
**           (CONTROL STATIONS).                                    **
**  XPST  =  MATRIX OF STANDARDIZED XOR MATRIX                      **
**                                                                  **
**********************************************************************
CC
      REAL XP(NOBS,IV), PATCH(NOBS)
CC
**********************************************************************
**   XP    =  MATRIX OF SELECTED SUBSET OF X.                       **
**   PATCH =  TEMPORATY VECTOR OF ESTIMATED Z MATRIX.               **
**                                                                  **
**********************************************************************
```

```
      REAL XPSTT(IV,NOBS), XPT(IV,NOBS)
CC
**********************************************************************
**    XPSTT = THE TRANSPOSED MATRIX OF XPST.                        **
**    XPT   = THE TRANSPOSED MATRIX OF XP                           **
**********************************************************************
CC
      REAL XPTXP(IV,IV), XPTYT(IV,DV), XPTXPI(IV,IV)
CC
**********************************************************************
**    XPTXP    =  XPSTT * XPST                                      **
**    XPTYT    =  XPSTT * YST                                       **
**    XPTXPI   =  THE INVERSE MATRIX OF XPTXP                       **
**********************************************************************
CC
      REAL MEANY, MEANX(DV,IV), MEANXT(IV,DV)
CC
**********************************************************************
**    MEANY     =  MEAN OF THE TARGET STATION.                      **
**    MEANX     =  MEANS OF CONTROL STATIONS.                       **
**    MEANXT    =  MEANX TRANSPOSED                                 **
**********************************************************************
CC
      REAL BETA(IV,DV), BETA2
CC
**********************************************************************
**    BETA, BBETA = LEAST SQUARES PARAMETER ESTIMATES.              **
**    BETA2       = MEANX * BETA                                    **
**********************************************************************
CC
      REAL BETA0
CC
**********************************************************************
**    BETA0       = INTERCEPT TERM.                                 **
**********************************************************************
CC
      REAL YSTBET(NOBS,DV), XSTBET(NOBS,DV)
CC
**********************************************************************
**    XSTBET   =  XPST * BETA                                       **
**    YSTBET   =  YST - XSTBET                                      **
**********************************************************************
CC
      REAL YSBETT(DV,NOBS), YTASTY(DV,DV)
CC
**********************************************************************
**    YSBETT   = THE TRANSPOSE OF YSTBET.                           **
**    YTASTY   = YSBETT * YST                                       **
**********************************************************************
```

```
CC
      REAL PVAR(NOBS), PVAR1, CV1, HAT
CC
**********************************************************************
**   PVAR = PREDICTIVE VARIANCE.                                    **
**********************************************************************
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER  L, COL, ROW, K
      INTEGER  NROW
      INTEGER  IND(IV), NPRED
CC
**********************************************************************
**   NROW      =   NUMBER OF ROWS                                   **
**   NPRED     =   NUMBER OF CONTROL STATIONS IN THE SUBSET         **
**   IND       =   INDICATOR VARIABLE                               **
**********************************************************************
CC
CC******* FORMAT STATEMENTS *******
CC
  10  FORMAT(9F8.0)
  20  FORMAT(20F6.0/5F6.0)
CC
**********************************************************************
** - THE FOLLOWING DO-LOOP READS AND WRITES A MATRIX OF ALL         **
**   THE RAINFALL STATIONS AND ALL THE OBSERVATIONS                 **
**********************************************************************
CC
      DO 100 ROW = 1, NOBS
         READ(13,10) (Z(ROW,COL), COL = 1, NSTAT)
         WRITE(6,10) (Z(ROW,COL), COL = 1, NSTAT)
 100  CONTINUE
CC
**********************************************************************
** - IN THE FOLLOWING DO-LOOP, WE PARTITION THE Z-MATRIX            **
**   INTO THE VECTOR OF THE TARGET STATION AND THE MATRIX           **
**   OF CONTROL STATIONS.                                           **
**********************************************************************
CC
      DO 120 ROW = 1, NOBS
         Y(ROW,1) = Z(ROW,1)
         DO 110 COL = 2, NSTAT
            X(ROW,COL-1) = Z(ROW,COL)
 110     CONTINUE
 120  CONTINUE
CC
      CALL COPY(Z,NOBS,NSTAT,ZOR,NOBS,NSTAT,NOBS,NSTAT)
```

```
      DO 33300 K = 1, NOBS
CC
**********************************************************************
**   Set PVAR(K) to a very big positive number                      **
**********************************************************************
CC
         IF (Y(K,1) .EQ. -999.0) THEN
            PVAR(K)= 9999999.
         ELSE
            PVAR(K)= 0.0
         ENDIF
CC
**********************************************************************
**    Check for a missing record                                    **
**********************************************************************
CC
       IF (Y(K,1) .EQ. -999.0) THEN
CC
            DO 130 L = 1, IV
130         IND(L) = 0
            L = 1
140         IND(L) = 1

            NPRED = 0
            DO 160 COL = 1, IV
               IF (IND(COL) .LT. 1) GO TO 160
                  NPRED = NPRED + 1
                  DO 150 ROW = 1, NOBS
                     XP(ROW,NPRED) = X(ROW,COL)
150                  CONTINUE
160         CONTINUE

            DO 180 ROW = 1, NOBS
               Z(ROW,1) = Y(ROW,1)
               DO 170 COL = 1, NPRED
                  Z(ROW,COL+1) = XP(ROW,COL)
CC
**********************************************************************
**   - Check if the selected control stations are capable of        **
**     estimating the k-th missing observation.                     **
**********************************************************************
CC
                  IF (XP(K,COL) .EQ. -999.0) GO TO 280
CC
 170            CONTINUE
 180         CONTINUE
```

```
**********************************************************************
**  - Check if there are missing observations in any of the         **
**    included control stations.  If there is any, let all the      **
**    observations in that row be equal to "-999.0".                **
**********************************************************************
CC
            NROW = 0
            DO 220 ROW = 1, NOBS
               COL = 1
 190           IF (COL .LE. (NPRED+1)) THEN
                  IF (Z(ROW,COL) .EQ. -999.0) THEN
                     DO 200 COL = 1, NPRED+1
                        Z(ROW,COL) = -999.0
 200                 CONTINUE
                  ENDIF
               COL = COL + 1
               GO TO 190
               ENDIF
CC
**********************************************************************
**  - ELIMINATE FROM Z ALL THOSE ROWS WHICH HAVE "-999.0"           **
**    ENTRIES.  COUNT THE REMAINING NUMBER OF ROWS.                 **
**********************************************************************
CC
               IF (Z(ROW,1) .NE. -999.0) THEN
                  NROW = NROW + 1
                  YOR(NROW,1) = Z(ROW,1)
                  DO 210 COL = 2, NPRED+1
                     XOR(NROW,COL-1) = Z(ROW,COL)
 210              CONTINUE
               ENDIF
CC
 220        CONTINUE
CC
**********************************************************************
**  -  Check for sufficient concurrent records to fit a             **
**     regression model                                             **
**********************************************************************
CC
            IF (NROW .LT. IV+2) GO TO 280
CC
**********************************************************************
** - When called, these subroutines find the least squares          **
**   estimates  and the covariance matrix for  A*, which is         **
**   defined as A* = X*(X*TX*)-1(X*T)                               **
**********************************************************************
CC
            CALL CNTRAL(XPST,NOBS,IV,XOR,NOBS,IV,NROW,NPRED)
            CALL CNTRAL(YST,NOBS,DV,YOR,NOBS,DV,NROW,DV)
            CALL TRANP(XPST,NOBS,IV,XPSTT,IV,NOBS,NROW,NPRED)
            CALL MULT(XPSTT,IV,NOBS,XPST,NOBS,IV,XPTXP,IV,IV,
     &                  NPRED,NROW,NPRED)
```

```
            CALL IDVERT(XPTXP,IV,XPTXPI,NPRED)
            CALL MULT(XPSTT,IV,NOBS,YST,NOBS,DV,XPTYT,IV,DV,NPRED,
     &                NROW,DV)
            CALL MULT(XPTXPI,IV,IV,XPTYT,IV,DV,BETA,IV,DV,NPRED,
     &                NPRED,DV)
            CALL MULT(XPST,NOBS,IV,BETA,IV,DV,XSTBET,NOBS,DV,NROW,
     &                NPRED,DV)
            CALL DIFFS(YSTBET,NOBS,DV,YST,NOBS,DV,XSTBET,NOBS,DV,
     &                NROW,DV)
            CALL TRANP(YSTBET,NOBS,DV,YSBETT,DV,NOBS,NROW,DV)
            CALL MULT(YSBETT,DV,NOBS,YST,NOBS,DV,YTASTY,DV,DV,DV,
     &                NROW,DV)
CC
            MEANY = 0.0
            DO 230 ROW = 1, NROW
               MEANY = MEANY + YOR(ROW,1)
 230        CONTINUE
            MEANY = MEANY / NROW
CC
            BETA2 = 0.0
            DO 250 COL = 1, NPRED
               MEANX(1,COL) = 0.0
               DO 240 ROW = 1, NROW
                  MEANX(1,COL) = MEANX(1,COL) + XOR(ROW,COL)
 240           CONTINUE
               MEANX(1,COL) = MEANX(1,COL) / NROW
               BETA2 = BETA2 + MEANX(1,COL) * BETA(COL,1)
 250        CONTINUE

            CALL TRANP(MEANX,DV,IV,MEANXT,IV,DV,DV,NPRED)
            CALL TRANP(XP,NOBS,IV,XPT,IV,NOBS,NOBS,NPRED)

*****************************************************************
**    Find the intercept term                                  **
*****************************************************************
CC
            BETA0 = MEANY - BETA2
CC
            CV1 = YTASTY(1,1) / (NROW - IV - 1)
CC
            HAT = 0.0
            DO 260 I = 1, NPRED
            DO 260 J = 1, NPRED
CC
               HAT = HAT + ((XP(K,I) - MEANX(1,I))
     &                 * XPTXPI(I,J) * (XPT(J,K) - MEANXT(J,1)))
CC
 260        CONTINUE
CC
            PVAR1 = (1.0 +(1.0 / FLOAT(NOBS)) + HAT) * CV1
```

```
**********************************************************************
**  - Compare the obtained predictive variances.  If the new        **
**    variance is smaller than the previous variance, then          **
**    substitute the old variance by the new variance.              **
**    Estimate the missing value by using the subset                **
**    corresponding to the smallest predictive variance.            **
**********************************************************************
CC
            IF (PVAR(K).GT. PVAR1) THEN
               PVAR(K)= PVAR1
               PATCH(K) = 0.0
               DO 270 I = 1, NPRED
                  PATCH(K) = PATCH(K) + XP(K,I) * BETA(I,1)
 270           CONTINUE
               PATCH(K) = PATCH(K) + BETA0
            ENDIF
CC
**********************************************************************
**    Consider the next subset.                                     **
**********************************************************************
CC
 280        DO 290 L = 1, IV
               IF(IND(L) .EQ. 0) GO TO 140
 290        IND(L) = 0
CC
         ENDIF
CC
33300 CONTINUE
CC
      DO 300 ROW = 1, NOBS
         IF (Y(ROW,1) .EQ. -999.0) THEN
            Z(ROW,1) = PATCH(ROW)
         ENDIF
 300  CONTINUE
CC
      CALL PMAT(Z,NOBS,NSTAT,NOBS,NSTAT)
CC
      STOP
      END
```

PROGRAM 2

```
CC*****************************************************************
CC******                                                          **
CC******  A PROGRAM TO ESTIMATE MISSING RAINFALL DATA BY          **
CC******  MAKING USE OF ONE OF THE SELECTION OF VARIABLES         **
CC******  PROCEDURES - SELECTING CONTROL RECORDS FOR SEVERAL      **
CC******  MISSING VALUES.                                        **
CC******                                                          **
CC*****************************************************************
CC*****************************************************************
CC                                                                **
CC   THIS PROGRAM IS USED TO ESTIMATE DATA MATRICES WHICH         **
CC   CONTAIN MISSING OBSERVATIONS IN ALMOST ALL THE RAINFALL      **
CC   STATIONS.                                                    **
CC                                                                **
CC   THE STATIONS ARE READ AS ONE BIG MATRIX WHICH CONSISTS       **
CC   OF A COLUMN OF THE TARGET STATION - WHICH, IN THIS           **
CC   PROGRAM, SHOULD ALWAYS BE THE FIRST COLUMN, AND THE          **
CC   REMAINING COLUMNS BEING THE MATRIX OF CONTROL STATIONS.      **
CC                                                                **
CC   EACH ROW OF DATA REPRESENTS AN OBSERVATION (ANNUAL           **
CC   RAINFALL TOTAL). MISSING RAINFALL DATA POINTS ARE            **
CC   REPRESENTED BY A "-999".                                     **
CC                                                                **
CC   THE DATA IS STORED IN A MATRIX CALLED THE Z-MATRIX,          **
CC   AND THAT IS PARTITIONED INTO THE:                            **
CC              Y-VECTOR = A VECTOR OF THE TARGET STATION         **
CC              X-MATRIX = A MATRIX OF THE CONTROL STATIONS.      **
CC                                                                **
CC   THE MAXIMUM DIMENSIONS OF THE MATRICES ARE:                  **
CC              TARGET STATION          :  1                      **
CC              CONTROL STATIONS        :  25                     **
CC              OBSERVATIONS            :  100                    **
CC                                                                **
CC   NOTE THAT SOME OF THE ROUTINES WHICH ARE IN THIS             **
CC   PROGRAM WERE COPIED FROM THE PROGRAMS WRITTEN BY             **
CC   DR ROSS S. SPARKS.                                           **
CC*****************************************************************
CC*****************************************************************
CC
```

```
CC******* VARIABLES DECLARATIONS *******
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER NOBS, NSTAT, IV, DV, MAX
CC
**********************************************************************
**  IV     =  NUMBER OF CONTROL STATIONS.                             **
**  DV     =  NUMBER OF TARGET STATIONS.                              **
**  NSTAT  =  NUMBER OF ALL THE STATIONS - TARGET + CONTROL.          **
**  NOBS   =  NUMBER OF ALL THE OBSERVATIONS.                         **
**  MAX    =  MAXIMUM NUMBER OF SUBSETS TO BE CONSIDERED              **
**            AT A TIME.                                              **
**********************************************************************
CC
CC******* PARAMETER STATEMENTS *******
CC
      PARAMETER (NOBS   =   50)
      PARAMETER (NSTAT  =    5)
      PARAMETER (IV     =    4)
      PARAMETER (DV     =    1)
      PARAMETER (MAX    =   15)
      PARAMETER(MSUB    =  MAX + NSTAT)
CC
CC******* REAL VARIABLES *******
CC
      REAL Z(NOBS,NSTAT), ZOR(NOBS,NSTAT)
CC
**********************************************************************
**  Z    =  MATRIX OF ALL THE STATIONS.                              **
**  ZOR  =  MATRIX OF THE ORIGINAL DATA MATRIX Z.                    **
**********************************************************************
CC
      REAL Y(NOBS,DV), YOR(NOBS,DV), YST(NOBS,DV)
CC
**********************************************************************
**  Y    =  VECTOR OF THE TARGET STATION                             **
**  YOR  =  VECTOR OF THE ORIGINAL TARGET STATION.                   **
**  YST  =  VECTOR OF CONCURRENT OBSERVATIONS FROM THE TARGET        **
**          STATION.                                                 **
**********************************************************************
CC
      REAL X(NOBS,IV), XOR(NOBS,IV), XPST(NOBS,NSTAT)
CC
**********************************************************************
**  X     =  MATRIX OF CONTROL STATIONS.                             **
**  XOR   =  MATRIX OF THE ORIGINAL CONTROL STATIONS.                **
**  XPST  =  MATRIX OF CONCURRENT OBSERVATIONS                       **
**           (CONTROL STATIONS).                                     **
**********************************************************************
```

```
      REAL XP(NOBS,IV,MSUB)
CC
*****************************************************************
**    XP    = MATRIX OF SELECTED SUBSET OF X.                    **
*****************************************************************
CC
      REAL XPSTT(IV,NOBS)
CC
*****************************************************************
**    XPSTT = THE TRANSPOSED MATRIX OF XPST.                     **
*****************************************************************
CC
      REAL XPTXP(IV,IV), XPTYT(IV,DV)
CC
*****************************************************************
**    XPTXP = XPSTT * XPST                                       **
**    XPTYT = XPSTT * YST                                        **
*****************************************************************
CC
      REAL MEANY, MEANX(1,IV)
CC
*****************************************************************
**    MEANY = MEAN OF THE TARGET STATION.                        **
**    MEANX = MEANS OF CONTROL STATIONS.                         **
*****************************************************************
CC
      REAL BETA(IV,DV), BBETA(IV,DV,MSUB), BETA2
CC
*****************************************************************
**    BETA, BBETA = LEAST SQUARES PARAMETER ESTIMATES.           **
**    BETA2       = MEANX * BETA                                 **
*****************************************************************
CC
      REAL BBETAO(MSUB)
CC
*****************************************************************
**    BBETAO      = INTERCEPT TERM.                              **
*****************************************************************
CC
      REAL YSTBET(NOBS,DV), XSTBET(NOBS,DV)
CC
*****************************************************************
**    XSTBET  =  XPST * BETA                                     **
**    YSTBET  =  YST - XSTBET                                    **
*****************************************************************
CC
      REAL YSBETT(DV,NOBS), YTASTY(DV,DV)
CC
*****************************************************************
**    YSBETT  = THE TRANSPOSE OF YSTBET.                         **
**    YTASTY  = YSBETT * YST                                     **
*****************************************************************
```

```
      REAL XPBETA(NOBS), E(NOBS)
CC
*****************************************************************
**    XPBETA  =  XP * BETA                                     **
**    E       =  VECTOR OF RESIDUALS                           **
*****************************************************************
CC
      REAL WKSPCE(IV), CV1(MAX)
CC
      REAL PVAR(MSUB), VARP(MSUB), PREVAR(NSTAT)
CC
*****************************************************************
**    PVAR = PREDICTIVE VARIANCE.                              **
*****************************************************************
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER L, COL, ROW
      INTEGER NOROW
      INTEGER NROW(NOBS), IND(IV), NPRED(MSUB)
      INTEGER IMIN(MSUB), IMINJ(NSTAT)
CC
*****************************************************************
**    NOROW, NROW = NUMBER OF ROWS                             **
**    NPRED       = NUMBER OF CONTROL STATIONS IN THE SUBSET    **
**    IND         = INDICATOR VARIABLE                         **
*****************************************************************
CC
CC******* FORMAT STATEMENTS *******
CC
  10  FORMAT( 9F8.0)
  20  FORMAT(20F6.0/5F6.0)
CC
*****************************************************************
** - THE FOLLOWING DO-LOOP READS AND WRITES A MATRIX OF ALL     **
**    THE RAINFALL STATIONS AND ALL THE OBSERVATIONS            **
*****************************************************************
CC
      DO 100 ROW = 1, NOBS
         READ(13,10) (Z(ROW,COL), COL = 1, NSTAT)
         WRITE(6,10) (Z(ROW,COL), COL = 1, NSTAT)
 100  CONTINUE
*****************************************************************
** - IN THE FOLLOWING DO-LOOP, WE PARTITION THE Z-MATRIX        **
**    INTO THE VECTOR OF THE TARGET STATION AND THE MATRIX      **
**    OF CONTROL STATIONS.                                      **
*****************************************************************
CC
      DO 120 ROW = 1, NOBS
         Y(ROW,1) = Z(ROW,1)
         DO 110 COL = 2, NSTAT
            X(ROW,COL-1) = Z(ROW,COL)
 110     CONTINUE
 120  CONTINUE
```

```
      CALL COPY(Z,NOBS,NSTAT,ZOR,NOBS,NSTAT,NOBS,NSTAT)
CC
**********************************************************************
**    I = 0 IMPLIES THAT WE ARE NOT CONSIDERING ANY SUBSET          **
**********************************************************************
CC
      I = 0
      J = 0
CC
      DO 130 L = 1, IV
 130  IND(L) = 0
      L = 1
 140  IND(L) = 1
CC
**********************************************************************
**    COUNT THE NUMBER OF SUBSETS WHICH ARE BEING CONSIDERED        **
**********************************************************************
CC
      I = I + 1
CC
      NPRED(I) = 0
      DO 160 COL = 1, IV
         IF (IND(COL) .LT. 1) GO TO 160
            NPRED(I) = NPRED(I) + 1
            DO 150 ROW = 1, NOBS
               XP(ROW,NPRED(I),I) = X(ROW,COL)
               Z(ROW,NPRED(I)+1) = X(ROW,COL)
 150        CONTINUE
 160  CONTINUE
CC
**********************************************************************
** - CHECK IF THERE ARE MISSING OBSERVATIONS IN ANY OF THE          **
**    INCLUDED PREDICTORS.  IF THERE ARE, LET ALL THE               **
**    OBSERVATIONS IN THAT ROW BE EQUAL TO "-999".                  **
**********************************************************************
CC
      NROW(I) = 0
      DO 200 ROW = 1, NOBS
CC
         Z(ROW,1) = Y(ROW,1)
         COL = 1
 170     IF (COL .LE. (NPRED(I)+1)) THEN
            IF (Z(ROW,COL) .EQ. -999.0) THEN
               DO 180 COL = 1, NPRED(I)+1
                  Z(ROW,COL) = -999.0
 180           CONTINUE
            ENDIF
         COL = COL + 1
         GO TO 170
         ENDIF
```

```
**********************************************************************
** - ELIMINATE FROM Z ALL THOSE ROWS WHICH HAVE "-999.0"            **
**   ENTRIES. COUNT THE REMAINING NUMBER OF ROWS.                   **
**********************************************************************
CC
         COL = 1
         IF (Z(ROW,COL) .NE. -999.0) THEN
            NROW(I) = NROW(I) + 1
            YOR(NROW(I),COL) = Z(ROW,COL)
            DO 190 COL = 2, NPRED(I)+1
               XOR(NROW(I),COL-1) = Z(ROW,COL)
 190        CONTINUE
         ENDIF
CC
 200  CONTINUE
CC
**********************************************************************
** - CHECK FOR SUFFICIENT CONCURRENT RECORDS TO FIT A                **
**   REGRESSION MODEL                                               **
**********************************************************************
CC
      IF (NROW(I) .LT. IV+2) GO TO 320
CC
**********************************************************************
**   - WHEN CALLED, THESE SUBROUTINES FIND THE LEAST SQUARES        **
**     ESTIMATES AND THE COVARIANCE MATRIX FOR A*, WHERE            **
**    A* = X*(X*TX*)-1(X*T)                                         **
**********************************************************************
CC
      CALL CNTRAL(XPST,NOBS,IV,XOR,NOBS,IV,NROW(I),NPRED(I))
      CALL CNTRAL(YST,NOBS,1,YOR,NOBS,1,NROW(I),DV)
      CALL TRANP(XPST,NOBS,IV,XPSTT,IV,NOBS,NROW(I),NPRED(I))
      CALL MULT(XPSTT,IV,NOBS,XPST,NOBS,IV,XPTXP,IV,IV,NPRED(I),
     &          NROW(I), NPRED(I))
      CALL MULT(XPSTT,IV,NOBS,YST,NOBS,1,XPTYT,IV,1,NPRED(I),
     &          NROW(I),DV)
      CALL F04ABE(XPTXP,IV,XPTYT,IV,NPRED(I),1,BETA,IV,WKSPCE,E,
     &          NROW(I),IFAIL)
      CALL MULT(XPST,NOBS,IV,BETA,IV,1,XSTBET,NOBS,1,NROW(I),
     &          NPRED(I), DV)
      CALL DIFFS(YSTBET,NOBS,1,YST,NOBS,1,XSTBET,NOBS,1,NROW(I),
     &          DV)
      CALL TRANP(YSTBET,NOBS,1,YSBETT,1,NOBS,NROW(I),DV)
      CALL MULT(YSBETT,1,NOBS,YST,NOBS,1,YTASTY,1,1,DV,NROW(I),
     &          DV)
CC
      DO 210 COL = 1, NPRED(I)
         BBETA(COL,1,I) = BETA(COL,1)
 210  CONTINUE
```

```
      MEANY = 0.0
      DO 220 ROW = 1, NROW(I)
         MEANY = MEANY + YOR(ROW,1)
 220  CONTINUE
      MEANY = MEANY / NROW(I)
CC
      BETA2 = 0.0
      DO 240 COL = 1, NPRED(I)
         MEANX(1,COL) = 0.0
         DO 230 ROW = 1, NROW(I)
            MEANX(1,COL) = MEANX(1,COL) + XOR(ROW,COL)
 230     CONTINUE
         MEANX(1,COL) = MEANX(1,COL) / NROW(I)
         BETA2 = BETA2 + MEANX(1,COL) * BETA(COL,1)
 240  CONTINUE
CC
*************************************************************************
**    CALCULATE THE INTERCEPT TERM                                     **
*************************************************************************
CC
      BBETA0(I) = MEANY - BETA2
CC
      CV1(I) = YTASTY(1,1) / (NROW(I) - IV - 1)
CC
      PVAR(I) = (NROW(I) * NPRED(I)) * CV1(I) / NROW(I)
CC
*************************************************************************
**   - CHECK FOR ALL SELECTED SUBSETS WHICH HAVE ONLY ONE               **
**    CONTROL STATION.                                                 **
*************************************************************************
CC
      IF (NPRED(I) .EQ. 1) THEN
CC
         J = J + 1
         JJ = MAX + J
         NPRED(JJ) = NPRED(I)
         PVAR(JJ) = PVAR(I)
         BBETA0(JJ) = BBETA0(I)
CC
         DO 260 COL = 1, NPRED(JJ)
            BBETA(COL,1,JJ) = BBETA(COL,1,I)
            DO 250 ROW = 1, NOBS
               XP(ROW,COL,JJ) = XP(ROW,COL,I)
 250        CONTINUE
 260     CONTINUE
CC
      ENDIF
```

```
**********************************************************************
** - CHECK IF THE NUMBER OF SUBSETS EXCEEDS THE REQUIRED            **
**     NUMBER.                                                      **
**********************************************************************
CC
      IF (I .EQ. MAX) THEN
CC
**********************************************************************
** - COPY THE PREDICTIVE VARIANCES TO ANOTHER VECTOR.               **
** - IMIN(.) KEEPS TRACK OF THE NUMBER OF SUBSETS.                  **
**********************************************************************
CC
         DO 270 ROW = 1, MAX
            VARP(ROW) = PVAR(ROW)
            IMIN(ROW)= ROW
 270     CONTINUE
CC
**********************************************************************
**    SWOP THE NEW PREDICTIVE VARIANCE VECTOR                       **
**********************************************************************
CC
         CALL SWOPPY(VARP,IMIN,MAX)
CC
**********************************************************************
** - THE FOLLOWING STATEMENTS EXCHANGE TWO DIFFERENT ROW            **
**    NUMBERS OF THE PREDICTIVE VARIANCE VECTOR, IN SUCH A WAY      **
**    THAT THE ROW NUMBER MAX CORRESPONDS TO THE LARGEST NUMBER     **
**    IMIN(MAX).                                                    **
**********************************************************************
CC
         DO 290 ROW = 1, MAX
            IF (IMIN(ROW) .EQ. MAX) NOROW = ROW
 290     CONTINUE
CC
         IF (MAX .NE. IMIN(MAX)) THEN
            IMIN(NOROW) = IMIN(MAX)
            IMIN(MAX) = MAX
CC
            NPRED(IMIN(NOROW)) = NPRED(IMIN(MAX))
CC
            TEMMM = PVAR(IMIN(NOROW))
            PVAR(IMIN(NOROW)) = PVAR(IMIN(MAX))
            PVAR(IMIN(MAX)) = TEMMM
CC
            DO 310 COL = 1, NPRED(IMIN(MAX))
               BBETA(COL,1,IMIN(NOROW)) = BBETA(COL,1,IMIN(MAX))
               DO 300 ROW = 1, NOBS
                  XP(ROW,COL,IMIN(NOROW)) = XP(ROW,COL,IMIN(MAX))
 300           CONTINUE
 310        CONTINUE
CC
            BBETA0(IMIN(NOROW)) =BBETA0(IMIN(MAX))
CC
         ENDIF
```

```
**********************************************************************
** - THE FOLLOWING STATEMENT MAKES SURE THAT THE NUMBER OF          **
**   SUBSETS DOES NOT EXCEED THE REQUIRED NUMBER = MAX.             **
**********************************************************************
CC
      I = MAX-1
CC
      ENDIF
CC
 320  DO 330 L = 1, IV
         IF (IND(L) .EQ. 0) GO TO 140
 330  IND(L) = 0
CC
**********************************************************************
** - WHEN ALL THE SUBSETS HAVE BEEN CONSIDERED, NOW WE SWOP         **
**   THE ONE-PREDICTOR SUBSETS.                                     **
**********************************************************************
CC
      DO 340 ROW = 1, IV
         PREVAR(ROW) = PVAR(MAX+ROW)
         IMINJ(ROW) = MAX+ROW
 340  CONTINUE
CC
      CALL SWOPPY(PREVAR,IMINJ,IV)
CC
      DO 350 ROW = 1, IV
         IMIN(MAX+ROW) = IMINJ(ROW)
 350  CONTINUE
CC
      DO 360 ROW = 1, MAX+IV
         PRINT*, VARP(ROW), PVAR(IMIN(ROW)), IMIN(ROW),
     &           BBETA0(IMIN(ROW))
 360  CONTINUE
CC
**********************************************************************
** - IN THE FOLLOWING DO-LOOP, WE CHECK FOR MISSING                 **
**   OBSERVATIONS AND THEN ESTIMATE THEM. WE FIRST TRY TO           **
**   ESTIMATE THE MISSING OBSERVATIONS BY CONSIDERING THAT          **
**   SUBSET WHICH HAS THE SMALLEST PREDICTIVE VARIANCE.             **
**   IF WE CAN'T ESTIMATE WITH IT, THEN WE CONSIDER                 **
**   THE FOLLOWING SUBSET WITH THE NEXT SMALLEST PREDICTIVE         **
**   VARIANCE, ETC.                                                 **
**********************************************************************
CC
      DO 390 ROW = 1, NOBS
          IF (Y(ROW,1) .EQ. -999.0) THEN
             J = 0
             II = 0
CC
  370        II = II + 1
             IF (II .EQ. MAX) GO TO 370
```

```
               XPBETA(ROW) = 0.0
               DO 380 COL = 1, NPRED(IMIN(II))
CC
                  X(ROW,COL) = XP(ROW,COL,IMIN(II))
CC
                  IF (X(ROW,COL) .EQ. -999.0) GO TO 370
CC
                  BETA(COL,1) = BBETA(COL,1,IMIN(II))
CC
                  XPBETA(ROW) = X(ROW,COL)
     &                        * BETA(COL,1) + XPBETA(ROW)
 380           CONTINUE
CC
               Y(ROW,1) = BBETA0(IMIN(II)) + XPBETA(ROW)
            ENDIF
CC
*****************************************************************
** - COPY THE ESTIMATED Y-VECTOR INTO THE FIRST COLUMN OF      **
**   THE Z-MATRIX.                                             **
*****************************************************************
CC
            Z(ROW,1) = Y(ROW,1)
CC
 390     CONTINUE
CC
         CALL PMAT(Z,NOBS,NSTAT,NOBS,NSTAT)
CC
         STOP
         DEBUG SUBCHK
         END
```

PROGRAM 3

```
CC*****************************************************************
CC******                                                          **
CC******  A PROGRAM TO ESTIMATE MISSING RAINFALL DATA BY          **
CC******  MAKING USE OF ONE OF THE SELECTION OF VARIABLES         **
CC******  PROCEDURES - FORWARD SELECTION.                         **
CC******                                                          **
CC*****************************************************************
CC*****************************************************************
CC                                                                **
CC   THIS PROGRAM IS USED TO ESTIMATE DATA MATRICES WHICH         **
CC   CONTAIN MISSING OBSERVATIONS IN ALMOST ALL THE RAINFALL      **
CC   STATIONS.                                                    **
CC                                                                **
CC   THE STATIONS ARE READ AS ONE BIG MATRIX WHICH CONSISTS       **
CC   OF A COLUMN OF THE TARGET STATION - WHICH, IN THIS           **
CC   PROGRAM, SHOULD ALWAYS BE THE FIRST COLUMN, AND THE          **
CC   REMAINING COLUMNS BEING THE MATRIX OF CONTROL STATIONS.      **
CC                                                                **
CC   EACH ROW OF DATA REPRESENTS AN OBSERVATION (ANNUAL           **
CC   RAINFALL TOTAL). MISSING RAINFALL DATA POINTS ARE            **
CC   REPRESENTED BY A "-999".                                     **
CC                                                                **
CC   THE DATA IS STORED IN A MATRIX CALLED THE Z-MATRIX,          **
CC   AND THAT IS PARTITIONED INTO THE:                            **
CC              Y-VECTOR = A VECTOR OF THE TARGET STATION         **
CC              X-MATRIX = A MATRIX OF THE CONTROL STATIONS.      **
CC                                                                **
CC   THE MAXIMUM DIMENSIONS OF THE MATRICES ARE:                  **
CC              TARGET STATION          :  1                      **
CC              CONTROL STATIONS        :  25                     **
CC              OBSERVATIONS            :  100                    **
CC                                                                **
CC   NOTE THAT SOME OF THE ROUTINES WHICH ARE IN THIS             **
CC   PROGRAM WERE COPIED FROM THE PROGRAMS WRITTEN BY             **
CC   DR ROSS S. SPARKS.                                           **
CC*****************************************************************
CC*****************************************************************
```

```
CC******* VARIABLES DECLARATIONS *******
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER NOBS, NSTAT, NCON, NTARG, MAX
CC
*****************************************************************
**  NCON   =  Number of control stations.                      **
**  NTARG  =  Number of target stations.                       **
**  NSTAT  =  Number of all the stations - target + control.   **
**  NOBS   =  Number of all the observations.                  **
**  MAX    =  Indicator                                        **
*****************************************************************
CC
CC******* PARAMETER STATEMENTS *******
CC
      PARAMETER (NOBS     =   50)
      PARAMETER (NSTAT    =    5)
      PARAMETER (NCON     =    4)
      PARAMETER (NTARG    =    1)
CC
CC******* REAL VARIABLES *******
CC
      REAL Z(NOBS,NSTAT), ZOR(NOBS,NSTAT)
CC
*****************************************************************
**  Z    =  MATRIX OF ALL THE STATIONS.                        **
**  ZOR  =  MATRIX OF THE ORIGINAL DATA MATRIX Z.              **
*****************************************************************
CC
      REAL Y(NOBS,NTARG), YOR(NOBS,NTARG), YST(NOBS,NTARG)
CC
*****************************************************************
**  Y    =  VECTOR OF THE TARGET STATION                       **
**  YOR  =  VECTOR OF THE ORIGINAL TARGET STATION.             **
**  YST  =  VECTOR OF CONCURRENT OBSERVATIONS FROM THE TARGET  **
**          STATION.                                           **
*****************************************************************
CC
      REAL X(NOBS,NCON), XOR(NOBS,NCON), XPST(NOBS,NSTAT)
CC
*****************************************************************
**  X     =  MATRIX OF CONTROL STATIONS.                       **
**  XOR   =  MATRIX OF THE ORIGINAL CONTROL STATIONS.          **
**  XPST  =  MATRIX OF CONCURRENT OBSERVATIONS                 **
**           (CONTROL STATIONS).                               **
*****************************************************************
CC
      REAL XP(NOBS,NCON), PATCH(NOBS)
CC
*****************************************************************
**   XP   = MATRIX OF SELECTED SUBSET OF X.                    **
*****************************************************************
```

```
      REAL XPSTT(NCON,NOBS), XPT(NCON,NOBS)
CC
*****************************************************************
**    XPSTT = THE TRANSPOSED MATRIX OF XPST.                   **
**    XPT = THE TRANSPOSED MATRIX OF XP.                       **
*****************************************************************
CC
      REAL XPTXP(NCON,NCON), XPTYT(NCON,NTARG)
      REAL XPTXPI(NCON,NCON)
CC
*****************************************************************
**    XPTXP    =  XPSTT * XPST                                 **
**    XPTYT    =  XPSTT * YST                                  **
**    XPTXPI   =  THE INVERSE MATRIX OF XPTXP                  **
*****************************************************************
CC
      REAL MEANY, MEANX(1,NCON), MEANXT(NCON,1)
CC
*****************************************************************
**    MEANY = MEAN OF THE TARGET STATION.                      **
**    MEANX    =  MEANS OF CONTROL STATIONS.                   **
**    MEANXT   =  THE TRANSPOSED VECTOR OF MEANX.              **
*****************************************************************
CC
      REAL BETA(NCON,NTARG), BBETA(NCON,NTARG,0:NSTAT), BETA2
CC
*****************************************************************
**    BETA, BBETA = LEAST SQUARES PARAMETER ESTIMATES.         **
**    BETA2       = MEANX * BETA                               **
*****************************************************************
CC
      REAL BBETA0(0:NSTAT), BETA0
CC
*****************************************************************
**    BBETA0, BETA0 = INTERCEPT TERM.                          **
*****************************************************************
CC
      REAL YSTBET(NOBS,NTARG), XSTBET(NOBS,NTARG)
CC
*****************************************************************
**    XSTBET   =  XPST * BETA                                  **
**    YSTBET   =  YST - XSTBET                                 **
*****************************************************************
CC
      REAL YSBETT(NTARG,NOBS), YTASTY(NTARG,NTARG)
CC
*****************************************************************
**    YSBETT   = THE TRANSPOSE OF YSTBET.                      **
**    YTASTY   = YSBETT * YST                                  **
*****************************************************************
```

```
      REAL XPBETA(NOBS,NTARG), E(NOBS)
      REAL C(NOBS,NCON,0:NSTAT)
CC
*****************************************************************
**    XPBETA  =  XP * BETA                                     **
**    E       =  VECTOR OF RESIDUALS                           **
**    C       =  MATRIX OF DIFFERENT SELECTED CONTROL STATIONS **
*****************************************************************
CC
      REAL WKSPCE(NCON), CV1(NSTAT), HAT(NSTAT)
CC
      REAL PVAR(NSTAT), VARC, V(0:NSTAT)
CC
*****************************************************************
**    PVAR = PREDICTIVE VARIANCE.                              **
*****************************************************************
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER L, COL, ROW
      INTEGER NOROW, NCHOS(NSTAT)
      INTEGER NROW, IND(NCON), NPRED
CC
*****************************************************************
**    NOROW, NROW = NUMBER OF ROWS                             **
**    NPRED       = NUMBER OF CONTROL STATIONS IN THE SUBSET    **
**    IND         = INDICATOR VARIABLE                         **
**    NCHOS       = NUMBER OF INCLUDED CONTROL STATIONS        **
*****************************************************************
CC
CC******* FORMAT STATEMENTS *******
CC
  10  FORMAT( 9F8.0)
  20  FORMAT(20F6.0/5F6.0)
CC
*****************************************************************
** - THE FOLLOWING DO-LOOP READS AND WRITES A MATRIX OF ALL    **
**    THE RAINFALL STATIONS AND ALL THE OBSERVATIONS           **
*****************************************************************
CC
      DO 100 ROW = 1, NOBS
         READ(13,10) (Z(ROW,COL), COL = 1, NSTAT)
         WRITE(6,10) (Z(ROW,COL), COL = 1, NSTAT)
 100  CONTINUE
```

```
**********************************************************************
** - IN THE FOLLOWING DO-LOOP, WE PARTITION THE Z-MATRIX            **
**   INTO THE VECTOR OF THE TARGET STATION AND THE MATRIX           **
**   OF CONTROL STATIONS.                                           **
**********************************************************************
CC
      DO 120 ROW = 1, NOBS
         Y(ROW,1) = Z(ROW,1)
         DO 110 COL = 2, NSTAT
            X(ROW,COL-1) = Z(ROW,COL)
 110     CONTINUE
 120  CONTINUE
CC
       CALL COPY(Z,NOBS,NSTAT,ZOR,NOBS,NSTAT,NOBS,NSTAT)
CC
 130          DO 160 ROW = 1, NOBS
CC
                 COL = 1
  140            IF (COL .LE. NSTAT) THEN
                    IF (Z(ROW,COL) .EQ. -999.0) THEN
                       DO 150 COL = 1, NSTAT
                          Z(ROW,COL) = -999.0
  150                  CONTINUE
                    ENDIF
                    COL = COL + 1
                    GO TO 140
                 ENDIF

  160         CONTINUE

              NROW = 0
              DO 170 ROW = 1, NOBS
                 COL = 1
                 IF (Z(ROW,COL) .NE. -999.0) THEN
                    NROW = NROW + 1
                    YOR(NROW,COL) = Y(ROW,COL)
                 ENDIF
  170         CONTINUE
CC
**********************************************************************
**  - Check if there are sufficient concurrent records to fit       **
**    a regression model.                                           **
**********************************************************************
CC
              IF (NROW .LT. (NCON + 2)) GO TO 210
CC
**********************************************************************
**  - Compute the mean for the "degenerate model".                  **
**********************************************************************
CC
              MEANY = 0.0
              DO 180 ROW = 1, NROW
                 MEANY = MEANY + YOR(ROW,1)
  180         CONTINUE
              MEAN = MEANY / NROW
```

```
**********************************************************************
**  - Compute the predictive variance for the "degenerate           **
**    model".                                                       **
**********************************************************************
CC
              VARC = 0.0
              DO 190 ROW = 1, NROW
                 VARC = VARC + (YOR(ROW,1) - MEAN) ** 2
 190          CONTINUE
              V(0) = VARC / (NROW - 1)

              PRINT*, 'THIS IS THE VALUE OF V(0) WHICH IS = ',  V(0)

       DO 450 K = 1, NOBS

          IF (Y(K,1) .EQ. -999.0) THEN
CC
              I = 0
CC
              DO 200 L = 1, NCON
 200          IND(L) = 0
CC
 210          I = I + 1
CC
              V(I) = 99999999.0
CC
              L = 1
CC
 220          NPRED = 0
CC
              IF (IND(L) .EQ. 0) THEN
                 NPRED = NPRED + 1
                 DO 230 ROW = 1, NOBS
                    XP(ROW,NPRED) = X(ROW,L)
 230             CONTINUE
CC
**********************************************************************
**  - Check if the currently selected control station is            **
**    capable of estimating the k-th missing observation.           **
**    If not, consider selecting the  next control station,         **
**    otherwise include it to the already selected stations.        **
**********************************************************************
CC
                 IF (XP(K,1) .EQ. -999.0) GO TO 260
CC
                 DO 250 COL = 1, NCON
                    IF (IND(COL) .NE. 0) THEN
                       NPRED = NPRED + 1
                       DO 240 ROW = 1, NOBS
                          XP(ROW,NPRED) = C(ROW,NPRED-1,I-1)
  240                  CONTINUE
                    ENDIF
  250            CONTINUE
```

```
            ELSE
CC
*****************************************************************
**  - Check if all the stations have been considered.          **
*****************************************************************
CC
 260            IF (L .GE. NCON) GO TO 420
                L = L + 1
                GO TO 220
CC
            ENDIF
CC
            DO 270 ROW = 1, NOBS
               Z(ROW,1) = Y(ROW,1)
               DO 280 COL = 1, NPRED
                  Z(ROW,COL+1) = XP(ROW,COL)
 280           CONTINUE
 270        CONTINUE
CC
*****************************************************************
**  - Check if there are missing observations in any of the    **
**    included predictors.  If there are, let all the          **
**    observations in that row be equal to "-999".             **
*****************************************************************
CC
            DO 310 ROW = 1, NOBS
CC
               COL = 1
 290           IF (COL .LE. (NPRED+1)) THEN
                  IF (Z(ROW,COL) .EQ. -999.0) THEN
                     DO 300 COL = 1, NPRED+1
                        Z(ROW,COL) = -999.0
 300                 CONTINUE
                  ENDIF
                  COL = COL + 1
                  GO TO 290
               ENDIF
CC
 310        CONTINUE
CC
*****************************************************************
**  - Eliminate from Z all those rows which have "-999.0"      **
**    entries.  Count the remaining number of rows.            **
*****************************************************************
CC
            NROW = 0
            DO 330 ROW = 1, NOBS
               COL = 1
               IF (Z(ROW,COL) .NE. -999.0) THEN
                  NROW = NROW + 1
                  YOR(NROW,COL) = Z(ROW,COL)
                  DO 320 COL = 2, NPRED+1
                     XOR(NROW,COL-1) = Z(ROW,COL)
 320              CONTINUE
               ENDIF
 330        CONTINUE
```

```
**********************************************************************
**  - Check for sufficient concurrent records to fit a              **
**    regression model.                                             **
**********************************************************************
CC
            IF (NROW .LT. (NCON + 2)) GO TO 410
CC
**********************************************************************
**  - When called, these subroutines find the least squared         **
**    estimates and eh covariance matrix for A*, where A* is        **
**    defined as A* = X*(X*TX*)-1(X*T).                             **
**********************************************************************
CC
            CALL CNTRAL(XPST,NOBS,NCON,XOR,NOBS,NCON,NROW,NPRED)
            CALL CNTRAL(YST,NOBS,NTARG,YOR,NOBS,NTARG,NROW,NTARG)
            CALL TRANP(XPST,NOBS,NCON,XPSTT,NCON,NOBS,NROW,NPRED)
            CALL MULT(XPSTT,NCON,NOBS,XPST,NOBS,NCON,XPTXP,NCON,
     &                NCON,NPRED,NROW,NPRED)
            CALL IDVERT(XPTXP,NCON,XPTXPI,NPRED)
            CALL MULT(XPSTT,NCON,NOBS,YST,NOBS,NTARG,XPTYT,NCON,
     &                NTARG,NPRED,NROW,NTARG)
            CALL MULT(XPTXPI,NCON,NCON,XPTYT,NCON,NTARG,BETA,NCON,
     &                NTARG,NPRED,NPRED,NTARG)
            CALL MULT(XPST,NOBS,NCON,BETA,NCON,NTARG,XSTBET,NOBS,
     &                NTARG,NROW,NPRED,NTARG)
            CALL DIFFS(YSTBET,NOBS,NTARG,YST,NOBS,NTARG,XSTBET,
     &                NOBS,NTARG,NROW,NTARG)
            CALL TRANP(YSTBET,NOBS,NTARG,YSBETT,NTARG,NOBS,NROW,
     &                NTARG)
            CALL MULT(YSBETT,NTARG,NOBS,YST,NOBS,NTARG,YTASTY,
     &                NTARG,NTARG,NTARG,NROW,NTARG)
CC
            MEANY = 0.0
            DO 340 ROW = 1, NROW
               MEANY = MEANY + YOR(ROW,1)
 340        CONTINUE
            MEANY = MEANY / NROW
CC
            BETA2 = 0.0
            DO 360 COL = 1, NPRED
               MEANX(1,COL) = 0.0
               DO 350 ROW = 1, NROW
                  MEANX(1,COL) = MEANX(1,COL) + XOR(ROW,COL)
 350           CONTINUE
               MEANX(1,COL) = MEANX(1,COL) / NROW
               BETA2 = BETA2 + MEANX(1,COL) * BETA(COL,1)
 360        CONTINUE
CC
**********************************************************************
**  - Compute the intercept term.                                   **
**********************************************************************
CC
            BETA0 = MEANY - BETA2
```

```
            CALL TRANP(MEANX,NTARG,NCON,MEANXT,NCON,NTARG,NTARG,
     &                 NPRED)
            CALL TRANP(XP,NOBS,NCON,XPT,NCON,NOBS,NOBS,NPRED)

            CV1(I) = YTASTY(1,1) / (NROW - NCON - 1)
CC
            HAT(I) = 0.0
            DO 370 P = 1, NPRED
            DO 370 J = 1, NPRED
CC
            HAT(I) = HAT(I)
     &                + ((XP(K,P) - MEANX(1,P)) * XPTXPI(P,J)
     &                * (XPT(J,K) - MEANXT(J,1)))
CC
 370        CONTINUE
CC
**********************************************************************
**   - Compute the value of the predictive variance.                **
**********************************************************************
CC
            PVAR(I) = (1.0 + (1.0 / FLOAT(NOBS)) + HAT(I))
     &                                          * CV1(I)
CC
**********************************************************************
**  - Check if the currently calculated predictive variance         **
**    is less than the present smallest predictive variance         **
**    and substitute the old one by the current one.                **
**********************************************************************
CC
            IF (V(I) .GT. PVAR(I)) THEN
               V(I) = PVAR(I)
               NCHOS(I) = NPRED
               MAX = L
CC
**********************************************************************
**  - MAX is an indicator of those control stations which           **
**    led to the smallest predictive variance.                      **
**********************************************************************
CC
               BBETA0(I) = BETA0
CC
**********************************************************************
**  - This DO-LOOP copies the selected control stations into        **
**    a new matrix called C.                                        **
**********************************************************************
CC
               DO 390 ROW = 1, NOBS
                  DO 380 COL = 1, NCHOS(I)
                     C(ROW,COL,I) = XP(ROW,COL)
 380                 CONTINUE
 390           CONTINUE
```

```
                  DO 400 COL = 1, NCHOS(I)
                     BBETA(COL,1,I) = BETA(COL,1)
 400              CONTINUE
CC
               ENDIF
CC
*****************************************************************
**  - Check if all the control stations have been considered.  **
*****************************************************************
CC
 410           IF (L .LT. NCON) THEN
                  L = L + 1
                  GO TO 220
               ENDIF
CC
 420           L = MAX
               IND(L) = 1
CC
               PRINT*, 'V(I) = ', V(I), 'I = ', I
CC
*****************************************************************
**  - Check if the predictive variance obtained for the subset **
**    which has (I-1) predictors is smaller than the one       **
**    obtained for the subset with (I) predictors.  Estimate   **
**    the missing values by using the (I-1) predictors.        **
*****************************************************************
CC
               IF (V(I-1) .LT. V(I)) THEN
CC
*****************************************************************
**  - If the variance for the "degenerate model" is smaller    **
**    than the variance obtained when one control station      **
**    was considered, then estimate the missing observation    **
**    by the mean of the target station.                       **
*****************************************************************
CC
                  IF (I .EQ. 1) THEN
                     PATCH(K) = MEAN
                     GO TO 450
                  ENDIF
CC
                  DO 430 COL = 1, NCHOS(I-1)
                     XP(K,COL) = C(K,COL,I-1)
                     BETA(COL,1) = BBETA(COL,1,I-1)
                     XPBETA(K,1) = XPBETA(K,1) + XP(K,COL)
     &                                         * BETA(COL,1)
 430              CONTINUE
CC
                  BETA0 = BBETA0(I-1)
                  PATCH(K) = BETA0 + XPBETA(K,1)
                  GO TO 450
```

```
               ELSEIF (I .EQ. NCON) THEN
                    DO 440 COL = 1, NCHOS(I)
                       XP(K,COL) = C(K,COL,I)
                       BETA(COL,1) = BBETA(COL,1,I)
                       XPBETA(K,1) = XPBETA(K,1) + XP(K,COL)
     &                                     * BETA(COL,1)
 440                 CONTINUE
CC
                    BETA0 = BBETA0(I)
CC
                    PATCH(K) = BETA0 + XPBETA(K,1)
                    GO TO 450
               ENDIF
CC
             GO TO 210
CC
             ENDIF
CC
 450   CONTINUE
CC
*****************************************************************
**  - Copy the estimated target station into the first column  **
**    of the Z-matrix.                                         **
*****************************************************************
CC
       DO 460 ROW = 1, NOBS
CC
          IF (Y(ROW,1) .EQ. -999.0) THEN
             Z(ROW,1) = PATCH(ROW)
          ENDIF
CC
 460   CONTINUE
CC
       CALL PMAT(Z,NOBS,NSTAT,NOBS,NSTAT)
CC
       STOP
       END
```

PROGRAM 4

```
CC*****************************************************************
CC******                                                          **
CC******  A PROGRAM TO ESTIMATE MISSING ANNUAL RAINFALL DATA      **
CC******  BY MAKING USE OF THE EM ALGORITHM                       **
CC******  - "Standard" EM algorithm.                              **
CC******                                                          **
CC*****************************************************************
CC*****************************************************************
CC                                                                **
CC   This program is used to estimate data matrices which         **
CC   contain missing observations in almost all the rainfall      **
CC   stations.                                                    **
CC                                                                **
CC   The stations are read as one big matrix which consists       **
CC   of a column of the TARGET station and the remaining          **
CC   columns being the matrix of control stations.                **
CC                                                                **
CC   Each row of data represents an observation (Annual           **
CC   rainfall total). Missing rainfall data points are            **
CC   represented by a "-999".                                     **
CC                                                                **
CC   The data is stored in a matrix called the Z-matrix,          **
CC   which can be partitioned into the:                           **
CC              Y-vector = A vector of the target station         **
CC              X-matrix = A matrix of the control stations.      **
CC                                                                **
CC   The maximum dimensions of the matrices are:                  **
CC              Target station          :  1                      **
CC              Control stations        :  25                     **
CC              Observations (annual)   :  100                    **
CC                                                                **
CC                                                                **
CC   Note that some of the routines which are in this             **
CC   program were copied from the programs written by             **
CC   DR ROSS S. SPARKS.                                          **
CC                                                                **
CC*****************************************************************
CC*****************************************************************
```

```
CC******* VARIABLES DECLARATION *******
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER NOBS, NSTAT, IV, DV
CC
CC------- PARAMETER STATEMENTS ----
CC
      PARAMETER (NOBS  =  28)
      PARAMETER (NSTAT =   6)
      PARAMETER (IV    =   5)
      PARAMETER (DV    =   1)
CC
*****************************************************************
**  NOBS   =   Number of all the observations.                 **
**  NSTAT  =   Number of all the stations - Target + Control   **
**  IV     =   Number of control stations                      **
**  DV     =   Number of target stations                       **
*****************************************************************
CC
CC******* REAL VARIABLES *******
CC
      REAL Z(NOBS,NSTAT), ZOR(NOBS,NSTAT), ZOR1(NOBS,NSTAT)
      REAL ZORR(NOBS,NSTAT), ZOROR(NOBS,NSTAT)
CC
*****************************************************************
**  Z      =  Matrix of all the stations.                      **
**  ZOR    =  Matrix of the original Z-matrix.                 **
**  ZOR1   =  Matrix of the original Z-matrix.                 **
**  ZORR   =  Matrix of the original Z-matrix.                 **
**  ZOROR  =  Matrix of the original Z-matrix.                 **
*****************************************************************
CC
      REAL TMAT(NOBS,NSTAT), PATCH(NOBS,NSTAT,500)
CC
*****************************************************************
**  TMAT   = A temporary matrix with all the estimated values. **
**  PATCH  = A matrix with all the estimated values.           **
*****************************************************************
CC
      REAL X(NOBS,IV), XOR(NOBS,IV), XXREG(NOBS,IV)
      REAL XX(NOBS,IV), XXOR(NOBS,NSTAT)
CC
*****************************************************************
**  X      = A matrix of all the stations.                     **
**  XOR    = A matrix of original X matrix.                    **
**  XXOR   = A matrix containing the "relevant" control        **
**           stations only.                                    **
**  XXREG  = A standardized XXOR matrix.                       **
**  XX     = A matrix of the "relevant" control stations       **
**           (all observations).                               **
*****************************************************************
```

```
      REAL Y(NOBS,DV), YREG(NOBS,DV), YYOR(NOBS,DV)
CC
**********************************************************************
**  Y      = A vector of the target station.                          **
**  YREG   = A vector of standardized Y vector.                       **
**  YYOR   = A vector of the original Y vector.                       **
**********************************************************************
CC
      REAL XXT(IV,NOBS), XXTXX(IV,IV), XXREGT(IV,NOBS)
CC
**********************************************************************
**  XXT      = A transposed XX matrix.                                **
**  XXREG    = A transposed XXREG matrix.                             **
**  XXTXX    = XXT * XX                                               **
**********************************************************************
CC
      REAL XRTYR(IV,DV), XXTYY(IV,DV), XRTXR(IV,IV)
CC
**********************************************************************
**  XRTYR  = XXREGT * YREG                                            **
**  XXTYY  = XXT * Y                                                  **
**  XRTXR  = XXREGT * XXREG                                           **
**********************************************************************
CC
      REAL MEANXX(DV,IV), MEANYY(1,1), WKSPCE(IV), E(NOBS)
CC
**********************************************************************
**  MEANXX  = The means of control stations.                          **
**  MEANYY  = The mean of the target station.                         **
**  E       = A vector of the residuals.                              **
**********************************************************************
CC
      REAL UNIT(NOBS,DV), UNITT(DV,NOBS)
CC
**********************************************************************
**  UNIT    = A vector of ones.                                       **
**  UNITT   = The transposed vector of UNIT.                          **
**********************************************************************
CC
      REAL UNXXOR(DV,IV), UNYYOR(1,1), CRIT
CC
**********************************************************************
**  UNXXOR  = UNITT * XXOR                                            **
**  UNYYOR  = UNITT * YYOR                                            **
**********************************************************************
CC
      REAL BETA2(1,1), BETA(IV,DV), BETA0
CC
**********************************************************************
**  BETA   = Least squares parameter estimates.                       **
**  BETA0  = The intercept term.                                      **
**********************************************************************
```

```
CC******* INTEGER VARIABLES *******
CC
      INTEGER ROW, COL, ROUND, NOROW, K
      INTEGER NROUND, NOCOL, NROW
CC
CC******* FORMAT SATEMENTS *******
CC
   1  FORMAT(8F9.0)
  11  FORMAT(/,' ',6(F12.3))
  33  FORMAT(20F6.0)
  22  FORMAT(1X,5F10.4)
CC
**********************************************************************
**  This DO-LOOP reads and writes a matrix of all the rainfall      **
**  stations and all the observations.                              **
**********************************************************************
CC
      DO 10 ROW = 1, NOBS
         READ(13,1) (Z(ROW,COL), COL = 1, NSTAT)
         WRITE(6,11) (Z(ROW,COL), COL = 1, NSTAT)
  10  CONTINUE
CC
**********************************************************************
**  In the following DO-LOOP, we partition the Z-matrix into        **
**  the vector of the target station and the matrix of              **
**  control stations.                                               **
**********************************************************************
CC
      DO 30 ROW = 1, NOBS
         Y(ROW,1) = Z(ROW,1)
         DO 20 COL = 2, NSTAT
            X(ROW,COL-1) = Z(ROW,COL)
  20     CONTINUE
  30  CONTINUE
CC
      CALL COPY(Z,NOBS,NSTAT,ZOR,NOBS,NSTAT,NOBS,NSTAT)
      CALL COPY(Z,NOBS,NSTAT,ZORR,NOBS,NSTAT,NOBS,NSTAT)
      CALL COPY(Z,NOBS,NSTAT,ZOROR,NOBS,NSTAT,NOBS,NSTAT)
      CALL COPY(Z,NOBS,NSTAT,ZOR1,NOBS,NSTAT,NOBS,NSTAT)
      CALL COPY(X,NOBS,IV,XOR,NOBS,IV,NOBS,IV)
CC
**********************************************************************
**  Consider the first column of the Z-matrix.                      **
**********************************************************************
CC
      ROUND = 1
CC
   2  DO 60 K = 1, NOBS
CC
**********************************************************************
**  Check if the k-th observation of the considered target          **
**  station is missing.                                             **
**********************************************************************
CC
         IF (ZOR(K,ROUND) .EQ. -999.0) THEN
```

```
**********************************************************************
**  If the k-th observation is missing, then count the number       **
**  of columns of the control stations with k-th observations       **
**  not missing, and form a new control stations matrix.            **
**********************************************************************
CC
              NOCOL = 0
              DO 50 COL = 1, IV
                 IF (XOR(K,COL) .NE. -999.0) THEN
                     NOCOL = NOCOL + 1
                     DO 40 ROW = 1, NOBS
                        XX(ROW,NOCOL) = X(ROW,COL)
   40                CONTINUE
                  ENDIF
   50         CONTINUE
CC
**********************************************************************
**  Find the concurrent records of the considered matrices.         **
**********************************************************************
CC
              NROW = 1
              DO 80 ROW = 1, NOBS
                 IF (ZOR(ROW,ROUND) .NE. -999.0) THEN
                    DO 70 COL = 1, NOCOL
                       IF (XX(ROW,COL) .EQ. -999.0) GO TO 80
                          XXOR(NROW,COL) = XX(ROW,COL)
   70                 CONTINUE
                      YYOR(NROW,1) = ZOR(ROW,ROUND)
                      NROW = NROW + 1
                 ENDIF
   80         CONTINUE
CC
              NOROW = NROW - 1
CC
**********************************************************************
**  When called, these subroutines find the least squares           **
**  parameter estimates.                                            **
**********************************************************************
CC
              CALL CNTRAL(YREG,NOBS,DV,YYOR,NOBS,DV,NOROW,DV)
              CALL CNTRAL(XXREG,NOBS,IV,XXOR,NOBS,IV,NOROW,NOCOL)
              CALL TRANP(XXREG,NOBS,IV,XXREGT,IV,NOBS,NOROW,NOCOL)
              CALL MULT(XXREGT,IV,NOBS,XXREG,NOBS,IV,XRTXR,IV,IV,
     &                  NOCOL,NOROW,NOCOL)
              CALL MULT(XXREGT,IV,NOBS,YREG,NOBS,DV,XRTYR,IV,DV,
     &                  NOCOL,NOROW,DV)
              CALL F04ABE(XRTXR,IV,XRTYR,IV,NOCOL,1,BETA,IV,WKSPCE,
     &                  E,NOROW,IFAIL)
```

```
**********************************************************************
**  Form a vector of ones.                                          **
**********************************************************************
CC
              DO 90 ROW = 1, NOROW
                 UNIT(ROW,1) = 1
  90          CONTINUE
CC
              CALL TRANP(UNIT,NOBS,DV,UNITT,DV,NOBS,NOROW,DV)
              CALL MULT(UNITT,DV,NOBS,XXOR,NOBS,IV,UNXXOR,DV,IV,
     &                  DV,NOROW,NOCOL)
              CALL MULT(UNITT,DV,NOBS,YYOR,NOBS,DV,UNYYOR,1,1,DV,
     &                  NOROW,DV)
              CALL COPY(YYOR,NOBS,DV,YREG,NOBS,DV,NOROW,DV)
CC
**********************************************************************
**  Find the mean of the current target station.                    **
**********************************************************************
CC
              MEANYY(1,1) = UNYYOR(1,1) / NOROW
CC
**********************************************************************
**  Find the means of the current control stations.                 **
**********************************************************************
CC
              DO 100 COL = 1, NOCOL
                 MEANXX(1,COL) = UNXXOR(1,COL) / NOROW
 100          CONTINUE

              CALL MULT(MEANXX,1,IV,BETA,IV,DV,BETA2,1,1,DV,
     &                  NOCOL,DV)
CC
**********************************************************************
**  Find the intercept term.                                        **
**********************************************************************
CC
              BETA0 = MEANYY(1,1) - BETA2(1,1)
CC
**********************************************************************
**  Estimate the missing record.                                    **
**********************************************************************
CC
              PATCH(K,ROUND,1) = 0.0
              DO 110 COL = 1, NOCOL
                 PATCH(K,ROUND,1) = XX(K,COL) * BETA(COL,1)
                                  + PATCH(K,ROUND,1)
     &
 110          CONTINUE
CC
              PATCH(K,ROUND,1) = BETA0 + PATCH(K,ROUND,1)

CC
          ELSE
CC
              PATCH(K,ROUND,1) = ZOR(K,ROUND)
CC
          ENDIF
```

```
          TMAT(K,ROUND) = PATCH(K,ROUND,1)
CC
*****************************************************************
**  TMAT is a temporary matrix which contains the new values   **
**  of the target station. It should be of the same size as    **
**  the  Z-matrix.                                             **
*****************************************************************
CC
  60  CONTINUE
CC
*****************************************************************
**  Consider the next column of the Z-matrix.                  **
*****************************************************************
CC
      ROUND = ROUND + 1
CC
*****************************************************************
**  Check if there are any stations which are not yet          **
**  estimated.                                                 **
*****************************************************************
CC
      IF (ROUND .LE. NSTAT) THEN
CC
*****************************************************************
**  If there are, then swop the different stations so that the **
**  one which needs to be estimated is always in the first     **
**  column of the Z-matrix.                                    **
*****************************************************************
CC
       CALL SWOP(ZORR,NOBS,NSTAT,NOBS,ROUND)
       CALL SWOP(ZOROR,NOBS,NSTAT,NOBS,ROUND)
CC
       CALL COPY(ZORR,NOBS,NSTAT,Z,NOBS,NSTAT,NOBS,NSTAT)
CC
       DO 120 ROW = 1, NOBS
          Y(ROW,1) = Z(ROW,1)
          DO 130 COL = 2, NSTAT
             X(ROW,COL-1) = Z(ROW,COL)
             XOR(ROW,COL-1) = ZOROR(ROW,COL)
 130      CONTINUE
 120   CONTINUE
CC
       GO TO 2
CC
      ELSE
CC
*****************************************************************
**  If all the missing observations have been estimated, then  **
**  consider the first iteration. NROUND counts the number     **
**  of alterations which are made.                             **
*****************************************************************
CC
      NROUND = 1
```

```
      CALL COPY(TMAT,NOBS,NSTAT,Z,NOBS,NSTAT,NOBS,NSTAT)
      CALL COPY(ZOR,NOBS,NSTAT,ZOROR,NOBS,NSTAT,NOBS,NSTAT)
CC
      DO 19 ROW = 1, NOBS
         Y(ROW,1) = Z(ROW,1)
         DO 18 COL = 2, NSTAT
            X(ROW,COL-1) = Z(ROW,COL)
            XOR(ROW,COL-1) = ZOROR(ROW,COL)
  18     CONTINUE
  19  CONTINUE
CC
12121 CALL COPY(Z,NOBS,NSTAT,ZORR,NOBS,NSTAT,NOBS,NSTAT)
CC
*****************************************************************
**  Consider the first column of the Z-matrix.                 **
*****************************************************************
CC
      ROUND = 1
CC
13131 DO 500 K = 1, NOBS
CC
*****************************************************************
**  Check if the k-th observation of the considered target     **
**  station is missing.                                        **
*****************************************************************
CC
         IF (ZOR(K,ROUND) .EQ. -999.0) THEN
CC
*****************************************************************
**  If the k-th observation is missing, then count the number  **
**  of columns of the control stations with k-th observations  **
**  not missing, and form a new control stations matrix.       **
*****************************************************************
CC
            NOCOL = 0
            DO 710 COL = 1, IV
               IF (XOR(K,COL) .NE. -999.0) THEN
                  NOCOL = NOCOL + 1
                  DO 700 ROW = 1, NOBS
                     XXOR(ROW,NOCOL) = X(ROW,COL)
 700              CONTINUE
               ENDIF
 710        CONTINUE
CC
*****************************************************************
**  When called, these subroutines find the least squares      **
**  parameter estimates.                                       **
*****************************************************************
CC
          CALL COPY(Y,NOBS,DV,YYOR,NOBS,DV,NOBS,DV)
          CALL CENTRE(Y,NOBS,DV,NOBS,DV)
          CALL CNTRAL(XX,NOBS,IV,XXOR,NOBS,IV,NOBS,NOCOL)
          CALL TRANP(XX,NOBS,IV,XXT,IV,NOBS,NOBS,NOCOL)
```

```
      CALL TRANP(Y,NOBS,DV,YT,DV,NOBS,NOBS,DV)
      CALL MULT(XXT,IV,NOBS,XX,NOBS,IV,XXTXX,IV,IV,NOCOL,NOBS,
     &          NOCOL)
      CALL MULT(XXT,IV,NOBS,Y,NOBS,DV,XXTYY,IV,DV,NOCOL,NOBS,
     &          DV)
      CALL F04ABE(XXTXX,IV,XXTYY,IV,NOCOL,1,BETA,IV,WKSPCE,E,
     &          NOBS,IFAIL)
CC
**********************************************************************
**  Form a vector of ones.                                          **
**********************************************************************
CC
      DO 740 ROW = 1, NOBS
         UNIT(ROW,1) = 1
 740  CONTINUE
CC
      CALL TRANP(UNIT,NOBS,DV,UNITT,DV,NOBS,NOBS,DV)
      CALL MULT(UNITT,DV,NOBS,XXOR,NOBS,IV,UNXXOR,DV,IV,DV,
     &          NOBS,NOCOL)
      CALL MULT(UNITT,DV,NOBS,YYOR,NOBS,DV,UNYYOR,1,1,DV,NOBS,
     &          DV)
      CALL COPY(YYOR,NOBS,DV,Y,NOBS,DV,NOBS,DV)
CC
      MEANYY(1,1) = UNYYOR(1,1) / NOBS
CC
      DO 750 COL = 1, NOCOL
         MEANXX(1,COL) = UNXXOR(1,COL) / NOBS
 750  CONTINUE
CC
      CALL MULT(MEANXX,DV,IV,BETA,IV,DV,BETA2,1,1,DV,NOCOL,
     &          DV)
CC
      BETA0 = MEANYY(1,1) - BETA2(1,1)
CC
**********************************************************************
**  Re-estimate the missing observation.                            **
**********************************************************************
CC
      PATCH(K,ROUND,NROUND) = 0.0
      DO 995 COL = 1, NOCOL
         PATCH(K,ROUND,NROUND) = PATCH(K,ROUND,NROUND)
     &                           + XXOR(K,COL) * BETA(COL,1)
 995  CONTINUE
CC
      PATCH(K,ROUND,NROUND) = BETA0 + PATCH(K,ROUND,NROUND)
CC
**********************************************************************
**  Check if the estimated value converge.                          **
**********************************************************************
CC
      IF (NROUND .GT. 1) THEN
         CRIT = (PATCH(K,ROUND,NROUND)
     &                           - PATCH(K,ROUND,NROUND-1))
     &                              / PATCH(K,ROUND,NROUND)
```

```
            CRIT = CRIT ** 2
CC
            IF (CRIT .LT. 0.00000001) THEN
               ZOR(K,ROUND) = PATCH(K,ROUND,NROUND)
            ENDIF
CC
         ENDIF
CC
      ELSE
         PATCH(K,ROUND,NROUND) = ZOR(K,ROUND)
      ENDIF
CC
      TMAT(K,ROUND) = PATCH(K,ROUND,NROUND)
CC
**********************************************************************
**  TMAT is a temporary matrix which contains the new values        **
**  of the target station. It should be of the same size as         **
**  the Z-matrix.                                                   **
**********************************************************************
 500  CONTINUE
CC
**********************************************************************
**  Consider the next column of the Z-matrix.                       **
**********************************************************************
CC
14141 ROUND = ROUND + 1
CC
**********************************************************************
**  Check if there are any stations which needs to be               **
**  estimated.                                                      **
**********************************************************************
CC
      IF (ROUND .GT. NSTAT) THEN
CC
         CALL COPY(TMAT,NOBS,NSTAT,Z,NOBS,NSTAT,NOBS,NSTAT)
         CALL COPY(ZOR1,NOBS,NSTAT,ZOROR,NOBS,NSTAT,NOBS,NSTAT)
CC
         DO 402 ROW = 1, NOBS
            Y(ROW,1) = Z(ROW,1)
            DO 401 COL = 2, NSTAT
               X(ROW,COL-1) = Z(ROW,COL)
               XOR(ROW,COL-1) = ZOROR(ROW,COL)
 401        CONTINUE
 402     CONTINUE
CC
**********************************************************************
**  Check if there are some of the estimates which have not         **
**  yet converged.                                                  **
**********************************************************************
CC
         DO 460 ROW = 1, NOBS
            DO 450 COL = 1, NSTAT
               IF (ZOR(ROW,COL) .NE. -999.0) GO TO 450
               NROUND = NROUND + 1
               GO TO 12121
 450        CONTINUE
 460     CONTINUE
```

```
**********************************************************************
**  Print the estimated matrix ZOR.                                 **
**********************************************************************
CC
        PRINT*, 'These are the patched values after',  NROUND
        DO 911 ROW = 1, NOBS
           WRITE(6,11) (ZOR(ROW,COL), COL = 1, NSTAT)
 911    CONTINUE
CC
        GO TO 998
CC
      ELSE
CC
**********************************************************************
**  If there are missing values, then swop the different            **
**  stations.                                                       **
**********************************************************************
CC
         CALL SWOP(ZORR,NOBS,NSTAT,NOBS,ROUND)
         CALL SWOP(ZOROR,NOBS,NSTAT,NOBS,ROUND)
CC
         CALL COPY(ZORR,NOBS,NSTAT,Z,NOBS,NSTAT,NOBS,NSTAT)
CC
         DO 610 ROW = 1, NOBS
            Y(ROW,1) = Z(ROW,1)
            DO 600 COL = 2, NSTAT
                X(ROW,COL-1) = Z(ROW,COL)
                XOR(ROW,COL-1) = ZOROR(ROW,COL)
 600        CONTINUE
 610     CONTINUE
CC
**********************************************************************
**  Check if the current target station has missing values.         **
**********************************************************************
CC
         DO 537 ROW = 1, NOBS
            IF (ZOR(ROW,ROUND) .EQ. -999.0) GO TO 13131
 537     CONTINUE
CC
         GO TO 14141
CC
      ENDIF
CC
      ENDIF
CC
 998  STOP
      DEBUG SUBCHK,SUBTRACE
      END
```

PROGRAM 5

```
CC*****************************************************************
CC******                                                          **
CC******  A PROGRAM TO ESTIMATE MISSING ANNUAL RAINFALL DATA      **
CC******  BY MAKING USE OF THE EM ALGORITHM                       **
CC******  - (Modification of the "Standard" EM algorithm.)        **
CC******                                                          **
CC*****************************************************************
CC*****************************************************************
CC                                                                **
CC   This program is used to estimate data matrices which         **
CC   contain missing observations in almost all the rainfall      **
CC   stations.                                                    **
CC                                                                **
CC   The stations are read as one big matrix which consists       **
CC   of a column of the TARGET station and the remaining          **
CC   columns being the matrix of control stations.                **
CC                                                                **
CC   Each row of data represents an observation (Annual           **
CC   rainfall total). Missing rainfall data points are            **
CC   represented by a "-999".                                     **
CC                                                                **
CC   The data is stored in a matrix called the Z-matrix,          **
CC   which can be partitioned into the:                           **
CC              Y-vector = A vector of the target station         **
CC              X-matrix = A matrix of the control stations.      **
CC                                                                **
CC   The maximum dimensions of the matrices are:                  **
CC              Target station          :  1                      **
CC              Control stations        :  25                     **
CC              Observations            :  100                    **
CC                                                                **
CC                                                                **
CC   Note that some of the routines which are in this             **
CC   program were copied from the programs written by             **
**   DR ROSS S. SPARKS.                                           **
CC                                                                **
CC*****************************************************************
CC*****************************************************************
CC
CC******* VARIABLES DECLARATION *******
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER NOBS, NSTAT, IV, DV
CC
*****************************************************************
**  NOBS   =  Number of all the records                         **
**  NSTAT  =  Number of all the stations - target + control.    **
**  IV     =  Number of control stations.                       **
**  DV     =  Number of target stations.                        **
*****************************************************************
```

```
CC******* PARAMETER STATEMENTS *******
CC
      PARAMETER (NOBS   =   28)
      PARAMETER (NSTAT  =    6)
      PARAMETER (IV     =    5)
      PARAMETER (DV     =    1)
CC
CC******* REAL VARIABLES *******
CC
      REAL Z(NOBS,NSTAT), ZOR(NOBS,NSTAT), ZCEN(NOBS,NSTAT)
CC
**********************************************************************
**  Z    =  Matrix of all the stations.                               **
**  ZOR  =  Matrix of the original data matrix Z.                     **
**  ZCEN =  Matrix of the centralised matrix Z.                       **
**********************************************************************
CC
      REAL TMAT(NOBS,NSTAT), PATCH(NOBS,NSTAT,500)
CC
**********************************************************************
**  TMAT     =  Temporary matrix of estimated Z matrix               **
**  PATCH    =  Matrix containing estimated values of Z.             **
**********************************************************************
CC
      REAL ZT(NSTAT,NOBS), ZTZ(NSTAT,NSTAT)
CC
**********************************************************************
**  ZT   =  The transposed matrix of ZCEN                           **
**  ZTZ  =  ZT * ZCEN                                               **
**********************************************************************
CC
      REAL MEANZZ(DV,NSTAT), MEAN1(NSTAT), MEAN2(NSTAT)
CC
**********************************************************************
**  MEANZZ  = A vector of the means of all control stations.        **
**  MEAN1   = A vector of the sum of values which were              **
**            observed from the control stations                    **
**  MEAN2   = A vector of the sum of values which were missing      **
**            from the control stations.                            **
**********************************************************************
CC
      REAL BETA(NSTAT), BETA0, BETA2
CC
**********************************************************************
**  BETA   =  A vector of least squares parameter estimates         **
**  BETA0  =  The intercept term                                    **
**********************************************************************
CC
      REAL CRIT(NSTAT), CONV
CC
**********************************************************************
**   CRIT  = The convergence criterion used.                        **
**   CONV  checks if all the the observations have converged.       **
**********************************************************************
```

```
CC******* INTEGER VARIABLES *******

      INTEGER ROW, COL, ROUND, NROW
      INTEGER NROUND
CC
*****************************************************************
**  NROUND  =  Number of iterations performed.                 **
**  ROUND   =  Number of current station considered.           **
*****************************************************************
CC
CC******* FORMAT SATEMENTS *******

   1  FORMAT(8F9.0)
  22  FORMAT(1X,10F11.0)
 222  FORMAT(13F9.4)
 333  FORMAT(11F11.4)
CC
*****************************************************************
**  The following DO-LOOP reads and writes a matrix of all     **
**  the rainfall stations and all the observations in the data **
*****************************************************************
CC
      DO 100 ROW = 1, NOBS
         READ(13,1) (Z(ROW,COL), COL = 1, NSTAT)
         WRITE(6,1) (Z(ROW,COL), COL = 1, NSTAT)
 100  CONTINUE
CC
      CALL COPY(Z,NOBS,NSTAT,ZOR,NOBS,NSTAT,NOBS,NSTAT)
CC
*****************************************************************
**  Check if there are missing observations in any of the      **
**  included predictors.  If there are, let all the            **
**  observations in that row be equal to "-999".               **
*****************************************************************
CC
      DO 60 ROW = 1, NOBS
         COL = 1
  40     IF (COL .LE. NSTAT) THEN
            IF (Z(ROW,COL) .EQ. -999.0) THEN
               DO 50 COL = 1, NSTAT
                  Z(ROW,COL) = -999.0
  50           CONTINUE
            ENDIF
         COL = COL + 1
         GO TO 40
         ENDIF
  60  CONTINUE
```

```
**********************************************************************
**  Eliminate from Z all those rows which have "-999.0"             **
**  entries.  Count the remaining number of rows.                   **
**********************************************************************
CC
      NROW = 0
      DO 80 ROW = 1, NOBS
         COL = 1
         IF (Z(ROW,COL) .NE. -999.0) THEN
            NROW = NROW + 1
            DO 85 COL = 1, NSTAT
               Z(NROW,COL) = Z(ROW,COL)
 85         CONTINUE
         ENDIF
 80   CONTINUE
CC
**********************************************************************
**  Check if there are sufficient concurrent records to fit a       **
**  regression model.                                               **
**********************************************************************
CC
      IF (NROW .EQ. NOBS) GO TO 998
CC
**********************************************************************
**  Standardize the Z matrix.                                       **
**********************************************************************
CC
      CALL CNTRAL(ZCEN,NOBS,NSTAT,Z,NOBS,NSTAT,NROW,NSTAT)
      CALL TRANP(ZCEN,NOBS,NSTAT,ZT,NSTAT,NOBS,NROW,NSTAT)
      CALL MULT(ZT,NSTAT,NOBS,ZCEN,NOBS,NSTAT,ZTZ,NSTAT,NSTAT,
     &          NSTAT,NROW,NSTAT)
      CALL INVERT(ZTZ,NSTAT,NSTAT)
CC
**********************************************************************
**  Calculate the means of the concurrent observations of           **
**  all the stations.                                               **
**********************************************************************
CC
      DO 130 COL = 1, NSTAT
         MEANZZ(1,COL) = 0.0
         DO 120 ROW = 1, NROW
            MEANZZ(1,COL) = MEANZZ(1,COL) + Z(ROW,COL)
 120     CONTINUE
         MEANZZ(1,COL) = MEANZZ(1,COL) / NROW
 130  CONTINUE
CC
**********************************************************************
**  Consider the first station for estimation.                      **
**********************************************************************
CC
      ROUND = 1
```

```
**********************************************************************
**  Calculate the  least squares parameter estimates.               **
**********************************************************************
CC
31313 DO 121 ROW = 1, NSTAT
         BETA(ROW) = (-1.0) * ZTZ(ROW,ROUND) / ZTZ(ROUND,ROUND)
 121  CONTINUE
CC
      BETA2 = 0.0
      DO 113 ROW = 1, NSTAT
         IF (ROW .NE. ROUND) THEN
            BETA2 = BETA2 + MEANZZ(1,ROW) * BETA(ROW)
         ENDIF
 113  CONTINUE
CC
**********************************************************************
**  Find the intercept term                                         **
**********************************************************************
CC
      BETA0 = MEANZZ(1,ROUND) - BETA2
CC
**********************************************************************
**  Estimate the missing values in the current station.             **
**********************************************************************
CC
      DO 105 ROW = 1, NOBS
         IF (ZOR(ROW,ROUND) .EQ. -999.0) THEN
            PATCH(ROW,ROUND,1) = 0.0
            DO 103 COL = 1, NSTAT
               IF (COL .EQ. ROUND) GO TO 103
               IF (ZOR(ROW,COL) .EQ. -999.0) GO TO 103
               PATCH(ROW,ROUND,1) = PATCH(ROW,ROUND,1)
     &                              + ZOR(ROW,COL) * BETA(COL)
 103           CONTINUE

            PATCH(ROW,ROUND,1) = PATCH(ROW,ROUND,1) + BETA0

         ELSE
            PATCH(ROW,ROUND,1) = ZOR(ROW,ROUND)
         ENDIF
CC
**********************************************************************
**  Copy the estimated station into a temporary matrix TMAT         **
**********************************************************************
CC
         TMAT(ROW,ROUND) = PATCH(ROW,ROUND,1)
CC
 105  CONTINUE
CC
**********************************************************************
**  Consider estimating the next station.                           **
**********************************************************************
CC
      ROUND = ROUND + 1
```

```
      IF (ROUND .LE. NSTAT) GO TO 31313
CC
*****************************************************************
**  Calculate the sum of all the observed observations in each **
**  station.                                                   **
*****************************************************************
CC
      DO 900 COL = 1, NSTAT
         MEAN1(COL) = 0.0
         DO 890 ROW = 1, NOBS
            IF (ZOR(ROW,COL) .NE. -999.0) THEN
               MEAN1(COL) = MEAN1(COL) + ZOR(ROW,COL)
            ENDIF
 890     CONTINUE
 900  CONTINUE

      NROUND = 1

12121 CALL COPY(TMAT,NOBS,NSTAT,Z,NOBS,NSTAT,NOBS,NSTAT)
CC
*****************************************************************
**  Standardize the Z matrix.                                  **
*****************************************************************
CC
      CALL CNTRAL(ZCEN,NOBS,NSTAT,Z,NOBS,NSTAT,NOBS,NSTAT)
      CALL TRANP(ZCEN,NOBS,NSTAT,ZT,NSTAT,NOBS,NOBS,NSTAT)
      CALL MULT(ZT,NSTAT,NOBS,ZCEN,NOBS,NSTAT,ZTZ,NSTAT,NSTAT,
     &          NSTAT,NOBS,NSTAT)
      CALL INVERT(ZTZ,NSTAT,NSTAT)
CC
*****************************************************************
**  Calculate the means    the estimated values for each of    **
**  station.                                                   **
*****************************************************************
CC
      DO 167 COL = 1, NSTAT
         MEAN2(COL) = 0.0
         DO 163 ROW = 1, NOBS
            IF (ZOR(ROW,COL) .EQ. -999.0) THEN
               MEAN2(COL) = MEAN2(COL) + Z(ROW,COL)
            ENDIF
 163     CONTINUE
         MEANZZ(1,COL) = (MEAN1(COL) + MEAN2(COL)) / NOBS
 167  CONTINUE
CC
      ROUND = 1
CC
13131 DO 810 ROW = 1, NSTAT
         BETA(ROW) = (-1.0) * ZTZ(ROW,ROUND) / ZTZ(ROUND,ROUND)
 810  CONTINUE
```

```
      BETA2 = 0.0
      DO 830 ROW = 1, NSTAT
         IF (ROW .NE. ROUND) THEN
            BETA2 = BETA2 + MEANZZ(1,ROW) * BETA(ROW)
         ENDIF
 830  CONTINUE
CC
      BETA0 = MEANZZ(1,ROUND) - BETA2
CC
**********************************************************************
**  Re-estimate the missing observations in the current             **
**  station.                                                        **
**********************************************************************
CC
      CRIT(ROUND)= 0.0
      DO 200 ROW = 1, NOBS
         IF (ZOR(ROW,ROUND) .EQ. -999.0) THEN
            PATCH(ROW,ROUND,NROUND) = 0.0
            DO 192 COL = 1, NSTAT
               IF (COL .EQ. ROUND) GO TO 192
               PATCH(ROW,ROUND,NROUND) = PATCH(ROW,ROUND,NROUND)
     &                                   + Z(ROW,COL) * BETA(COL)
 192        CONTINUE

            PATCH(ROW,ROUND,NROUND) = BETA0
     &                                + PATCH(ROW,ROUND,NROUND)
CC
**********************************************************************
**  Check if the current estimated value has converged.             **
**********************************************************************
CC
            IF (NROUND .GT. 1) THEN
               CRIT(ROUND) = CRIT(ROUND)
     &                       + ABS((PATCH(ROW,ROUND,NROUND)
     &                         - PATCH(ROW,ROUND,NROUND-1))
     &                           / PATCH(ROW,ROUND,NROUND))
            ENDIF
CC
         ELSE
            PATCH(ROW,ROUND,NROUND) = ZOR(ROW,ROUND)
         ENDIF
CC
         TMAT(ROW,ROUND) = PATCH(ROW,ROUND,NROUND)
CC
 200  CONTINUE
CC
      CONV = CONV + CRIT(ROUND)
CC
**********************************************************************
**  Consider the next cycle for re-estimation.                      **
**********************************************************************
CC
 223  ROUND = ROUND + 1
```

```
      IF (ROUND .GT. NSTAT) THEN
CC
         IF (NROUND .GT. 1) THEN
CC
*****************************************************************
**  Check for convergence of all the estimated observations.   **
*****************************************************************
CC
            IF (CONV.LT. 0.0001) THEN
               DO 392 ROW = 1, NOBS
                  DO 382 COL = 1, NSTAT
                     ZOR(ROW,COL) = TMAT(ROW,COL)
 382              CONTINUE
 392           CONTINUE
CC
*****************************************************************
**  Print the estimated matrix ZOR.                            **
*****************************************************************
CC
               PRINT*, 'These are the patched values after',NROUND
               CALL PMAT(ZOR,NOBS,NSTAT,NOBS,NSTAT)
CC
               GO TO 998

            ENDIF
         ENDIF
CC
*****************************************************************
**  Consider the next iteration.                               **
*****************************************************************
CC
         NROUND = NROUND + 1
CC
         CONV = 0.0
CC
         GO TO 12121
CC
      ELSE
         GO TO 13131
      ENDIF

 998  STOP
CC    DEBUG SUBCHK,SUBTRACE
      END
```

PROGRAM 6

```
CC*****************************************************************
CC******                                                          **
CC******  A PROGRAM TO ESTIMATE MISSING MONTHLY RAINFALL DATA     **
CC******  BY MAKING USE OF THE EM ALGORITHM                       **
CC******  - (Modification of the "Standard" EM algorithm.)        **
CC******                                                          **
CC*****************************************************************
CC*****************************************************************
CC                                                                **
CC   This program is used to estimate data matrices which         **
CC   contain missing observations in almost all the rainfall      **
CC   stations.                                                    **
CC                                                                **
CC   The stations are read as one big matrix which consists       **
CC   of a column of the TARGET station and the remaining          **
CC   columns being the matrix of control stations.                **
CC                                                                **
CC   Each row of data represents an observation (Annual           **
CC   rainfall total). Missing rainfall data points are            **
CC   represented by a "-999".                                     **
CC                                                                **
CC   The data is stored in a matrix called the Z-matrix,          **
CC   which can be partitioned into the:                           **
CC              Y-vector = A vector of the target station         **
CC              X-matrix = A matrix of the control stations.      **
CC                                                                **
CC   The maximum dimensions of the matrices are:                  **
CC              Target station          :  1                      **
CC              Control stations        :  25                     **
CC              Observations (annual)   :  100                    **
CC                                                                **
CC                                                                **
CC   Note that some of the routines which are in this             **
CC   program were copied from the programs written by             **
CC   DR ROSS S. SPARKS.                                          **
CC                                                                **
CC*****************************************************************
CC*****************************************************************
CC
```

```
CC******* VARIABLES DECLARATION *******
CC
CC******* INTEGER VARIABLES *******
CC
      INTEGER NOBS, NSTAT, IV, DV, NYEAR
      INTEGER NMONTH
CC
*****************************************************************
**  NOBS   =  Number of all the monthly records.                **
**  NSTAT  =  Number of all the stations - target + control.    **
**  IV     =  Number of control stations.                       **
**  DV     =  Number of target stations.                        **
**  NYEAR  =  Number of all the annual records.                 **
**  NMONTH =  Number of the months being considered.            **
*****************************************************************
CC
CC******* PARAMETER STATEMENTS *******
CC
      PARAMETER (NYEAR  =  28)
      PARAMETER (NSTAT  =   6)
      PARAMETER (IV     =   5)
      PARAMETER (DV     =   1)
      PARAMETER (NMONTH =  12)
      PARAMETER (NOBS   =  NYEAR * NMONTH)
CC
CC******* REAL VARIABLES *******
CC
      REAL Z(NYEAR,NSTAT), ZOR(NYEAR,NSTAT), ZZ(NOBS,NSTAT)
CC
*****************************************************************
**  Z    =  Matrix of all the stations and annual observations **
**  ZOR  =  Matrix of the original data matrix Z.              **
**  ZZ   =  Matrix of all the stations and monthly             **
**          observations.                                      **
*****************************************************************
CC
      REAL TMAT(NYEAR,NSTAT), PATCH(NYEAR,NSTAT,500)
CC
*****************************************************************
**  TMAT     =  Temporary matrix of estimated Z matrix         **
**  PATCH    =  Matrix containing estimated values of Z.       **
*****************************************************************
CC
      REAL ZT(NSTAT,NYEAR), ZTZ(NSTAT,NSTAT)
CC
*****************************************************************
**  ZT   =  The transposed matrix of Z                         **
**  ZTZ  =  ZT * Z                                             **
*****************************************************************
CC
      REAL MEANZZ(DV,NSTAT)
CC
*****************************************************************
**  MEANZZ  = A vector of the means of all control stations.   **
*****************************************************************
```

```
      REAL BETA(NSTAT)
CC
*****************************************************************
**  BETA    =  A vector of least squares parameter estimates    **
*****************************************************************
CC
      REAL CRIT(NSTAT), CONV
CC
*****************************************************************
**   CRIT = The convergence criterion used.                     **
**   CONV  checks if all the stations have converged.           **
*****************************************************************
CC
CC******* INTEGER VARIABLES *******

      INTEGER ROW, COL, ROUND, NROW, MONTH
      INTEGER NROUND, YEAR
CC
*****************************************************************
**  NROUND   =  Number of iterations performed.                 **
**  ROUND    =  Number of current station considered.           **
*****************************************************************
CC
CC******* FORMAT SATEMENTS *******

   1  FORMAT(8F9.0)
  22  FORMAT(1X,10F11.0)
 222  FORMAT(13F9.4)
 333  FORMAT(11F11.4)
CC
*****************************************************************
**  The following DO-LOOP reads and writes a matrix of all     **
**  the rainfall stations and all the observations in the data **
*****************************************************************
CC
      DO 100 ROW = 1, NOBS
         READ(13,1) (ZZ(ROW,COL), COL = 1, NSTAT)
CC       WRITE(6,1) (ZZ(ROW,COL), COL = 1, NSTAT)
 100  CONTINUE
CC
      DO 1920 MONTH = 1, NMONTH
CC
*****************************************************************
**  Group the same months together in such a way that we form  **
**  twelve different matrices for each month separately.       **
*****************************************************************
CC
         CONV = 0.0
         DO 776 ROW = 1, NYEAR
            YEAR = NMONTH * (ROW -1) + (1 * MONTH)
            DO 771 COL = 1, NSTAT
               Z(ROW,COL) = ZZ(YEAR,COL)
 771        CONTINUE
 776     CONTINUE
```

```
      CALL COPY(Z,NYEAR,NSTAT,ZOR,NYEAR,NSTAT,NYEAR,NSTAT)
CC
**********************************************************************
**  Check if there are missing observations in any of the           **
**  included predictors.  If there are, let all the                 **
**  observations in that row be equal to "-999".                    **
**********************************************************************
CC
      DO 60 ROW = 1, NYEAR
         COL = 1
 40      IF (COL .LE. NSTAT) THEN
            IF (Z(ROW,COL) .EQ. -999.0) THEN
               DO 50 COL = 1, NSTAT
                  Z(ROW,COL) = -999.0
 50            CONTINUE
            ENDIF
            COL = COL + 1
            GO TO 40
         ENDIF
 60   CONTINUE
CC
**********************************************************************
**  Eliminate from Z all those rows which have "-999.0"             **
**  entries.  Count the remaining number of rows.                   **
**********************************************************************
CC
      NROW = 0
      DO 80 ROW = 1, NYEAR
         COL = 1
         IF (Z(ROW,COL) .NE. -999.0) THEN
            NROW = NROW + 1
            DO 85 COL = 1, NSTAT
               Z(NROW,COL) = Z(ROW,COL)
 85         CONTINUE
         ENDIF
 80   CONTINUE
CC
**********************************************************************
**  Check if there are sufficient concurrent records to fit a       **
**  regression model.                                               **
**********************************************************************
CC
      IF (NROW .EQ. NYEAR) GO TO 1920
CC
      CALL TRANP(Z,NYEAR,NSTAT,ZT,NSTAT,NYEAR,NROW,NSTAT)
      CALL MULT(ZT,NSTAT,NYEAR,Z,NYEAR,NSTAT,ZTZ,NSTAT,NSTAT,
     &          NSTAT,NROW,NSTAT)
      CALL INVERT(ZTZ,NSTAT,NSTAT)
```

```
**********************************************************************
**  Calculate the means of the concurrent observations of all       **
**  the stations.                                                   **
**********************************************************************
CC
         DO 130 COL = 1, NSTAT
            MEANZZ(1,COL) = 0.0
            DO 120 ROW = 1, NROW
               MEANZZ(1,COL) = MEANZZ(1,COL) + Z(ROW,COL)
 120        CONTINUE
            MEANZZ(1,COL) = MEANZZ(1,COL) / NROW
 130     CONTINUE
CC
**********************************************************************
**  Consider the first station for estimation.                      **
**********************************************************************
CC
         ROUND = 1
CC
**********************************************************************
**  Calculate the least squares parameter estimates.                **
**********************************************************************
CC
31313    DO 121 ROW = 1, NSTAT
            BETA(ROW) = (-1.0) * ZTZ(ROW,ROUND) / ZTZ(ROUND,ROUND)
 121     CONTINUE
CC
**********************************************************************
**  Estimate the missing values in the current station.             **
**********************************************************************
CC
         DO 105 ROW = 1, NYEAR
            IF (ZOR(ROW,ROUND) .EQ. -999.0) THEN
                PATCH(ROW,ROUND,1) = 0.0
                DO 103 COL = 1, NSTAT
                   IF (COL .EQ. ROUND) GO TO 103
                   IF (ZOR(ROW,COL) .EQ. -999.0) GO TO 103
                   PATCH(ROW,ROUND,1) = PATCH(ROW,ROUND,1)
     &                                  + ZOR(ROW,COL) * BETA(COL)
 103            CONTINUE

             ELSE
                PATCH(ROW,ROUND,1) = ZOR(ROW,ROUND)
             ENDIF
CC
**********************************************************************
**  Copy the estimated station into a temporary matrix TMAT.        **
**********************************************************************

             TMAT(ROW,ROUND) = PATCH(ROW,ROUND,1)
CC
 105      CONTINUE
```

```
**********************************************************************
**  Consider estimating the next station.                           **
**********************************************************************
CC
          ROUND = ROUND + 1
CC
          IF (ROUND .LE. NSTAT) GO TO 31313
CC
**********************************************************************
**  Consider the first iteration.                                   **
**********************************************************************
CC
          NROUND = 1
CC
12121     CALL COPY(TMAT,NYEAR,NSTAT,Z,NYEAR,NSTAT,NYEAR,NSTAT)
CC
          CALL TRANP(Z,NYEAR,NSTAT,ZT,NSTAT,NYEAR,NYEAR,NSTAT)
          CALL MULT(ZT,NSTAT,NYEAR,Z,NYEAR,NSTAT,ZTZ,NSTAT,
     &               NSTAT,NSTAT,NYEAR,NSTAT)
          CALL INVERT(ZTZ,NSTAT,NSTAT)
CC
**********************************************************************
**  Consider the first station for estimation.                      **
**********************************************************************
CC
          ROUND = 1
CC
**********************************************************************
**  Calculate the least squares parameter estimates.                **
**********************************************************************
CC
13131     DO 810 ROW = 1, NSTAT
             BETA(ROW) = (-1.0) * ZTZ(ROW,ROUND) / ZTZ(ROUND,ROUND)
 810      CONTINUE
CC
**********************************************************************
**  Estimate the missing values in the current station.             **
**********************************************************************
CC
          CRIT(ROUND) = 0.0
          DO 200 ROW = 1, NYEAR
             IF (ZOR(ROW,ROUND) .EQ. -999.0) THEN
                PATCH(ROW,ROUND,NROUND) = 0.0
                DO 192 COL = 1, NSTAT
                   IF (COL .EQ. ROUND) GO TO 192
                   PATCH(ROW,ROUND,NROUND) =
     &                                      PATCH(ROW,ROUND,NROUND)
     &                                      + Z(ROW,COL) * BETA(COL)
 192            CONTINUE
```

```
**********************************************************************
**  Check if the current estimated missing value has converged **
**********************************************************************
CC
                IF (NROUND .GT. 1) THEN
                CRIT(ROUND) = CRIT(ROUND)
     &                              + ABS((PATCH(ROW,ROUND,NROUND)
     &                                - PATCH(ROW,ROUND,NROUND-1))
     &                                  / PATCH(ROW,ROUND,NROUND))
                ENDIF
CC
             ELSE
                PATCH(ROW,ROUND,NROUND) = ZOR(ROW,ROUND)
             ENDIF
CC
**********************************************************************
**  Copy the estimated station into a temporary matrix TMAT.    **
**********************************************************************
CC
                TMAT(ROW,ROUND) = PATCH(ROW,ROUND,NROUND)
CC
 200       CONTINUE
CC
           CONV = CONV + CRIT(ROUND)
CC
**********************************************************************
**  Consider the next station.                                      **
**********************************************************************
CC
           ROUND = ROUND + 1
CC
           IF (ROUND .GT. NSTAT) THEN
CC
              IF (NROUND .GT. 1) THEN
CC
**********************************************************************
**  Check for convergence of all the estimated observations.    **
**********************************************************************
CC
                IF (CONV .LT. 0.0001) THEN

                   CALL COPY(TMAT,NYEAR,NSTAT,ZOR,NYEAR,NSTAT,
     &                       NYEAR,NSTAT)
CC
**********************************************************************
**  Print the complete matrix.                                      **
**********************************************************************
CC
                   PRINT*, 'THESE ARE THE PATCHED VALUES AFTER',
     &                     NROUND,'ITERATIONS :',  MONTH
                   CALL PMAT(ZOR,NYEAR,NSTAT,NYEAR,NSTAT)
```

```
                    DO 7076 ROW = 1, NYEAR
                       YEAR = NMONTH * (ROW-1) + (1 * MONTH)
                       DO 7071 COL = 1, NSTAT
                          ZZ(YEAR,COL) = ZOR(ROW,COL)
7071                    CONTINUE
7076                 CONTINUE
CC
                     GO TO 1920
CC
                  ENDIF
CC
               ENDIF
CC
*****************************************************************
**  Consider the next iteration.                               **
*****************************************************************
CC
               NROUND = NROUND + 1
               CONV = 0.0
CC
               GO TO 12121
CC
            ELSE
CC
               GO TO 13131
CC
            ENDIF
CC
1920  CONTINUE
CC
      CALL PMAT(ZZ,NOBS,NSTAT,NOBS,NSTAT)
CC
998   STOP
      END
```

SUBROUTINES

```
      SUBROUTINE COPY(MAT1,M1,N1,MAT2,M2,N2,DIM1,DIM2)
CC
**********************************************************************
**   Given the matrix MAT1 (M1,N1), copies it into a matrix         **
**   MAT2 (M2,N2) and loses previous MAT2, where DIM1 and           **
**   DIM2 are real dimensions of MAT1.                              **
**********************************************************************
CC
      INTEGER DIM1,DIM2
      REAL MAT1(M1,N1), MAT2(M2,N2)
CC
      DO 10020 I = 1,M2,1
         DO 10030 J = 1,N2,1
            MAT2(I,J) = 0.0
10030    CONTINUE
10020 CONTINUE
CC
      DO 10000 I = 1,DIM1,1
         DO 10010 J = 1,DIM2,1
            MAT2(I,J) = MAT1(I,J)
10010    CONTINUE
10000 CONTINUE
CC
      RETURN
      END


      SUBROUTINE PMAT(MAT,M,N,DIM1,DIM2)
CC
**********************************************************************
**  Prints out a matrix MAT of size M by N.                         **
**********************************************************************
CC
      REAL MAT(M,N)
      INTEGER DIM1,DIM2
CC
      WRITE(6,5020)
CC
      DO 5000 I = 1,DIM1,1
         WRITE(6,5010) (MAT(I,J), J = 1,DIM2)
5010     FORMAT(20F6.0)
5015     FORMAT(/,' ',13(F9.3))
5000  CONTINUE
      WRITE(6,5020)
5020  FORMAT(/)
CC
      RETURN
      END
```

```
      SUBROUTINE DIFFS (DIFF,MD,ND,MAT1,M1,N1,MAT2,M2,N2,DIM1,DIM2)
CC
*****************************************************************
**  Given the matrix MAT1 (M1,N1) and matrix MAT2 (M2,N2),     **
**  returns their difference as DIFF (MD,ND).                  **
**  DIM1 and DIM2 are real dimensions for the three matrices.  **
*****************************************************************
CC
      INTEGER    DIM1,DIM2
      REAL       DIFF (MD,ND), MAT1(M1,N1), MAT2(M2,N2)
      INTEGER I, J
CC
      DO 11000 I = 1,DIM1,1
         DO 11010 J = 1,DIM2,1
            DIFF(I,J) = MAT1(I,J) - MAT2(I,J)
11010    CONTINUE
11000 CONTINUE
CC
      RETURN
      END


      SUBROUTINE MULT(MAT1,M1,N1,MAT2,M2,N2,PROD,M3,N3,II,KK,JJ)
CC
*****************************************************************
**  Given matrices MAT1 (M1,N1) and MAT2 (M2,N2), returns      **
**  their product as PROD (M3,N3) where II, KK, JJ are         **
**  are real real dimensions of the first two matrices.        **
*****************************************************************
CC
      REAL       MAT1(M1,N1), MAT2(M2,N2), PROD(M3,N3)
CC
      DO 7000 I = 1,II,1
         DO 7010 J = 1,JJ,1
            PROD(I,J) = 0.0
            DO 7020 K = 1,KK,1
               PROD(I,J) = PROD(I,J) + MAT1(I,K) * MAT2(K,J)
7020        CONTINUE
7010     CONTINUE
7000  CONTINUE
CC
      RETURN
      END
```

```
      SUBROUTINE SQROOT(AMAT,M5,T,M6)
CC
**********************************************************************
**  This subroutine performes the Cholesky Decomposition            **
**********************************************************************
CC
      DIMENSION AMAT(M5,M5),T(M5,M5),BMAT(10,10)
CC
        DO 45554 I=1,M5
        DO 45554 J=1,M5
45554   T(I,J)=0.
        DO 35553 I=1,M6
        DO 35553 J=1,M6
        IF(J.GT.I)GO TO 25552
        T(I,I)=0.
        IF(I.EQ.1)GO TO 65556
        DO 75557 K=1,I-1
        T(I,I)=T(I,I)+T(K,I)*T(K,I)
75557   CONTINUE
65556   T(I,I)=(AMAT(I,I)-T(I,I))**.5
        GO TO 35553
25552   T(I,J)=0.
        IF(I.EQ.1)GO TO 95559
        DO 53335 K=1,I-1
            T(I,J)=T(I,J)+T(K,I)*T(K,J)
53335   CONTINUE
95559   T(I,J)=(AMAT(I,J)-T(I,J))/T(I,I)
35553   CONTINUE
       DO 74447 I=1,M6
       DO 74447 J=1,M6
       BMAT(I,J)=0.
       DO 64446 K=1,M6
       BMAT(I,J)=BMAT(I,J)+T(K,I)*T(K,J)
64446  CONTINUE
CC       WRITE(6,84448)BMAT(I,J)
84448   FORMAT(' ',F15.6)
74447  CONTINUE
        RETURN
        END
```

```
      SUBROUTINE TRANP(MAT1,M1,N1,MAT2,M2,N2,DIM1,DIM2)
CC
**********************************************************************
**  Given the matrix MAT1 (M1,N1), returns its transposed           **
**  matrix MAT2 (M2,N2) where DIM1 and DIM2 are real                **
**  dimension of the matrix MAT1.                                   **
**********************************************************************
CC
      REAL        MAT1(M1,N1), MAT2(M2,N2)
      INTEGER     DIM1,DIM2
CC
      DO 8000 I = 1,DIM1,1
      DO 8000 J = 1,DIM2,1
         MAT2(J,I) = MAT1(I,J)
8000  CONTINUE
CC
      RETURN
      END


      SUBROUTINE IDVERT(MAT,MM,MATT,IIM)
CC
**********************************************************************
**  Given the matrix MAT (MM,MM), returns its inverse as            **
**  MATT (MM,MM), where IIM is the real dimension of MAT.           **
**********************************************************************
CC
      EXTERNAL F01ACE
CC
      REAL        MAT(MM,MM), MATR(17,16), ZF(16), BMAT(16,16),
     &            MATT(MM,MM)
      INTEGER IIM, AA, ALL
CC
      IFAIL = 1
      DO 16005 I = 1,IIM
         DO 16005 J = 1,IIM
            MATR(I,J) = MAT(I,J)
16005 CONTINUE
CC
      CALL F01ACE(IIM,X02AAE(AA),MATR,   17,BMAT, 16,ZF,ALL,IFAIL)
      WRITE(6,16006) IFAIL
16006 FORMAT(' ',I3)
CC
      DO 16007 I = 1,IIM
      DO 16007 J = 1,I
         MATT(I,J) = MATR(I+1,J)
      IF (I .EQ. J) GO TO 16007
         MATT(J,I) = MATR(I+1,J)
16007 CONTINUE
CC
      RETURN
      END
```

```
      SUBROUTINE SWOP(MATT,N,M,DIM1,NUM)
CC
**********************************************************************
**  Given the matrix MAT (N,M), exchanges the first column      **
**  and the one in column NUM and returns the swopped matrix    **
**  MATT, where DIM1 is the real row dimension.                 **
**********************************************************************
CC
      REAL MATT(N,M), TEMP(100)
      INTEGER DIM1
CC
      DO 73730 ROW = 1, DIM1
         TEMP(ROW) = 0.0
         TEMP(ROW) = MATT(ROW,1)
         MATT(ROW,1) = MATT(ROW,NUM)
         MATT(ROW,NUM) = TEMP(ROW)
73730 CONTINUE
CC
      RETURN
      END


      SUBROUTINE SWOPPY(ARRAY1,ARRAY2,SUBS)
CC
**********************************************************************
**  Given the arrays ARRAY1 and ARRAY2 both of dimension SUBS,  **
**  returns the arrays ARRAY1 and ARRAY2 with values in         **
**  ascending order.                                            **
**********************************************************************
CC
      INTEGER SUBS
CC
      REAL ARRAY1(SUBS), TEMM
CC
      INTEGER TEMM2, SWOPS, ROW, J, ARRAY2(SUBS)
CC
      J = 0
      SWOPS = 1
3000  IF (SWOPS .NE. 1 .AND. J .GT. SUBS-1) GO TO 3020
      SWOPS = 0
      J = J + 1
CC
      DO 3010 ROW = 1, SUBS-1
CC
         IF (ARRAY1(ROW) .GT. ARRAY1(ROW+1)) THEN
            TEMM = ARRAY1(ROW)
            ARRAY1(ROW) = ARRAY1(ROW+1)
            ARRAY1(ROW+1) = TEMM
```

```
            TEMM2 = ARRAY2(ROW)
            ARRAY2(ROW) = ARRAY2(ROW+1)
            ARRAY2(ROW+1) = TEMM2
CC
            SWOPS = SWOPS + 1
         ENDIF
CC
3010  CONTINUE
CC
      GO TO 3000
3020  CONTINUE
CC
      RETURN
      END


      SUBROUTINE CNTRAL(MAT,M,N,MATOR,M1,N1,DIM1,DIM2)
CC
*****************************************************************
**  Given the matrix MATOR (M1,N1), returns the standardized   **
**  matrix MAT (M,N), where DIM1 and DIM2 are real dimensions  **
**  of the two matrices.                                       **
*****************************************************************
CC
      REAL        MAT(M,N), MATOR(M1,N1)
      INTEGER     DIM1, DIM2
      REAL        AVE(25)
CC
      DO 6000 J = 1,DIM2,1
         AVE(J) = 0.0
         DO 6010 I = 1,DIM1,1
            AVE(J) = AVE(J) + MATOR(I,J)
6010     CONTINUE
         AVE(J) = AVE(J) / FLOAT(DIM1)
6000  CONTINUE
CC
      DO 6020 I = 1,DIM1,1
         DO 6030 J = 1,DIM2,1
            MAT(I,J) =MATOR(I,J) - AVE(J)
6030     CONTINUE
6020  CONTINUE
CC
      RETURN
      END
```

```
      REAL FUNCTION INDEX(RATTAR,RATCON,NOBS,NMONTH)
CC
**********************************************************************
**  This function calculates the index between the stations.    **
**********************************************************************
CC
      REAL RATTAR(NOBS), RATCON(NOBS)
CC
      INTEGER NMONTH, YEAR, NROW
CC
      NROW = 0
      YEAR = 0
      INDEX = 0.0
 900  NROW = NROW + 1
      IF (NROW .GT. NOBS) GO TO 1030
      IF (NROW .GT. (NMONTH*YEAR)) YEAR = YEAR + 1
         IF (RATTAR(NROW) .EQ. -999.0 .OR.
     &       RATCON(NROW) .EQ. -999.0) THEN
             NROW = NMONTH * YEAR
             GO TO 900
          ELSE
             INDEX = INDEX + (RATTAR(NROW) - RATCON(NROW)) ** 2
             GO TO 900
          ENDIF
CC
1030  RETURN
      END



      SUBROUTINE CENTRE(MAT,M,N,DIM1,DIM2)
CC
**********************************************************************
**  Given a matrix MAT (M,N), returns it being standardized     **
**********************************************************************
CC
      REAL        MAT(M,N)
      INTEGER     DIM1, DIM2
      REAL        AVE(25)
      DO 6000 J = 1,DIM2,1
         AVE(J) = 0.0
         DO 6010 I = 1,DIM1,1
            AVE(J) = AVE(J) + MAT(I,J)
6010     CONTINUE
         AVE(J) = AVE(J) / FLOAT(DIM1)
6000  CONTINUE
CC
      DO 6020 I = 1,DIM1,1
         DO 6030 J = 1,DIM2,1
            MAT(I,J) = MAT(I,J) - AVE(J)
6030     CONTINUE
6020  CONTINUE
      RETURN
      END
```

```
      SUBROUTINE AGGREG(Z,RATIO,NOBS,ZZ,NYEAR,NSTAT,NMONTH)
CC
**********************************************************************
**  Given an annual rainfall totals matrix Z and a monthly          **
**  rainfall totals matrix ZZ, returns a matrix RATIO, which        **
**  contains the ratios in each station.                            **
**********************************************************************
CC
      INTEGER NROW, YEAR, NMONTH, COL, ROW, MMONTH, PMONTH
CC
      REAL Z(NOBS,NSTAT), ZZ(NYEAR,NSTAT), RATIO(NOBS,NSTAT)
CC
      DO 100 COL = 1, NSTAT
         NROW = 0
         YEAR = 0
         DO 150 ROW = 1, NYEAR
            ZZ(ROW,COL) = 0.0
 150     CONTINUE
CC
 300     NROW = NROW + 1
         IF (NROW .GT. NOBS) GO TO 100
         IF (NROW .GT. (NMONTH * YEAR)) YEAR = YEAR + 1
         MMONTH = NMONTH * YEAR
         PMONTH = MMONTH - NMONTH + 1
         IF (Z(NROW,COL) .LT. 0.0) THEN
            ZZ(YEAR,COL) = -999.0
            NROW = MMONTH
            DO 200 ROW = PMONTH , MMONTH
               RATIO(ROW,COL) = -999.0
 200        CONTINUE
            GO TO 300
         ELSE
            ZZ(YEAR,COL) = ZZ(YEAR,COL) + Z(NROW,COL)
            IF (NROW .EQ. MMONTH) THEN
               DO 250 ROW = PMONTH, MMONTH
                  RATIO(ROW,COL) = Z(ROW,COL) / ZZ(YEAR,COL)
 250           CONTINUE
            ENDIF
            GO TO 300
         ENDIF
CC
 100  CONTINUE
CC
      RETURN
      END
```

```
      REAL FUNCTION GRANN(ISED,SDV,AMEAN)
      GRANN=0.
      DO 3033 I=1,12
         GRANN=GRANN+URAND(ISED)
3033  CONTINUE
      GRANN=(GRANN-6.0)*SDV+AMEAN
      RETURN
      INCLUDE UCT*ASCII.URAND
      END


      SUBROUTINE INDMAT(RATIO,NOBS,VECTOR,NSTAT,NMONTH)
CC
**********************************************************************
**  Given the ratio RATIO of monthly observations in each       **
**  station, returns a vector VECTOR of ordered index           **
**  between the stations.                                       **
**********************************************************************
CC
      REAL  RATTAR(1200), RATCON(1200)
      REAL RATIO(NOBS,NSTAT)
      REAL IND(15), INDEX
CC
      INTEGER NOBS, ROW, COL, NMONTH, NSTAT
      INTEGER J, JJ, NUMBER(15), VECTOR(NSTAT,NSTAT)
CC
      DO 900 COL = 1, NSTAT
         DO 910 ROW = 1, NOBS
            RATTAR(ROW) = RATIO(ROW,COL)
 910     CONTINUE

         JJ = 0
         DO 920 J = 1, NSTAT
            IF (J .NE. COL) THEN
               JJ = JJ + 1
               DO 930 ROW = 1, NOBS
                  RATCON(ROW) = RATIO(ROW,J)
 930           CONTINUE
               IND(J) = INDEX(RATTAR,RATCON,NOBS,NMONTH)
            ELSE
               IND(J) = 999.0
            ENDIF
            NUMBER(J) = J
 920     CONTINUE
CC
      CALL SWOPPY(IND,NUMBER,NSTAT)
CC
       DO 879 J = 1, NSTAT
          VECTOR(COL,NUMBER(J)) = J
 879   CONTINUE
CC
 900  CONTINUE
CC
      RETURN
      END
```

```
      SUBROUTINE DATGEN(Y,DISTAN,MINIM,A1,A2,ISEED)
CC
*****************************************************************
**  Given a matrix of distances between the stations DISTAN,   **
**  the minimum distance MINIM, correlations A1 and A2, and    **
**  an integer ISEED, returns a multivariate normally          **
**  distributed data matrix Y with mean 1000 and standard      **
**  deviation 200.                                             **
*****************************************************************
CC
      REAL SIGMA(10,10), Z(1200,10), V(10,10)
      REAL Y(1200,10), ALPHA, BETA
      REAL DISTAN(10,10), MINIM
CC
      INTEGER DV, ISEED
CC
      DATA NOBS, DV/1200, 10/
CC
      BETA = (-1.0) * (ALOG(A2) - ALOG(A1)) / (70.0 - MINIM)
      ALPHA = EXP(ALOG(A1) + (BETA * MINIM))
CC
      CALL SIGG(SIGMA,10,10,DV,DV,DISTAN,ALPHA,BETA)
CC
      CALL SQROOT(SIGMA,10,V,DV)
CC
      NNN = 0
      DO 122 I = 1, NOBS
      NNN = NNN + 1
      DO 100 J = 1, DV
         Z(I,J)=GRANN(ISEED,1.,0.)
 100  CONTINUE
CC
      DO 135 J = 1, DV
         Y(NNN,J) = 0.0
         DO 131 K = 1, DV
            Y(NNN,J) = Y(NNN,J) + Z(I,K) * V(K,J)
 131     CONTINUE
 135  CONTINUE
CC
      DO 110 J = 1, DV
         Y(NNN,J) = 1000 * Y(NNN,J) + 200
 110  CONTINUE
CC
 122  CONTINUE
CC
      RETURN
      END
```

```
      SUBROUTINE SIGG(MATT,NN,MM,N1,M1,DISTAN,ALPHA,BETA)
CC
*****************************************************************
**  Given a matrix DISTAN which contains the distances between **
**  the stations, ALPHA and BETA, returns a matrix MATT of     **
**  correlation coefficients between the stations.             **
*****************************************************************
CC
      INTEGER L, K, N1, M1
CC
      REAL CORR(10,10), DISTAN(10,10)
      REAL MATT(NN,MM), ALPHA, BETA
CC
      DO 530 K = 1, N1
      DO 530 L = 1, M1
CC
      IF (K .EQ. L) THEN
         CORR(K,L) = 1.0
      ELSE
         CORR(K,L) = ALPHA * EXP(((-1.0) * BETA) * DISTAN(K,L))
      ENDIF
CC
530   CONTINUE
CC
      CALL COPY(CORR,10,10,MATT,NN,MM,N1,M1)
CC
      RETURN
      END


      SUBROUTINE STANCE(DIST,NOSTAT,MINIM)
CC
*****************************************************************
**  Given the number of stations NOSTAT, returns a matrix      **
**  DIST (NOSTAT,NOSTAT) which contains the distances between  **
**  the different stations.                                    **
*****************************************************************
CC
      REAL X(10), DIST(NOSTAT,NOSTAT), Y(10)
      REAL MINIM
CC
      DO 897 I = 1, NOSTAT
         X(I) = URAND(35)
         Y(I) = URAND(155)
 897  CONTINUE
```

```
      MINIM = 999.0
      DO 600 I = 1, NOSTAT
      DO 600 J = 1, NOSTAT
CC
         DIST(I,J) = 70. * ((X(I) - X(J)) ** 2.
     &             + (Y(I) - Y(J)) ** 2.) ** 0.5
CC
         IF (I .EQ. J) GO TO 600
          MINIM = AMIN1(DIST(I,J),MINIM)
CC
 600  CONTINUE
CC
      RETURN
      END


      SUBROUTINE INVERT(MATT,NN,MM)
CC
**********************************************************************
**  Given a matrix MATT (NN,MM), returns it being an inverse        **
**********************************************************************
CC
      REAL MATT(NN,NN), INVER(25,25)
      REAL MATR1(25,25)
CC
      II = 0
  20  II = II + 1
CC
      MATR1(II,II) = 1.0 / MATT(II,II)
CC
      DO 40 J = 1, MM
         DO 30 I = 1, MM
CC
            IF (J .EQ. II .AND. I .EQ. II) THEN
               INVER(I,J) = (-1.0) * MATR1(II,II)
CC
            ELSEIF (J .EQ. II .AND. I .NE. II) THEN
               INVER(I,J) = MATT(I,J) * MATR1(II,II)
CC
            ELSEIF (I .EQ. II .AND. J .NE. II) THEN
               INVER(I,J) = MATT(I,J) * MATR1(II,II)
CC
            ELSE
               INVER(I,J) = MATT(I,J) - ((MATT(I,II) *
     &                      MATT(II,J)) * MATR1(II,II))
            ENDIF
CC
  30     CONTINUE
  40  CONTINUE
CC
      CALL COPY(INVER,25,25,MATT,NN,NN,MM,MM)
CC
      IF (II .LT. MM) GO TO 20
CC
      RETURN
      END
```