A Manual for EOF and SVD Analyses of Climatic Data

H. Björnsson and S.A. Venegas

C^2GCR Report No. 97-1
February 1997

A Manual for EOF and SVD Analyses of Climatic Data

H. Björnsson and S. A. Venegas
Department of Atmospheric and Oceanic Sciences
and
Centre for Climate and Global Change Research
McGill University

February 1997

Contents

Chapter 1

Introduction

1.1 What's it all about

This manual contains a fairly detailed discussion of two methods for analyzing the spatial and temporal variability of geophysical fields. They are the method of Empirical Orthogonal Functions (EOFs), also known as Principal Component Analysis[1] (PCA), and the method of Singular Value Decomposition (SVD).

The description given here is by no means complete. Those who want a more detailed description should read some of the references. For the EOF/PCA method, the "big book" is Preisendorfer (1988). It contains a wealth of information about the method, ranging from its history to detailed analyses of the method. For the SVD method the prime reference is the article by Bretherton et al. (1992), who compare several different methods for detecting coupled patterns between different components of the climate system. Another reference that everyone who is going to study climatic variability should be familiar with, is the text book edited by von Storch and Navarra (1995). This book not only contains a review of methods for the analysis of patterns in geophysical fields, but has also examples of the application of those methods (see chapters 11 - 15) . The scope of the book is quite wide, and many methods not mentioned here are covered there thoroughly. Furthermore, the book contains an entertaining chapter on the misuse of statistical analysis in climate research. For

[1] See §1.1.2 for terminology.

readers with little experience in climatic data analyses, a very good introduction of the basic methods is given in the text book by Thiebaux (1994).

All of the references mentioned above are fairly mathematical, and do not give detailed descriptions of how to perform the analyses. Thus these texts may be somewhat "unfriendly" for beginners. This manual is written for those who are beginning to use the EOF and SVD methods, and has the explicit aim of giving a step-by-step "How to do" recipe for the methods. For those wanting an intuitive understanding, we try to give hand-waving explanations of how the methods work. For beginners with mathematical inclinations we also include some mathematical results, but those should be easy for everyone familiar with matrix algebra. Those who do not care much for the mathematics can simply omit the sections that they feel are too mathematical.

As well as giving recipes for the methods, we also give Matlab scripts for them. Matlab is in many ways the ideal tool for matrix-based analysis methods. Matlab is a product of The MathWorks Company and is widely used among researchers in science and engineering. Matlab is not a free product, but there are several look-a-likes available for free over the Internet. Among these is Octave which is highly compatible with Matlab, and the French SciLab which is a comprehensive toolkit available for free from Institut National de Recherche en Informatique et en Automatique (INRIA). These packages are similar enough to Matlab so that the scripts given in the manual should be easily translated.

1.1.1 When to use each method

The EOF/PCA method is the method of choice for analyzing the variability of a single field, i.e. a field of only one scalar variable (SLP, SST, etc). The method finds the spatial patterns of variability, their time variation, and gives a measure of the "importance" of each pattern.

For examining the coupled variability of two fields we recommend using the SVD method. Coupled field analyses can be performed using coupled EOFs, but these methods have not been popular, since the output can be difficult to interpret. The SVD analysis of two fields together will identify *only* those modes of behavior in which the variations of the two fields are strongly coupled.

It should be stressed that even though the methods presented below break the data into

"modes of variability", these modes are primarily *data modes*, and not necessarily *physical modes*. Whether they are physical will be a matter of subjective interpretation. A focussed discussion of this for the SVD method can be found in Newman and Sardeshmukh (1995).

The methods described herein are fairly advanced. Before using them it might be sensible to first use some simple methods, such as composite analysis, analysis of anomaly maps, time series of chosen grid points, correlation analyses etc., to get an idea of the variability in the data. Maybe the most useful first step is to plot a time sequence of contour plots for a geophysical field, to get "some intuition" for the variability in the data. Furthermore, once patterns have been found using the EOF or SVD methods, it is wise to re-examine the data with simpler methods and verify that the signal is indeed there, and not just a result of the advanced method of analysis. This is emphasized in von Storch and Navarra (p. 25):

> I have learned the following rule to be useful when dealing with advanced methods. Such methods are often needed to find a signal in a vast noisy phase space, i.e. the needle in the haystack. But after having the needle in our hand, we should be able to identify the needle by simply looking at it. Whenever you are unable to do so there is a good chance that something is rotten in the analysis.

1.1.2 Terminology

The literature is quite confusing when it comes to differences between the EOF method and the PCA method. Although some authors (Richman, 1985) define the two methods differently, others seem to use PCA and EOF to describe the same method. As an example, the EOF method described in the appendix of Peixoto and Oort (1992) is identical to the PCA method described by Preisendorfer (1988). Since the literature is more or less in a state of confusion on the difference of these methods, we choose to use the phrases EOF and PCA interchangeably.

The EOF method finds both time series and spatial patterns. Most authors refer to the patterns as the "EOFs", but some refer to them as the "Principal component loading patterns" or just "principal components". (Confusing? Just read on!) The time series are referred to as "EOF time series", "expansion coefficient time series", "expansion coefficients",

"principal component time series" or just "principal components". In this manual we will refer to the patterns as the EOFs and the time series as "principal components" or "expansion coefficients".

EOF analysis has been extensively used to examine variability of scalar (simple) fields, such as SLP, SST, SAT, 500Z, etc. Even though the method is easily extended to include coupled fields (e.g. matrices containing maps of SST and SLP simultaneously) this has not enjoyed much popularity. However, it has recently become popular to use the SVD analysis on two coupled fields. This usage of the term "SVD" is in conflict with the original meaning of the term as a general name for a particular form of matrix decomposition. We will see in §2.3.1 how the mathematical operation of singular value decomposition can yield insight into EOF analyses. In this manual we will use the abbreviation SVD *only* for the analysis of two fields. In §2.3.1, where we discuss the connections between EOF analyses and singular value decomposition, we will not use the abbreviated name.

In this manual we only present methods to analyze real valued fields. This means that the patterns of variability given by the EOF or SVD methods represent standing oscillations, and not propagating patterns. It is easy to extend the EOF method to include complex fields, which is useful for finding moving patterns in data (see Preisendorfer(1988) for details). Other methods which are also useful for finding propagating patterns include the "Principal Oscillation Patterns" (POPs) and "Principal Interaction Patterns" (PIPs). For details on these see von Storch and Navarra (1995).

1.2 The Data

The analysis methods to be described are essentially matrix methods. The following section describes how the manual assumes that the data to be analyzed are already arranged into matrices. The procedure for doing this is similar whether we are preparing to analyze a single field (using the EOF method) or are preparing to analyze the coupling between two fields (e.g. by using SVD).

Let us assume that we have measurements of some variable at locations $x_1, x_2, \ldots x_p$ taken at times $t_1, t_2, \ldots t_n$. For each time t_j $(j = 1, \ldots n)$ we can think of the measurements

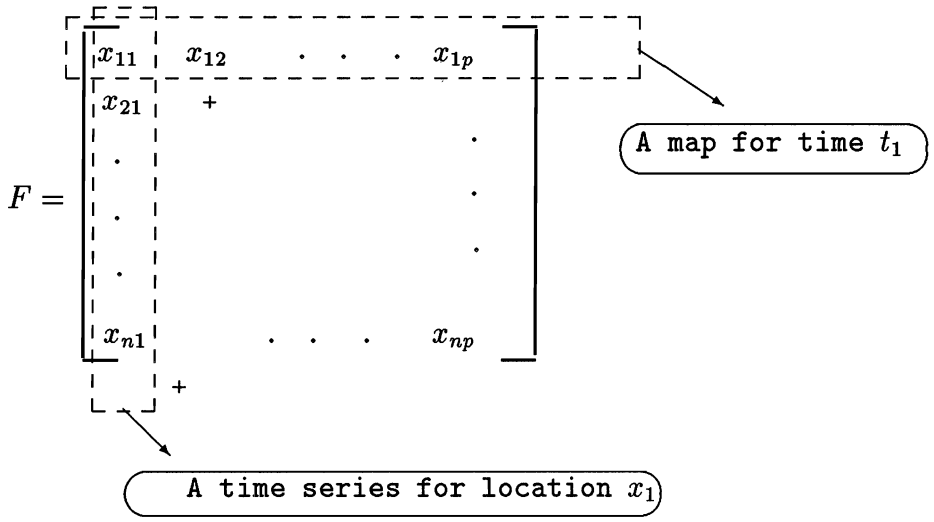

Figure 1.1: The matrix F. Each row is one map, and each column is a time series of observations for a given location..

x_i $(i = 1, \ldots p)$ as a map or field.

We store these measurements in a matrix F as n maps each being p points long. We arrange each map into a row vector in F so that the size of F becomes n by p. We can then interpret each of the p columns in F as a time series for a given location (see Figure 1.2). The EOF analysis is performed using F as the data matrix.

In the above example the variable in F might be sea level pressure (SLP) over an ocean basin. We might also be interested in examining the coupling between the SLP and sea surface temperature (SST) measured at locations $y_1, y_2, \ldots y_r$ taken at times $t_1, t_2, \ldots t_n$. These SST measurements are then ordered into another matrix S, in the same fashion as before, and the two matrices, F and S, are analyzed using the SVD method. Notice that the location of these measurements of the SLP and the SST does not have to be the same, but in this manual we only discuss analyses where the times t_j are the same (the fields are measured simultaneously).

This way of ordering (time,position) data into a matrix is referred to as S-mode analyses. This is the only mode of EOF analyses discussed in this manual. For definitions of other modes of analysis the reader is referred to the literature (Preisendorfer, 1988; Richman, 1985).

Chapter 2

EOF of a single field

2.1 How to do it

Let us assume that we have removed the mean from each of the p time series in F, so that each column has zero mean.[1]

We form the *covariance matrix* [2] of F by calculating $R = F^t F$, and then we solve the eigenvalue problem [3]

$$RC = C\Lambda. \tag{2.1}$$

Λ is a diagonal matrix containing the eigenvalues λ_i of R. The c_i column vectors of C are the eigenvectors of R corresponding to the eigenvalues λ_i. Both Λ and C are of the size p by p.

For each eigenvalue λ_i chosen we find the corresponding eigenvector c_i. Each of these eigenvectors can be regarded as a map. These eigenvectors are the EOFs we were looking

[1] Removing the time means has nothing to do with the process of finding eigenvectors, but it allows us to interpret R as the covariance matrix, and hence understand the results. Strictly speaking you can find EOFs without removing any mean, or you can remove the mean of each map, but not each time series.

[2] There are several variations of this definition, Peixoto & Oort (1992, appendix B) use $R = \frac{1}{N} F^t F$ (actually $R = \frac{1}{N-1} F^t F$ is more correct) but Preisendorfer uses the one given above. This does not matter much, since the EOFs and their time series will be the same, to within a constant factor

[3] Since R is a quadratic matrix, C will have the property that $C^{-1} = C^t$ so eqn. 2.1 can just as well be written $R = C\Lambda C^t$

for. In what follows we always assume that the eigenvectors are ordered according to the size of the eigenvalues[4]. Thus, EOF1, is the eigenvector associated with the biggest eigenvalue, and the one associated with the second biggest eigenvalue is EOF2, etc. Each eigenvalue λ_i, gives a measure of the fraction of the total variance in R explained by the mode. This fraction is found by dividing the λ_i by the sum of all the other eigenvalues (the trace of Λ).

The eigenvector matrix C has the property that $C^tC = CC^t = I$ (where I is the identity matrix). This means that the EOFs are *uncorrelated* over space. Another way of stating this is to say that the eigenvectors are orthogonal to each other. Hence the name Empirical Orthogonal Functions (EOF).

The pattern obtained when an EOF is plotted as a map, represents a standing oscillation. The time evolution of an EOF shows how this pattern oscillates in time. To see how EOF1 'evolves' in time we calculate:

$$\vec{a_1} = F\vec{c}_1 \tag{2.2}$$

The n components of the vector $\vec{a_1}$ are the projections of the maps in F on EOF1, and the vector is a time series for the evolution of EOF1. In general, for each calculated EOFj, we can find a corresponding $\vec{a_j}$. These are the *principal component time series* (PC's) or the *expansion coefficients* of the EOFs. Just as the EOFs where uncorrelated in space, the expansion coefficients are uncorrelated in time[5].

We can reconstruct the data from the EOFs and the expansion coefficients as follows:

$$F = \sum_{j=1}^{p} \vec{a}_j(EOF_j) \tag{2.3}$$

A common use of EOFs is to reconstruct a 'cleaner' version of the data by truncating this sum at some $j = N \ll p$; that is, we only use the EOFs of the first (largest) few eigenvalues.

[4]Computer routines that solve Eqn. 2.1 do not necessarily order the output in this manner. As an example the Matlab `eig` routine puts the eigenvector associated with the largest eigenvalue in the last column.

[5]A note on terminology: Some authors refer to the EOFs as the PC's, and the expansion coefficients as "the eigenvector time series". Preisendorfer makes a distinction between the EOFs and the PC's and uses the term principal components (PC's) for the expansion coefficients. We will follow his definition, but mostly use the the term "expansion coefficients" throughout the manual and reserve the use of the words "time series" for time series of the actual data.

The rationale is that the first N eigenvectors are capturing the dynamical behavior of the system, and the other eigenvectors (corresponding to the smallest eigenvalues) are just due to random noise. Needless to say this assumption does not always have to be true.

We have a simple procedure for the EOF method:

- Form matrix F from the observations, and remove the time mean of each time series.
- Find the covariance matrix $R = F^t F$
- Find the eigenvalues and eigenvectors of R by solving $RC = C\Lambda$
- Find the biggest eigenvalues and their corresponding eigenvectors, the EOFs.
- Find the expansion coefficients by calculating $\vec{a_j} = F \times \text{EOF}j$ (the projection of F onto the j-th EOF).

Usually the EOFs are presented as dimensionless maps, often normalized so that the highest value is 1, or 100. This means that if the associated expansion coefficients are also to be presented, then they have to be adjusted correspondingly. The simplest way to do this is to calculate the expansion coefficient after having normalized the eigenvector. Another way to represent an EOF is by calculating the correlation map between the expansion coefficients associated with the eigenvector, and the data F. This method has some advantages over the other method, and is discussed in more detail in §4.1.

Doing it with Matlab

1 Put the data is in a matrix `M`, with each row as one map, and each column a time series for a given station.

2 Remove the mean of each column:

```
F=detrend(M,0);
```

3 Form the covariance matrix:

```
R=F'*F;
```

4 Calculate the eigenvalues and eigenvectors of the covariance matrix:

```
[C,L] = eig(R)
```

The eigenvalues will be on the diagonal of `L` and the corresponding eigenvectors will be the column vectors of `C`.

5 For a shortcut of the above, do:

```
[C,L] = eig(cov(M))
```

If this is done then the Matlab definition of a covariance matrix is used $\left(R = \frac{1}{N-1}F^tF\right)$. This will show up as a constant factor in the eigenvalues.

6 Find the expansion coefficients corresponding to eigenvalue number `i`:

```
PCi= F * C(:,i)
```

7 A vector giving the 'amount of variance explained' for each eigenvalue is found by:

```
diag(L)/trace(L)
```

2.2 How to understand it

Even though the method is straightforward to apply, how it works is not as easy to comprehend. This section will try to explain the idea behind the method. Here we present two explanations, one technical, and one of the hand-waving kind.

2.2.1 The technical part

If every row in F is one map, think of that row vector f_n, as the position vector for a point in p dimensional space. Every observation (there are n of them) makes one point in that p-dimensional space. If the observations are totally random, the points would describe a 'blob' in p-space. Any regularities in the data would make the points organized into clusters, or along preferred directions. We might be able to define a new coordinate system for our p-dimensional space so that each cluster of data points would have a coordinate axis of the new system going right through it. This is what we try to do with the EOF analysis. What

the method does is find a set of orthonormal basis vectors e_i that maximizes the projection of the f_n onto the basis vectors.

Mathematically, the problem is to maximize

$$\sum_{i=1}^{n}(e_m f_i)^2 \tag{2.4}$$

for $m = 1, \cdots, p$, subject to an orthonormality condition on e_m:

$$e_i^t e_j = \delta_{ij}. \tag{2.5}$$

First notice that

$$\sum_{i=1}^{n}(e_m f_i)^2 = e_m F^t F e_m^t = e_m R e_m^t.$$

The maximization of the quantity (2.4) subject to the constraint (2.5) gives

$$\nabla(\vec{x^t} R \vec{x}) - \lambda \nabla(\vec{x^t}\vec{x}) = 0. \tag{2.6}$$

where ∇ is the gradient operator and λ is a Lagrange multiplier. A little bit of manipulation (where we use the symmetry[6] of R) then yields

$$R\vec{x} = \lambda\vec{x}, \tag{2.7}$$

which is just another form of (2.1), the eigenvalue problem for R. Thus we see how the eigenvectors of R arise naturally when we try to find a new coordinate system 'along' regularities in the data. (For what to do when the regularities in the data are not aligned in orthogonal directions, see section §2.4.)

The above does not explain how we interpret the EOFs. That is done by focusing on R as the covariance matrix. Since R is symmetric, it follows[7] that the eigenvalues (λ_i) and eigenvectors (the EOFs, call them c_i) decompose R according to:

$$R = \lambda_1 c_1 c_1^t + \lambda_2 c_2 c_2^t + \cdots + \lambda_p c_p c_p^t \tag{2.8}$$

[6]To see why R must be symmetric see section 2.3.

[7]This follows from a famous theorem by Hilbert, that is commonly referred to as the *spectral representation theorem*

It is this decomposition that is the basis for statements like "EOF 1 explains 30% of the variance", (in that case $\lambda_1 / \sum_i \lambda_i = 0.3$).

Often the first few (in numerical order) of the λ_i dominate the others. This means that most of the behavior in our data matrix can be explained by just a few basis vectors. This is exactly what we hoped for in EOF calculations; i.e. we wanted to use the method to reduce the data to a few different modes of variability. When this happens, let's say that of the p eigenvalues of R, only $k \ll p$ eigenvalues are large. We know that the data is organized into a small sub-space of the p-dimensional space. Technically we could say that the phase-space of the system that generated the data is k rather than p dimensional. EOF analyses, and all its relatives (such as Empirical Normal Modes, Fourier decomposition, Wavelet - decomposition etc.) are all methods that try to reduce the size of a phase space. To examine this correspondence between the methods in more detail would take us far beyond the purpose of this manual.

2.2.2 The hand-waving part

The frustrated reader is now thinking: "This is not an explanation, this is just a mathematical way around an explanation!"- So, let's do some real hand-waving: Imagine a contour map of a field that evolves in time. As time passes the contours change, and dance around the map. The EOF method is a 'map-series' method of analysis that takes all the variability in the time evolving field and breaks it into (hopefully) a few standing oscillations and a time series to go with each oscillation. Each of these oscillations (each EOF) is often referred to as a *mode of variability* and the expansion coefficients of the mode (the PC) show how this mode oscillates in time.

Let us take a physical (and unrealistic) example. Suppose we have Pacific SST data, one SST map per month sampled over a period of few decades. We perform an EOF analysis and find that most of the data is explained by 2 eigenvectors only. The first eigenvector (EOF1) is a map of contours that are positive in the northern hemisphere and negative in the southern hemisphere. The corresponding PC is periodic with period of about 12 months. We therefore identify EOF1 as the annual cycle. EOF2 shows many contour lines close to the equator but few elsewhere and the corresponding PC is quasi periodic with a period of

few years. We identify this eigenvector to be associated with El Niño. From this data we would get the following picture of Pacific SSTs and their evolution in time: Most of what happens is described by the SST change associated with the annual cycle and the slower SST change associated with El Niño. And since no other eigenvectors contribute much (all other eigenvalues are small) we assume that everything else found in the observations is just noise. The EOF method thus allows us to view all the complicated variability in the original SST data and explain it by two processes.

In practice it would be inadvisable to keep the seasonal cycle in our data, since that signal is likely to dominate everything else. Therefore, we would begin our analysis by removing the seasonal cycle from the data using an appropriate filtering method.

2.3 Some useful matrix algebra

This section deals with some matrix relations that are fundamental to EOF analyses. Readers who are not keen on matrix algebra can omit this section.

From the definition of the covariance matrix, namely,

$$R = F^t F,$$

it should be obvious that R is symmetric. Another fact which follows from F being real is that R is positive definite. What this means is that all the eigenvalues of R are greater than or equal to zero. Apart from having mathematical implications, this fact also has practical consequences:

If you are getting negative eigenvalues you are certainly doing something wrong!!! [8]

From the symmetry of R we get

$$C^t C = I,$$

where I is the identity matrix. This is actually the orthogonality constraint of Eqn. (2.5). For real matrices this property only holds if the matrix is symmetric, which implies that the

[8] To see that negative eigenvalues would be wrong think of the following situation: R has the eigenvalues $10, 2$ and -10. Then the amount of variance explained by EOF1 is $10/(10+2-10) = 10/2 = 5$ (or 500%), which is clearly absurd.

maximization of (2.4) subject to (2.5) is only possible due to the symmetry of R. Another consequence of the orthogonality of C is that the eigenvalue equation for R, (2.1), can be written as $R = C\Lambda C^t$. This can, of course, also be written as $\Lambda = C^t R C$, a seemingly trivial point, except that since Λ is a diagonal matrix, we see that the EOF basis gives a diagonalizing linear transformation of R.

If we arrange the expansion coefficients $\vec{a}_j$ as the columns of a matrix A, we can rewrite Eqn. (2.2) as

$$A = FC.$$

Multiplying this by C^t and using Eqn (2.5), we obtain

$$F = AC^t.$$

This formula is the basis for the reconstruction of the data[9]:

$$F(t,x) = \sum_{allEOFs} a_j(t) EOF_j(x).$$

Another important formula is the *PCA property* of R (Preisendorfer, 1988). It arises when we look at the covariance structure of A:

$$A^tA = (FC)^t(FC) = C^tF^tFC = C^tRC = C^tC\Lambda C^tC = \Lambda \tag{2.9}$$

Here we have shown that because the EOFs are an orthogonal diagonalizing transformation, this implies that the expansion coefficients of the EOFs are uncorrelated in time. In general, there are infinitely many different sets of orthonormal basis vectors for R. The eigenvector basis is, however, the only one that has the PCA property (the only diagonalizing one). Thus if we change our basis vectors by rotating the eigenvectors, for instance (see §2.4), we loose the PCA property.

We have another test of our calculations:

If A^tA is not equal to Λ you are doing something wrong !!!

[9]This formula is sometimes referred to as a *generalized Fourier decomposition.* Some readers will without doubt be familiar with a special case of the Fourier decomposition (the original one) involving cos and sin functions.

Note also that $AA^t = FC(FC)^t = FCC^tF^t = FF^t$. Since we know that F^tF is the temporal covariance of F, we can refer to FF^t as the "spatial covariance" [10]. Thus we see that the spatial covariance of the expansion coefficients is the same as the spatial covariance of the data.

We can take the decomposition of R a little bit further by noting that since the eigenvalues λ_i are all non-negative, the PCA property suggests we might define

$$\phi_j \equiv \frac{\vec{a}_j}{\sqrt{\lambda_j}} \tag{2.10}$$

and order the ϕ_j as the column vectors of a matrix Φ. One can think of the ϕ_j's as a way of 'normalizing' $\vec{a}$, the expansion coefficients of the EOFs. Note that we then have

$$\Phi^t\Phi = I,$$

and if we define a diagonal matrix D with $\sqrt{\lambda_j}$ on the diagonal (so $A = \Phi D$), we can write

$$F = AC^t = \Phi DC^t.$$

With the last equation we have derived what is called the *singular value decomposition* of F. This operation breaks F into a matrix with 'normalized' time series (Φ), a diagonal matrix (D) and a matrix of eigenvectors C. This operation is so important that we will dedicate a special section to it.

2.3.1 Singular value decomposition and the EOF method

It has been said that at the heart of the EOF method lies the technique of singular value decomposition. Singular value decomposition is a general decomposition, that decomposes any $n \times m$ matrix F into the form:

$$F = U\Gamma V^t, \tag{2.11}$$

where U is an $n \times n$ orthonormal matrix, V is an $m \times m$ orthonormal matrix and Γ is a diagonal $n \times m$ matrix with ρ elements down the diagonal ($\Gamma_{i,j} = \delta_{i,j}\gamma_{i,i}$ for $i = 1, \rho$ with

[10] Strictly speaking it is not a covariance, since F is centered in time, not space.

$\rho \leq min(n, m)$). The numbers $\gamma_{i,i}$ are called the singular values of the matrix. The columns of the matrices U, V contain the singular vectors of F.

Note that if we have ρ singular values, and ρ is less than both n and m, then some of the singular vectors are redundant, and we can just as well write:

$$F = U_\rho \Gamma_\rho V_\rho^t.$$

where $U = (U_\rho, U_0)$, $V = (V_\rho, V_0)$, and Γ_ρ is a $\rho \times \rho$ diagonal matrix with the singular values on the diagonal. The matrices U_0 and V_0 are the *kernel*, or *null space* of F.

If the matrix F is square and invertible then $U = V$ and Γ is the diagonal matrix containing the eigenvalues.

To see further the connection between the EOF method and singular value decomposition, consider the following. F is the data matrix, with the means removed, and the covariance matrix R is defined as $R = F^t F$. The EOF method then yields Λ such that

$$R = C \Lambda C^t.$$

But if we first do a singular value decomposition on F, and then form R we get

$$R = F^t F = (U \Gamma V^t)^t (U \Gamma V^t) = V \Gamma^t U^t U \Gamma V^t = V \Gamma^t \Gamma V^t.$$

By comparing the two formulas above it should be clear that $C = V$ and $\Lambda = \Gamma^t \Gamma$. The relationship between the eigenvalues λ_i of R and the singular values $\gamma_{i,i}$ of F is obviously

$$\lambda_i = \gamma_{i,i}^2$$

.

We should also note that while the column vectors of V contain the eigenvectors for $F^t F$, the column vectors of the matrix U contain the eigenvectors for $F F^t$, which also happens to be the normalized time series of the preceding section. So we have a another option for calculating the EOFs:

- Use singular value decomposition to find U, Γ and V such that $F = U \Gamma V^t$.
- The eigenvalues λ_i of $R = F^t F$ are $\gamma_{i,i}^2$.

- The eigenvectors of $R = F^t F$ are the column vectors of V.

Doing it with Matlab

1 Assume the data are in a matrix `M`, with each row as one map, and each column a time series for a given station.

2 To remove the mean of each column, compute:

```
F=detrend(M,0);
```

3 To find the eigenvectors and singular values, compute:

```
[C,Lambda,CC] = svd(F) .
```

The eigenvectors are in the matrix `CC` and the squared diagonal values of `Lambda` are the eigenvalues of the covariance matrix $R = F^t F$.

(To see this, compare the result with:

```
R=F'*F
```

```
[C,L] = eig(R) )
```

4 To find the PC corresponding to eigenvalue number `i`, do:

```
PCi= F * C(:,i)
```

2.4 Rotated EOFs

It has been noted (Richman, 1985) that often the map for the first EOF of a data set will be a unimodal field (and is interpreted as a unimodal oscillation), while the structure of the second EOF is bimodal (and is interpreted as a di-polar oscillation). If the data field is split into two domains (for instance a northern and a southern domain) and the EOF analysis is done separately for each domain, the same results (unimodal and dipolar structures) may be obtained for each subdomain. The dominant unimodal structure of the total domain, has then vanished as the domain was split into two. If this happens the result of the EOF analysis has shown to be sensitive to the model domain. Any attempt at a physical explanation for the EOFs is then difficult (or just plain foolish). There are several other known difficulties

that may arise in EOF analyses. Sometimes these can be 'fixed' by *rotating* the eigenvectors. The rotation of eigenvectors is a rather controversial topic, with authors tending to fall into two camps depending on whether they support rotation or not. This topic is briefly discussed below.

When rotation of eigenvectors is performed, it is common to first conduct a regular EOF analysis, retain some of the eigenvectors (and corresponding expansion coefficients), and reconstruct the data using this "truncated basis". Starting off from this truncated basis[11] a new basis is found by maximizing some "function of simplicity" (von Storch, 1995; p. 234). There are several different "functions of simplicity" in the literature, "varimax, promax, vartmax, equimax" etc. This subject is covered in Preisendorfer (1988) and also in von Storch and Navarra (1995). Richman (1985) has written a detailed review article on rotated eigenvectors. Richman maintains that unrotated EOFs often:

> exhibit four characteristics which hamper their utility to isolate individual modes of variation. These four characteristics are: *domain shape dependence*, *subdomain instability*, *sampling problems* and *inaccurate portrayal of the physical relationships embedded within the input data.*

The first of two of these relate to the fact that for the same data we can sometimes obtain completely different EOFs if parts of the domain are omitted (shape of domain changed), or if the domain is split in two and the EOF analysis is done separately on each subdomain. The sampling problem refers to what happens when eigenvalues of the unrotated EOFs are closely spaced (see also §4.3 for more discussion on this). The fourth characteristic cited by Richman is the logical conclusion when we find any of the others characteristics in our data. If we want to interpret the EOFs as physical modes of variability in our data, at least we should not be overly sensitive to the area, or sampling errors. It is easy for the EOF analysist to check if any of the three first problems arise in the data. By rotating the EOFs sometimes these problems may be overcome.

[11] It is not necessary to start from a truncated eigenvector basis; the methods also work when starting from the full eigenvector basis (or any other basis of the same size). This is, however, more expensive computationally than using the truncated basis.

As mentioned above, there are many different rotation methods. These can be grouped into orthogonal methods (that find a new orthogonal basis), and oblique methods (in which the new basis does not have to be orthogonal). It should be mentioned that an orthogonal rotation (like varimax) will find a new orthogonal basis, but the new basis will not have the PCA property, i.e., the expansion coefficients in the rotated basis will not be uncorrelated. Whether "to rotate" or "not to rotate" will depend on the data, and on the intentions of the analysist. EOF analysis is a great tool for reducing the variability in the data into a few patterns (or modes). If the modes are not to be physically interpreted, but to be used for other purposes (prediction, pattern recognition, noise reduction, etc.) rotation is probably not necessary. If the modes are to be given physical interpretation then the analysist may find that rotation is necessary. It should always be kept in mind that often there is no a priori reason to suspect that the data to be analyzed was generated by physical orthogonal modes of variability.

2.4.1 A "synthetic" example

The following example is based on one given in Richman (1985). The Matlab code used in these examples is given in the Appendix. We generate the gridded data from the three pressure patterns shown in Fig. 2.1 (and their "reflections", obtained by reflecting each pattern about the mean of 1012 mb). In understanding Fig. 2.1 think of a time sequence where the atmospheric flow patterns alternate in the following manner: anti-cyclonic $\longrightarrow$ zonal (west to east) $\longrightarrow$ meridional (south to north) $\longrightarrow$ cyclonic $\longrightarrow$ zonal (east to west) $\longrightarrow$ meridional (west to east).[12]. The form of the data is chosen such that the mean field for the data is 1012 mb, and the input patterns (anti-cyclonic, zonal and meridional) are orthogonal to each other (once the constant mean of 1012 mb has been removed).

From this time sequence a data matrix is constructed and the EOF analysis is performed. The results are shown in Fig. 2.2. The first three EOFs explain all the variance in the data, which implies that the data can be completely reconstructed using only three patterns, and three time series. This shows that the EOF method correctly finds the right number of

[12]This assumes we are in the Northern hemisphere.

independent patterns that make up the variability in the field. However, Fig. 2.1 and Fig. 2.2 do not show the same patterns. The anti-cyclonic pattern is correctly reproduced, but instead of retrieving the zonal and meridional patterns, we get eigenvectors that are a linear combination of these. Thus, even when the EOF method is applied on orthogonal input fields, it is not guaranteed that the EOFs produced will resemble the data.

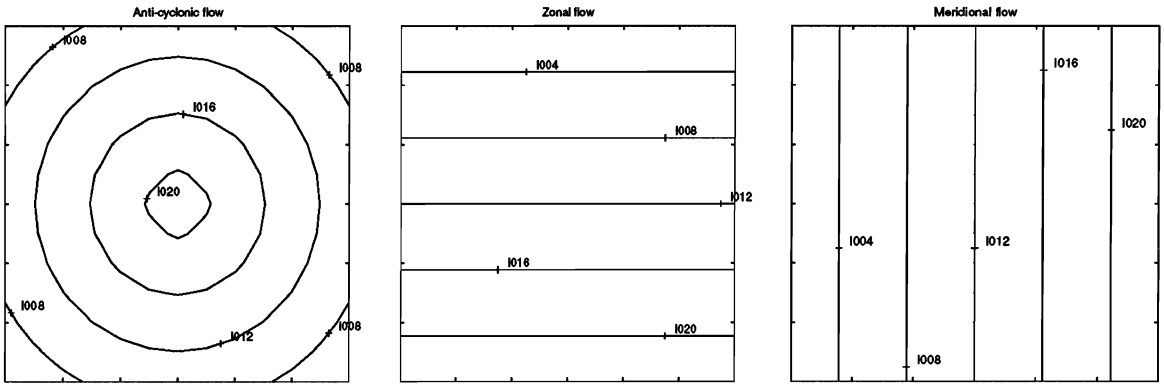

Figure 2.1: The input patterns: three sea-level pressure patterns (in mb) that are used to generate the data.

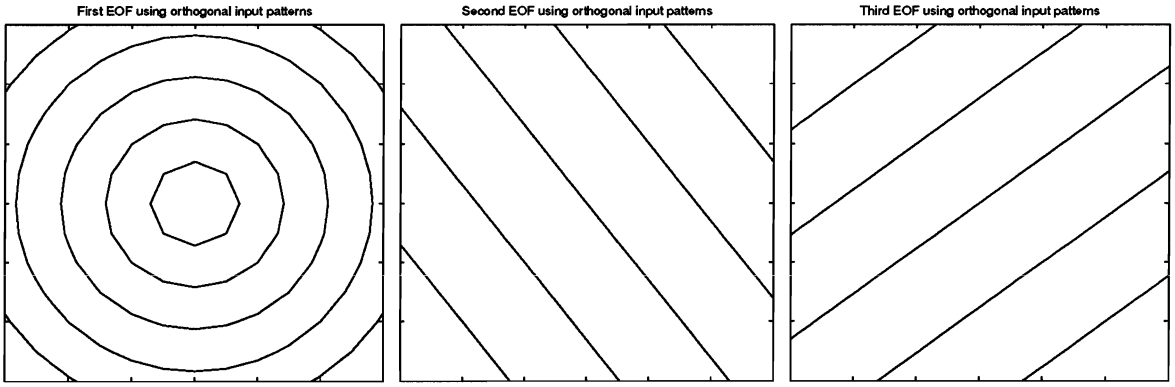

Figure 2.2: The first three EOFs (which explain **all** of the variance in the data). Contour levels are in non-dimensional units. Notice that the two diagonal EOFs are linear combinations of the zonal and meridional height fields..

We now rotate the data using the "varimax" criterion. First we form a truncated eigenvector basis from the first three eigenvectors already obtained, and then we perform the varimax rotation. The resulting rotated eigenvectors can be seen in Fig. 2.3. We see that the rotated eigenvectors retrieve the three types of input height fields. Thus, in this case, varimax rotation outperformed unrotated EOFs.

Instead of rotating the eigenvectors, it is also possible to perform rotation on the expan-

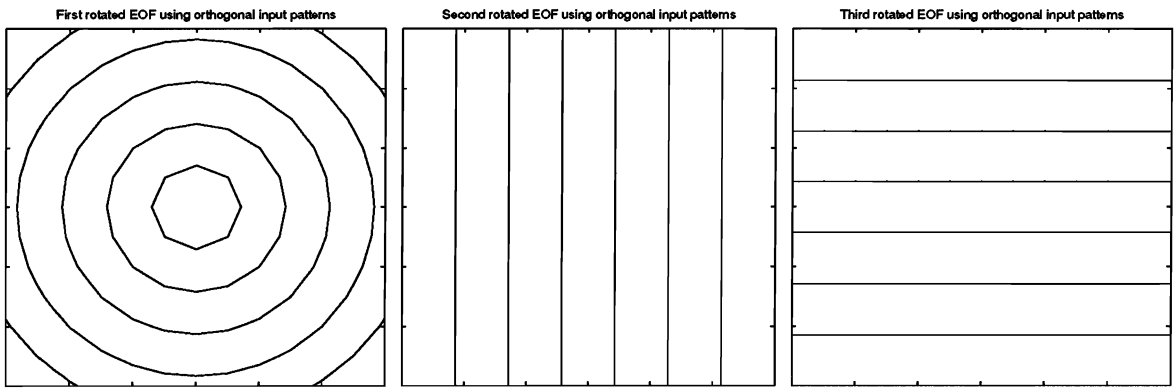

Figure 2.3: The rotated eigenvectors. These look very similar to the original input patterns. Voila!

sion coefficient time series. This will result in a new set of orthogonal expansion coefficients, but the eigenvectors associated with each of the new expansion coefficients will not be orthogonal. We begin by normalizing each of the expansion coefficient time series by the associated eigenvalue, and then we perform a varimax rotation on the matrix of normalized expansion coefficients. This rotation is referred to as "rotation in sample space" or "dual space rotation" in Preisendorfer (p. 282). In our example this method does not improve much on the unrotated egenvectors (Fig. 2.2), but in cases where the input patterns are not mutually orthogonal we have found this method to be useful. An example of its application is given in the Appendix. The reader is encouraged to try the routines given in the Appendix, splitting the data into two domains (and repeating the analyses), using different fields as input data, etc.

Chapter 3

SVD of coupled fields

3.1 The Singular Value Decomposition Method

The Singular Value Decomposition (SVD) method can be thought of as a generalization to rectangular matrices of the diagonalization of a square symmetric matrix (like in EOF analysis). It is usually applied in geophysics to two combined data fields, such as SLP and SST. The method identifies pairs of coupled spatial patterns and their temporal variation, with each pair explaining a fraction of the covariance between the two fields. Hence, to perform the SVD method, we construct the temporal cross-covariance matrix between two space and time dependent data fields. This matrix need not be square as the two fields may be defined on a different number of grid points. However, the variables need to span the same period of time[1]. As in the EOF method, the temporal means for each variable are removed from the time series at all grid points.

The SVD of the cross-covariance matrix yields two spatially orthogonal sets of singular vectors (spatial patterns analogous to the eigenvectors or EOFs, but one for each variable) and a set of singular values associated with each pair of vectors (analogous to the eigenvalues). Each pair of spatial patterns describe a fraction of the square covariance (SC) between the two variables. The first pair of patterns describes the largest fraction of the SC and each succeeding pair describes a maximum fraction of the SC that is unexplained by the previous

[1]We do assume that the variables are measured simultaneously. One could of course make one variable lag the other, and look at lagged cross-covariance structures.

pairs. The square covariance fraction (SCF) accounted for by the k-th pair of singular vectors is proportional to the square of the k-th singular value. The k-th expansion coefficient for each variable is computed by projecting the k-th singular vector onto the corresponding original data field. The correlation value ($\mathbf{r}$) between the k-th expansion coefficients of the two variables indicates how strongly related the k-th coupled patterns are.

3.2 How to do it

A climatic application of the method is given in Bretherton et al. (1992). Let us assume we have two data matrices, S and P. S is the sea surface temperature (SST) off the east coast of South America measured n times at p locations (S is n by p) and P the overlying sea level pressure (SLP) also measured n times but at q locations (P is n by q). Just as in the F data matrix described in §1.2, each column of S (or P) contains a time series for a particular location, and each row contains a map of SST (or SLP) for a given time.

We assume that both S and P are centred in time, that is, the mean of each time series in S and P has been removed. We begin by forming the covariance matrix[2]

$$C = S^t P.$$

At this point it should be noted that some investigators choose to scale the variability in S and P, by dividing each time series by its standard deviation (see for instance Wallace et al. (1992)). The result is that the cross-correlation matrix is calculated, rather than the cross-covariance matrix. The reasoning behind this kind of scaling is that if one set of variables has much higher amplitude variability than the other, the variability in that field might dominate the covariance structure between the matrices. However, not all authors prefer to use this scaling. In what follows we will stick with non-scaled time series, and hence use covariances.

Once the covariance matrix C has been formed, we perform the singular value decomposition of C. That is, we find matrices U and V and a diagonal matrix L so that

$$C = ULV^t.$$

[2]We could also choose to use $C = S^t P/(n-1)$.

The singular vectors for S are the columns of U, and the singular vectors for P are the columns of V. Each pair of singular vectors is a mode of co-variability between the fields S and P. The columns of U are sometimes called the *left patterns* and the columns of V the *right patterns.* These names originate from the way that the covariance matrix C was formed. Just as with the EOFs, these patterns represent standing oscillations in the data fields.

We can now find the expansion coefficients, i.e. time series describing how each mode of variability oscillates in time. For S we calculate

$$A = SU,$$

and for P we calculate

$$B = PV.$$

The columns of the matrices A and B contain the expansion coefficients of each mode[3], and since U and V are both orthogonal, we can reconstruct the data matrices, using $S = AU^t$ and $P = BV^t$. The diagonal of L contains the singular values. The total squared covariance in C is given by the sum of the squared diagonal values of L. This gives a simple way of assessing the relative importance of the singular modes, through the squared covariance fraction (SCF) explained by each mode. If $l_i = L(i,i)$ is the $i-th$ singular value, the fraction of squared covariance (SCF) explained by the corresponding singular vectors $\vec{u}_i$ and $\vec{v}_i$ is given by

$$SCF_i = \frac{{l_i}^2}{\Sigma {l_i}^2}$$

We calculate the SCF for each singular value and decide how many of these we want to keep. For each mode we deem worthy of attention, we can plot the mode of variability (a map of the SST variability associated with the mode, and a map of SLP variability associated with the mode) and the time series of expansion coefficients showing how the maps vary in time. Thus if we are interested in the mode corresponding to the singular value l_i, the SST map for the mode would be $\vec{u}_i$ and the SLP map for the mode would be $\vec{v}_i$. The expansion

[3]In von Storch and Navarra (p. 223) the corresponding formulae for A and B are incorrect, and therefore not in agreement with ours.

coefficients would be $\vec{a}_i$ (column vector number i in A) for the SST and $\vec{b}_i$ (column vector number i in B) for the SLP.

We have the following recipe:

- Remove the time mean from S and P and form the covariance matrix $C = S^t P$.
- Calculate the singular value decomposition of C, by solving $C = ULV^t$.
- Calculate the expansion coefficients $A = SU$ and $B = PV$.
- Use the singular values to find the modes that explain most of the covariance.

Doing it with Matlab

1 Remove the mean of `S` and `P`:

```
S = detrend(S,0); P = detrend(P,0);
```

Another way to accomplish the same is by the following:

```
[n,m] = size(S);

S = S - ones(n,1) * mean(S);
```

And similarly for `P`.

2 Form the covariance matrix by computing:

```
C=S'*P
```

OR

```
C=S'*P/(n-1);
```

3 Perform the singular value decomposition:

```
[U,L,V] = svd(C) ;
```

4 Calculate the expansion coefficients (the time series):

```
A=S*U ; B=P*V;
```

5 Calculate the squared covariance fraction SCF:

```
l = diag(L);  scf = l.^2 / sum(l.^2);
```

6 For singular mode `i` the two patterns are `U(:,i)` for the SST and `V(:,i)` for the SLP. The expansion coefficients are `A(:,i)` for SST and `B(:,i)` for the SLP.

3.3 How to understand it

Imagine a case where only one singular value l_1 is non-zero. In this case the vectors $\vec{u}_1$ and $\vec{v}_1$ (the first columns in U and V respectively) are sufficient to completely reconstruct the covariance matrix, i.e.

$$C = l_1 \vec{u}_1 * \vec{v}_1^t.$$

In general, we can reconstruct the covariance matrix using

$$C = \sum_{i=1}^{N} \vec{u}_i L(i,i) \vec{v}_i^t.$$

If N equals the total number of non-zero singular values then this reconstruction is complete. If N is a smaller number and only the biggest singular values are used then our reconstructed C represents "most" of the covariance in the data. Just as Eqn. (2.8) is the key to understanding how eigenvectors "explain" a certain fraction of the variance in the data, this equation helps us to understand how pairs of singular vectors "explain" a certain fraction of the covariance between the data fields.

Since the S data matrix is n by p and the P matrix is n by q our covariance matrix, C, will be p by q. On performing the SVD we find that U is p by p, L is p by q and V is q by q. It is therefore natural to use the matrix U as an orthonormal basis for the data in S and to use V as an orthonormal basis for the data in P (there is no other way to do it). Unlike in the EOF method, we cannot retrieve the expansion coefficients directly from the SVD of C, but we can still calculate them by projecting the data onto the basis of singular vectors U or V ($A = SU$ and $B = PV$). The analogy with the PCA property can then be found by calculating

$$A^t B = (SU)^t (PV) = U^t S^t P V = U^t U L V^t V = L.$$

It is interesting to note that for A, we have $AA^t = SU(SU)^t = SUU^tS = SS^t$ (and similarly for B). So just like for the EOF method, we find that the spatial covariance of the expansion coefficients is the same as the spatial covariance of the data.

Chapter 4

Discussion of the EOF and SVD Methods

4.1 Presenting EOF and SVD patterns

The spatial patterns corresponding to the different EOF and SVD modes can be presented in several different ways. One possibility is to plot the values of the eigenvector itself, but the amplitudes of the contours plotted this way are not easy to interpret in terms of useful quantities. However, there are other ways of presenting the spatial patterns which provide more information. These include homogeneous and heterogeneous correlation maps, and maps of local variance explained. These are described in the following sections.

4.1.1 Correlation maps

The k-th homogeneous correlation map is defined as the vector of correlation values between the expansion coefficient of the k-th mode of a field and the values of the *same* field at each grid point. It is a useful indicator of the spatial localization of the co-varying part between the field and its k-th mode. The k-th heterogeneous correlation map is defined as the vector of correlation values between the expansion coefficient of the k-th mode of a field and the grid point values of *another* field, which we want to relate to the first field. It indicates how well the grid point values of the second field can be predicted from the knowledge of the

expansion coefficient of the first.

In SVD studies, a correlation map for the k-th mode usually refers to the correlation map between the k-th expansion coefficient of one variable with the grid point values of the *same* variable (homogeneous map) or the *other* variable involved in the SVD (heterogeneous map). In all the cases, the contours plotted show the distribution of centres of action of the mode, scaled as correlation coefficients.

Vector correlation maps can also be constructed by correlating the k-th expansion coefficient of one variable with the two components of a vector field *separately* (*e.g.*,u and v components of vector wind), and then plotting the two correlation values as the components of a 'vector correlation'. The orientation of the arrows indicate the direction of the wind, and their length is proportional to the magnitude of the correlation.

4.1.2 Variance maps

If instead of plotting the correlation values as above, we plot the square of the correlations, we obtain maps of local variance explained by the k-th mode at every grid point. If we multiply the contours by 100, these maps display the spatial distribution of the percentage of variance accounted for by a mode.

4.2 Trouble shooting

4.2.1 EOFs for very non-square matrices

The following section is a rather technical discussion about what to do when number of spatial points in each map greatly exceeds the number of maps. If you are not interested in situations like this, just skip to the next section.

A good example is the January SLP from 1951 to 1985 read from the U.S. National Meteorological Center (NMC) CD-ROM disc. This would make $n = 35$ records, and each record has $p = 1977$ points. In this case we would read this into a matrix of size 35 by 1977. Our covariance matrix R would then be 1977 by 1977 in size, and the computer available might have difficulty with solving the eigenvalue equation.

Solving the Eigenproblem

In finding the EOFs we calculate the eigenvectors and eigenvalues of the covariance matrix of F. The covariance matrix is found by calculating $R = F^tF$. The size of R is p by p, and solving the eigenproblem can be a formidable task if p is a large number. Many eigensolver routines run into difficulty when the matrix becomes larger than 1000 by 1000 (this depends on the program, and the machine you are running it on). There is, however a clever way around this problem when the number of observation times (n) is much less than the number of gridpoints (p). Since the rank of F is at most n, the rank of R cannot exceed n. So the number of zero eigenvalues of R is at least $p-n$. We can use this fact to get at the eigenvalues without having to work with a p by p matrix.

Solving a Smaller Problem

As before, let the eigenvalues λ_i of R be arranged in a diagonal matrix Λ and the respective eigenvectors $\vec{v}_i$ of R be the column vectors in a matrix C. Then we have

$$RC = C\Lambda. \tag{4.1}$$

Now let $L = FF^t$. The size of L is n by n, which in this case is much less than the size of R. Multiplying the right side of (4.1) by F we get $FRC = FF^tF = LFC$. The left hand side becomes $FC\Lambda$.

Now we define $B = FC$ and write Eqn. (4.1) as:

$$LB = B\Lambda. \tag{4.2}$$

Notice that Eqn. (4.2) is exactly the eigenproblem for L. Thus Λ contains also the eigenvalues of L. The eigenvectors $\vec{b}_i$ of B are NOT the same as the eigenvectors of R. The important thing here is that since the size of L is n by n, the Λ found by solving Eqn. (4.2) will also be n by n, much smaller than the matrix Λ in Eqn. (4.1) which is p by p. The good news is that a computer that cannot solve Eqn. (4.1) may be able to solve Eqn. (4.2).

We solve the eigenvalues for L (and R). Now it remains to find the eigenvectors of R. Let us assume that we find that the total amount of variance explained by the k largest eigenvalues is large enough so that we can just discard the rest of the eigenvalues. Just as we

could find the PCs by taking the projection of F onto the eigenvectors from C, we can find the EOF by taking the projection of F^t on the vectors from B. It turns out that the vectors thus obtained are proportional to the EOFs of Eqn. (2.1). The proportionality factor is $1/\sqrt{\lambda_i}$ [1]. If we are looking for the eigenvector associated with eigenvalue λ_i, we calculate $D = F^tB$. We then find column vector d_i of matrix D, and then our EOF number i is $d_i/\sqrt{\lambda_i}$. This formalism is referred as the "dual" formalism, or "sample space" formalism in Preisendorfer (1988).

In this case, we have the following deviation from the scheme given in §2.1.

- Calculate $L = FF^t$.
- Solve $LB = B\Lambda$.
- Calculate $D = F^tB$.
- Choose the eigenvalues to retain and find the corresponding eigenvectors by dividing d_i with $\sqrt{\lambda_i}$.

Doing it with Matlab

1 Put the data in a matrix `M`, with each row as one map, and each column as a time series for a given station.

2 Remove the mean of each column:

```
F=detrend(M,0);
```

3 Form the small covariance matrix:

```
L=F* F'
```

4 Calculate the eigenvalues and eigenvectors of the small covariance matrix:

```
[B,Lambda] = eig(L)
```

(The eigenvalues in `Lambda` are also eigenvalues of `R` = F^tF.)

[1] If we are going to normalize the eigenvector this constant will not matter. The proof that the constant is $1/\sqrt{\lambda_i}$ is easy. Simply use singular value decomposition on F (see §2.3.1).

5 Find the matrix whose column vectors are proportional to the eigenvalues of the large covariance matrix `R = F`^{t}`F`:

```
D=F'*B
```

6 Divide through each vector by the square root of the corresponding eigenvalue:

```
sq=sqrt(diag(Lambda))+eps]';sq=sq(ones(1,p),:)
CC=D./sq
```

The `eps` is just added to make sure that when dividing by small numbers, the outcome is not a NAN ("Not-A-Number"). The vector `sq` is expanded to the same size as the matrix `D` so that it can be divided (element by element) by D.

7 Find the PC corresponding to eigenvalue number i, i.e., do:

```
PCi= F * CC(:,i)
```

Comparing this method with the one given previously reveals that though Matlab gives the same eigenvalues, it may have altered the sign of some eigenvectors. Such a change will also change the sign of the PC for that eigenvector, so there is no problem really.

4.2.2 Missing data

Both EOF and SVD analysis methods assume the data to be complete. Nevertheless, if there are gaps in the data, these can be taken into account in the construction of the covariance matrix. When there are gaps, each element computed in the covariance matrix can be divided by the actual number of data available for its calculation. This contrasts with all elements being divided by the same number, as would be the case if the data were complete. The resulting covariance matrix is symmetric, but not positive definite, and we thus obtain several (small) negative eigenvalues. A consequence of this loss of definiteness is a slight correlation between the principal components. Hence, the resulting modes are no longer strictly orthogonal. A way of overcoming this problem is to fill the gaps in the data using

an adequate interpolation method. If the gaps are not too large, however, the differences in the results using the original and the interpolated data sets are small.

4.3 Significance tests

4.3.1 Selection Rules

A popular method for deciding which eigenvectors to keep and which to discard is to use *selection rules.* Preisendorfer devotes a whole chapter to this subject and derives several different rules. Some of these rules are based on Monte Carlo experiments, and others are based on a theoretical distribution of eigenvalues from random covariance matrices. There are three classes of selection rules depending on whether they focus on the eigenvalues, the expansion coefficients or the eigenvectors[2]. Firstly, the *dominant variance* rules are based on the amount of variance explained (or the size of λ_i). Of this class, *rule N* is popular, as it claims to find the physically significant eigenvalues. (Most of the discussion of EOFs in the preceding sections was implicitly based on dominant variance thinking). Secondly, the *time-history* rules examine the expansion coefficients for non-noisy temporal behavior. The data is then reconstructed and analyzed using only eigenvectors and eigenvalues corresponding to those expansion coefficients that passed this test. Thirdly, the *space-map* rules select eigenvectors based on some pre-specified form of the eigenvector map (for example, in this case we may know that the eigenvectors should resemble known normal modes of some dynamical system).

4.3.2 North's "rule of thumb"

Using scaling arguments, North et al.(1982) defined the "typical errors" between two neighboring eigenvalues λ and between two neighboring eigenvectors ψ as follows:

$$\Delta\lambda_k \approx \sqrt{\frac{2}{n}}\lambda_k$$

$$\Delta\psi_k \approx \frac{\Delta\lambda_k}{\lambda_j - \lambda_k}\psi_j$$

[2]See Preisendorfer for details.

where λ_j is the eigenvalue closest to λ_k and n is the number of independent samples. The estimation of the EOF ψ_k (associated with eigenvalue λ_k) is mainly contaminated by the pattern of the closest neighboring EOF ψ_j (associated with eigenvalue λ_j). The closer the eigenvalues, the smaller the difference $\lambda_j - \lambda_k$ and the more severe the contamination. North's rule of thumb says that if the typical error $\Delta\lambda$ of a particular eigenvalue is comparable or larger than the difference between the eigenvalue and its closest neighbor, then the typical error $\Delta\psi$ of the eigenvectors will be comparable to the size of the neighboring eigenvector.

The condition of *independent* samples, however, is practically never valid in geophysical fields. Hence, care should be taken on the choice of the number of degrees of freedom n, which is generally less than the number of data points. The rule of thumb is very useful as a simple but powerful tool to use for deciding which eigenvectors to keep.

4.3.3 Monte Carlo approach

To assess the statistical robustness of the results obtained from EOF and SVD analyses, significance tests can be performed using a Monte Carlo approach. Examples of this can be found in Peng and Fyfe(1996) and Venegas et al.(1996a). They performed a dominant variance-type test and focussed on the total square covariance (TSC, the sum of the squares of all the singular values) and on the square covariance corresponding to each mode (SC, the square of each singular value), rather than the square covariance fractions (SCF=SC/TSC) or the **r** coupling coefficients. The SC is a direct measure of the coupling between the two fields, while the SCF and **r** are only meaningful when they are associated with a significant SC (Wallace *et al.*, 1992). After performing a series of SVD calculations using surrogate data, they established significance levels for the singular values $lambda_i$. A comparison of the results for their original data with these significance levels told them which singular vectors to retain. A similar method is outlined below.

Consider the case of an SVD analysis on monthly data, between "field one" and "field two", using 40 years of data. We create the surrogate data, a randomized data set of field one by scrambling the monthly maps of the 40 years in the time domain, in order to break the chronological order of field two relative to field one. The field to be scrambled is chosen as the one having the smaller month-to-month autocorrelation ('memory') of the two fields, in

order to minimize the increase of degrees of freedom inherent in the scrambling procedure. The scrambling should be done on the years and not on the months, *i.e.*,the 12 maps of field two of a given year will be paired with 12 maps of field one of another (randomly chosen) year, but the order of the months inside the year is maintained. In such a way, we link January with January, February with February, etc, and do not deteriorate the intra-seasonal variability inherent in the coupling, which would lower the significance levels and make the results appear more significant than they really are.

We then perform an SVD analysis on the scrambled data sets. The same procedure of scrambling the data set and performing the analysis is repeated 100 times, each time keeping the values of the TSC and SC of each mode. A SC value from the original data set will be statistically significant at the 95% level if it is not exceeded by more than five values of the corresponding SC obtained using the scrambled data sets.

In the case of an EOF analysis, the quantities tested are the eigenvalues, which are proportional to the fraction of variance explained by each mode. The procedure described above can be reproduced, scrambling the order of the data that is being analyzed.

Finally, it should be mentioned that even more sophisticated methods can be used to establish the significance of the EOFs. An example of this can be found in Thacker (1996), where in the formulation for the EOF method, uncertainties in the data are explicitly accounted for. This allows more reliable parts of the data field to be given more weight than the less reliable parts.

4.3.4 Significance levels for the correlations

The significance level for a correlation varies according to the integral time scale as determined by the autocorrelation function, even when the nominal number of degrees of freedom (given by the length of the series) does not change. Therefore, variables having shorter time scales (or 'small memory', such as wind speed) have more real degrees of freedom and thus their significance levels are lower than those corresponding to variables with 'long memory' (such as SST). Consequently, correlations for variables averaged over large areas (such as regional indices) or over long time periods (such as seasonal means) have longer time scales, fewer effective degrees of freedom and higher significance levels than those for variables over

individual grid points and times.

The following method (suggested by Sciremammano, 1979) calculates the significance level for any particular correlation, accounting for the autocorrelation ('memory') inherent in the two series involved. The Large-Lag Standard Error σ (Davis, 1976) between two time series $X(t)$ and $Y(t)$ is computed as follows:

$$\sigma^2 = N^{-1} \sum_{i=-M}^{M} Cxx(it)Cyy(it)$$

, where Cxx and Cyy are the autocorrelation functions of $X(t)$ and $Y(t)$ respectively, N is the length of both series, and M is large compared with the lag number at which both Cxx and Cyy are statistically zero. Thus, to a good approximation, for time series involving at least 10 degrees of freedom, the 90%, 95% and 99% significance levels are equivalent to:

$$C_{90} = 1.7 * \sigma$$

$$C_{95} = 2.0 * \sigma$$

$$C_{99} = 2.6 * \sigma$$

Significance levels for correlations can also be found using Monte Carlo methods similar to the ones described above. In general these methods, which also go by the name of "bootstrapping methods", establish the significance for a particular quantity (correlation coefficient, magnitude of eigenvalue, TSC, etc.) by first generating surrogate data, then performing the analysis on the surrogate data, repeating this enough times to get a distribution of the desired quantity and finally using that distribution to assess the significance levels for the calculated quantity.

As an example of this, we might like to assess the significance of the correlation between two time series, Ts_1 and Ts_2. We would generate 100 different surrogate time series, and correlate Ts_1 with each of them. This would give us 100 different correlation coefficients, and from their distribution we could assess the desired significance level (5th largest value would be the 95% significance level, etc).

How the surrogate time series are formed is of course crucial for this method to make any sense. We want the surrogate time series to have the same statistical properties as

the original data time series (Ts_2 in this example). Often it is considered sufficient for the surrogate data and the original data to have the same auto-correlation ("memory") and the same distribution. To achieve this, there are several different ways to proceed.

One is to randomly scramble the original time series. The scrambled time series have obviously the same distribution as the original time series, but the scrambling may have destroyed the memory in the data (unless special care is taken in the scrambling procedure, as in the example in §4.3.3). If this happens, then some authors filter the surrogate data to "re-install" the memory. This filtering may then change the distribution of the data, and that has then to be fixed.

In their wide ranging book on nonlinear dynamics, Kaplan and Glass [3] (1995) review the time series analysis methods used for the data analyses of systems with nonlinear dynamics. The most general way of generating surrogate data they present is based on a method called "phase randomization". This method uses the fact that there is a unique correspondence (through the Fourier transform) between the autocorrelation of a time series and its power spectrum. The method generates the surrogate time series by ensuring that they have the same spectrum as the original data, and thus the same autocorrelation. The surrogate data generated using this method will have a Gaussian distribution, and if the original data is not Gaussian distributed we must "amplitude adjust" the surrogate data to make sure it has the right distribution. The interested reader is referred to Kaplan and Glass (1995) for further details.

[3]This book is a highly readable beginners'guide to nonlinear dynamics.

Chapter 5

Examples

5.1 EOF analysis

As an example of application of the EOF method, we show here the analysis of 40 years of sea surface temperature (SST) in the South Atlantic. Further details can be found in Venegas et al.(1996a). The data analyzed are monthly SST anomalies from COADS (Comprehensive Ocean-Atmosphere Data Set), covering the ocean from the Equator to 50°S over a 2° by 2° grid, and spanning the period 1953-1992.

The three leading EOF modes together account for 47% of the total monthly SST variance. Individually, they explain 30%, 11% and 6% of the variance. The spatial patterns associated with these three SST modes are shown in Figure 5.1 as homogeneous correlation maps E1(SST), E2(SST) and E3(SST). Figure 5.2 shows the temporal variations of each eigenvector, represented by the expansion coefficients e1(SST), e2(SST) and e3(SST), respectively.

E1(SST) exhibits a mono-pole pattern extending over the entire domain. Since the square of the correlations represents the variance explained locally, this mode accounts for up to 64% of the variance in the region of largest amplitude, namely, off the coast of southern Africa. The fraction of local variance explained decreases towards the south and west. The expansion coefficients e1(SST) associated with this pattern is dominated by a mixture of interannual and decadal fluctuations. E2(SST) displays an out-of-phase relationship between temperature anomalies north and south of about 25°S-30°S. The mode explains up to 20%-

40% of the variance near its centres of action. The associated expansion coefficient time series e2(SST) consists of a combination of interannual and lower frequency oscillations. E3(SST) exhibits three latitudinal bands of centres of action with alternating signs. A 20°-wide band centred around 25°S is surrounded by two bands of the opposite polarity north of 15°S and south of 35°S. Interestingly, the middle band displays two distinct centres of action off the coasts of South America and South Africa. The temperature fluctuations near the latter centre of action account for up to 20% of the total field variance. This mode oscillates on mainly interannual timescales, as can be seen from the time series e3(SST).

We performed a Varimax rotation on the SST field (reconstructed using the three first EOFs), and the spatial patterns and time series obtained for the rotated EOFs are essentially the same as for the unrotated case. This indicates that the EOF decomposition is robust.

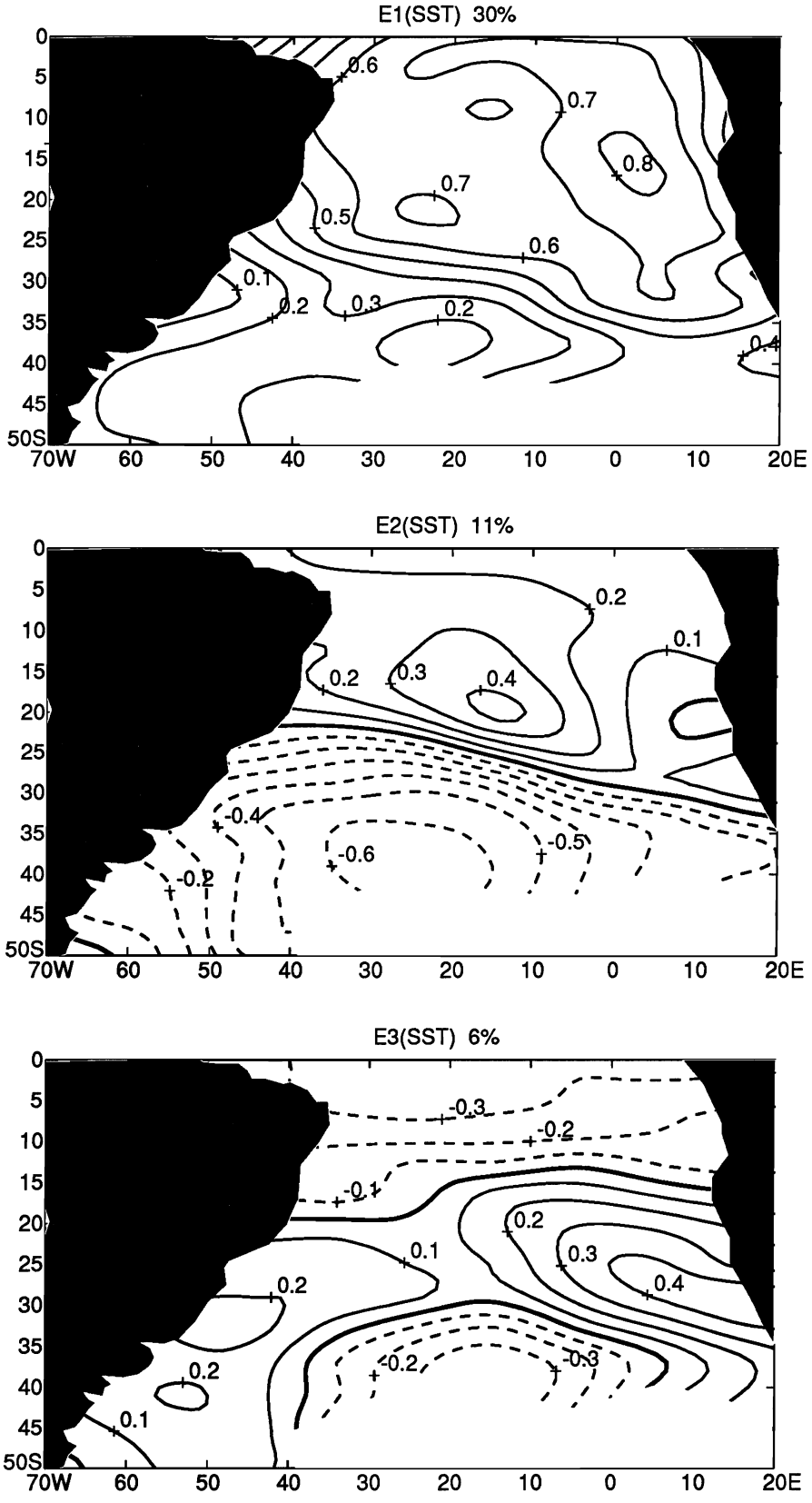

Figure 5.1: Spatial patterns of the first three EOF modes of SST, presented as homogeneous correlation maps. Contour interval is 0.1. Negative contours are dashed. Zero line is thicker. From Venegas et al. (1996a).

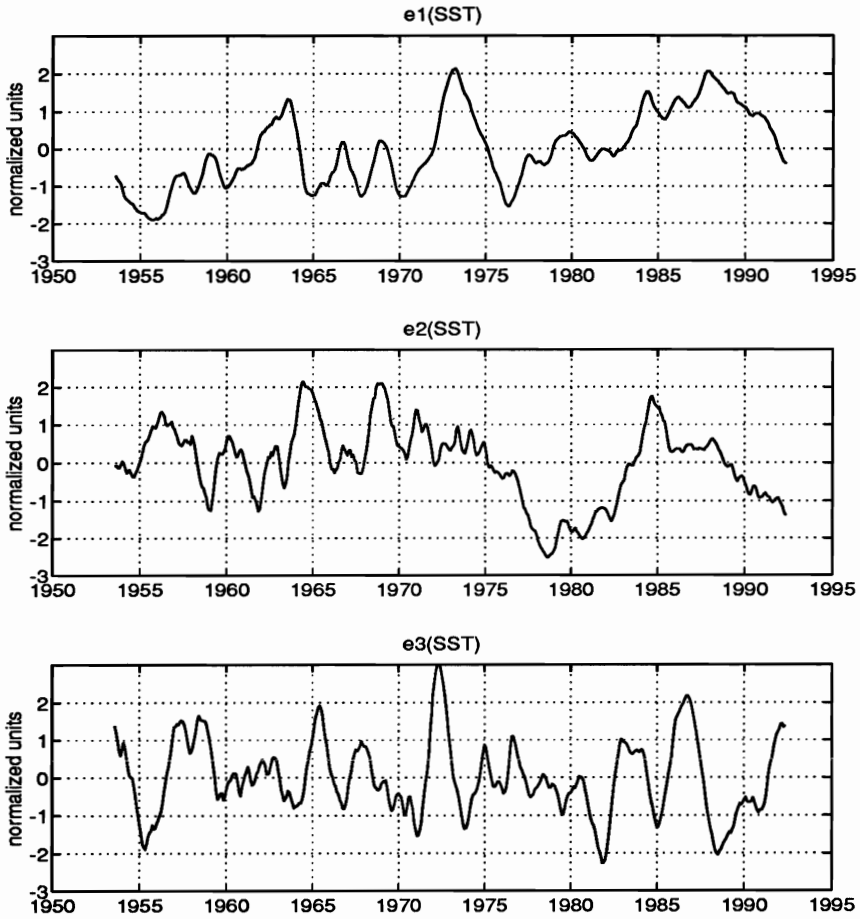

Figure 5.2: Time series of expansion coefficient of the first three modes of SST. The time series are smoothed using a 13-month running mean, and their amplitudes are normalized by the standard deviation. From Venegas et al. (1996a).

5.2 SVD analysis

An example of the application of the SVD analysis method, which appeared in Venegas et al. (1996b), is now given. Monthly anomalies of SST and sea level pressure (SLP) are analyzed together over the same spatial and temporal domain as described in section 5.1. In contrast to the individual EOF analysis performed on the SST (section 5.1), the SVD analysis on the two combined fields will identify *only* those modes of behavior in which the SST and SLP variations are strongly coupled.

The three leading SVD modes of the coupled SST and SLP variations account for 89% of the total square covariance. We label the spatial patterns as Sk, and the expansion coefficients as sk, k=1,3. Table 5.1 displays the square covariance fractions (SCF) explained by each mode and the correlation coefficient (**r**) between the expansion coefficients of both variables (sk(SST) and sk(SLP)), as indicators of the strength of the coupling. The first mode exhibits a strong and highly significant coupling between SST and SLP. By contrast, the coupling coefficient associated with the second mode is smaller than those of the other two, which suggests a relatively weaker coupling. Even though the third SVD mode explains the least variance, its corresponding coupling coefficient is higher than that of the second mode, which highlights the relative importance of the third mode.

Mode	SCF	**r**
1	63%	0.46 (0.29)
2	20%	0.30 (0.20)
3	6%	0.41 (0.21)

Table 5.1: Square covariance fraction (SCF) and coupling correlation coefficient between the expansion coefficients of both variables, corresponding to the three leading SVD modes. In parentheses are the 95% significance levels for the correlations.

The coupled spatial patterns and expansion coefficients corresponding to each variable and each mode are displayed in Figures 5.3, 5.4 and 5.5. Note that the SST patterns associated with the first and second SVD modes repeat the SST patterns of the first and second EOF modes (section 5.1), but they are interchanged. The first coupled mode of

variability between SST and SLP in the South Atlantic (Figure 5.3) can be described as a strengthening and weakening of the subtropical anticyclone, which is accompanied by fluctuations in a north-south dipole structure in the ocean temperature. Both time series are dominated by interdecadal oscillations. The second SVD mode is characterized by east-west displacements of the subtropical anticyclone centre, accompanied by large SST fluctuations over a broad region off the coast of Africa. The time series shows low frequency interannual oscillations. The third SVD mode is characterized by north-south displacements of the subtropical anticyclone, accompanied by SST fluctuations over a latitudinal band in the central South Atlantic ocean. Relatively high frequency interannual timescales dominate the SST and SLP oscillations.

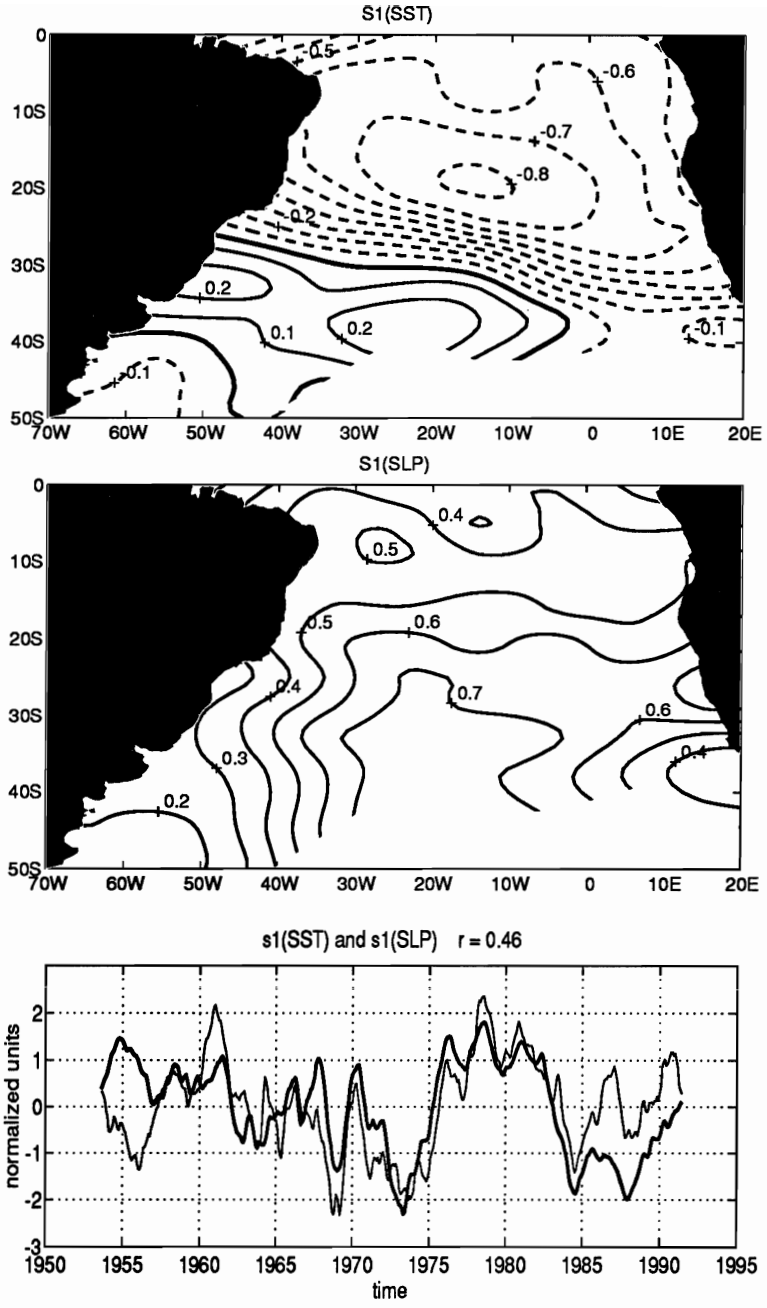

Figure 5.3: Spatial patterns (S1) and time series of expansion coefficients (s1) of the first SVD mode. Spatial patterns are presented as homogeneous correlation maps. Negative contours are dashed. Zero line is thicker. Contour interval is 0.1. Time series are smoothed by a 13-month running mean and amplitudes are normalized by the relevant standard deviation. Thick line is s1(SST), thin line is s1(SLP). From Venegas et al. (1996b).

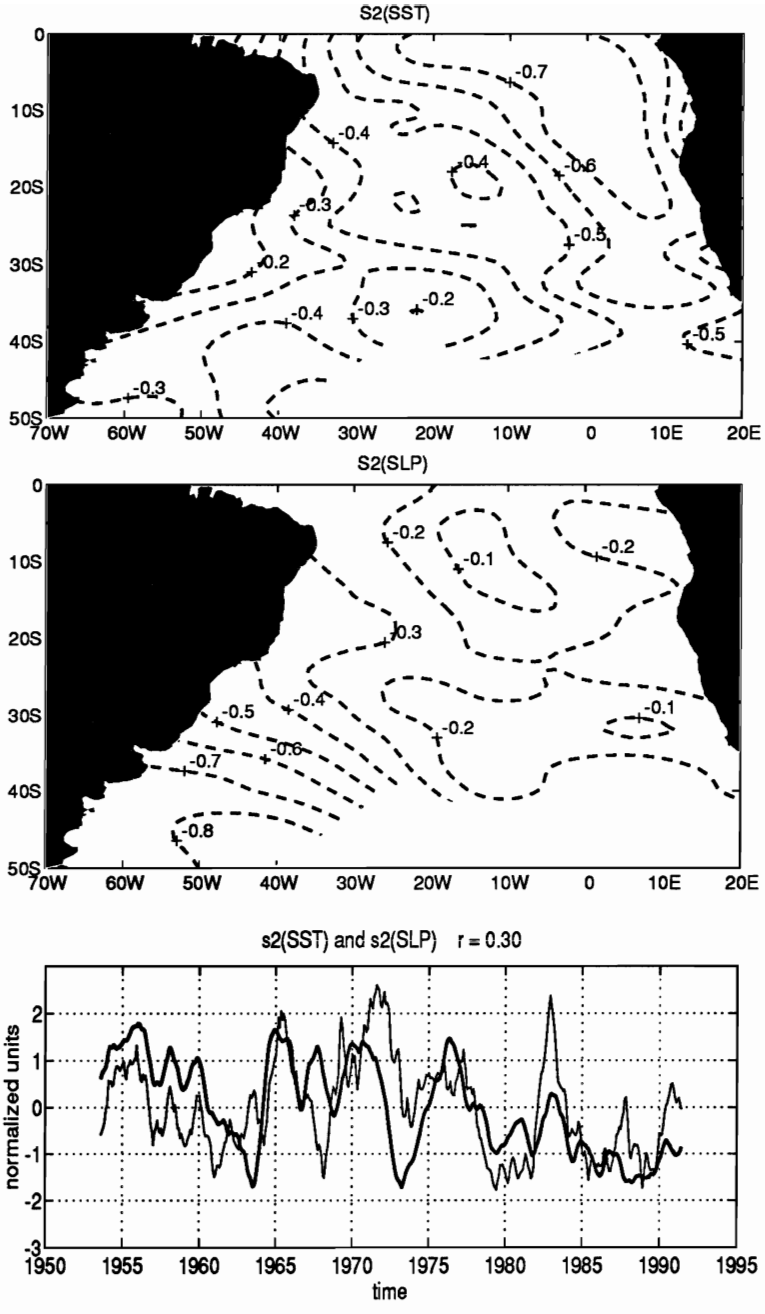

Figure 5.4: Spatial patterns (S2) and time series of expansion coefficient (s2) of the second SVD mode. Conventions as in Figure 5.3. From Venegas et al. (1996b).

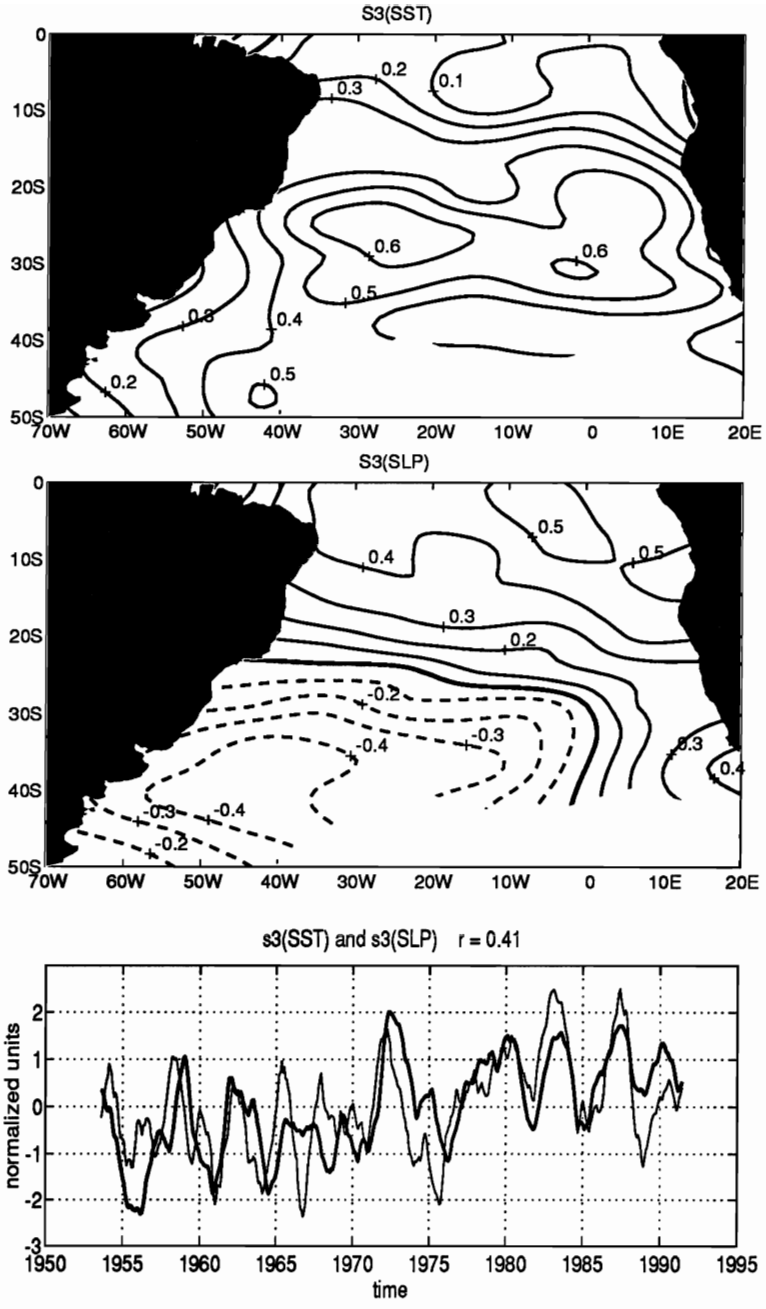

Figure 5.5: Spatial patterns (S3) and time series of expansion coefficient (s3) of the third SVD mode. Conventions as in Figure 5.3. From Venegas et al. (1996b).

Chapter 6

Acknowledgments

Discussions with Professors L. A. Mysak and D. N. Straub led to significant contributions to this document. We thank Professor Mysak for carefully reading two draft versions of this report. Helpful correspondence with Dr. D.B. Enfield and discussions with Dr. G. Schmidt and Dr. B. Tremblay are also gratefully acknowledged. During the writing of this manual, H.B. and S.A.V. were supported in part by grants from NSERC and FCAR awarded to Professors L. A. Mysak and D. N. Straub, and also by the FCAR grant awarded to the Centre for Climate and Global Change Research (C^2GCR).

Bibliography

BRETHERTON, C. S., SMITH, C., AND WALLACE, J. M. (1992). An intercomparison of methods for finding coupled patterns in climate data. *J. Clim.*, 5:541–560.

DAVIS, R. E. (1976). Predictability of sea surface temperature and sea level pressure anomalies over the North Pacific Ocean. *J. Phys. Oceanogr.*, 6:249–266.

KAPLAN, D. AND GLASS, L. (1995). *Understanding Nonlinear Dynamics.* Springer.

NEWMAN, M. AND SARDESHMUKH, P. D. (1995). A caveat concerning singular value decomposition. *J. Clim.*, 8:352–360.

NORTH, G. R., BELL, T. L., CAHALAN, R. F., AND MOENG, F. J. (1982). Sampling errors in the estimation of empirical orthogonal functions. *Mon. Wea. Rev.*, 110:699–706.

PEIXOTO, J. P. AND OORT, A. H. (1992). *Physics of Climate.* American Institute of Physics, New York.

PENG, S. AND FYFE, J. (1996). The coupled patterns between sea level pressure and sea surface temperature in the mid-latitude North Atlantic. *J. Clim.*, 9:1824–1839.

PREISENDORFER, R. W. (1988). *Principal Component Analyses in Meteorology and Oceanography.* Elsevier.

RICHMAN, M. B. (1985). Rotation of principal components. *Journal of Climatology*, 6:293–335.

SCIREMAMMANO, F. (1979). A suggestion for the presentation of correlations and their significance levels. *J. Phys. Oceanogr.*, 9:1273–1276.

THACKER, W. C. (1996). Metric-based principal components. *Tellus*, 46A:584–592.

THIEBAUX, H. J. (1994). *Statistical data analyses for ocean and atmospheric sciences.* Academic Press.

VENEGAS, S. A., MYSAK, L. A., AND STRAUB, D. N. (1996a). Atmosphere-ocean coupled variability in the South Atlantic. C2CGR Report 96-13, Centre for Climate and Global Change Research, McGill University, Montreal, Canada.

VENEGAS, S. A., MYSAK, L. A., AND STRAUB, D. N. (1996b). Evidence for interannual and interdecadal climate variability in the South Atlantic. *Geophys. Res. Lett.*, 23(19):2673–2676.

VON STORCH, H. AND NAVARRA, A., editors (1995). *Analyses of Climate Variability - Applications of Statistical Techinques.* Springer.

WALLACE, J. M., SMITH, C., AND BRETHERTON, C. S. (1992). Singular value decomposition of wintertime sea surface temperature and 500-mb height anomalies. *J. Clim.*, 5:561–576.

Appendix A

Matlab code for the examples

The Matlab code used in the example in §2.4.1.

1 Generate the data:

```
x=0:0.5:6;y=x;[X,Y]=meshgrid(x,y);

H1=12*(1.2-0.35*sqrt((X-3).*(X-3)+(Y-3).*(Y-3)));
H1=1012+H1- mean(mean(H1));
H2=1022.8-3.6*Y;
H3=1001.2+3.6*X;

H1b=1012+(1012-H1);
H2b=1012+(1012-H2);
H3b=1012+(1012-H3);
```

Each field can be examined, e.g. by using:
`val=[992:4:1032 ];`
`c=contour(X,Y,H3,val);clabel(c);` (See Fig. 2.1)

2 Move the data into a matrix:

```
F=zeros(6,169);
F(1,:)=reshape(H1,1,169);
F(2,:)=reshape(H1b,1,169);
F(3,:)=reshape(H2,1,169);
F(4,:)=reshape(H2b,1,169);
F(5,:)=reshape(H3,1,169);
F(6,:)=reshape(H3b,1,169);
mean(mean(F))
Z=detrend(F,0);
```

3 Find the eigenvectors:

```
S=Z'*Z;
[E,d] = eof(S);
dsum=diag(d)/trace(d)
```

4 The fractional amount of variance explained by each eigenenvector is listed in `dsum`. This list shows that only three eigenvectors are significant and they explain all the variance in the data. We look closer at these:

```
eof1=E(:,169); eof2=E(:,168);eof3=E(:,167);
pc1=Z*eof1; pc2=Z*eof2; pc3=Z*eof3;
E1=zeros(size(X));E2=zeros(size(X));E3=zeros(size(X));
E1(:)=eof1(:); E2(:)=eof2; E3(:)=eof3(:);
```

Contour graphs of the `E1`, `E2` and `E3` can be seen in Fig. 2.2

5 Make a truncated eigenvector basis and reconstruct the data on that basis:

```
E=[eof1 eof2 eof3];
A=[pc1 pc2 pc3];
Z=A*E';
```

(This step is included here for completeness. Here it is strictly speaking not neccessary since the three EOF's completely explain all the variance in the data).

Next we call the `varimax` routine (see below) that finds a new orthogonal basis `U`. The columnvectors of `U` are the rotated patterns:

```
varimax;
U1=zeros(size(X)); U2=zeros(size(X)); U3=zeros(size(X));
U1(:)=U(:,1); U2(:)=U(:,2); U3(:)=U(:,3);
```

Contour graphs of the `U1`, `U2` and `U3` can be seen in Fig. 2.3

6 State space (or dual space) rotation:
First we generate an orthogonal matrix of normalized expansion coefficients:

```
Phi=[A(:,1)/sqrt(d(169,169)) A(:,2)/sqrt(d(168,168)) A(:,3)/sqrt(d(167,167))];
```

Next we perform the rotation (see below for the code for the routine `dualvarimax`):

```
U=Phi; Z=A*E'; dualvarimax
```

The routine `dualvarimax` stores the spatial patterns associated with each rotated principal component in the matrix `B`.

```
B1=zeros(size(X)); B2=zeros(size(X)); B3=zeros(size(X));
B1(:)=B(:,1); B2(:)=B(:,2); B3(:)=B(:,3);
```

Contour graphs of the `B1`, `B2` and `B3` can then be examined.

The routines **varimax** and **dualvarimax** are based on an algorithm given in Preisendorfer (p. 273 - 283).

```
% M-script VARIMAX. Performs varimax rotation in state space.
% Assumes that started is from a basis E (e.g. truncated eigenvector basis)
% and that Z is the data reconstructed with that basis, e.g. Z=A*E'
% (A is the matrix of retained expansion coefficients).
U=E;
B=Z*U;
Unew=zeros(size(U));
tol=1e-10 % the tolerance parameter; set this to desired
limit=1 % convergence level
while limit >tol
D=diag(mean(B.^2));
C=B.^3;
V=Z'*(C-B*D);
        [w,k]=eig(V'*V);
% The following is according to 7.26 in Preisendorfer (and now the w are not wrong)
% This formulation of k^(-1/2) is numerically sensitive, and can be rewritten:
%   kin=diag(diag(k.^(-1/2)));
%   kin(~finite(kin))=zeros(size( kin(~finite(kin))));
%   Ginv=w*kin*w';
Ginv=w*k^(-1/2)*w';
Unew=V*Ginv;
limit=max(max(Unew-U))
Bold=B;
U=Unew;
B=Z*U;
end
```

```
% M-script DUALVARIMAX. Performs varimax rotation in sample space.
% Assumes that started is from a matrix Phi of orthogonal principal
% component time series. And the data is is in the matrix Z
% After convergence the new rotated PCs are stored in U, but
% the new patterns are stored (which are not orthogonal) are stored in B.
U=Phi;
B=Z'*U;
Unew=zeros(size(U));
tol=1e-7
limit=1
while limit >tol
D=diag(mean(B.^2));
C=B.^3;
V=Z*(C-B*D);
        [w,k]=eig(V'*V);
Ginv=w*k^(-1/2)*w';
Unew=V*Ginv;
limit=max(max(Unew-U))
Bold=B;
U=Unew;
B=Z'*U;
end
```