DATA REDUCTION BY SEPARATION OF SIGNAL AND NOISE COMPONENTS FOR MULTIVARIATE SPATIAL IMAGES

by

GEIR STORVIK
STANFORD UNIVERSITY

TECHNICAL REPORT NO. 125
JULY 1989

STUDY ON STATISTICS AND ENVIRONMENTAL FACTORS IN HEALTH (SIMS)

PREPARED UNDER A GRANT TO SIMS FROM
ENVIRONMENTAL PROTECTION AGENCY (EPA)
SLOAN FOUNDATION
NATIONAL SCIENCE FOUNDATION (NSF)

DEPARTMENT OF STATISTICS
DEPARTMENT OF APPLIED EARTH SCIENCES
STANFORD UNIVERSITY
STANFORD, CALIFORNIA
Abstract

The methods developed for reconstruction of signal from one- or two-dimensional images blurred by noise are often computationally expensive even for univariate data at each pixel location. For multivariate data, the computation can be to enormous to be carried out with all the data in hand and some preliminary data-reduction must be done.

Principal components (PC) is a widely used method for this purpose. However, PC has the drawback of not being invariant to rescaling or other linear transformations of the data. Without knowledge of the structure of the covariance matrix for the noise, it is therefore impossible to know how PC will perform.

Switzer and Green(1984) introduced the Min/Max Autocorrelation Factors (MAF) process for transforming the data to a new set of vectors where the components are ordered according to their amount of autocorrelation. Since signal is usually associated to high autocorrelation, the components with low autocorrelation would be expected to contain mainly noise and could be discarded or down-weighted. Assuming each pixel belongs to one of a finite number of classes, it is shown that MAF is optimal, in the sense of maximizing the average of the mahalanobis distances between the classes, when the probability of two neighbor locations belonging to the same class is independent of the underlying class and that, given they are different, the class at one location does not give any information about the class in the other location. However, when the classes occur at different spatial scales or some classes tend to appear together, MAF does not perform well.

This paper presents a new method, the Restricted Min/Max Autocorrelation Factors (RMAF) process, for dimension reduction which is a modification of MAF. A simulation study is presented for comparing MAF and RMAF, which shows the superiority of RMAF when the features in the image are small. The calculation of RMAF will also involve an interesting nonstandard use of estimation techniques from robust statistics.
1. Introduction

Problems involving observations from locations on a spatial grid frequently occur in the natural sciences. Some examples are medical image analysis and remote sensing, where channel data are available on a two-dimensional array of discrete picture elements, and from well log analysis where logs are available at equal distance levels down the well. The observations will generally consist of a signal blurred by noise, and the aim of an analysis of the data may be to recover the signal or some functional of the signal. In this paper we will consider situations where the underlying image consists of pixels belonging to one of a finite number of classes and where the final goal is to recover the underlying image.

Often the observation at each location is a high dimensional vector. Extensive correlation between the components in the observation vector is usually the case for such data, which suggests the possibility of reducing the dimensions of the multivariate data by separating out some of the noise. Reducing the dimension is desirable for many reasons. The most important is that one often wish to do some time-consuming computation, for example probabilistic clustering, which may be too expensive with the amount of data available. In remote sensing one often wish to reduce the number of channels to three, so that one could combine them to form a colored picture.

Separating noise from signal involves defining an operator that transforms the vectors into other vectors. In this paper we will only consider pointwise, in particular linear, operators on the data. Such pointwise operators have the important property that they do not further blur the signal. This is in contrast to many other methods for separating out noise, such as spatial smoothing, which do not have this property. Since a linear operation on a vector gives a new vector with components being projections in different directions, we will often use the term direction for the components of the transformed data, meaning the direction of the line onto which the data are projected.

Transformation by Principal Component analysis (PCA) is the most commonly used method for removing or reducing redundancy in multivariate data. Although it is very popular, PCA has the drawback that it is not invariant to rescaling or other linear transformations of the data. Further, Chang (1983) shows that, in general, the components with the larger eigenvalues do not necessarily contain most information about the signal. In fact, if the data were pretransformed to be uncorrelated and with the same variance, PCA would not be able to distinguish between any of the directions.

The Min/Max Autocorrelation Factors (MAF) process is a method introduced by Switzer and Green (1984). This method produces uncorrelated variables that are ordered according to the degree of spatial autocorrelation. Since noise is commonly associated with low spatial autocorrelation, MAF provides a method for noise separation which is more rational than the variance criterion associated with standard PCA. Section 3 contains some results for when MAF is optimal in the sense of maximizing the variance of signal to variance of noise ratio. These criteria are quite restrictive, however some heuristic arguments are given for that MAF should perform reasonable when the features in the image are large.

In section 4 we will introduce a new method for separating signal from noise, called Restricted Min/Max Autocorrelation Factors (RMAF). This method also produces uncorrelated variables that are ordered according to the degree of spatial autocorrelation. We
are, however, restricting ourselves to observations lying in the interior of the features in the image. This method will be optimal in the sense of maximizing signal to noise ratio under quite general assumptions.

Calculating these factors would in principle require knowledge of the features in the image, which will not be the case. Still, approximate estimates can be obtained using techniques from robust statistical theory. Simulation experiments have shown that large improvements from MAF, in the sense of giving a larger average mahalanobis distance between the classes, can be obtained by using RMAF, especially when the features in the image are small.

The outline of the paper is as follows: Section 2 contains the assumptions that will be made throughout this paper. Section 3 reviews the MAF-method and derive some of its properties. In section 4 we will introduce the RMAF method and discuss several techniques for calculation it. Section 5 contains a simulation experiment where the different methods are compared, while section 6 make some final remarks.

2. Model assumptions and notations

Constructing methods for separating noise from a signal is done under some assumption of how the signal behaves compared to noise. Optimality in special cases is then proved in form of some performance measure. PCA makes the assumption that the signal will vary much more than noise along any direction in the sample space. Although this may seem to be a reasonable assumption, it is easy to see that in many situations this will not be the case. Chan(1983) considers this in more detail. In the special case where each variable has equal variance and the correlation between two variables is the same for all pair of variables, PCA actually will be optimal in the sense of maximizing variance of signal to variance of noise ratio.

Switzer and Green's MAF process takes another approach in that noise is typically weakly autocorrelated compared with the signal and seeks to find the directions that have low autocorrelation, expecting these will contain mainly noise. Under the proportional covariance model, that is the cross-variance of the signal and noise each attenuates at different rates as a function of spatial lag $\Delta$, Switzer and Green(1984) shows that MAF is optimal (again in the sense of maximizing the variance of signal to variance of noise ratio).

In this paper we start by first choosing a performance measure. Under some general model assumptions we will then try to optimize this performance measure. This will lead to the RMAF process. The performance measure we will use is the variance of signal to variance of noise ratio.

The assumptions we will make are mainly that the underlying image, that is the spatial grid containing the observations, is (by an unknown stochastic process) built up by a finite number of classes and that the signal within each class is constant. Also assumptions about stationarity are assumed.

Assumptions:

1. The data consist of $p$-dimensional observation vectors measured on each location in a $m \times n$ grid. We allow for the possibility of $n = 1$, which correspond to the time series case. We have restricted ourselves to images in one or two dimensions, although all the methods discussed in this paper may be generalized to images of any dimensions.
2. The observation $Z(x)$ at point $x$ consists of a signal $S(x)$ contaminated by noise $\epsilon(x)$;

$$Z(x) = S(x) + \epsilon(x).$$

(2.1)

The signal and noise parts of the observation vector are assumed uncorrelated.

3. The noise has mean zero and covariance matrix $\Sigma_\epsilon$. Noise at different locations are uncorrelated. This assumption may in many cases be unreasonable, but is made here mainly to simplify the evaluation of the methods. The crucial assumption is that the autocorrelation in the signal is larger than the autocorrelation in the noise, although the problem of separating noise from signal will increase with autocorrelation in the noise.

4. Each location is assumed to belong to one of a finite number, $K$, of classes. The proportion of class $k$ is $\pi_k$. Furthermore, $S(x) = \mu_C(x)$, where $C(x)$ is the class at location $x$ and $\mu_C$ is the expected value of the observation given the underlying class is $C$. The expected value of a randomly chosen point will then be $\sum_k \mu_k \pi_k$ which we will denote by $\bar{\mu}$.

5. $\Pr\{C(x_i) = k, C(x_j) = l\} = \pi_{kl}$ for $d(x_i, x_j) = 1$. That is, $\pi_{kl}$ is the proportion of neighbor locations belonging to classes $k$ and $l$.

Under these assumptions, it is possible to get an explicit expression for the signal to noise ratio along a direction $a$ (that is the observations $Z$ are transformed to $a^T Z$):

$$\frac{\text{var}(a^T S(x))}{\text{var}(a^T \epsilon(x))} = \frac{\sum_{k=1}^K (a^T \mu_k - a^T \bar{\mu})^2 \pi_k}{(a^T \Sigma_\epsilon a)}$$

$$= \sum_k (a^T \mu_k - a^T \bar{\mu})(a^T \Sigma_\epsilon a)^{-1}(a^T \mu_k - a^T \bar{\mu}) \pi_k$$

$$= \sum_{k \neq l} (a^T \mu_k - a^T \mu_l)(a^T \Sigma_\epsilon a)^{-1}(a^T \mu_k - a^T \mu_l) \pi_k \pi_l,$$

which is recognized as the weighted sum of the square of the mahalanobis distances between the classes along the direction $a$, the weights being the corresponding class probabilities. Written in this form, it is possible to generalize the variance to ratio measure to a $q$-dimensional space. Let $T$ be a linear operator that transform $Z$ from $\mathcal{R}^p$ to $\mathcal{R}^q$. The information in the transformed data is then defined to be

$$\text{Inf}_T = \sum_{k \neq l} (T^T \mu_k - T^T \mu_l)(T^T \Sigma_\epsilon T)^{-1}(T^T \mu_k - T^T \mu_l) \pi_k \pi_l.$$

(2.2)

The goal of this paper is to find a transformation of the data to a lower dimensional space without loosing too much information. If the parameters $K, \mu_k, k = 1, \ldots, K$ and $\Sigma_\epsilon$ were known, the optimal directions could actually be found as the first $q$ principal components of the matrix (we will call this the information matrix)

$$\sum_{k \neq l} \Sigma_\epsilon^{-1/2}(\mu_k - \mu_l)(\mu_k - \mu_l)^T \Sigma_\epsilon^{-1/2} \pi_k \pi_l.$$
However, in general, these parameters will be unknown. Estimation of these parameters when there is no autocorrelation in the data is extensively considered in the literature. Titterington, Smith and Makov (1985) and McLachlan and Basford (1988) discuss procedures for finding the maximum likelihood estimators for $K, \mu_k, k = 1, \ldots, K$ and $\Sigma_e$. These procedures involve some iterative algorithms, e.g. the EM-algorithm, that are computationally expensive for even low-dimensional data. In our situation, with autocorrelated observations of high dimensions, finding the maximum likelihood estimators is a formidable task, both involving a complicated modeling step and expensive computation. Methods that do not involve estimation of the parameters are therefore desirable. Actually, in many cases the aim for reducing the dimension of the data is to make it possible to compute such estimators.

The next section will evaluate the MAF procedure in terms of (2.2) while in section 4 we will see how the matrix (2.3) can be approximately estimated without going through any complicated modeling step.

3. Min/Max Autocorrelation Factors process

The MAF transform is defined in Switzer and Green (1984) as follows:

Let $Z = (Z_1, \ldots, Z_p)$ be the original observation vector. Then $Z^M = (Z^M_1, \ldots, Z^M_p)$ is called MAF if it has the following properties:

\[ Z^M_i(x) = a'_i Z(x), \quad i = 1, \ldots, p, \]
\[ \text{corr} \ (Z^M_i(x), Z^M_i(x + \Delta)) = \min \ a \ \text{corr} \ (a'_i Z(x), a'_i Z(x + \Delta)), \]
\[ \text{under the restriction that} \quad \text{corr} \ (a'_i Z(x), Z^M_j(x)) = 0 \quad \text{for} \quad j < i. \]  

Here $\Delta$ is the spatial lag.

The motivation for MAF is that noise is often identified as quantities with low spatial autocorrelation, while signal will usually have high spatial autocorrelation. It is then reasonable to expect that the low numbered factors will contain mainly noise. Unlike PCA, MAF has the important property that it is invariant to linear transformation.

The MAF procedure has other formulations which permit the use of standard multivariate routines to extract the vectors. Specifically, the factors are obtained as the eigenvectors of a matrix $S_2 = S_2 S_1^{-1}$ where

\[ S_1 = \text{cov} \ \{Z(x)\}, \]
\[ S_\Delta = \text{cov} \ \{Z(x) - Z(x + \Delta)\} \]  

This suggest the following computational procedure for two dimensional images:

(i) The original data $Z(x)$ are linearly transformed to any $Z^*(x)$ where $\text{cov} \ (Z^*(x)) = I_p \times p$.

(ii) Form two sets of differences of the orthogonalized data,

\[ [Z^*(x) - Z^*(x + \Delta')] \quad \text{and} \quad [Z^*(x) - Z^*(x + \Delta'')], \]  

\[ [Z^*(x) - Z^*(x + \Delta')] \quad \text{and} \quad [Z^*(x) - Z^*(x + \Delta'')], \]
where $\Delta'$ is a unit horizontal shift and $\Delta''$ is a unit vertical shift; calculate the corresponding two global covariance matrices $S_2'$ and $S_2''$ and pool them to form $S_2$.

(iii) Obtain the principal components corresponding to the pooled covariance matrix $S_2$. This is the MAF solution. Notice that the components with the highest autocorrelations correspond to the components with the smallest eigenvalues.

To evaluate MAF, we will look at its asymptotical performance. MAF involves computation of two matrices (we will use the notation $Z_{i,j} = Z(x)$ for $x = (i,j)$ and $\bar{Z}$ for the average of all $Z$'s):

\[
S_1 = \frac{1}{mn} \sum_{i,j}(Z_{i,j} - \bar{Z})(Z_{i,j} - \bar{Z})'
\]

\[
S_2 = \frac{1}{2nm - n - m} \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} (Z_{i,j} - Z_{i+1,j})(Z_{i,j} - Z_{i+1,j})' + \frac{1}{2nm - n - m} \sum_{j=1}^{n-1} \sum_{i=1}^{m-1} (Z_{i,j} - Z_{i,j+1})(Z_{i,j} - Z_{i,j+1})',
\]

where $Z_{i,j} = S_1^{-1/2}Z_{i,j}$ and $Z'$ is the transpose of $Z$.

Under the assumptions in section 2, for large $nm$,

\[
S_1 \simeq A_1 = \Sigma_\epsilon + \sum_{k=1}^{K} \mu_k \mu_k' \pi_k
\]

\[
= \Sigma_\epsilon + \frac{1}{2} \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l,
\]

and ($\Sigma_\epsilon^* = A_1^{-1/2}\Sigma_\epsilon A_1^{-1/2}$ and $\mu_k^* = A_1^{-1/2}\mu_k$)

\[
S_2 \simeq A_2 = 2\Sigma_\epsilon^* + \sum_{k \neq l} (\mu_k^* - \mu_l^*)(\mu_k^* - \mu_l^*)' \pi_{kl}
\]

\[
= A_1^{-1/2}[2\Sigma_\epsilon + \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_{kl}]A_1^{-1/2}
\]

\[
= A_1^{-1/2}BA_1^{-1/2},
\]

where

\[
B = 2\Sigma_\epsilon + \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_{kl}.
\]

Since the principal components of $A_1^{-1/2}BA_1^{-1/2}$ is equivalent to the principal components using $A_1^{-1}B$, we will sometimes use the latter. The following theorem shows the optimal performance of MAF in a special case.
Theorem 3.1

Suppose

\[ \pi_{ki} = \begin{cases} \frac{p \pi_k}{1 - \sum_k \pi_k}, & \text{if } k = i; \\ \frac{(1-p) \pi_{ki}}{1 - \sum_k \pi_k}, & \text{if } k \neq i, \end{cases} \quad (3.7) \]

and \( p > \sum_{m=1}^{K} \pi_m^2 \). Then

\[ A^{-1}_1 B = 2I - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})P' \Lambda^* P, \quad (3.8) \]

where \( P' \Lambda P \) is the spectral decomposition of \( \sum_{k \neq i} \Sigma_e^{-1} (\mu_k - \mu_i)(\mu_k - \mu_i)' \pi_k \pi_i \), and \( \Lambda^* = [I + \frac{1}{2} \Lambda]^{-1} \Lambda \).

Furthermore, the principal components of \( A^{-1}_1 B \) is equivalent to the principal components of the information matrix (2.3).

The assumption (3.7) corresponds to the assumption that the probability of two neighbor locations belonging to the same class is independent of the underlying class and that, given they are different, the class in one location does not give any information about the class in the other location. The constant \( p \) is the probability for two neighbor pixels belonging to the same class. If the classes are randomly distributed over the image, that is if there is no structure, \( p \) would be equal to \( \sum_m \pi_m^2 \), and an increase in \( p \) correspond to an increase in the sizes of the features in the image. The assumption \( p > \sum_{m=1}^{K} \pi_m^2 \) is therefore quite reasonable.

Proof:

Under (3.7), \( B \) can be written as

\[ B = 2\Sigma_e + \frac{1-p}{1 - \sum_m \pi_m^2} \sum_{k \neq i} (\mu_k - \mu_i)(\mu_k - \mu_i)' \pi_k \pi_i \]

Then
\[ A_1^{-1}B = [\Sigma_e + \frac{1}{2} \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l]^{-1} \times \\
[2 \Sigma_e + \frac{1-p}{1 - \sum_m \pi_m^2} \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l] \\
= [\Sigma_e + \frac{1}{2} \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l]^{-1} \times \\
[2 \Sigma_e + (1 - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})) \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l] \\
= 2I - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})[\Sigma_e + \frac{1}{2} \sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l]^{-1} \times \\
\sum_{k \neq l} (\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l \\
= 2I - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})[I + \frac{1}{2} \sum_{k \neq l} \Sigma_e^{-1}(\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l]^{-1} \times \\
\sum_{k \neq l} \Sigma_e^{-1}(\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l. \\

By using the spectral decomposition, we get

\[ A_1^{-1}B = 2I - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})[I + \frac{1}{2} P'\Lambda P]^{-1}P'\Lambda P \\
= 2I - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})P'[I + \frac{1}{2} I]^{-1}P \\
= 2I - (1 - \frac{1-p}{1 - \sum_m \pi_m^2})P'\Lambda^*P. \]

Now since \( \Lambda_{ii} \geq 0 \) for all \( i \) and \( f(\lambda) = (1 + \frac{\lambda}{2})^{-1}\lambda \) is an increasing function of \( \lambda \) for \( \lambda > 0 \), finding the principal components of \( A_1^{-1}B \) is equivalent to finding the principal components of \( \sum_{k \neq l} \Sigma_e^{-1}(\mu_k - \mu_l)(\mu_k - \mu_l)' \pi_k \pi_l \), which again is equivalent to finding the principal components of \( (2.3) \). The restriction \( 1 - (1-p)/(1 - \sum_m \pi_m^2) > 0 \), that is \( p > \sum_m \pi_m^2 \), is needed to give the right order of the directions.

Notice that if \( \pi_k = 1/K \), the model in theorem 3.1 will satisfy Switzer and Green's proportional covariance model, and their optimality result corresponds to the result above.

Although reasonable in some cases, the model assumptions in theorem 3.1 will usually not be satisfied. However, MAF has another nice property that follows from the next theorem.
Theorem 3.2
Suppose \( \mu_k = (\mu_k^1, \ldots, \mu_k^q)' \), where \( \mu_k^i \) is a \( q \)-dimensional vector, \( q < p \). Suppose, further, that \( \pi_{kl} < \pi_k \pi_l \), for \( k \neq l \). Then

\[
A^{-1}_1B = \begin{pmatrix}
A^{-1}_{11}B_{11} & 0 \\
0 & 2I_{p-q}
\end{pmatrix},
\]

(3.9)

and the eigenvalues of \( A^{-1}_{11}B_{11} \) are all \( \leq 2 \).

Since MAF is invariant to linear transformations, this result actually says that if there are some directions that only contains noise, MAF will asymptotically find these as the first \( q \) factors, provided \( \pi_{kl} < \pi_k \pi_l \) for \( k \neq l \). This last restriction says that two different classes should appear less frequently as neighbors as if they were randomly distributed, which seems to be a reasonable assumption when there is some structure in the image.

Proof:
Without loss of generality, we can assume \( \Sigma_\epsilon = I \). It is then easy to see from (3.4) and (3.6) that \( A^{-1}_1B \) is of the form (3.9) with

\[
A_{11} = I_q + \frac{1}{2} \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'\pi_k \pi_l,
\]

and

\[
B_{11} = 2I_q + \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'\pi_{kl}
= 2I_q + \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'\pi_k \pi_l - \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'(\pi_k \pi_l - \pi_{kl}).
\]

Then

\[
A^{-1/2}_{11}B_{11}A^{-1/2}_{11} = 2I_q - A^{-1/2}_{11} \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'(\pi_k \pi_l - \pi_{kl})A^{-1/2}_{11}.
\]

We are done if the eigenvalues of the last term in the expression above are greater than or equal to 0. Let \( P'_{11}A_{11}P_{11} \) be the spectral decomposition of \( A_{11} \). Then

\[
A^{-1/2}_{11} \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'(\pi_k \pi_l - \pi_{kl})A^{-1/2}_{11}
= P'_{11}A^{-1/2}_{11}P_{11} \sum_{k \neq l} (\mu_k^1 - \mu_l^1)(\mu_k^1 - \mu_l^1)'(\pi_k \pi_l - \pi_{kl})P_{11}A^{-1/2}_{11}P_{11}
= \sum_{k \neq l} [P_{11}A^{-1/2}_{11}P_{11}(\mu_k^1 - \mu_l^1)][P_{11}A^{-1/2}_{11}P_{11}(\mu_k^1 - \mu_l^1)']'(\pi_k \pi_l - \pi_{kl}),
\]

8
which is positive semidefinite under our assumptions, and therefore has non-negative eigenvalues.

In practice, we will only have estimates of the quantities involved, and some variability will occur. An important question is how this will influence the MAF-components. Of course, when the class differences are small, it will not be possible to distinguish the signal from the noise, and reducing the dimension will be impossible for any method. This effect on MAF is seen from (3.4) and (3.5), because with small \( \mu_k \)'s, \( S_2 \simeq 2I \). Equation (3.8) shows another undesirable property of MAF. When the class differences are large, which corresponds to large eigenvalues \( \Lambda_{ii} \), then \( \Lambda_{ii}^* \simeq 2 \) for all \( i \), and

\[ A_1^{-1} B \simeq (1 + 2 - \frac{1 - p}{1 - \sum_m \pi_m^2}) I \]

under the assumption (3.7). This effect will occur also outside the model (3.7). Figure 3.1 illustrate these effects on an simulation experiment with \( n = 100, m = 1, K = 2, \mu_1 = (0, 0)', \mu_2 = (d, 0)', \pi_1 = \pi_2 = .5, \pi_{11} = \pi_{22} = .8 \) and gaussian noise with expectation zero and covariance matrix equal to \( I \). The simulations were done with \( d \)-values equal to 1, 2, 3, 4, 6, however the \( d \)-values in the figure are the mahalanobis distances between the two classes calculated by using the simulated data. At each \( d \)-value, 20 simulations were performed, and the figure shows the percentage of information in the first component. We see the effect on MAF for small values on \( d \). For large values of \( d \) the effect on MAF does not seem to influence the variability in the range of interest, at least for this particular example.

Another feature of MAF is that its performance decreases, in that it becomes more variable, with decreasing spatial correlation. This can be seen from (3.8) when \( p \) decreases towards \( \sum_m \pi_m^2 \). This situation is more serious when the assumption (3.7) is not satisfied. If there are high autocorrelation in the signal, this will correspond to small \( \pi_{kl} \)'s, and \( S_2 \) given in (3.5) will be approximately equal to (2.3). This may be used as an argument for the performance of MAF in situations not covered by theorems 3.1 and 3.2. However, when the autocorrelation decreases, this approximation will get worse, and it is difficult to predict how MAF will perform for our information criterion. Figure 3.2 show this effect on the same experiment as for Figure 3.1, but with \( d = 2 \) and \( p = .575, .720, .815, .860 \).

Although theorem 3.2 shows that MAF asymptotically gives the optimal directions in some interesting cases, there are situations where we would like to reduce the dimensions, but MAF does not perform well. This will be the case when some directions mainly consist of noise. The following example illustrate this.

**Example 3.1**

Consider the situation with three classes and bivariate observations. Let \( \pi_1 = \pi_2 = \pi_3 = 1/3 \). Let \( \Pi = (\pi_{kl}) \) be given by

\[
\Pi = \begin{pmatrix}
.300 & .017 & .017 \\
.017 & .233 & .083 \\
.017 & .083 & .233 \\
\end{pmatrix}
\]

(3.10)

Let further \( \mu_1 = (-2, 1)', \mu_2 = (-3, -1)', \mu_3 = (5, 0) \) and let the covariance matrix of the noise be the identity matrix. In this case, the optimal direction, in the sense that maximizes
(2.2), is (.9996, .0277) and the information contained along this direction is 25.36 of a total of 26.66 in the full 2-dimensional case. MAF, however finds the direction (.82, -.58) which contains only 16.8 of the information.

In the next section we improve the MAF-method to handle situations like this.

4. Restricted Min/Max Autocorrelation Factors

This section presents a new method for noise separation. Although we will see that the method is similar to MAF, it takes another approach in estimating the information matrix

$$\text{INF} = \sum_{k \neq l} \Sigma_e^{-1/2} (\mu_k - \mu_l)(\mu_k - \mu_l)' \Sigma_e^{-1/2} \pi_k \pi_l$$

(4.1)

(or actually $(I + \text{INF})^{-1}$) directly. After the transformation of the data so that the covariance matrix equals the identity (i) on page 6), differences between observations from the same class will have expectation 0 and covariance matrix approximately $(I + \text{INF})^{-1}$. MAF uses the differences between observations 1 unit distance apart to calculate a covariance matrix (3.3). When the image contains large features, these differences will mainly contain differences between observations from the same class, while a small proportion will be differences of observations from different classes. As we discussed in the previous section, this may be an argument for why MAF can give reasonable results in this case. However, as we saw in example 3.1, the proportion of differences from different classes may in some cases give a “bias” term in the covariance matrix that will move the directions obtained by MAF far apart from the optimal ones. Although also these differences will have a covariance matrix equal to $(I + \text{INF})^{-1}$, they will have different expectations in that differences of observations from the same class will have expectation 0, while the other differences will have non-zero expectations corresponding to the distances between the classes. If we can separate out the latter differences, and calculated the covariance matrix of the rest, estimates of the optimal directions will be found. Now

$$\text{var}(Z^* (X) - Z^* (x + \Delta)) = \text{var}(Z^* (X)) + \text{var}(Z^* (x + \Delta)) - 2 \text{cov} (Z^* (x), Z^* (x + \Delta))$$

$$= 2I - 2 \text{cov} (Z^* (x), Z^* (x + \Delta)).$$

(4.2)

Finding the direction that minimizes $\text{var}(Z^* (X) - Z^* (x + \Delta))$ among differences of observations at locations from the same class is therefore equivalent to finding the direction maximizing the autocorrelation at spatial lag $\Delta$ restricted to observations in the interior of the classes appearing in the image. These directions will be denoted Restricted Min/Max autocorrelation factors (RMAF).

If we identify the differences with non-zero expectations as “outliers”, we are in a problem setting that has been considered by numerous scientists in the last 25 years, and a large amount of procedures are available which are robust toward outliers. This suggests the possibility of using a robust estimation technique for calculating the covariance of the differences. This approach gives the following computational procedure:
(i) The original data $Z(x)$ are linearly transformed to any $Z^*(x)$ where $\text{cov}(Z^*(x)) = I_p$.

(ii) Form two sets of differences of the orthogonalized data,

$$[Z^*(x) - Z^*(x + \Delta')] \quad \text{and} \quad [Z^*(x) - Z^*(x + \Delta'')]$$

where $\Delta'$ is a unit horizontal shift and $\Delta''$ is a unit vertical shift; calculate the corresponding two global covariance matrices $S^*_2'$ and $S^*_2''$ by a robust procedure and pool them to form $S^*_2$.

(iii) Obtain the principal components corresponding to the pooled covariance matrix $S^*_2$. This is the RMAF solution.

This procedure has, like MAF, the desirable property of being invariant to linear transformations of the data. Unlike MAF, however, it increases in performance as the distances between the classes increases, since the "outliers" then are easier to identify. It is therefore not difficult to construct examples where RMAF will perform better than MAF. The important question, however, is comparing their performance in practical situations.

The procedure is not totally specified, since we still have to choose what robust estimation method to use. We will now discuss some robust techniques for estimating covariance matrices. Notice that in our case, the location is known to be zero, which will simplify the estimation problem considerably. Estimation techniques for multivariate location and scatter are usually divided into affine-equivariant and non-affine-equivariant estimators. Since the latter group does not automatically produce positive definite covariance matrices, only affine-equivariant estimators will be considered. To simplify the notation, we will label the difference measurements as $Y_1, \ldots, Y_N$.

A. Affine Equivariant M-estimators

Maronna(1976) formally introduced affine equivariant M-estimators of scatter as the solution $C$ of the equation

$$N^{-1} \sum_{i=1}^{N} u(Y'_i C^{-1} Y_i) Y_i Y'_i = C,$$  \hfill (4.3)

where $u$ is a weight function satisfying a set of general assumptions, (see section 2 of Maronna(1976)). Huber(1981, p.238) describes an algorithm for the numerical calculation of the estimator.

In the case of spherically symmetric contaminated normal densities, the minimax estimate is found by choosing (Huber(1981), p.223)

$$u(r) = \begin{cases} \frac{a^2}{r^2}, & \text{for } 0 \leq r \leq a; \\ b, & \text{for } a \leq r \leq b; \\ \frac{b^2}{r^2}, & \text{for } b \leq r, \end{cases}$$  \hfill (4.4)

the constants $a$ and $b$ being determined by the amount of contamination, that is the proportion of "outliers". Since we are only concerned about "outliers" that are shifted in location, it seems reasonable to choose $a = 0$. 

11
Since the $b$-value corresponds to a truncation value for observations given full weight, we would like to choose $b$ small so the "outliers" will be given small weights. On the other hand, small $b$-values will cause a problem when the dimension $p$ is large, since the fraction of observations given full weight will be small.

Another problem occurs when the features in the image get smaller, since the number of "outliers" will increase. The breakdownpoint of an estimator is often used as a measure of the proportion of bad outliers the estimator can cope with. These estimators have a low breakdown point, $1/(p + 1)$. However, this breakdownpoint is calculated without knowledge of the distribution of the outliers. In our case the outliers are obtained from shifting the mean. The outliers are also symmetric around zero for stationary signals. Therefore we expect the estimator to have a higher breakdownpoint in our case. However, the proportion of "outliers" in the data may exceed the breakdownpoint when $p$ is large.

This discussion suggests the possibility of using estimators with higher breakdown points. There have been several attempts to find such estimators in the literature. We describe two of them below.

B. "Outlyingness-weighted estimators"

The first affine equivariant multivariate location and scatter estimator with a 50% breakdown point was obtained independently by Stahel(1981) and Donoho(1982). For each observation $Y_i$ one looks for a one-dimensional projection leaving it most exposed. This is done by computing the following measure of the "outlyingness" of $Y_i$:

$$t_i = \sup_{\|v\|=1} \frac{|Y_i v' - \text{med}_j(Y_j v')|}{\text{med}_k |Y_k v' - \text{med}_j(Y_j v')|},$$

(4.5)

where $\text{med}_j(Y_j v')$ is the median of the projections of the data points $Y_j$ on the direction $v$, and the denominator is the median absolute deviation of these projections. Then the covariance matrix is estimated by

$$C(Y) = \frac{\sum_j u(t_j) Y_i Y_i'}{\sum_j u(t_j)}$$

(4.6)

where $u(t)$ is a weight function satisfying a set of general assumptions.

To compute $t_i$ one must in general search over all possible directions. However, we are in the fortunate situation of knowing the location. The weights (4.5) can therefore be simplified to

$$t_i = \frac{|Y_i v'_i|}{\text{med}_j |Y_k v'_i|},$$

(4.7)

where $v_i$ is the direction between zero and $Y_i$.

C. Minimum volume ellipsoid estimator

Rousseeuw (1985) introduced a second affine equivariant estimator with high breakdown point. This estimator was defined as the minimal volume ellipsoid covering (at least) $h$ points of $Y_i$, multiplied by a suitable factor to obtain consistency. Rousseeuw
and Leroy (1987, section 7.1 c) contains a proof of the breakdown point being equal 
\((\lceil N/2 \rceil - p + 1)/N\). They also give an approximate calculation algorithm similar in spin 
to the bootstrap. Start by drawing a subsample of \(q\) different observations, indexed by 
\(J = \{i_1, \ldots, i_q\}\). For this subsample determine the corresponding covariance matrix (using 
the location is zero), given by

\[
C_J = \frac{1}{q - 1} \sum_{i \in J} Y_i Y_i'.
\] (4.8)

The corresponding ellipsoid is then inflated or deflated to contain exactly \(h\) points, which 
corresponds to computing

\[
m_J^2 = \text{med} \, Y_i' C_J^{-1} Y_i
\] (4.9)

The volume of the resulting ellipsoid, corresponding to \(m_J^2 C_J\), is proportional to

\[
(\det(m_J^2 C_J))^{1/2} = (\det(C_J))^{1/2}(m_J)^p.
\] (4.10)

This has to be repeated for many \(J\), after which the lowest objective function (4.10) is 
retained.

The number of repetitions can be obtained by using the probability that at least one 
out of \(m\) subsamples consists exclusively of “good” points is approximately 
\(1 - (1 - (1 - \eta)^{p+1})^m\) when the original data contains a fraction \(\eta\) of “outliers”.

5. Simulation experiments

In this section we will explore the methods discussed in sections 3 and 4 further by 
looking at their performance on simulated data. We will consider 3 different examples. In 
the first experiment, data-locations are equally spaced on the real line while in the second 
and third experiments data-locations are on a grid in \(\mathbb{R}^2\). In all the experiments the data 
were generated according to section 2 with independent gaussian noise and covariance 
the identity matrix. The choice of this covariance matrix is done without any loss of 
generality when evaluating MAF and RMAF, since both methods are invariant to linear 
transformations. PCA will actually be optimal in this case but bearing in mind that 
it is not invariant to linear transformations and that the covariance matrix of noise is 
rarely equal to the identity matrix, this optimality characteristic should not be taken too 
serious. Its performance can, however, be used as a measure for evaluating the MAF and 
RMAF simulations. Throughout this section we will refer to RMAF combined with the 
three methods for robust covariance matrix estimation described in section 4 as RMAF 
A, RMAF B and RMAF C, respectively. For each image, the proportions of classes and 
proportions of pair of classes in two neighbor pixels are calculated. We will refer to these as 
\(\pi_k\) and \(\pi_{kl}\). For each experiment, we have tried out different levels of signal, for exploring 
the effects on the methods for different variance of signal to variance of noise ratio levels.
Simulation experiment 1

In this experiment we consider the situation in example 3.1. There are three classes with \( \pi_1 = \pi_2 = \pi_3 = \frac{1}{3} \). A markov chain was generated with these class probabilities and transition probabilities given by (3.10). The average class proportions in the resulting chain where \( \pi_1 = .48, \pi_2 = .28, \pi_3 = .24 \), and

\[
\Pi = (\pi_{k\ell}) = \begin{pmatrix}
.449 & .015 & .020 \\
.015 & .200 & .060 \\
.020 & .060 & .160
\end{pmatrix}.
\]

The signal was given by \( S(x) = \mu_C(x) \), where

\[
\mu_1 = c \begin{pmatrix} 5 \\ 0 \end{pmatrix}, \quad \mu_2 = c \begin{pmatrix} -2 \\ 1 \end{pmatrix}, \quad \mu_3 = c \begin{pmatrix} -3 \\ -1 \end{pmatrix}.
\]

The constant \( c \) was chosen at 5 different levels, \( c = .25, .375, .5, .75, 1 \). In example 3.1 we calculated the theoretical performance of MAF for \( c = 1 \). Table 5.1 shows (in \%) the amount of information in the first direction for the MAF transform and the optimal transformation. The optimal direction is unaffected by \( c \), however MAF performs worse when the distances between the classes increases. Note that for \( c = 1 \) we get a different number compared to what we found in example 3.1 since we here have used the probabilities in the simulated chain and not the probabilities they were simulated from. Added to the table is also \( d = \sqrt{\text{Inf}/(1 - \sum_m \pi_m^2)} \), \( d^2 \) is a weighted average of the square of the mahalanobis distances between the classes.

**Table 5.1**

Theoretical performance of MAF for simulation experiment 1

<table>
<thead>
<tr>
<th>( c )</th>
<th>.25</th>
<th>.375</th>
<th>.5</th>
<th>.75</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAF</td>
<td>91.9</td>
<td>90.1</td>
<td>86.6</td>
<td>74.9</td>
<td>62.9</td>
</tr>
<tr>
<td>Opt</td>
<td>93.3</td>
<td>93.3</td>
<td>93.3</td>
<td>93.3</td>
<td>93.3</td>
</tr>
<tr>
<td>( d )</td>
<td>1.46</td>
<td>2.19</td>
<td>2.92</td>
<td>4.38</td>
<td>5.84</td>
</tr>
</tbody>
</table>

\( d = \) average mahalanobis distance

For each \( c \)-value, 15 simulations were performed and the factors were calculated for PCA, MAF and RMAF A, B and C. For the affine-equivariant M-estimator (RMAF A) we used the weight function given in (4.4) with \( a = 0 \) and \( b = 1.5 \) (\( b = .5 \) was also tried out but gave very similar results). For the “Outlyingness-weighted” estimator (RMAF B) we used the same weight-function but with \( a = 0 \) and \( b = .5 \). For the “minimum volume ellipsoid” estimator (RMAF C), we used \( q = 2 \ast (p + 1) = 6 \). Rousseeuw and Leroy(1987) actually recommended using \( q = p + 1 \), but this gave much worse results in our experiments. 200 samples were then taken to calculate the minimum volume ellipsoid.

Figure 5.1 shows the performance of the different methods (% information in the first component) plotted against average mahalanobis distance between the classes in the total data. Figure 5.1 a,b,c,d and e give plots of the methods individually (each star corresponds to one simulation). To each plot we have added a smoothed line (using the S-command lowess) to give an indication of the average performance of the methods. Figure 5.1 f
show these smoothed lines only for comparison of the methods. As we see from the figures, MAF performs almost as well as PCA when \( d = 4 \), and increasing or decreasing this value degrades the performance. Comparing Figures 5.1 a and b, we also see that MAF varies more than PCA. RMAF A, however, is almost identical to PCA for high \( d \)-values \((d > 3.5)\). For lower \( d \)-values, the performance decreases, which was to be expected, since then the "outliers" are more difficult to detect. The variability also increases with decreasing \( d \)-values. However, RMAF A performs better than MAF for \( d \)-values down to 2. RMAF B gave slightly worse results than RMAF A, however the differences are small. The last method, RMAF C (figure 5.1 d), did not perform well in this experiment. The average performance is worse than MAF for \( d \)-values up to 6, and the variability is also high. Some of this is caused by the fact that the algorithm did not succeed in finding the minimum volume ellipsoid. For some simulations, 1000 samples were tried. This gave some small improvements, but nothing significant. Also different values of \( q \) were tried without giving any improvements.

**Simulation experiment 2**

The second example we consider is an image consisting of pixels (data-locations, picture elements) in a two-dimensional grid. The image is shown in figure 5.2. The class proportions in this image are:

\[
\pi_1 = .180, \quad \pi_2 = .283, \quad \pi_3 = .235, \quad \pi_4 = .107, \pi_5 = .195
\]

\[
\Pi = (\pi_k l) = \begin{pmatrix}
.073 & .061 & .012 & .003 & .028 \\
.061 & .151 & .062 & .007 & .006 \\
.012 & .062 & .129 & .027 & .006 \\
.003 & .007 & .027 & .052 & .019 \\
.028 & .006 & .006 & .019 & .134
\end{pmatrix}
\]

The signal was given by \( S(x) = \mu C(x) \), where

\[
\mu_1 = c \begin{pmatrix} -3 \\ -1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mu_2 = c \begin{pmatrix} 4 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mu_3 = c \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \mu_4 = c \begin{pmatrix} 1 \\ 0 \\ -2 \\ 1 \\ 0 \end{pmatrix}, \quad \mu_5 = c \begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \\end{pmatrix}.
\]

White gaussian noise with covariance equal to the identity matrix was added to the signal to generate the data. The constant \( c \) was chosen to be \(.375, .5, .625, .75, .875, 1.0. \) Table 5.2 shows the theoretical performance of MAF compared to the optimal performance. Also in this example, MAF degrades with increasing \( d \)-values. This is especially the case for the information contained in the first component.
Table 5.2
Theoretical performance of MAF for simulation experiment 2

<table>
<thead>
<tr>
<th>c</th>
<th>.375</th>
<th>.5</th>
<th>.625</th>
<th>.75</th>
<th>.875</th>
<th>1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAF</td>
<td>67.6</td>
<td>54.5</td>
<td>29.6</td>
<td>23.6</td>
<td>25.2</td>
<td>29.9</td>
</tr>
<tr>
<td>Opt</td>
<td>80.1</td>
<td>80.1</td>
<td>80.1</td>
<td>80.1</td>
<td>80.1</td>
<td>80.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAF</td>
<td>80.8</td>
<td>75.6</td>
<td>70.6</td>
<td>66.4</td>
<td>63.2</td>
<td>60.7</td>
</tr>
<tr>
<td>Opt</td>
<td>92.5</td>
<td>92.5</td>
<td>92.5</td>
<td>92.5</td>
<td>92.5</td>
<td>92.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAF</td>
<td>95.2</td>
<td>91.2</td>
<td>86.0</td>
<td>81.2</td>
<td>77.3</td>
<td>74.4</td>
</tr>
<tr>
<td>Opt</td>
<td>98.6</td>
<td>98.6</td>
<td>98.6</td>
<td>98.6</td>
<td>98.6</td>
<td>98.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d</td>
<td>1.91</td>
<td>2.55</td>
<td>3.18</td>
<td>3.82</td>
<td>4.46</td>
<td>5.09</td>
</tr>
</tbody>
</table>

*d* = average mahalanobis distance

For each c-value, 5 simulations were performed. Figure 5.3 shows the amount of information in the first component calculated by the different methods. As we see from Figure 5.3 a and b, MAF does not perform well. While the optimal direction (PCA) capture around 80% of the information, MAF, on average, capture only 50% for small d-values and down to 20% for higher d-values. From figure 5.5 b, we also see that MAF has a high variability. RMAF A capture a little under 70% of the information, and is quite independent of d in the range 2 to 8. Also RMAF A has some variability in this case. RMAF B and RMAF C did not perform well in this case, although both of them do better than MAF for high d-values.

Figure 5.4 shows the same plots for the two first components. The optimal directions contain around 90% of the information. MAF has a peak in its performance when d equals 3 and is almost optimal. However for larger and smaller d-values, the amount of information contained in the two first components decreases and the variability increases. RMAF A captures around 80% of the information in the data, a bit less for small d's and a bit higher for large d's. The variability is also less than for MAF, especially for high d-values. RMAF B performs well in this case. The results looks very similar to RMAF A, although it performs worse for low d's. RMAF C did perform well.

The results for the first three components are shown in figure 5.5. This is perhaps the most interesting case, since for the optimal transformation, almost all of the information is captured by the first three components. When d equals 4, MAF is almost optimal and decreases only slightly for smaller and larger d-values. There is, however some variability in the results. RMAF A again performs better, especially for high d-values. The variability is also much smaller. RMAF B looks very similar to RMAF A, although with a bit more variability. Again, RMAF C did not perform well.

Simulation experiment 3

The third example is again an image consisting of pixels in a two-dimensional grid. The image is shown in figure 5.6. This is an example with 6 classes where 4 of the classes
contain large features, while the last two classes are mixed together. This is typical the
Case for images of cities. The class proportions in the image are

\[ \pi_1 = .25, \quad \pi_2 = .21, \quad \pi_3 = .18, \quad \pi_4 = .19, \pi_5 = .08, \pi_6 = .09 \]

\[ \Pi = (\pi_k l) = \begin{pmatrix}
0.118 & 0.098 & 0.012 & 0.004 & 0.011 & 0.009 \\
0.099 & 0.072 & 0.013 & 0.013 & 0.014 & 0.009 \\
0.012 & 0.013 & 0.139 & 0.018 & 0.000 & 0.000 \\
0.004 & 0.013 & 0.018 & 0.148 & 0.000 & 0.001 \\
0.011 & 0.014 & 0.000 & 0.000 & 0.054 & 0.000 \\
0.009 & 0.009 & 0.000 & 0.000 & 0.000 & 0.074
\end{pmatrix} \]

The signal is given by \( S(x) = \mu_{C(x)} \), where

\[ \mu_1 = c \begin{pmatrix}
0.32 \\
-1.23 \\
0.55 \\
0.04
\end{pmatrix}, \quad \mu_2 = c \begin{pmatrix}
-2.01 \\
-0.77 \\
0.24 \\
-0.17
\end{pmatrix}, \quad \mu_3 = c \begin{pmatrix}
0.52 \\
-0.88 \\
1.46 \\
0.13
\end{pmatrix}, \quad \mu_4 = c \begin{pmatrix}
-1.17 \\
0.62 \\
-1.14 \\
0.23
\end{pmatrix}, \quad \mu_5 = c \begin{pmatrix}
0.49 \\
-0.94 \\
0.49 \\
0.12
\end{pmatrix}, \quad \mu_6 = c \begin{pmatrix}
1.17 \\
1.25 \\
-1.25 \\
0.03
\end{pmatrix} \]

The constant \( c \) was chosen at 7 different levels, \( c = .25, .375, .5, .625, .75, 1.0, 1.5, 2.0 \). Table 5.3 show the theoretical performance of MAF compared to the optimal performance.

Table 5.3
Theoretical performance of MAF for simulation experiment 3

<table>
<thead>
<tr>
<th>c</th>
<th>.25</th>
<th>.375</th>
<th>.5</th>
<th>.625</th>
<th>.75</th>
<th>1.0</th>
<th>1.5</th>
<th>2.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAF</td>
<td>70.5</td>
<td>70.4</td>
<td>70.2</td>
<td>70.1</td>
<td>70.0</td>
<td>69.6</td>
<td>68.8</td>
<td>67.6</td>
</tr>
<tr>
<td>Opt</td>
<td>71.9</td>
<td>71.9</td>
<td>71.9</td>
<td>71.9</td>
<td>71.9</td>
<td>71.9</td>
<td>71.9</td>
<td>71.9</td>
</tr>
</tbody>
</table>

components 1 and 2

| MAF  | 77.9| 77.7 | 77.5| 77.3 | 77.1| 76.7| 75.8| 74.5|
| Opt  | 93.2| 93.2 | 93.2| 93.2 | 93.2| 93.2| 93.2| 93.2|

components 1, 2, and 3

| MAF  | 86.1| 84.7 | 83.4| 82.4 | 81.8| 81.0| 80.6| 80.5|

d = average mahalanobis distance

For each \( c \)-value, 5 simulations were performed. Figure 5.7 shows the amount of
information in the first component calculated by the different methods. MAF performs
quite well in this case. For small \( d \)-values, it is better than PCA (!), while for higher \( d \)-
values, it is only slightly worse. The variability of MAF is not larger than that of PCA.
RMADF A and RMADF B look very similar to MAF, although both of them are bad for two
cases of low \( d \)-values. As in the first experiments, RMADF C does not do well.
For the first two components (figure 5.8) MAF is not able to capture as much information as PCA, as is expected from the theoretical calculations in table 5.3. This is the case especially for higher \( d \)-values. For lower \( d \)-values, however, MAF does quite well. RMAF A and RMAF B are similar and do slightly better than MAF, again for the highest \( d \)-values. Although on average RMAF C only do slightly worse than MAF, its variability is bad.

The amount of information in the first three components is the interesting case, since we see from table 5.3 that the optimal components theoretically should be able to capture almost all the information in the data, while MAF is not expected to capture more than 80-87%. This is also the case for the simulated data. The percents of information captured by PCA are all between 90 and 100%, except for three cases with small \( d \)-values. MAF varies between 80 and 100%, with an average of 87%. It is interesting, however, that for the three bad cases of PCA, MAF performs quite well. In this case, RMAF A and RMAF B are, on the average, significantly better for higher \( d \)-values. RMAF A does slightly better than RMAF B and also has smaller variability. For small \( d \)-values, RMAF A and MAF do equally well on the average, but RMAF A has a little more variability. RMAF C again performed worse than the other methods.

6. Concluding remarks

In the preceding sections several methods for separating noise from signal by linear pointwise transformations have been compared. RMAF A and B seems to be superior to MAF when the classes are not mixed together too much. RMAF A seems to do better than RMAF B. However, our simulation experiments have been limited to gaussian noise, for which the robust estimator in RMAF A is especially constructed. This is not the case for RMAF B. On the other hand, RMAF B requires more computation. RMAF C did not perform well in any cases we considered and does not seem to be suited to this problem. It is also the computationally most expensive method.

All the RMAF-methods are computational expensive to calculate. The M-estimators involve using an iterative algorithm, the “outlyingness-weighted estimator” involve computing the median of \( N \) numbers \( N \) times, while the “minimum volume ellipsoid estimator” involve taking a large number of samples of size \( q \) from the \( Y_i \)’s, compute the empirical covariance matrix and flat the corresponding ellipsoid out to contain \( h \) points. Finding methods for reducing the computation is therefore desirable.

Reducing the number of observations for estimating the robust covariance matrix is possible using the nature of the data in hand. Suppose we instead of the weight function (4.4) used the following function:

\[
u(r) = \begin{cases} 
1, & \text{for } 0 \leq r \leq b; \\
\frac{b^2}{r^2}, & \text{for } b \leq r \leq c; \\
0, & \text{for } c \leq r. 
\end{cases}
\]  

(6.1)

Then observations far from zero would be given weight zero, and therefore not be used in the estimation. Because of the nature of the data, the variance of the \( Y \)’s will be less or equal to 2 for any direction in the \( \mathbb{R}^p \) space. This follows by the same arguments as in the proof of theorem 3.2. Therefore observations with euclidian distance larger than
$c^2$ will have mahalanobis distance at least $c^2$ (that is $r > c$) for every possible covariance matrix, and will with the weight function above be given weight zero. These observations can therefore be discarded at once. The estimation by minimum volume ellipsoid does not involve any weight function, but since in practice the $q$ observations that give the smallest ellipsoid will be inside the circle with radius $c$ with high probability, the results should be similar.

Acknowledgements

I am grateful to professor Paul Switzer for helpful discussions and encouragement during this work. I am also thankful to the department of statistics, Stanford University for using their computer facilities, and to professor Narinder Singh for carefully reading through this paper.
References


Donoho, D. L. (1982), Breakdown properties of multivariate location estimators, qualifying paper, Harvard University, Boston, M.A.


Information (in %) in the first MAF-component as a function of the distance between the classes. Each star corresponds to one simulation.
Figure 3.2

Information (in %) in the first MAF-component as a function of \( p \)
Each star corresponds to one simulation.
Information (ln ω) in the first component as a function of d for simulation experiment

a. PCA
b. RMRF A
c. RMRF B
d. RMRF C
e. Combined
f. Combined
Figure 5.2

Underlying image for simulation experiment 2
Classes ordered according to greylevel, class 1 being white
Figure 5.3: Information in the first component as a function of d for simulation experiments 2, 4, and 7.

- a. PCA
- b. MAP
- c. RMAP A
- d. RMAP B
- e. RMAP C
- f. Combined
Figure 5.4
Information in the first two components as a function of \( d \) for simulation experiment 2

a. PCA

b. MAF

c. RMAF A

d. RMAF B

e. RMAF C

f. Combined
Figure 5.5

Information in the first three components as a function of d for simulation experiment 2

a. PCA

b. MAP

c. RMSE A

d. RMSE B

e. RMSE C

f. Combined
Figure 5.7
Information in the first component as a function of $d$ for simulation experiment 3

a. PCA

b. MAF

c. RMAF A

d. RMAF B

e. RMAF C

f. Combined
Figure 5.8
Information in the first two components as a function of \( d \) for simulation experiment 3.

- a. PCA
- b. RMAF A
- c. RMAF A
- d. RMAF B
- e. RMAF C
- f. Combined
Figure 5.9

Information in the first three components as a function of d for simulation experiment 3

a. PCA
b. NAF
c. NAF A
d. NAF B
e. NAF C
f. Combined