Subspace Optimization Techniques for Classification Problems

David Lindgren
Division of Automatic Control
Department of Electrical Engineering
Linköpings universitet, SE-581 83 Linköping, Sweden
WWW: http://www.control.isy.liu.se
E-mail: david@isy.liu.se

2003

Report no.: LiTH-ISY-R-2534
Submitted to YYY

Technical reports from the Control & Communication group in Linköping are available at http://www.control.isy.liu.se/publications.
Subspace Optimization Techniques for Classification Problems

David Lindgren

2003

Abstract

The nonlinear conjugate gradients method is a very powerful program in the search for Bayes error optimal linear subspaces for classification problems. In this report, techniques to find linear subspaces where the classification error is minimized are surveyed. Summary statistics models of normal populations are used to form smooth, non-convex objective functions of a linear transformation that reduces the dimensionality. Objective functions that are based on the Mahalanobis or Bhattacharyya distances and that are closely related to the probability of misclassification are derived, as well as their subspace gradients. Different approaches to minimize those objective functions are investigated: Householder and Givens parameterizations as well as steepest descent and conjugate gradient methods. The methods are evaluated on experimental data with respect to convergence rate and subspace classification accuracy.
5 Conclusions

Operators

\( E \left[ \cdot \right] \) Expectation or mean of a random variable
\( \text{tr} \left[ \cdot \right] \) Matrix determinant
\( \det \left[ \cdot \right] \) Matrix trace
\( \sup \) Supremum
\( \inf \) Infimum
\( \| \cdot \|_2 \) Vector 2-norm
\( \nabla \) Gradient
\( P(\cdot) \) “Probability of occurrence”

Symbols

\( n \) Dimensionality of original variable
\( k \) Dimensionality of transformed variable
\( S \in \mathbb{R}^{n \times k} \) Linear transformation, \( \mathbb{R}^n \to \mathbb{R}^k \)
\( s \in \mathbb{R}^n \) Linear combination \( (k = 1) \)
\( e_i \) \( [0^{(1)} 0^{(2)} \ldots 0^{(i-1)} 1^{(i)} 0^{(i+1)} \ldots 0^{(g)}]^T \)
\( g \) Number of populations
\( p \) Population index (or parameter vector)
\( \Sigma \in \mathbb{R}^{n \times n} \) Covariance matrix equal for all distributions, homoscedastic case
\( \Sigma_p \in \mathbb{R}^{n \times n} \) Covariance matrix of the distribution associated with population \( p \), heteroscedastic case
\( \mu_p \in \mathbb{R}^n \) Mean vector of the distribution associated with population \( p \)
\( m \) \( \mu_1 - \mu_2 \)
\( N \) Number of observations
\( N_p \) Number of observations of population \( p \)
\( N_n(\mu, \Sigma) \) \( n \)-variate normal distribution with mean \( \mu \) and covariance matrix \( \Sigma \)
\( b \) Bhattacharyya distance
\( \delta \) Mahalanobis distance
\( : \) Parameter estimate
\( V \) Bayes error
\( V_p \) Bayes error approximated by pair-wise union
\( f_p(\cdot) \) Population-conditional probability density function
\( Q(\cdot) \) Normal distribution function
\( Q(\cdot) \) Complementary error function
\( I_n \in \mathbb{R}^{n \times n} \) Identity matrix
\( F(\cdot) \) Objective function \( \mathbb{R}^{n \times k} \to \mathbb{R} \)
\( H(\cdot) \) Householder reflection
\( R_n(\cdot) \) Composite reflection
\( r(\cdot) \) Vector (Givens) rotation
\( G(\cdot) \) Composite rotation
\( \mathcal{V}_{n,k} \) Set of \( n \)-by-\( k \) orthonormal matrices; \( \{X \in \mathbb{R}^{n \times k}|X^TX = I_k\} \)
\( \mathcal{O}_n \) \( \mathcal{V}_{n,n} \)
1 Introduction

“When two or more populations have been measured in several characters, $x_1, \ldots, x_n$, special interest attaches to certain linear functions of the measurements by which the populations are best discriminated”, [Fisher, 1936].

Given a measurement or an observation that we know stems from one of $g$ known distributions (populations), the question from which one is classical and has been answered for many cases. From the cite above it is evident that the data compression of the observation is also a classical problem, but, as it turns out, one with solutions only for a small set of special cases.

This report concerns summary statistics models of normal distributed populations in $\mathbb{R}^n$, and investigates how a linear transformation of an arbitrary observation $x$ down to $k$ dimensions best should be selected to retain the ability to decide to which population it belongs. In this respect, it will be assumed that all $g$ populations have the same prior probability. For a particular set of populations a particular decision rule $c$ will select the wrong population with the probability $V_c$, which defines the error rate. No decision rule can however fall short of a certain uncertainty limit denoted the Bayes error, which will be defined below. In the search for linear transformations of $x$ with minimum Bayes error we will consider objective functions as well as optimization techniques.

1.1 Probabilistic Framework

We say the observation $x$ is a sample of the random variable $X$. The conditional distribution of $X$ knowing the population $p$ has the probability density function $f_p(\cdot)$, which completely defines the probabilistic properties of population $p$. If the $p$-conditional distribution is normal, for instance, the $p$-conditional density function reads

$$f_p(x) = \frac{e^{-(x-\mu_p)^T\Sigma_p^{-1}(x-\mu_p)/2}}{(2\pi)^{n/2}\det[\Sigma_p]^{1/2}}, \quad p = 1, 2, \ldots, g.$$  

(1)

We see that the normal population $p$ is completely specified by two parameters: a mean vector $\mu_p$ and an $n$-by-$n$ invertible covariance matrix $\Sigma_p$. Since the actual population is unknown to us, we consider also $p$ to be a sample of a random variable $P$, which is discrete and has, as mentioned earlier, a uniform distributed prior; the probability of $p$ is

$$P(p) = \begin{cases} \frac{1}{g} & \text{if } p = 1, 2, \ldots, g \\ 0 & \text{else.} \end{cases}$$  

(2)

The purpose of a decision rule (or classifier) is to accurately guess which population $p$ a given observation $x$ belongs to. If we take the decision rule to be maximum likelihood (maximum of the posterior density of $P$) and thus estimate population according to

$$\hat{p}: \quad f_{\hat{p}}(x) \geq f_i(x), \quad \hat{p}, i = 1, 2, \ldots, g,$$  

(3)

then the error rate will coincide with the Bayes error, which we define as

$$V \triangleq 1 - \frac{1}{g} \int \sup_i f_i(x) \, dx, \quad i = 1, 2, \ldots, g.$$  

(4)
This rule is referred to as the Bayes rule; no decision rule gives lower error rate under the assumptions made. It should immediately be said, however, there is no general analytical expression for $V$; we are often resigned to use bounds and approximations.

Whether we assume the covariance matrices of the distributions to be equal ($\Sigma_p = \Sigma \forall p$) or not, we distinguish between a homoscedastic and a heteroscedastic model, respectively. If the (normal) populations are homoscedastic, the Bayes rule is a linear function of $\mathcal{X}$, else quadratic. These results are standard and found in most textbooks on the subject, for instance [Fukunaga, 1990, Johnson and Wichern, 1998]. A recommended text on normal models and some of their linear transformations is [McLachlan, 1992, ch. 3].

1.2 Linear Transformation

The linear transformation we define by the $n$-by-$k$ matrix $S$, and our interest attaches to the transformed variable

$$\mathcal{Y} = S^T \mathcal{X}$$

and consequently on the transformed observation

$$y = S^T x \in \mathbb{R}^k.$$  

Note that we are only interested in the case $k < n$, which means that the dimensionality is reduced (data compression). It is well-known that any linear combination of normal distributed variables is also normal distributed. It can easily be shown that if the $p$-conditional distribution of $\mathcal{X}$ is $N_n(\mu_p, \Sigma_p)$, that is, $n$-dimensional normal with mean vector $\mu_p$ and covariance matrix $\Sigma_p$, then the distribution of $\mathcal{Y}$ under the same condition is $N_k(S^T \mu_p, S^T \Sigma_p S)$. Consequently, the Bayes error for the transformed variable $\mathcal{Y}$ is dependent of the transformation $S$ and defined for $S$ with full column rank by

$$V(S) \equiv 1 - \frac{1}{g} \int \sup_p \left[ \frac{e^{-\frac{1}{2}(y-S^T \mu_p)(S^T \Sigma_p S)^{-1}(y-S^T \mu_p)/2}}{(2\pi)^k |\det(S^T \Sigma_p S)|} \right] dy.$$  

1.3 Problem Formulation

Obviously $V(S)$ would be an excellent quality assessment for the transformation $S$. It should be clear that we can not improve the error rate (add information) by a transformation; $V(S) \geq V(I_n)$. We can however hope to find an $n$-by-$k$ transformation $S^*$ by which the increase in error rate is minimal, $V(S^*) \leq V(S)$ for all $n$-by-$k$ matrices $S$. The pursuit of good linear transformations thus lead us to the problem formulation

$$S^* = \arg \min_S V(S), \quad \text{s.t. } \text{rank } S = k.$$  

The analytical solution of (8) is however known only for the case with $g = 2$ homoscedastic populations and $k = 1$, namely the Fisher’s linear discriminant,

$$s^* = \arg \max_S |s^T (\mu_1 - \mu_2)|, \quad \text{s.t. } s^T \Sigma s = \Delta$$
for a fixed positive constant $\Delta$, or in closed form

$$s^* = \Sigma^{-1}(\mu_1 - \mu_2),$$

(10)

see [Fisher, 1936, Fisher, 1938]. In this case the Bayes error is

$$V = Q \left( -\frac{1}{2}(\mu_1 - \mu_2)^T \Sigma^{-1}(\mu_1 - \mu_2) \right)$$

(11)

where $Q(\cdot)$ is the standard normal distribution function

$$Q(q) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{q} e^{-\frac{t^2}{2}} dt.$$  

(12)

Note that $k = 1$ is sufficient to reach the infimum Bayes error for $g = 2$ homoscedastic normal populations. No improvement can be made by taking $k > 1$. This is not the case for $g > 2$ when (9) is often extended to

$$\tilde{S} = \arg \max_S \sum_{p=1}^{g} ||S^T(\mu_p - \mu)||_2^2, \ \text{s.t.} \ \ S^T S = I_k$$

(13)

where $\mu$ is the mean of $\mathcal{X}$. The range space of $\tilde{S}$ in (13) is however not Bayes error optimal in general (unless $g = 2$). The columns of $\tilde{S}$ are called the Fisher’s linear discriminants and the technique to use them is sometimes referred to as linear discriminant analysis (LDA). The LDA problem (13) is equivalent to the generalized singular value problem, see [Park et al., 2002] for numerically sound programs.

Example, Fisher’s Linear Discriminant

In Figure 1 on page 7 are plotted $N = 300$ samples of $g = 3$ homoscedastic populations with $n = 2$ dimensions. The decision boundaries of Bayes rule are also drawn. The populations are apparently fairly separated. By a transformation

$$y^{(j)} = s^T x^{(j)}, \ j = 1, 2, \ldots, N,$$

for $s$ in $\mathbb{R}^n$ the observations are compressed to $k = 1$ dimension. Figure 2 compares two such transformations by histogram-like dot diagrams (same data set as above). In 2(a) the transformation is the Fisher’s linear discriminant (13) and in (b) it is simply the first component of the observations, $x_1^{(j)}$. By inspection we conclude that the Fisher’s linear discriminant is not optimal in general; two populations are almost indistinguishable. Monte Carlo simulations give that $V = 0.24$ for the Fisher’s linear discriminant and $V = 0.060$ for $x_1$.

Example, Decision Boundaries

We will visually compare the Bayes rule of a homoscedastic model to that of a heteroscedastic. The models are estimated from a data set (Image Segmentation data, see page 43) where each of the $N = 211$ observations represents an image. The $g = 7$ distinct populations are due to images of different surfaces. The images are compressed to $k = 2$ dimensions. In Figure 3 the decision boundaries of the homoscedastic model are drawn in the same diagram as 2099 validation images. As mentioned before, the decision rule (for any two populations) is a linear function, which is seen clearly. The decision boundaries of the heteroscedastic model is much more flexible, see Figure 4. In this instance the empirical error rate (test error) is however the same for both models, $\hat{V} = 0.28$.  

6
Figure 1: Three homoscedastic normal distributed populations with distinct means, together with the decision boundaries of Bayes rule. The Bayes error is very low.
Figure 2: In (a) the Fisher’s linear discriminant (9) of the data in Figure 1 visualized as a histogram-like dot diagram, $k = 1$. In (b) a simple projection of the same data onto $x_1$. In comparison, the Fisher’s linear discriminant is apparently not optimal with respect to the Bayes error; two populations are almost indistinguishable in (a). The Bayes error is 0.24 for the Fisher’s linear discriminant and 0.060 for $x_1$. 

(a) Fisher’s linear discriminant: $0.1x_1 + 0.99x_2$

(b) $x_1$
Figure 3: Bayes rule decision boundaries of a homoscedastic model of the Image Segmentation data. Empirical error rate $\hat{V} = 0.28$. 
Figure 4: Bayes rule decision boundaries of a heteroscedastic model of the Image Segmentation data. Empirical error rate $\bar{V} = 0.28$ (same as for the homoscedastic model).
1.4 Prior Work

For the special case $k = 1$, [Geisser, 1977] shows that for $g \geq 2$ homoscedastic populations, the Bayes error (4) is

$$V(s) = \frac{2}{g} \sum_{i=1}^{g-1} Q \left[ \frac{1}{2} (v_{i+1}(s) - v_i(s)) \right],$$

where

$$v_i(s) = \frac{s^T \mu_i}{\sqrt{s^T \Sigma s}}$$

are ordered as $v_1(s) \geq v_2(s) \geq \cdots \geq v_g(s)$. This result is useful in conjunction with numerical minimization over non-zero linear combinations $s$ in $\mathbb{R}^n$. [Schervish, 1984] shows how this can be done for the case $g = 3$ by minimizing a convex function over an angle (scalar). [McCulloch, 1986] also considers a linear combination ($k = 1$), but for an arbitrary number of homoscedastic populations, $g \geq 2$. He minimizes (14) over linear combinations of two or more of Fisher’s linear discriminants (13). See also [Hudlet and Johnson, 1977].

1.4.1 Two Heteroscedastic Populations

The pursuit of optimal $n$-by-$k$ transformations in the heteroscedastic case is much more difficult. A simplified version considering the performance of a (suboptimal) linear decision rule discriminating $g = 2$ heteroscedastic populations (thus $k = 1$) is addressed by [Clunies-Ross and Rienburgh, 1960, Rienburgh, 1960, Anderson and Bahadur, 1962]. They use the decision rule $s^T x \geq c$ for the linear combination

$$s = [\omega_1 \Sigma_1 + \omega_2 \Sigma_2]^{-1} m$$

and

$$c = s^T \mu_1 - \omega_1 s^T \Sigma_1 s = s^T \mu_2 + \omega_2 s^T \Sigma_2 s.$$  

Here and below we take

$$m = \mu_1 - \mu_2$$

for notational purposes. $\omega_1$ and $\omega_2$ are open parameters that can be selected in different ways. If we let $\omega_1 = \omega_2 = \omega_0$ where

$$\omega_0 = \arg \min_{0 < \omega < 1} \Delta(\omega)$$

where in turn

$$\Delta(\omega) = \frac{2s^T m}{\sqrt{s^T \Sigma_1 s} + \sqrt{s^T \Sigma_2 s}}$$

the maximum of the probability of misclassifying an observation of distribution 1 and an observation of distribution 2 is minimized. The error rate is

$$V_{\text{linear, minimax}} = Q \left[ -\frac{1}{2} \Delta(\omega_0) \right],$$

11
see also [Anderson, 1984, pp. 240]. If the populations are homoscedastic, $\Delta(\omega)$ is identical to the Mahalanobis distance,

$$\Delta = \sqrt{m^2 \Sigma^{-1} m},$$

(22)

see also [Mahalanobis, 1936, Maesschalck et al., 2000]. The Mahalanobis distance is a measure of divergence between two populations, that is, a dissimilarity measure that more or less relates to the Bayes error.

For two concentric populations ($\mu_1 = \mu_2$) [Lindgren and Spångs, 2002] define the entity

$$\xi(s) = \frac{s^T \Sigma_1 s}{s^T \Sigma_2 s}$$

(23)

and show that if $\xi(s) > 1$ for all non-zero $s$, then

$$s^* = \arg \max_s \xi(s)$$

(24)

is a Bayes error optimal linear transformation for the case $k = 1$. $s^*$ is calculated by solving a generalized eigenvalue problem.

### 1.4.2 Divergence Measures for Heteroscedastic Populations

Using $\Delta(\omega_0)$ as a measure of dissimilarity or divergence of two heteroscedastic populations has been considered by at least [Chadda and Marcus, 1968], [Chernoff, 1952, Marks and Dunn, 1974, McLachlan, 1975]. Divergence measures are interesting not in the sense that they always give the optimal transformation down to $k$ dimensions directly, but as criteria in numerical optimization. In this report we are mainly interested in what is often called allocatory measures, by which we understand measures that approximate the Bayes error $V$.

In our setting, $V$ defined in (4) is considered the optimal measure. Certainly there are cases where this may not be true, but they are not covered here. To our knowledge, the integral (4) or (7) can however only be computed analytically in the special case mentioned earlier: two homoscedastic populations. An expression equivalent to (4) when $g = 2$ is

$$V = \frac{1}{2} \int \inf \{f_1(y), f_2(y)\} dy,$$

(25)

which in the homoscedastic normal distribution case is identical to (11), which in turn is the Mahalanobis distance (22) inserted into (21).

[Bhattacharyya, 1943] describes the angle of divergence $\xi$ between two heteroscedastic populations as

$$\cos \xi = \frac{1}{2} \int \sqrt{f_1(y)f_2(y)} dy.$$

(26)

Clearly, the geometric mean $\sqrt{f_1(y)f_2(y)}$ is not less than $\inf \{f_1(y), f_2(y)\}$ in (25), why

$$\frac{1}{2} \cos \xi \geq \frac{1}{2} \int \inf \{f_1(y), f_2(y)\} dy = V.$$

(27)
For two normal populations this upper bound reads

\[ V \leq \frac{1}{2} \cos \xi = \frac{1}{2} e^{-b} \]  

(28)

where \( b \) is the Bhattacharyya distance,

\[ b = \frac{1}{8} m^T \left[ \frac{1}{2} (\Sigma_1 + \Sigma_2) \right]^{-1} m + \frac{1}{2} \log \frac{\det \left[ \frac{1}{2} (\Sigma_1 + \Sigma_2) \right]}{\sqrt{\det \Sigma_1 \det \Sigma_2}}. \]  

(29)

In fact

\[ \frac{1}{2} \int [f_1(y)]^\omega [f_2(y)]^{1-\omega} dy \geq \frac{1}{2} \int \inf \{f_1(y), f_2(y)\} dy = V \]  

(30)

for \( 0 < \omega < 1 \), see [Chernoff, 1952]. Based on this observation, [Chernoff, 1973] proposes the measure of divergence we today call the Chernoff distance,

\[ \tau = \sup_{0 \leq \omega \leq 1} \frac{1}{2} \left\{ \omega (1 - \omega) m^T \Sigma^{-1} m + \log \frac{\det \Sigma}{\det \Sigma_1 \det \Sigma_2} \right\}, \]  

(31)

where \( \Sigma = \omega \Sigma_1 + (1 - \omega) \Sigma_2 \). For two normal populations, \( \tau \) obviously induces a tighter bound on the error rate than the Bhattacharyya distance (Bhattacharyya is the special case \( \omega = \frac{1}{2} \)),

\[ \frac{1}{2} e^{-b} \geq \frac{1}{2} e^{-\tau} \geq V. \]  

(32)

The calculation of the Chernoff distance however involves numerical maximization, for instance by the Newton-Raphson method\(^2\). See also [Fukunaga, 1990, pp. 99] or [Theodoridis and Koutroumbas, 1999, pp. 153].

[Lee and Choi, 2000] make extensive simulations to compare the error rate of a quadratic classifier to the Bhattacharyya distance \( b \). By curve fitting they find that the polynomial

\[ \hat{V} = 40.219 - 70.019b + 63.578b^2 - 32.766b^3 + 8.717b^4 - 0.91875b^5 \]  

(33)

is a good error estimator (for \( b < 3 \)).

**Information Theory**

Apart from divergence measures motivated by the Bayes error, there are measures based on information theory. The concept of *amount of information supplied by an observation about an unknown parameter* was known already to [Fisher, 1925]. For the two hypotheses \( H_p, p = 1, 2 \), that \( x \) is from population \( p \), we may define

\[ I(2 : 1) = \int f_2(x) \log \frac{f_1(x)}{f_2(x)} dx \]  

(34)

as the mean information per observation from population 2 for discrimination in favor of \( H_1 \) against \( H_2 \), see [Kullback, 1959, pp. 6]. From this, the divergence

\[ \text{[Chernoff, 1973]} \] gives expressions for both the first and second order derivatives needed for Newton-Raphson method.
measure sometimes referred to as the *Kullback-Leibler information number* is defined as

\[ I(1:2) + I(2:1) \]  

which in the normal distributed case reads

\[ \frac{1}{2} \text{tr} \left[ (\Sigma_1 - \Sigma_2)(\Sigma_1^{-1} - \Sigma_2^{-1}) \right] + \frac{1}{2} \text{tr} \left[ (\Sigma_1^{-1} + \Sigma_2^{-1})mm^T \right] \]  

or alternatively

\[ \frac{1}{2} \text{tr} \left[ (\Sigma_1^{-1} + \Sigma_2^{-1})(\Sigma_1 + \Sigma_2 + mm^T) \right] - 2n, \]  

see [Kullback, 1959].

### 1.4.3 Many Heteroscedastic Populations

For the general case with an arbitrary number of heteroscedastic populations and \( k \geq 1 \), [Decell et al., 1981] show that the rank \( k \) of the matrix

\[ M = \begin{bmatrix} \mu_2 - \mu_1 & \mu_3 - \mu_1 & \ldots & \mu_g - \mu_1 \\ \Sigma_2 - \Sigma_1 & \Sigma_3 - \Sigma_1 & \ldots & \Sigma_g - \Sigma_1 \end{bmatrix} \]  

(with \( n \) rows and \( 2(g - 1) \) columns) is equivalent to the minimum value of \( k \) for which an \( n \times k \) linear transformation preserves the Bayes error. In other words, it is possible to reduce the dimensionality to \( k \) (but no less) without any increase in the Bayes error. Such a minimal transformation can, for instance, be calculated by the compact singular value decomposition

\[ M = UT S^T, \]  

where \( U \) and \( S \) are unitary. By *compact* is understood the factorization does not account for the null space of \( M \); \( T \) is square and diagonal. As before, \( S \) is the transformation matrix.

[Young et al., 1987] use the results of [Decell et al., 1981] and propose an approximative solution. To form the linear transformation down to \( k < \hat{k} \) dimensions, they take \( S \) to be a \( k \)-rank approximation of \( M \). This approximation is defined by the the \( k \) dominant singular vectors, which are known to be Frobenius-norm optimal. See also [Odell, 1979, Tubbs et al., 1982].

[Brunzell and Eriksson, 2000] propose

\[ \prod_{1 \leq i < j \leq g} (\mu_i - \mu_j)^T(\Sigma_i + \Sigma_j)^{-1}(\mu_i - \mu_j) \]  

as a *measure of separation* and show that a transformation that spans the column space of the matrix

\[ [(\Sigma_i + \Sigma_j)^{-1}(\mu_i - \mu_j)]_{\text{col} 1 \leq i < j \leq g} \]  

does not decrease this measure. A linear transformation down to \( k \) dimensions is proposed as a rank \( k \) approximation of (41) defined by the \( k \) dominant singular vectors. In [Spangeus, 2001, paper 1] the optimal linear transformation with respect to (40) is sought by numerical optimization.
1.5 Numerical Optimization

By numerical optimization, local minimizers of the Bayes error $V(S)$ defined in (4) can be obtained for some special cases. [Guseman and Peters, 1975] use this idea for the case $k = 1$, $g = 2$, for which the error rate $V(s)$ can be calculated analytically. They derive the Gateaux differential

$$\partial V(s; p) = \lim_{h \to 0} \frac{V(s) - V(s + hp)}{h}$$

(42)

and use $n$ such differentials to compose the subspace gradient at $s$, $\nabla V(s)$. By subspace gradient we refer to the matrix with partial derivatives at $s$,

$$\nabla V(S) \triangleq \left[ \frac{\partial V(S)}{\partial (e_i^T S e_j)} \right]_{\text{row } i=1,2\ldots n}^{\text{col } j=1,2\ldots k}. $$

(43)

The minimization program employs a so called Davidon-Fletcher-Powell procedure.

Since the Bayes error is intractable to calculate except for special cases, it is however common to search local minima of approximations. Many authors generalize the error rate for $g = 2$ populations $i$ and $j$ to more by the union $e V(S) = \bigcup_{1 \leq i < j \leq g} e V_{ij}(S)$:

$$e(V_{ij} = V_{ij}),$$

(44)

If $V_{ij} = V_{ij}$, this union induces an upper bound of the Bayes error $V$. For instance, [Decell and Marani, 1976] employ an iterative search for the minimum of (44), where $V_{ij}(S)$ is based on the Bhattacharyya distance. At iteration $i$, the transformation $S$ is parameterized by $v$ in $\mathbb{R}^n$ as

$$S = H(v_1)H(v_2)\cdots H(v_{i-1})H(v) \begin{bmatrix} I_k \\ 0 \end{bmatrix}. $$

(45)

$H(v)$ is the Householder reflection,

$$H(v) = I - 2 \frac{v v^T}{v^Tv},$$

(46)

and $H(v_1)\cdots H(v_{i-1})$ are reflections due to earlier iterates. The gradient of the Bhattacharyya distance with respect to the parameter $v$ is derived and used in a steepest descent procedure to search for the optimum $v_i = v^*$ at each iterate. [Decell and Mayekar, 1977] use the same optimization technique, but instead they maximize the union of the pair-wise Kullback-Leibler information numbers (36). [Bidasaria, 1987] use the same information numbers (37) but for $g = 2$. He employs a steepest descent minimization procedure to remove un-interesting linear combinations. [Peters, 1979] proposes a mean-squared error criterion. [Aladjem, 1998] maximizes the (non-parametric) Patrick-Fisher distance, for $g = 2$ by using the gradient function. The Patrick-Fisher distance induce an upper bound on the Bayes error.

[Lotlikar and Kothari, 2000] consider the homoscedastic case with $g \geq 2$. They use the Bayes error union (44) for $V_{ij} = V_{ij}$. They also derive the gradient function $\nabla V(S)$ under the assumption that the linear transformation is
orthonormal, $S^TS = I_k$. A large number of small steps with fixed length are taken in the negative gradient (steepest descent) direction. Every step is completed with orthonormalization (Gram-Schmidt) to retain $S^TS = I_k$ over the iterations. [Lotlikar and Kothari, 2000] use the same technique for other distributions by modeling them as a superposition of normal distributions (kernel estimator with Gaussian kernel).

[Choi and Lee, 2003] also use a union like (44) to generalize the two-population case, but their model is heteroscedastic. They approximate $V_{ij}$ by using the Bhattacharyya distance. The optimization technique is the same as the one used by [Lotlikar and Kothari, 2000], but the gradients are estimated by finite differences, cf. [Decell and Marani, 1976].

1.6 Related Methods

Principal Component Analysis

We would like to relate to one of the most common and fundamental method to extract a linear subspace, the principal component analysis (PCA). PCA fits a line or a plane to the main variation of a random variable $X$ (or to a set of points) so that the 2-norm of the residual is minimum,

$$S^* = \arg \min_{S \in \mathbb{V}_{n,k}} \mathbb{E} \left[ \| X - SS^T X \|_2^2 \right] \text{ for fixed positive } k < n \quad (47)$$

(the mean of $X$ is here assumed to be 0). It has been widely used for more than 100 years, see [Pearson, 1901, Hotellings, 1933]. PCA can of course be used to reduce the dimensionality in discriminant analysis. However, optimal in residual norm, the error rate convergence with respect to the number of principal components $k$ is slow ($k$ needs to be large). This is simply due to the fact that the relation between residual norm and discriminant error rate is very weak.

1.6.1 Partial Least Squares

Also a note on the partial least squares (PLS) algorithm which was originally formulated by [Wold, 1975]. PLS targets the magnitude of the prediction error $E$ in the inverse regression model (see [Eisenhart, 1939, Krutchkoff, 1967] etc.),

$$Y = BS^T X + \mathcal{E} \quad (48)$$

where $B$ in $\mathbb{R}^{g \times k}$ is the regression coefficient and $Y$ the explained variable (cf. reduced rank regression). PLS is widely used in chemometrics to process large data sets for which no prior knowledge of noise distribution is used. In short, PLS gives a linear subspace $S^TX$ (see [Ruscio, 2000]) that in a sense is highly covariant with $Y$. The main virtue of PLS is however not error rate convergence (see [Eldén, 2002]) but rather ability too coop with high-dimensional (and multicollinear) data sets. See also [Martens and Näs, 1989, Hastie et al., 2001].

[Sjöström et al., 1986] use PLS for discriminant analysis (PLS-DA) by defining the predicted variable ($Y$) to be a discrete population-indicator vector (dummy variable). The outcomes of $Y$ is $\{e_1, e_2, \ldots, e_g\}$ where

$$e_p \triangleq \begin{bmatrix} 0^{(1)} & 0^{(2)} & \ldots & 0^{(p-1)} & 1^{(p)} & 0^{(p+1)} & \ldots & 0^{(g)} \end{bmatrix}^T \quad (49)$$
for population \( p \). [Kemsley, 1996] shows interesting relations between PLS-DA and Fisher’s linear discriminants (13). He also compares PCA and PLS as a sample processing step prior to discriminant analysis, and concludes that PLS converges faster than PCA (fewer components needed). We acknowledge this way of using PCA and PLS in discriminant analysis, that is, as an initial dimensional reduction before model parameters are estimated from a finite sample set. See also [Stähle and Wold, 1987].

**Projection Pursuit**

Finally, a note on projection pursuit due to [Friedman and Tukey, 1974], an area where the aim is to find low-dimensional representations, or projections, of data. A projection is valued by a projection index, which can be seen as an objective function which optimum we pursue. Different indices have been proposed, for instance cluster revealing, entropy based and non-normality based, see [Jones and Sibson, 1987] for a review. The common denominator of projection pursuit and this work is of course the pursuit of projections.

[Friedman and Stuetzle, 1981] propose the projection pursuit regression model

\[
 f(x) = \sum_{m=1}^{M} g_m(s_m^T x) 
\]

(50)

for some set of scalar-valued smoothing functions \( g_m \) (cf. the one hidden layer neural network) and a set of projections \( s_m \). Like the PLS-DA, this is an inverse regression model (albeit nonlinear) since it estimates a predictor \( f(x) \) rather than a statistical model of the observations, that is, some \( y \) in \( \mathbb{R} \) is explained by

\[
 y = f(x) + e. 
\]

(51)

See also [Friedman, 1987, Huber, 1985, Glover and Hopke, 1992]. A recommended text is [Hastie et al., 2001, ch. 11].


**1.7 This Work**

In this work we focus on the problem

\[
 \min_{S \in V_{n,k}} V_u(S), \quad \text{for fixed positive } k < n. \tag{52}
\]

The demand that \( S \) should be a member of the set \( V_{n,k} \), which means that \( S \) is an \( n \)-by-\( k \) orthonormal matrix (56), is not obvious, but a numerically sound guarantee that \( S \) has rank \( k \). This constraint does not restrict the set of subspaces, it just demands that every subspace is represented by a normalized coordinate basis.

The objective function \( V_u(S) \) will be defined thoroughly in Section 3, but basically it is the union of pair-wise errors,

\[
 V_u(S) = c \sum_{1 \leq i < j \leq g} p_{ij}(S), \tag{53}
\]
where $c$ is a constant independent of $S$. In the homoscedastic case we let $p_{ij}(S) = V_{ij}(S)$, that is, the pair-wise Bayes error when discriminating between population $i$ and $j$. In the heteroscedastic case, $p_{ij}(S)$ will be an approximation thereof based on either the Mahalanobis or the Bhattacharyya distance.

In Section 2, parameterization and gradient methods to approach (52) are described. In Section 3, the objective functions we use are described and their subspace gradients derived. The objective functions are based on the Mahalanobis and Bhattacharyya distances. Finally, in Section 4 the methods are evaluated on numerical data with respect to convergence rate and sample error rate.

1.7.1 Contributions

1. To our knowledge, none of the four optimization techniques described in the report have been used earlier to reduce the dimensionality of normal populative models.

2. The derivations of the subspace gradients of the Mahalanobis and Bhattacharyya distances are original. The results are however already known, see [Decell and Quirein, 1973].

3. The exponential error estimator (138) based on the Bhattacharyya distance is original.
2 Subspace Optimization Techniques

The $k$-dimensional space spanned by the columns of $S$, where $S$ is an $n$-by-$k$ transformation matrix with full column rank, we denote subspace. The matrix representation of a subspace is not unique. For instance, $S$ and $SQ$, where $Q$ is a $k$-by-$k$ non-singular matrix, represent the same subspace, although the coordinate bases for $S^T\mathbf{x}$ and $Q^T S^T \mathbf{x}$ are different. The pursuit of a linear transformation $S$ that minimizes some homogeneous objective function $F(S) = F(SQ)$ we denote subspace optimization (optimization over the Grassman manifold). By subspace gradient we refer to the $n$-by-$k$ matrix with partial derivatives at $S$,

$$\nabla F(S) \triangleq \left[ \frac{\partial F(S)}{\partial (e_i^T S e_j)} \right]_{\text{row } i=1,2,..n}^{\text{col } j=1,2,..k} .$$

The rank constraint on $S$ together with the fact that the objective function is homogeneous is characteristic for the subspace optimization problem. Four techniques that for better or worse deal with these impediments will be investigated:

1. Householder reflection parameterization
2. Givens rotation parameterization
3. steepest descent, and
4. nonlinear conjugate gradients.

The Householder reflection and Givens rotation see [Golub and Loan, 1989], are full parameterizations in the sense that they map a parameter vector directly to an $n$-by-$k$ linear transformation matrix $S$ with column rank $k$ (in fact, $S$ is orthonormal). They are used together with standard programs for unconstrained minimization that do not require any analytical subspace gradient. Those minimization programs may use finite numerical approximations of the gradient (and possibly higher order derivatives) in the parameter space, but the subspace gradient is not used.

The steepest descent and conjugate gradients are line search methods that use line parameterizations that enforce the rank of the linear transformation (in fact, they enforce orthonormality). By line parameterization we understand a function $S = S(t)$ that allow us to perturb $S$ by changing a scalar parameter $t$. An example is the matrix flow $S(t) = S_0 + tG$ for an $n$-by-$k$ matrix $G$. For this simple parameterization we can however not guarantee that rank $S(t) = k$. The minimization procedure is iterative; at each iteration we stand at $S_i$ and determine a promising search direction. Minimization along this search direction (by the line parameterization) gives a new standpoint $S_{i+1}$, and so on. Line search methods can of course also be used in the parameter space of Householder reflections and Givens rotations, but in those spaces the minimization is unconstrained, a difference worth noting.

2.1 Notes on Differential Geometry

The constraint surface imposed by $S^T S = I$, $S$ in $\mathbb{R}^{n \times k}$, is termed the Stiefel manifold. Differentiating $S^T S = I$ gives $S^T \Delta + \Delta^T S = 0$, where $\Delta$ is tangent
vectors, that is, $S^T \Delta$ is skew-symmetric. This condition imposes $k(k+1)/2$ constraints on $\Delta$, which means the space of all tangent vectors has the dimensionality

$$nk - \frac{1}{2}k(k+1) = \frac{1}{2}k(2n - k - 1),$$

see [Edelman et al., 1998]. One approach to optimize over the Stiefel manifold is to parameterize $S$ by $S = S(p)$ such that $S(p)^T S(P) = I$ for every parameter vector $p$ (for instance parameterization by Householder reflections or Givens rotations). From the above follows, however, that if the dimensionality of $p$ exceeds $\frac{1}{2}k(2n - k - 1)$, then the vector space of all tangent vectors $\Delta$ is over parameterized (at the point $S$), and local solutions to the optimization problem are not unique in $p$, not even if the objective function is non-homogeneous. For the Stiefel manifold we use the notation

$$\mathcal{V}_{n,k} \triangleq \{ S \in \mathbb{R}^{n \times k} | S^T S = I \}. \quad (55)$$

The homogeneity (basis invariance) of $F(\cdot)$ means $F(SQ) = F(S)$ for every non-singular matrix $Q$ in $\mathbb{R}^{k \times k}$. An obvious constraint here is that $S$ should have full column rank. If also $S^T S$ is well-conditioned, then the perturbation $SQ = S + hST$ has full column rank as long as the magnitude of $h$ is sufficiently small. $T$ is a $k$-by-$k$ matrix that can be taken arbitrarily. From this follows that

$$\lim_{h \to 0} \frac{F(S + hST) - F(S)}{h} = 0, \quad (57)$$

which in turn implies that derivatives in directions taken from the range space of $S$ (that is, $ST$) are zero, or

$$S^T \nabla F(S) = 0. \quad (58)$$

The surface imposed by homogeneity as well as orthonormality constraints is termed the Grassmann manifold. The points on the Grassmann manifold form matrix equivalence classes, which means there is no unique matrix representation.

We also define the orthonormal set as

$$\mathcal{O}_n \triangleq \{ X \in \mathbb{R}^{n \times n} | X^T X = I \}. \quad (59)$$

$\mathcal{O}_n$ is obviously closed under multiplication, that is, if $A \in \mathcal{O}_n$ and $B \in \mathcal{O}_n$ then $A^T B \in \mathcal{O}_n$.

2.2 The Householder Reflection

The Householder reflection (or Householder transformation) is a well-known instrument used for numerically stable QR-factorizations, the standard text here is [Golub and Loan, 1989]. As mentioned earlier, [Decell and Marani, 1976] and [Decell and Mayekar, 1977] use a series of Householder reflections to search for optimal subspaces in discriminant analysis. They did however not use a full parameterization, but rather a series of local ones. Full parameterization is used by [Andersson, 1992] for subspace optimization in the processing of multiple antenna signals. Householder reflections have also been used for the tracking
of principal and minor subspaces of vector sequences in a modification of the Oja’s subspace algorithm, see [Oja, 1992, Abed-Meraim et al., 2000].

With a full parameterization like the Householder reflection, we can address optimization problems like

$$\min_{S \in V_{n,k}} F(S)$$

(60)

with standard programs for unconstrained optimizations by letting $S = S(p)$ such that $S(p)$ lies in $V_{n,k}$ for every parameter vector $p$ given by the program. Although the Householder reflection is well-used in numerical linear algebra, the technique with full parameterization has, to our knowledge, only been used earlier by [Andersson, 1992]. The very brief introduction below, which put the Householder reflection into our framework and high-light properties vital to subspace extraction, is therefore motivated.

**Definition 1 (Householder Reflection)**
The Householder reflection $H$ we define for all (real) vectors $v$ as

$$H(v) \triangleq \begin{cases} I - \frac{2vv^T}{v^Tv} & \text{if } v \neq 0 \\ I & \text{else} \end{cases}.$$  

(61)

$H(v)$ is a reflection in the plane defined by the normal $v$, it is easily verified that $H(v)v = -v$. That $H(v)$ is in $O_n$ ($H(v)^T H(v) = I_n$) for all $v$ in $\mathbb{R}^n$ is understood by

$$H^T(v)H(v) = (I - \frac{2vv^T}{v^Tv})^T (I - \frac{2vv^T}{v^Tv}) = I + 4 \frac{vv^T}{v^Tv} - 4 \frac{v(v^Tv)v^T}{(v^Tv)(v^Tv)} = I.$$  

(62)

Another property of the Householder reflection we will use is that for two vectors $a$ and $b$ with the same length ($a^Ta = b^Tb$) holds

$$H(a - b)b = a.$$  

(63)

When $a - b = 0$ this is given immediately by the definition above, else

$$H(a - b)b = b - 2 \frac{(a - b)(a - b)^Tb}{(a - b)^T(a - b)} = b - 2 \frac{(a - b)(a - b)^Tb}{2(b^Tb - a^Tb)}$$

$$= b + \frac{(a - b)(b - a)^Tb}{(b - a)^Tb} = a.$$  

(64)

Especially,

$$H(e_1 - P_1)P_1 = e_1 \quad \text{where} \quad e_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T$$  

(65)

for any unit length vector $P_1$. Now, let $P$ be any matrix in $V_{n,k}$ with the first column denoted $P_1$. From the above and the fact that the set $O_n$ is closed under multiplication follows that $H(e_1 - P_1)P$ has the structure

$$H(e_1 - P_1)P = \begin{bmatrix} 1 & 0 \\ 0 & P \end{bmatrix}.$$  

(66)
Definition 2 (Composite Reflection)

Define the $k$-fold composite reflection in $n$ dimensions, $0 < k < n$, recursively as

$$R_n(p) \triangleq \begin{cases} \begin{bmatrix} 1 & 0 \\ 0 & R_{n-1}(\tilde{p}) \end{bmatrix} H(v) & \text{if } \dim p > n \\ H(p) & \text{if } \dim p = n \end{cases}$$

(67)

for (real) parameter vectors $p$ composed as

$$p = \begin{bmatrix} v \\ \tilde{p} \end{bmatrix},$$

(68)

where $\dim v = n$ and $\dim p = n + (n - 1) + \cdots + (n - k + 1) = k(2n - k + 1)/2$.

Theorem 1 (Composite Reflection)

For every $P$ in $\mathcal{V}_{n,k}$, $0 < k < n$, there exists a parameter $p$ such that the $k$ first columns of the $k$-fold $R_n^T(p)$ equals the columns of $P$. Furthermore, $R_n(p)$ is in $\mathcal{O}_n$ for every $p$ in the definition set above.

Proof: Induction over $k$. For $k = 1$ we have that $\dim p = n$ why $R_n(p) = H(p)$ and the basis step follows from (62) and (66).

If the theorem is true for at least $k = K$ ($\forall n > K$), then

$$\begin{bmatrix} 1 & 0 \\ 0 & R_n(\tilde{p}) \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & R_n(\tilde{p}) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & R_n(\tilde{p})^T R_n(\tilde{p}) \end{bmatrix} = I$$

(69)

for the $K$-fold composite reflection $R_n(\tilde{p})$. The $K + 1$-fold composite reflection $R_{n+1}(p)$ is thus in $\mathcal{O}_n$, since $H(p)$ is in $\mathcal{O}_n$ and this set is closed under multiplication. By induction it is concluded that $R_n(p) \in \mathcal{O}_n$ for every positive $k$.

Again, assume the theorem is true for at least $k = K$ ($\forall n > K$). Then for every $\tilde{P}$ in $\mathcal{V}_{n,k}$ it exists a parameter vector $\tilde{p}$ such that

$$R_n(\tilde{p}) \tilde{P} = \begin{bmatrix} I_K \\ 0 \end{bmatrix},$$

(70)

for the $K$-fold composite reflection $R_n(\tilde{p})$. Since both $R_n(\tilde{p})$ and $\tilde{P}$ are orthonormal, (70) implies that the $K$ first columns of $R_n^T(\tilde{p})$ are equal to the columns of $\tilde{P}$.

Now, let $P_1$ be the first column of an arbitrary member $P$ of $\mathcal{V}_{n+1,k+1}$. If $v = e_1 - P_1$, then according to (64)

$$H(v)P = \begin{bmatrix} 1 & 0 \\ 0 & \tilde{P} \end{bmatrix}$$

(71)

where $\tilde{P}$ is obviously in $\mathcal{V}_{n,k}$. By the assumption, there is thus a $\tilde{p}$ such that

$$R_{n+1}(p) = \begin{bmatrix} 1 & 0 \\ 0 & R_n(\tilde{p}) \end{bmatrix} H(v)P = \begin{bmatrix} 1 & 0 \\ 0 & R_n(\tilde{p}) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \tilde{P} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 \\ 0 & R_n(\tilde{p}) \tilde{P} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & I_K \end{bmatrix},$$

(72)
which is to say the $K + 1$ first columns of $R_{n+1}^T(p)$ for $p = [v^T \; \vec{p}^T]^T$ are equal to the columns of $P$ since both $R_{n+1}(p)$ and $P$ are orthonormal. Thus, under the assumption that the theorem is valid with the restriction $k = K$, we have showed it valid also for $k = K + 1$, and it is thereby valid for every positive $k$.

2.2.1 Comments

The result above is not novel but serves two purposes. First it shows that any $k$-dimensional subspace in $n$ dimensions can be computed using the composite reflection, and the composite reflection subspace has always an orthonormal representation. The subspace is defined by the range space of the transformation matrix

$$S(p) = R_{n+1}^T(p) \begin{bmatrix} e_1 & e_2 & \cdots & e_k \end{bmatrix}. \tag{73}$$

Second, it gives an algorithm to identify a parameter vector $p$ that yields a composite reflection equal to a particular subspace (with orthonormal representation).

It is a fact that the matrix representation of a subspace is not unique, that is, for every subspace there is a whole set of matrices with coincident column space. The parameter vector of the composite reflection does in turn not define an orthonormal matrix uniquely. This is seen immediately in (61); the length of a non-zero $v$ is inconsequential to $H(v)$. Also, the dimensionality $k(2n-k+1)/2$ of the parameter vector $p$ exceeds the dimensionality of the vector space of all tangent vectors (55). This is of course a disadvantage in numerical optimization, since it implies a singular Hessian and potentially bad convergence. The ambiguity in $p$ could be overcome by letting, for instance, the first entry in every reflection normal vector $v$ be identical to 1. This means however that certain subspaces, for instance the subspace given by $H([0 \; 1]^T)$, can be represented only up to a certain degree of accuracy. We have not seen any improvement by this modification.

Another drawback of the Householder reflection as we have defined it here, is the discontinuity around $v = 0$. This discontinuity is seen clearly in the scalar case, when $H(v) = -1$ for all non-zero $v$, but $H(0) = 1$ for $v = 0$. We have not experienced any problems in our numerical evaluations that we can derive to this fact. Yet we recognize it as an important issue that deserves a treatment however not given in this report.

The advantage of the composite reflection is mainly its computational simplicity.

2.3 Givens Rotation

Like the Householder reflection, the Givens rotation, defined as a plane rotation in $n$ dimensions, is used in QR factorization, see [Golub and Loan, 1989]. As we know it, it has not been used very extensively for subspace optimization. One application, though, is the Optimal Discriminative Projection, which targets the classification accuracy in high-dimensional (multicollinear) data sets, see [Spangeus, 2001, paper 1]. A related method that should be mentioned was reported by [McCulloch, 1986], who parameterized a linear combination of two of Fisher’s linear discriminants by an angle.
We define below the vector rotation that plays the same role as the Householder reflection, and the composite rotation that is analogous to the composite reflection, although rotations are used instead of reflections.

**Definition 3 (Vector Rotation)**

Define an \( n \)-dimensional vector rotation \( n \geq 2 \) recursively as

\[
    r_n(v) \triangleq \begin{cases}
        r_{n-1}(\tilde{v}) & (I_{n-2} 0) \begin{bmatrix} 0 & 0 \\ 0 & r_2(v_1) \end{bmatrix} \quad n > 2 \\
        \begin{bmatrix} \cos v_1 & -\sin v_1 \\ \sin v_1 & \cos v_1 \end{bmatrix} & n = 2
    \end{cases}
\]  

(74)

where \( v = [v_1 \ v_T] \) in \( \mathbb{R}^{n-1} \) and \( v_1 \) in \( \mathbb{R} \), are angles with entries typically in the interval \([0, 2\pi]\).

**Lemma 1 (Vector Rotation)**

For every \( a \) in \( \mathbb{R}^n \), \( n \geq 2 \), it exists an angle \( v \) in \( \mathbb{R}^{n-1} \) such that

\[
    r_n(v)a = e_1 \cdot \|a\|.  \tag{75}
\]

Furthermore, \( r_n(v) \) is in \( O_n \) for every angle \( v \).

**Proof**

It is easily verified that \( r_2^T(v)r_2(v) = I \). why \( r_2(v) \in O_n \). Since \( O_n \) is closed under multiplication it follows that \( r_n(p) \in O_n \).

The rest of the proof is algorithmic. Given any vector \( a \), an explicit procedure to determine \( v \) such that (75) holds is described. Let \( a_i \) and \( v_j \) denote the \( i \)th and \( j \)th entry of \( a \) and \( v \), respectively. Define for notational purposes the two-dimensional vector

\[
    \tilde{a}_i = \begin{bmatrix} a_{n-i} \\ a_n \end{bmatrix} \quad (76)
\]

Now take \( v_i \) as the solution to

\[
    r_2(a)\tilde{a}_i = e_1 \cdot \|\tilde{a}_i\|.  \tag{77}
\]

If this solution always exists, we see that

\[
    \begin{bmatrix} I & 0 \\ 0 & r_2(v_1) \end{bmatrix} a = \begin{bmatrix} a_1 \\ \vdots \\ a_{n-2} \\ \tilde{a}_1 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} I & 0 \\ 0 & r_3(v_1) r_2(v_2) \end{bmatrix} a = \begin{bmatrix} a_1 \\ \vdots \\ a_{n-3} \\ \tilde{a}_2 \\ 0 \\ 0 \end{bmatrix},  \tag{78}
\]

and so on until we end up \( n - 1 \) zeros and \( r_n(v)a = e_1 \cdot \|a\| \).

Essentially the proof lies in showing that we can always solve (77), which is actually the simple trigonometric problem of finding the angle of a triangle.
1. If $\alpha_i = 0$ then every $v_i$ is a solution,
2. else if $\alpha_{i1} = 0$ (the first entry of $\alpha_i$) then $v_i = \pi$,
3. else

$$v_i = -\arctan \frac{\alpha_{i2}}{\alpha_{i1}} - \frac{\pi}{2} (\text{sgn} \alpha_{i1} - 1).$$  \hspace{1cm} (79)

where the second term uses the signum operator to subtract $\pi$ in the second and third quadrant.

That this problem is solvable (although not uniquely) is considered a fact. \hfill \Box

**Definition 4 (Composite Rotation)**

Define the $k$-fold composite rotation in $n$ dimensions, $0 < k < n$, recursively as

$$G_n(p) = \begin{cases} 1 & \text{dim } p > n \\
0 & \text{dim } p = n \\
r(p) & \text{dim } p = n \end{cases}$$ \hspace{1cm} (80)

for real parameter vectors $p$ composed as

$$p = \begin{bmatrix} v \\ \bar{p} \end{bmatrix},$$ \hspace{1cm} (81)

where $\text{dim } v = n-1$ and $\text{dim } p = (n-1) + (n-2) + \cdots + (n-k) = k(2n-k-1)/2$.

**Theorem 2 (Composite Rotation)**

For every $P$ in $\mathbb{V}_{n,k}$, $0 < k < n$, it exists a parameter $p$ such that the $k$ first columns of $G^T(p)$ equals the columns of $P$. Furthermore, $G(p)$ is in $\mathbb{O}_n$ for every $p$ in the definition set.

**Proof** Analogous to Proof 2.2 with the difference that we use the vector rotation instead of the Householder reflection. Since the first column of the orthonormal matrix $P$ denoted $P_1$ is unit length, it always exist a $v$ such that

$$r(v)P = \begin{bmatrix} 1 & 0 \\ 0 & \bar{P} \end{bmatrix}$$ \hspace{1cm} (82)

which follows from Lemma 1 and the orthonormal property of the matrices. \hfill \Box

**2.3.1 Comments**

As mentioned, the Givens rotation is a well-known technique in numerical linear algebra. However, Lemma 1 shows that we can use the vector rotation in the same manner as the Householder reflection to extract every $k$-dimensional subspace. Lemma 1 together with Theorem 1 also give means to calculate the parameter vector $p$ corresponding to any member of $\mathbb{V}_{n,k}$.

Like the composite reflection, there are infinitely many parameter vectors that give exactly the same $k$ first columns of the composite rotation. Even
if the angles are constrained to \([0, 2\pi]\) the angle representation is not always unique. This stands clear by the consideration of \(\tilde{v}_i = 0\) in (77). For instance,

\[
G \left( \begin{bmatrix} 0 \\ \lambda \end{bmatrix} \right) = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}
\]  

(83)

for all \(\lambda\). However, the composite rotation is a smooth function in the sense that all components are smooth sine and cosine functions. The inverse mapping is however discontinuous, the limit

\[
\lim_{\lambda \to 0} G^{-1} \left( \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right),
\]

(84)

for instance, does not exist.

The advantage of the composite rotation parameterization compared to the composite reflection is the smoothness and the smaller dimensionality of the parameter vector (\(k\) fewer dimensions). The composite rotation is however more complex to compute.

### 2.4 Steepest Descent

A steepest descent-type method is iterative; at iteration \(i\) we stand at the point \(S_i\) and calculate the \(n\)-by-\(k\) gradient matrix \(G_i = \nabla F(S_i)\) (54) and then move a positive step length \(\alpha_i\) (scalar) in the negative gradient direction (the direction of steepest descent) to produce an updated point \(S_{i+1}\). By choosing the step length \(\alpha_i\) with (at least) the demand that

\[
F(S_{i+1}) \leq F(S_i)
\]

(85)

one expect that a number of iterations will move an initial point \(S_0\) close to a local minimizer of \(F\). We assume in this section that the gradient \(G_i\) is available by analytical calculation or some finite-difference approximation, see [Nocedal and Wright, 1999, ch. 7].

An example of steepest descent-type algorithm targeting the classification accuracy in a \(k\)-dimensional linear subspace is found in [Lotlikar and Kothari, 2000], where the point \(S_i\) is basically updated as

\[
\tilde{S}_{i+1} = S_i - \alpha G_i
\]

(86)

for fixed \(\alpha\). An assumption made in [Lotlikar and Kothari, 2000] strictly constrains \(S_i\) to \(V_{n,k}\), why the update is completed by orthonormalization,

\[
S_{i+1} = \text{orth}[\tilde{S}_{i+1}],
\]

(87)

where \(\text{orth}[\cdot]\) is an orthonormalization operator (Gram-Schmidt, for instance). However, in [Lotlikar and Kothari, 2000] no discussion on how to calculate the step length \(\alpha\) is given. (There are more examples of additive update followed

---

We have tried a simple adaptive step length that behaves acceptable: Starting off with a very large \(\alpha_0\), we keep halving the step length at each iteration as long as (85) is not fulfilled. Although fast convergence is attained from time to time, the performance is very varying between problem instances.
by orthogonalization, for instance Oja’s subspace algorithm for principal component extraction and derivatives thereof, see [Oja, 1992].

Below we propose a simple line search method wherein each iteration the optimal step length is sought. We parameterize the line so that an orthonormal linear transformation is guaranteed ($S(\alpha) \in \mathcal{V}_{n,k}$ for all scalar $\alpha$). Unconstrained line search methods are described in [Nocedal and Wright, 1999].

2.4.1 ON-constrained Line Search

We now formulate our problem of finding the optimal linear subspace as minimization over the Grassmann manifold:

$$\min_{S \in \mathcal{V}_{n,k}} F(S)$$

(88)

where $F$ is homogeneous, that is, $F(S) = F(SQ)$ for every square rank $k$ matrix $Q$. The homogeneity follows from the fact that we target the information content in a subspace, not the particular representation (basis) of it, see Section 2.1. The constraint $S \in \mathcal{V}_{n,k}$ guarantees the $k$-rank of $S$, but also simplifies the calculation of many objective functions and their corresponding gradients, see [Lotlikar and Kothari, 2000]. We insist that for a new search direction $p_i \in \mathbb{R}^{n \times k}$ at $S_i$ should hold that

1. $p_i^T S_i = 0$ and
2. $p_i \in \mathcal{V}_{n,k}$.

The first condition means that $p_i$ points at new subspaces linearly independent of $S_i$, and holds naturally for the gradient $\nabla F(S)$ as a consequence of the homogeneity. The second condition demands that the representation of the search direction is normalized, we will return to this latter condition soon. If the conditions above hold, it is easily verified that for every point on the line

$$T_i(\alpha) = S_i \cos \alpha + p_i \sin \alpha$$

(89)

holds

$$T_i(\alpha)^T T_i(\alpha) = I,$$

(90)

as well as that

$$\frac{dT_i(\alpha)}{d\alpha} = p_i \quad \text{at} \quad \alpha = 0.$$  

(91)

At $S_i$ the unconstrained local minimizer of

$$\alpha_i = \arg \min_{\alpha} F(T_i(\alpha))$$

(92)

updates the linear transformation at each iteration as

$$S_{i+1} = T(\alpha_i),$$

(93)
where $S_{i+1}$ thus is orthonormal. The search direction $p_i$ is given at each point $S_i$ by the gradient $\nabla F(S_i)$. However the second condition above, $p_i \in \mathbb{V}_{n,k}$, forces us to normalize the gradient. This we do by a Gram-Schmidt algorithm\(^4\):

$$[\bar{S}_i \quad p_i] = \text{orth}[S_i \quad \nabla F(S_i)].$$

(94)

If no roundoff errors due to finite numerical representation is introduced, $\bar{S}_i$ and $S_i$ are identical. If we replace $S_i$ by $\bar{S}_i$ in (89), however, the normalization above guarantees that the orthonormality of $S_i$ is not degenerated over the iterations.

2.4.2 Line Derivative

The minimization of (92) can be facilitated by using the derivative, which reads

$$\frac{d}{d\alpha} T_i(\alpha) = \text{tr} \left[ \nabla F^T (-S_i \sin \alpha + p_i \cos \alpha) \right].$$

(95)

2.4.3 Comments

The line parameterization (89) is not the only possible one, as will come clear in the next section, but simple and intuitive.

It is seen in [Nocedal and Wright, 1999, pp. 48] that one can easily construct toy minimization problems where it is evident that steepest descent-type algorithms will need a large number of iterations to reach a minimizer, even if the objective function is a simple convex quadratic (which is not our case). The numerical evaluations later on will give a hint to if the simplicity of the algorithm may compensate for poor convergence with respect to iteration count.

In practice, we will have to settle with a numerical approximation to the minimizer (92). An issue is how to relate the accuracy of the approximation to the convergence rate of the method. Can we apply the Wolfe conditions (103,104) well-known in numerical optimization here? Convergence results for steepest descent algorithms fulfilling the Wolfe conditions are found in [Nocedal and Wright, 1999], but we consider it left to be investigated if we can apply those results to our particular problems.

2.5 Conjugate Gradients Method

The conjugate gradients method is known to be a very efficient method for unconstrained minimization of convex quadratic functions and has been known since the 1950s, see [Hestenes and Stiefel, 1952, Luenberger, 1984]. It is a line search method, but with search directions that is likely to give better convergence rate compared to the directions of steepest descent.

Originally the algorithm was developed to solve systems of linear equations by minimizing the quadratic function whose normal equations coincided with the system of linear equations, the linear conjugate gradient method. Extensions to non-quadratic functions were however soon developed, see [Fletcher and Reeves, 1964, 1964].

\(^4\)However, if $[S_i \quad \nabla F(S_i)]$ lacks full column rank, this will evidently not work, and we cannot find a search direction that fulfills the two conditions stated above. This will happen exactly when the column rank of $\nabla F(S)$ is less than $k$. This is necessarily the case if $2k > n$, when we suggest minimizing over the orthogonal complement instead. Other cases are handled by reducing linearly dependent columns in a fairly obvious manner.
Polak and Ribière, 1969. Not too long ago, a nonlinear conjugate gradient algorithm was formulated for optimization over the Grassmann manifold and thus designed to solve

$$\min_{S \in V_{n,k}} F(S)$$

(96)

for homogeneous, nonlinear $F$, which from our viewpoint is a subspace optimization problem, see [Edelman et al., 1998]. The conjugate gradient method has however been used under many years for subspace extraction in the partial least squares algorithm designed to solve large linear regression problems, see [Wold, 1975]. Here the linear conjugate gradient method is used, see [Eldén, 2002] and [Hastie et al., 2001, pp. 68], but usually interrupted after $k \ll n$ iterations (latent variables), which is many iterations before convergence is guaranteed. The formulation (96) is however far more general than the problems solved by the partial least squares algorithm. The conjugate gradients method is also used for extracting principal components, see [Yang et al., 1989] for a survey.

The conjugate gradients over the Grassmann manifold algorithm given below is due to [Edelman et al., 1998] to which we refer for a complete background to the algorithm. It is assumed that the gradient $r^T F(S)$ is available and as we discussed earlier, we demand it to be orthogonal to $S$, $S^T r^T F(S) = 0$.

2.5.1 The Conjugate Gradients over the Grassmann Manifold

- Given $S_0$ such that $S_0^T S_0 = I$, set $H_0 = -G_0$ where $G_0$ is the gradient computed at $S_0$.
- For $i = 1, 2, \ldots$,
  1. Minimize $F(S_i(\alpha))$ over $\alpha$ where
     $$S_i(\alpha) = S_i L \cos(Ta)L^T + U \sin(Ta)L^T$$
     (97)
     and $UTL^T$ the compact singular value decomposition of $H_i$.
  2. Set $\alpha_i = \alpha_{\text{min}}$ and $S_{i+1} = S_i(\alpha_i)$.
  3. Compute the gradient $G_{i+1}$ at $S_{i+1}$.
  4. Compute the parallel transport for $H_i$ and $G_i$ to the new point $S_{i+1}$ as
     $$\tau H_i = [-S_i L \sin \Sigma \alpha_i + U \cos \Sigma \alpha_i] U^T G_i$$
     (98)
     $$\tau G_i = G_i - [S_i L \sin \Sigma \alpha_i + U (I - \cos \Sigma \alpha)] U^T G_i$$
     (99)
  5. Update the search direction
     $$H_{i+1} = -G_{i+1} + \frac{\langle G_{i+1} - \tau G_i, G_{i+1} \rangle}{\langle G_i, G_i \rangle} \tau H_i$$
     (100)
     where the scalar product is defined as $\langle A, B \rangle \triangleq \text{tr} [A^T B]$.
  6. If $i + 1 = 0 \mod (n - k)$ then reset $H_{i+1} = -G_{i+1}$.
2.5.2 Line Derivative

The derivative of (97) reads
\[
\frac{d}{d\alpha} S_i(\alpha) = \text{tr} \left[ \nabla F^T \left( -S_i L \Sigma \sin(\Sigma \alpha) L^T + U \Sigma \cos(\Sigma \alpha) L^T \right) \right].
\] (101)

2.5.3 Comments

One critical point of the algorithm appears to be the line (geodesics) minimization in step 1. When minimizing a union bound measure of the classification error rate in a subspace, the step length \( \alpha \) diverged to infinity. Interestingly, this problem was solved completely by replacing \( F(S) \) by \( \log F(S) \). Anyhow, we have not seen any discussion on how to do this minimization and what termination criteria to use, see however next section.

2.6 Line Minimization

Line search algorithms like the steepest descent and conjugate gradients described above requires that we at each iteration \( i \) can address a scalar minimization problem, which we denote
\[
\alpha_i^* = \min_{\alpha \in \mathbb{R}} \phi_i(\alpha).
\] (102)

\( \alpha_i \) is the step length at iteration \( i \). For the non-convex optimization problems we are concerned with in this report, it is not reasonable to demand the exact global minimizer of \( \phi_i(\alpha) \). Instead we use an iterative program that terminates when two conditions hold, namely
\[
\phi(\alpha) \leq \phi(0) + c_1 \alpha \phi'(0)
\]
and
\[
|\phi'(\alpha)| \leq -c_2 \phi'(0)
\]
(103) demands that the objective function is sufficiently decreased with respect to the step length \( \alpha \) and (104) that \( \alpha \) is close to a local minimizer. A program for the Wolfe conditions is proposed in [Nocedal and Wright, 1999].

The Wolfe conditions primarily guide the selection of step length. For the non-linear conjugate gradients due to [Fletcher and Reeves, 1964], a bounded level set and Lipschitz continuous objective function yields the convergence result
\[
\lim_{i \to \infty} \inf \|\nabla f_i\| = 0
\]
if the step length are taken to fulfill the strong Wolfe conditions with \( 0 < c_1 < c_2 < \frac{1}{2} \), see [Nocedal and Wright, 1999] for details. We have not showed (and have no reason to believe or not), however, that these assumptions on the objective function hold in our cases.
3 Subspace Error Rate

This section describes three approximators of the error rate in a linear subspace. They are used as objective functions in numerical optimization, as have been explained in the introduction. The approximators are unions of pair-wise divergence measures that build on the Mahalanobis and Bhattacharyya distances. First we will describe the pair-wise divergence measures and their subspace gradients, and after that the union of them. Thus, bear in mind that all pair-wise measures of \( g > 2 \) populations will be aggregated at the end of the section.

3.1 Mahalanobis Distance, Homoscedastic Case

As mentioned in the introduction, the Mahalanobis distance is a measure of divergence between two homoscedastic populations, \( \delta \triangleq \sqrt{m^T \Sigma^{-1} m} \),

where \( m = \mu_1 - \mu_2 \) is the mean difference and \( \Sigma \) the (common) covariance matrix. The Mahalanobis distance is related to the Bayes error as

\[
V = \frac{1}{2} \tilde{Q} \left( \frac{\delta}{2\sqrt{2}} \right),
\]

where

\[
\tilde{Q}(q) \triangleq 2Q(-\sqrt{2}q) = \frac{2}{\sqrt{\pi}} \int_{q}^{\infty} e^{-t^2} dt \quad 0 \leq q \in \mathbb{R}
\]

is the complementary error function and \( Q(\cdot) \) in turn the standard normal distribution function.

3.1.1 Subspace Gradient

The Mahalanobis distance for \( \mathcal{X} \) projected onto the \( k < n \) columns of the matrix \( S \) is

\[
\delta(S) = \sqrt{m^T S (S^T \Sigma S)^{-1} S^T m}
\]

and the Bayes error consequently

\[
V(S) = \frac{1}{2} \tilde{Q} \left( \frac{\delta(S)}{2\sqrt{2}} \right).
\]

Here we must assume that \( S \) has full column rank so that the inverse \( (S^T \Sigma S)^{-1} \) exists. In the search for the matrix \( S \) that gives the minimum Bayes error \( V(S) \), the subspace gradient of \( V(S) \), denoted \( \nabla V(S) \), is important, and some results will now be presented. It should be mentioned, that the subspace gradient of the Mahalanobis distance has previously been derived by [Decell and Quirein, 1973].

\[5\] For a given subspace \( S \), the Bayes error is the minimum error rate in that subspace.
**Theorem 3 (Mahalanobis Distance Subspace Gradient)**

For the function $\delta : \mathbb{R}^{n \times k} \to \mathbb{R}$ defined by

$$
\delta(S) = \sqrt{m \, S (S^T \Sigma S)^{-1} S^T m}
$$

(111)

holds for the operator $\nabla$ defined in (54) that

$$
\nabla \delta(S) = \frac{(m - \Sigma (S^T \Sigma S)^{-1} S^T m) \, m^T S (S^T \Sigma S)^{-1}}{\delta(S)}.
$$

(112)

**Proof** We define for notational purposes a differential operator as

$$
\square \alpha = \frac{\partial}{\partial \alpha}.
$$

(113)

We also refer to the two well-known calculus rules

$$
[AB]_\alpha = [A]_\alpha \, B + A [B]_\alpha
$$

(114)

$$
[A^{-1}]_\alpha = A^{-1} [A]_\alpha \, A^{-1}.
$$

(115)

We thus have that

$$
\left[\delta^2(S)\right]_\alpha = \left[ m^T S \left( S^T \Sigma S \right)^{-1} S^T m \right]_\alpha
\begin{align*}
= & m^T \left( [S]_\alpha \left( S^T \Sigma S \right)^{-1} S^T \right) m \\
+ & m^T \left( S \left( \left( S^T \Sigma S \right)^{-1} \right)_\alpha S^T \right) m \\
+ & m^T \left( S \left( S^T \Sigma S \right)^{-1} [S^T]_\alpha \right) m \\
= & m^T \left( [S]_\alpha \left( S^T \Sigma S \right)^{-1} S^T \right) m \\
+ & m^T \left( S \left( S^T \Sigma S \right)^{-1} \left( [S]_\alpha + [S]_\alpha \right) \left( S^T \Sigma S \right)^{-1} S^T \right) m \\
+ & m^T \left( S \left( S^T \Sigma S \right)^{-1} [S^T]_\alpha \right) m
\end{align*}
$$

(116)

where we have assumed that $[\Sigma]_\alpha = 0$. With the vectors

$$
A = (S^T \Sigma S)^{-1} S^T m \quad \text{and} \quad B = \Sigma S (S^T \Sigma S)^{-1} S^T m
$$

(117)

defined, it is easily verified that (116) can be simplified to

$$
\left[\delta^2(S)\right]_\alpha = 2 \left( m^T \left[ S \right]_\alpha A - A^T \left[ S^T \right]_\alpha B \right).
$$

(118)

Note that $S^T \Sigma S$ is a positive symmetric matrix. Trivially,

$$
\frac{\partial S}{\partial s_{ij}} = e_i e_j^T
$$

(119)
where \( s_{ij} \) is the \( j \)th entry on the \( i \)th row of \( S \) and \( e_i \) and \( e_j \) are base vectors of appropriate dimension. Thus

\[
\frac{\partial \delta^2(S)}{\partial s_{ij}} = 2(m_iA_j - A_jB_i), \tag{120}
\]

where the indices \( i \) and \( j \) select vector entries. From this we conclude that

\[
\nabla \delta^2(S) = 2(mA^T - BA^T)
\]

\[
= 2(m - \Sigma S(S^T \Sigma S)^{-1}S^T m) m^T S(S^T \Sigma S)^{-1}. \tag{121}
\]

The lemma now follows from the chain rule,

\[
d\bar{\delta} = d\sqrt{\delta^2} = \frac{d\delta^2}{2\delta}. \tag{122}
\]

\[\square\]

**Theorem 4 (Bayes Error Subspace Gradient)**

For the function \( V : \mathbb{R}^{n \times k} \to \mathbb{R} \) defined by

\[
V(S) = \frac{1}{2} \tilde{Q} \left( \frac{\delta(S)}{2\sqrt{2}} \right) \tag{123}
\]

where \( \tilde{Q}(\cdot) \) is the complementary error function (108) and

\[
\delta(S) = \sqrt{m^T S(S^T \Sigma S)^{-1}S^T m}, \tag{124}
\]

holds for the operator \( \nabla \) defined in (54) that

\[
\nabla V(S) = \frac{e^{-\delta(S)}}{\delta(S) \sqrt{8\pi}} (m - \Sigma S(S^T \Sigma S)^{-1}S^T m) m^T S(S^T \Sigma S)^{-1}. \tag{125}
\]

**Proof** The differentiation of \( \tilde{Q}(\cdot) \) in (108) is straightforward,

\[
d\tilde{Q}(q) = -\frac{2}{\sqrt{\pi}} e^{-q^2} \cdot dq, \tag{126}
\]

and thus

\[
dV = -\frac{1}{4\sqrt{2}} \cdot \frac{2}{\sqrt{\pi}} e^{-\delta^2} \cdot d\bar{\delta}. \tag{127}
\]

or

\[
\nabla V(S) = -\frac{1}{\sqrt{8\pi}} e^{-\delta^2(S)} \cdot \nabla \delta(S) \tag{128}
\]

from which the theorem follows with reference to Theorem 3. \[\square\]
3.1.2 Sphering

It is seen that if the \( p \)-conditional distribution of \( X \) is \( N_n(\mu_p, \Sigma) \), then the change of basis

\[
Y = \Sigma^{-\frac{1}{2}}X
\]

(129)
gives the \( p \)-conditional distribution of \( Y \)

\[
N_n(\Sigma^{-\frac{1}{2}}\Sigma\Sigma^{-\frac{1}{2}}, \Sigma^{-\frac{1}{2}}\mu_p) = N_n(I, \Sigma^{-\frac{1}{2}}\mu_p),
\]

(130)
that is, the covariance matrix equals \( I \). Since the Mahalanobis distance is basis invariant, see for instance [Lindgren, 2002, pp. 21], the Bayes error is unaffected by this transformation. This simplification, which means that the Mahalanobis distance coincides with the Euclid distance \( \sqrt{m^T S \cdot m} \), is called sphering.

If we can assume that the subspace has an orthonormal representation, \( S \in \mathbb{R}^{n \times k} (S^T S = I) \), the sphering simplifies the gradient (125) to

\[
\nabla V(S) = \frac{e^{-\delta(S)}}{\delta(S) \sqrt{8\pi}}(I - SS^T)mm^T S
\]

(131)
where \( \delta(S) = \sqrt{m^T SS^T} m \).

3.1.3 Comments

Note that \( S^T \nabla \delta(S) = 0 \), which is should come as no surprise since \( \delta(S) \) is basis invariant. If we assume that \( S \) is orthonormal and that \( \Sigma = I \) then \( (S^T \Sigma S)^{\pm 1} = I \) and the analysis appears simple,

\[
\delta^2(S) = m^T SS^T m \quad \Rightarrow \quad \nabla \delta^2(S) = 2mm^T S,
\]

(132)
see [Lotlikar and Kothari, 2000]. Note however that this function is not homogeneous, and the gradient is not orthogonal to \( S \). This fact renders some naive steepest decent minimizers, for instance \( S_{k+1} = S_k + c \cdot \nabla F(S_k) \), unnecessary poor, at least if we search the optimal subspace. Assuming \( S^T S = I \) and \( \Sigma = I \) in (121) gives

\[
\nabla \delta^2(S) |\Sigma = I, S^T S = I) = 2(I - SS^T)mm^T S,
\]

(133)
that is, \( 2mm^T S \) projected onto the null space of \( S \). This gradient points to a subspace independent of \( S \) (the spanning vectors are linearly independent).

3.2 Mahalanobis Distance, Heteroscedastic Case

The expression for the Bayes error (107) is not defined for distributions with distinct covariance matrices, \( \Sigma_1 \neq \Sigma_2 \). If we can assume that they are “almost equal”, an intuitive approach would be to use the matrix mean,

\[
\Sigma = \frac{1}{2} (\Sigma_1 + \Sigma_2),
\]

(134)
and consider (107) based on this mean to be a simple approximation of the Bayes error. The following experiment gives a hint on how good or bad this approximation can be.
Monte Carlo, Mahalanobis

Let $\Sigma_1 = I$ be fixed, and $r(j)$ a sequence of independent observations of $N(0, 1)$. We will repeat an experiment 500 times, and for each experiment generate random $\Sigma_2$ and mean difference $m$. 20000 samples are taken from each distribution and classified according to Bayes rule (3). The empirical error rate using Bayes rule is compared to (107) using (134). The experiment is conducted in three dimensions.

The mean difference is generated by

$$m^{(i)} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 0.4 \begin{bmatrix} r(j) \\ r(j+1) \\ r(j+3) \end{bmatrix}. \quad (135)$$

and the covariance matrix by

$$\Sigma_2^{(i)} = aa^T, \quad a = \begin{bmatrix} r(j+4) \\ r(j+6) \\ r(j+7) \end{bmatrix}. \quad (136)$$

However, a procedure is used to avoid $\text{cond}(\Sigma_2^{(i)}) > \lambda$. The condition number is used as a measure of dissimilarity between $\Sigma_1$ and $\Sigma_2^{(i)}$. In each experiment we thus know that the dissimilarity in some sense is bounded by $\lambda$.

Figure 5 shows simulation results for $\lambda = 16$. Apparently, reasonable approximation can be expected for small errors, say $V < 0.1$. In Figure 5 we have $\lambda = 10000$, and we see that the approximation is now very loose. The conclusion is that we can expect severely overestimated errors unless the errors are small and the covariance matrices not too dissimilar.

3.2.1 Subspace Gradient

Although the simulation above shows on poor conformity between the Bayes error and the approximation thereof based on (134), the Mahalanobis distance is simple to calculate compared to many other measures of divergence, and we will investigate subspaces where the approximation is targeted for minimization. However, for $q \geq 2$ populations with distinct covariance matrices, the simplifying sphering can not be used. Therefore we use (125) to calculate the subspace gradient in this case.

3.3 Bhattacharyya Distance

For two normal distributions with distinct covariance matrices, the aforementioned Bhattacharyya distance induces a better approximation of the Bayes error than the Mahalanobis distance. The Bhattacharyya distance (29) is rewritten here for convenience,

$$b \triangleq \frac{1}{8} m^T \left[ \frac{1}{2} (\Sigma_1 + \Sigma_2) \right]^{-1} m + \frac{1}{2} \log \frac{\det \left[ \frac{1}{2} (\Sigma_1 + \Sigma_2) \right]}{\sqrt{\det \Sigma_1 \det \Sigma_2}}. \quad (137)$$

Inspired by the bound (28), we have found by Monte Carlo simulations and curve fitting that

$$\bar{V} = 0.35 e^{-1.35b} \quad (138)$$
Figure 5: Error estimated by (134) inserted in (107) versus empirical error using Bayes rule (3) for 500 random data sets. The dissimilarity between $\Sigma_1$ and $\Sigma_2$ is bounded by $\lambda = 16$.

Figure 6: Error estimated by (134) inserted in (107) versus empirical error using Bayes rule (3) for 500 random data sets. The dissimilarity between $\Sigma_1$ and $\Sigma_2$ is bounded by $\lambda = 10000$. 
Figure 7: Error estimated by the Bhattacharyya distance (137) inserted in (138) versus empirical error using Bayes rule (3) for 500 random data sets. The dissimilarity between $\Sigma_1$ and $\Sigma_2$ is bounded by $\lambda = 10000$.

is a rather good Bayes error approximation, see simulation below.

An estimate of the error rate based on the Bhattacharyya distance has previously been suggested by [Lee and Choi, 2000], see (33). Figure 8 depicts a comparison between this polynomial fit and (138). For small distances the two functions are very similar. The exponential has acceptable behavior for large distances $b$, which can not be said about the polynomial. The polynomial estimate can apparently not be used unless we can state that $b < 3$.

Monte Carlo, Bhattacharyya

The same data as in Monte Carlo 1 is used, but (138) now estimates the Bayes error. The result for $\lambda = 10000$ is depicted in Figure 7. The approximation (138) appears to be a sensible alternative to (107), when we expect difference in the covariance matrices.

3.3.1 Subspace Gradient

By the chain rule, the subspace gradient of (138) is

$$\nabla\tilde{V}(S) = \tilde{V}(S) \cdot [-1.35\nabla b(S)]$$

(139)

where $\nabla b(S)$ in turn is the subspace gradient of the Bhattacharyya distance. Below we will give a result on how to compute this latter gradient analytically, but first mention that a derivation has previously been made by [Decell and Quirein, 1973].
Figure 8: Comparison by the polynomial error estimate suggested by [Lee and Choi, 2000] (33) and the exponential (138). The polynomial estimate obviously has undesirable properties for large distances.


**Theorem 5 (Bhattacharyya Distance Subspace Gradient)**

For the function $b: \mathbb{R}^{n \times k} \rightarrow \mathbb{R}$ defined by

$$b(S) = \frac{1}{8} m^T S \left[ \frac{1}{2} S^T (\Sigma_1 + \Sigma_2) S \right]^{-1} S^T m$$

$$+ \frac{1}{2} \log \frac{\det \left[ \frac{1}{2} S^T (\Sigma_1 + \Sigma_2) S \right]}{\sqrt{\det [S^T \Sigma_1 S] \det [S^T \Sigma_2 S]}}. \quad (140)$$

holds for the operator $\nabla$ defined in (54) that

$$\nabla b(S) = \frac{1}{2} \left[ m - (\Sigma_1 + \Sigma_2) S (S^T (\Sigma_1 + \Sigma_2) S)^{-1} S^T m \right] m^T S (S^T (\Sigma_1 + \Sigma_2) S)^{-1}$$

$$+ 2[\Sigma_1 + \Sigma_2] S (S^T [\Sigma_1 + \Sigma_2] S)^{-1} - \frac{1}{2} \Sigma_1 S (S^T \Sigma_1 S)^{-1} - \frac{1}{2} \Sigma_2 S (S^T \Sigma_2 S)^{-1}. \quad (141)$$

**Proof**  The Bhattacharyya distance is a sum,

$$b(S) = \frac{1}{8} \delta^2(S) + L(S) \quad (142)$$

where

$$\delta^2(S) = m^T S \left[ \frac{1}{2} S^T (\Sigma_1 + \Sigma_2) S \right]^{-1} S^T m \quad (143)$$
and

\[ L(S) = \frac{1}{2} \log \frac{\det \left( \frac{1}{2} S^T (\Sigma_1 + \Sigma_2) S \right)}{\sqrt{\det [S^T \Sigma_1 S] \det [S^T \Sigma_2 S]}} \]

\[ = \frac{1}{2} \log \det \left( \frac{1}{2} S^T (\Sigma_1 + \Sigma_2) S \right) - \frac{1}{4} \log \det [S^T \Sigma_1 S] - \frac{1}{4} \log \det [S^T \Sigma_2 S]. \]  

(144)

Since differentiation is a linear operation, clearly

\[ \nabla b(S) = \frac{1}{2} \nabla \delta^2(S) + \nabla L(S). \]  

(145)

By letting \( \Sigma = \frac{1}{2} (\Sigma_1 + \Sigma_2) \), the first term is given almost immediately by Theorem 3, which also identifies the first term in (141).

By the calculus rule

\[ \frac{\partial}{\partial A} \log \det [A] = (A^T)^{-1} \]  

(146)

we have that

\[ \frac{\partial}{\partial S} \log \det [S^T \Sigma S] = 2 \Sigma S (S^T \Sigma S)^{-1} \]  

(147)

and thus

\[ \nabla L(S) = 2 [\Sigma_1 + \Sigma_2] S (S^T [\Sigma_1 + \Sigma_2] S)^{-1} \]

\[ - \frac{1}{2} \Sigma_1 S (S^T \Sigma_1 S)^{-1} - \frac{1}{2} \Sigma_2 S (S^T \Sigma_2 S)^{-1} \]  

(148)

which identifies the three final terms in (141).

\[ \square \]

### 3.4 Union for Many Classes

We will use the union of pair-wise divergence measures,

\[ V_u(S) = \sum_{1 \leq i < j \leq q} \tilde{V}_{ij}(S), \]  

(149)

as a bound (when \( \tilde{V}_{ij}(S) \geq V_{ij}(S) \)) or approximation of the Bayes error. We will take \( \tilde{V}_{ij}(S) \) to be the three earlier described in this section (one at a time). We will thus search for the solution to

\[ \min_{S \in \mathbb{V}_{n,k}} V_u(S), \]  

(150)

where \( S \in \mathbb{V}_{n,k} \) means that \( S \) is an \( n \)-by-\( k \) matrix with \( S^T S = I_k \). It has however been found that the numerical minimization is greatly enhanced if we instead minimize \( \log V_u(S) \)\(^6\). We will not discuss why this is so, but we can also motivate it by that taking the logarithm gives a more convenient scale, \( \log V_u(S) \) usually lies in the interval \(-100\) to \(0\). It should be mentioned that \( \log V_u(S) \) is a smooth, but usually very complex function with many local minima.

\(^6\)In fact, the step length of the conjugate gradient method will diverge without this modification.
3.4.1 Subspace Gradient

The subspace gradient of $V_u(S)$ is of course

$$\nabla V_u(S) = \sum_{1 \leq i < j \leq q} \nabla V_{ij}(S),$$

(151)

and for $\log V_u(S)$

$$\nabla \log V_u(S) = \frac{\nabla V_u(S)}{V_u(S)}.$$
4 Numerical Experiments

The concepts that have been described so far will be tested on a number of data sets. The mean vectors and covariance matrices of the populations will be taken as the standard maximum likelihood estimates. We will study two aspects:

1. Convergence rate of the four different programs described in Section 3:
   (a) Householder reflections
   (b) Conjugate gradients
   (c) Steepest descent
   (d) Givens rotations

   The programs minimize $\log V_u(S)$ (149). Three types of pair-wise error rate estimates are tested:
   (i) Homoscedastic model using the Mahalanobis distance by estimating the covariance matrix common to all populations as
   
   $$\Sigma = \frac{1}{N - g} \sum_{p=1}^{g} (N_p - 1) \hat{\Sigma}_p$$  \hspace{1cm} (153)

   where $\hat{\Sigma}_p$ is the standard maximum likelihood estimate of the covariance matrix of population $p$, $N_p$ the number of observations of population $p$ and $N$ the total number of observations, for details, see [Johnson and Wichern, 1998, pp. 671]. The error rate is estimated by (107).
   (ii) Heteroscedastic model using the approximative Mahalanobis distance (134) inserted in (107).
   (iii) Heteroscedastic model using the Bhattacharyya distance in (138).

   For each dataset, 60 random start solutions are generated, and the mean of the function value of those 60 runs are plotted as a function of program execution time. The motivation for studying execution time instead of number of function evaluations is that we want to compare the efficiency of programs that calculate the analytical gradients with those who do not. The analytical gradient, when used, is calculated in conjunction with the function evaluation. It takes between 20-100% additional time to do this. The function and gradient evaluation stands together for the major part of the execution time, 75-90%. In turn, the major part of the evaluation time is spent on the error function (108) and the matrix inversion that stems from (109).

2. Empirical error rate for the linear transformations of the data sets found by minimizing $\log V_u(S)$ for the three different types of pair-wise error rate estimates (i)-(iii) above. The empirical error rate is is given by the application of a decision rule optimal for normal populations (a quadratic classifier).
4.1 Notes on the Implementation

The programs that will be evaluated are implemented in MATLAB. Some notes follow.

Householder Reflection

The implementation uses a standard program for unconstrained minimization, the MATLAB function `fminunc` in the `optimization toolbox`. Also the inverse of the composite reflection has been implemented. The minimization program terminates if the function tolerance (TolFun) $10^{-5}$ is achieved. The number of iterations is however maximized to 200.

Givens Rotation

The implementation uses a standard program for unconstrained minimization, the MATLAB function `fminunc` in the `optimization toolbox`. Also the inverse of the Givens rotation has been implemented. The code is somewhat optimized by using precalculated matrix products. This however limits the maximum number of dimensions to 10. The minimization program terminates if the function tolerance (TolFun) $10^{-5}$ is achieved. The number of iterations is however maximized to 200.

Steepest Descent

The program terminates when the absolute value of the line derivative $|\phi'_i(0)|$ is less than $10^{-4}$. The number of iterations is however maximized to 200.

Conjugate Gradients

The program terminates when the absolute value of the scalar product between $h$ and $\nabla F(Y)$ is less than $10^{-6}$. The number of iterations is however maximized to 200.

Line Minimizer

The line minimizer program used by the steepest descent and conjugate gradients algorithms terminates when the strong Wolfe conditions with $c_1 = 0.1$ and $c_2 = 0.5$ are fulfilled. Starting from an initial guess,

$$\alpha_i^0 = \alpha_{i-1} \frac{\phi'_i(0)}{\phi''_i(0)}, \quad (154)$$

the program tries as a first step to bracket the minimum of $\phi_i(\alpha)$ closest to $\alpha = 0$. Then it uses one step of cubic interpolation to locate the minimum. If the point so found fails to fulfill the conditions of termination, the program resigns to search by dividing intervals in half. Usually around five function/gradient evaluations are sufficient, but if the initial guess is bad, as many as 20 evaluations have been observed.
4.2 Artificial Data

A toy data set is computer generated for pedagogical purposes. Six distributions \( N_5(\mu_j, I) \) are given the six distinct means

\[
[0 \ 0 \ \pm 12 \ 0 \ 0]^T \quad \text{and} \quad [\pm 6 \ \pm 3 \ 0 \ 0 \ 0]^T.
\]

Clearly the populations are homoscedastic, \( \Sigma_j = I \). We now seek a 2-dimensional projection of 240 samples of the 5-dimensional distributions that retains the ability to discriminate among the six populations.

Figure 9 shows the projection given by LDA (13). Of course, LDA favors the major mean difference projections

\[
[0 \ 0 \ 1 \ 0 \ 0]^T \quad \text{and} \quad [1 \ 0 \ 0 \ 0 \ 0]^T,
\]

leaving four classes totally mixed up. Using the LDA as a start solution taking one step with the conjugate gradients program gives the projection in Figure 10.

A homoscedastic model is used where the pair-wise Bayes error is estimated by (107). The objective function decreased from \( -0.58 \) to \( -5.45 \). 10 function and gradient evaluations were done.

This example was clearly designed to fool the LDA and favor the more clever numerical optimization. This is perhaps not entirely fair since there are instances where there actually is very little improvement to be made by numerical optimization. For instance, LDA always finds the optimal subspace for two homoscedastic populations. However, it is very hard (if possible) to find a counter example that speaks in favor for the LDA.

4.3 Experimental Data

4.3.1 Image Segmentation Data Set

This data set is downloaded from the UCI Repository of machine learning databases [Blake and Merz, 1998]. The observations are drawn randomly from a database of 7 outdoor images of brick face, sky, foliage, cement, window, and grass, thus constituting 7 classes. The images have been hand-segmented to create a classification for every pixel, where each observation is a 3x3 pixel region. The data set contains 2310 observations. The dimensionality is \( n_0 = 19 \).

We use principal component analysis (see Section 1.6) as a preprocessing step, \( n = 10 \) components are used for further analysis.

Homoscedastic Model using the Mahalanobis Distance

Figure 11 on page 47 shows the convergence rates of programs minimizing the estimated error rate of a homoscedastic model based on the Mahalanobis distance. In average the programs take about 1 second to converge. Interestingly, all runs starting from different random points approach the same function value; \( -0.63 \) (no local minima?). The programs that use the gradient shows a slightly better performance (conjugate gradients and steepest descent). Figure 12 on page 48 shows the observations transformed by the best transformation found in all runs. After the transformation the empirical error rate is 11%. Compare this to LDA (13) with an empirical error rate of 18%. The plot of observations transformed by LDA is shown in Figure 13 on page 49.
Figure 9: A diagram with the six 5-dimensional populations in the artificial data set projected onto the $k = 2$ Fisher linear discriminants (13). A coordinate basis has been chosen so the axes in the diagram are uncorrelated.
Figure 10: A diagram with the six 5-dimensional populations in the artificial data set projected onto the \( k = 2 \) columns of \( S \). A coordinate basis has been chosen so the axes in the diagram are uncorrelated. \( S \) is given by one step of the conjugate gradients program starting from the projection in Figure 9. A homoscedastic model is used where the pair-wise Bayes error is estimated by (107).
Table 1: Best empirical error rates and average time to convergence for log \( V_u(S) \) (149) with different estimates of the pair-wise error rates.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp. error</td>
<td>18%</td>
<td>11%</td>
<td>10%</td>
<td>27%</td>
</tr>
<tr>
<td>Conv. time ≈</td>
<td>-</td>
<td>1 s</td>
<td>15 s</td>
<td>25 s</td>
</tr>
</tbody>
</table>

Heteroscedastic Model using the Approximative Mahalanobis Distance

Figure 14 on page 50 shows the convergence rate of programs minimizing the estimated error rate of a heteroscedastic model based on the approximative Mahalanobis distance using (134) in (107). In average the programs take about 15 second to converge. However, in average the conjugate gradients program converge to a better function value. Here the conjugate gradients program is superior. Figure 12 on page 48 shows the observations transformed by the best transformation found in all runs. After the transformation the empirical error rate is 10%. This slight improvement compared to the homoscedastic model does in this case not motivate the extra computation time.

Heteroscedastic Model using the Bhattacharyya Distance

Figure 14 on page 50 shows the convergence rate of programs minimizing the estimated error rate of a heteroscedastic model based on the Bhattacharyya distance (138). In average the programs take about 25 second to converge. The steepest descent algorithm converges to worse function values compared to the other algorithms. Figure 12 on page 48 shows the observations transformed by the best transformation found in all runs. After the transformation the empirical error rate is 27%. In short, this objective function is very complex to minimize and gives in the end an error rate worse than LDA.

Summary

Table 1 summarizes average convergence time and best empirical error rates. For the Image Segmentation data we see that the error rate of LDA can be improved if some extra time of computation is acceptable. But using the heteroscedastic models give little, if any, gain in error rate despite slow convergence. The preferred minimization program is the conjugate gradients.
Figure 11: The evolution of the objective function $\log V_u(S)$ (149) due to different minimization programs. The curves are the mean of 60 runs with random starting points. A homoscedastic model is used where the pair-wise Bayes error is calculated using (107). The data set is the Image Segmentation data, projection from $n = 10$ to $k = 2$ dimensions. Minimum function value found (by steepest descent): $-0.63$. Average function value at convergence: $-0.58 \pm 0.1$ std.
Figure 12: A diagram with the observations of *Image Segmentation* data projected onto the $k = 2$ columns of $S$. A coordinate basis has been chosen so the axes in the diagram are uncorrelated. $S$ has been taken as a local minimizer of $\log V_u(S)$ (149). A homoscedastic model is used where the pair-wise Bayes error is estimated by (107).
Figure 13: A diagram with the observations of Image Segmentation data projected onto the $k = 2$ Fisher linear discriminants (13). A coordinate basis has been chosen so the axes in the diagram are uncorrelated.
Figure 14: The evolution of the objective function $\log V_u(S)$ (149) due to different minimization programs. The curves are the mean of 60 runs with random starting points. A heteroscedastic model is used where the pair-wise Bayes error is approximated using the covariance matrix mean (134) together with (107). The data set is the Image Segmentation Data, projection from $n = 10$ to $k = 2$ dimensions. Minimum function value found (by Householder reflections): $-0.97$. Average function value at convergence: $-0.82 \pm 0.25$ std.
Figure 15: A diagram with the observations of Image Segmentation data projected onto the $k = 2$ columns of $S$. A coordinate basis has been chosen so the axes in the diagram are uncorrelated. $S$ has been taken as a local minimizer of $\log V_u(S)$ (149). A heteroscedastic model is used where the pair-wise Bayes error is approximated using the covariance matrix mean (134) together with (107).
Figure 16: The evolution of the objective function $\log V_u(S)$ (149) due to different minimization programs. The curves are the mean of 60 runs with random starting points. A heteroscedastic model is used where the pair-wise Bayes error is approximated using the Bhattacharyya distance (138). The data set is the Image Segmentation Data, projection from $n = 10$ to $k = 2$ dimensions. Minimum function value found (by Householder reflections): $-1.17$. Average function value at convergence: $-0.84 \pm 0.23$ std.
Figure 17: A diagram with the observations of Image Segmentation data projected onto the $k = 2$ columns of $S$. A coordinate basis has been chosen so the axes in the diagram are uncorrelated. $S$ has been taken as a local minimizer of $\log V_u(S)$ (149). A heteroscedastic model is used where the pair-wise Bayes error is approximated using the Bhattacharyya distance (138).
4.3.2 Grain Quality Data

This data set stems from measurements by an array of semiconductor gas sensors. 800 measurements on 8 different grain qualities gives a data set with 8 populations, 100 measurements in each. The signals from the sensors are recorded under an interval of time when a gas mixer switches from clean dry air to gas picked up immediately above a grain sample. The switching induces pulses in the signals that are slightly different in magnitude and shape for different samples. The recorded signals due to one sample are stacked in a measurement vector $x_i$. The dimensionality of this measurement vector depends on the number of sensors as well as on the time duration of the recording and the sampling rate of the analogue-to-digital converter and is usually very high. We use however principal component analysis (see Section 1.6) as a preprocessing step, $n = 7$ components are used for further analysis.

Convergence Rate

Comparing the convergence rates between the objective function given by the Mahalanobis distance in Figure 19, the approximative Mahalanobis distance in Figure 21, and the Bhattacharyya distance in Figure 23, we see that the two latter heteroscedastic models take at least 10 times as much time to converge. We see in Figure 19 that the objective function is likely to converge to a global optima since all curves approach the best value found: $-6.17$. Other authors have made the same observation, although no evidence that this is so in general exists.

Comparing the different optimization programs in the same figures, we see that the conjugate gradient program is the most time efficient.

Projection Quality

For reference, the data set is projected onto the Fisher linear discriminants (13), see Figure 18. By visual inspection we see that the error rate is rather high in this diagram.

Figure 20 is a diagram of linear combinations found by minimizing a function based on the Mahalanobis distance, Figure 22 is due to the approximative Mahalanobis distance, and Figure 24 to the Bhattacharyya distance. By visual inspection we conclude that the empirical error rates are all zero for these linear combinations, why this measure can not be used for evaluation here. The evaluation will thus be a subjective inspection of the diagrams.

Summary

The most time efficient optimization program here is the conjugate gradient algorithm.
Figure 18: A diagram with the observations of Grain Quality data projected onto the $k = 2$ Fisher linear discriminants (13). A coordinate basis has been chosen so the axes in the diagram are uncorrelated.
Figure 19: The evolution of the objective function \( \log V_u(S) \) (149) due to different minimization programs. The curves are the mean of 60 runs with random starting points. A homoscedastic model is used where the pair-wise Bayes error is calculated using (107). The data set is the Grain Quality data, projection from \( n = 7 \) to \( k = 2 \) dimensions. Minimum function value found (by steepest descent): \(-6.17\). Average function value at convergence: \(-6.2 \pm 0.003\) std.
Figure 20: A diagram with the observations of Grain Quality data projected onto the $k = 2$ columns of $S$. A coordinate basis has been chosen so the axes in the diagram are uncorrelated. $S$ has been taken as a local minimizer of $\log V_u(S)$ (149). A homoscedastic model is used where the pair-wise Bayes error is estimated by (107).
Figure 21: The evolution of the objective function log $V_u(S)$ (149) due to different minimization programs. The curves are the mean of 60 runs with random starting points. A heteroscedastic model is used where the pair-wise Bayes error is approximated using the covariance matrix mean (134) together with (107). The data set is the Grain Quality data, projection from $n = 7$ to $k = 2$ dimensions. Minimum function value found (by Givens rotations): $-17.2$. Average function value at convergence: $-14 \pm 3$ std.
Figure 22: A diagram with the observations of \textit{Grain Quality} data projected onto the $k = 2$ columns of $S$. A coordinate basis has been chosen so the axes in the diagram are uncorrelated. $S$ has been taken as a local minimizer of $\log V_u(S)$ (149). A heteroscedastic model is used where the pair-wise Bayes error is approximated using the covariance matrix mean (134) together with (107).
Figure 23: The evolution of the objective function \( \log V_u(S) \) (149) due to different minimization programs. The curves are the mean of 60 runs with random starting points. A heteroscedastic model is used where the pair-wise Bayes error is approximated using the Bhattacharyya distance (138). The data set is the Grain Quality data, projection from \( n = 7 \) to \( k = 2 \) dimensions. Minimum function value found (by Givens rotations): \(-20.5\). Average function value at convergence: \(-17 \pm 3 \) std.
Figure 24: A diagram with the observations of *Grain Quality* data projected onto the $k = 2$ columns of $S$. A coordinate basis has been chosen so the axes in the diagram are uncorrelated. $S$ has been taken as a local minimizer of $\log V_u(S)$ (149). A heteroscedastic model is used where the pair-wise Bayes error is approximated using the Bhattacharyya distance (138).
5 Conclusions

In the search for linear subspaces of normal populations, some different objective functions and optimization techniques have been studied. For the data sets in the study, marginal improvements were gained by using heteroscedastic models. A heteroscedastic model allows for distinct population covariance matrices. The heteroscedastic model is much more complex and take at least 10 times as much time to minimize compared to the homoscedastic model. For many applications, we would thus probably settle with a simple homoscedastic model.

The nonlinear conjugate gradients algorithm for the Grassmann manifold is the most time efficient alternative among the optimization programs in the study. If no analytical gradient is available, parameterization by Householder reflections is a good alternative.
References


