# LECTURE NOTE #10

PROF. ALAN YUILLE

## 1. Principle Component Analysis (PCA)

One way to deal with the curse of dimensionality is to project data down onto a space of low dimensions, see figure (1).

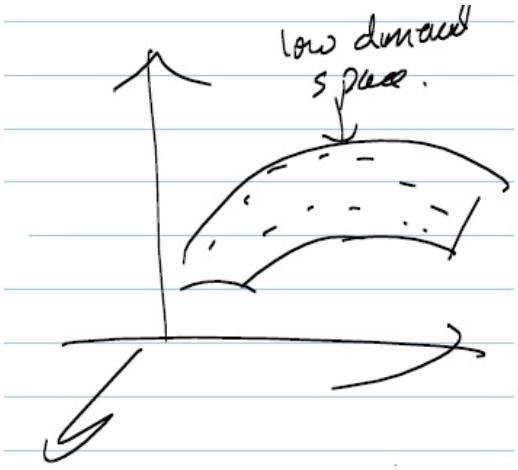


Figure 1

There are a number of different techniques for doing this. Now, we discuss the most basic method - Principle Component Analysis(PCA)

## 2. Convention

$\underline{\mu}^T \underline{\mu}$ is a scalar $\quad \mu_1^2 + \mu_2^2 + \cdots + \mu_D^2$

$\underline{\mu}\,\underline{\mu}^T$ is a matrix $\begin{pmatrix} \mu_1^2 & \mu_1\mu_2 & \mu_1\mu_3 & \cdots \\ \vdots & \mu_2^2 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix}$

Data samples $\underline{x}_1, \ldots, \underline{x}_N$ in D-dimension space.

Compute the mean

$\vec{\mu} = \frac{1}{N} \sum_{i=1}^{N} \underline{x}_i$

Compute the covariance

$\underline{\underline{K}} = \frac{1}{N} \sum_{i=1}^{N} (\underline{x}_i - \underline{\mu})(\underline{x}_i - \underline{\mu})^T$

Next compute the eigenvalues and eigenvector of $\underline{\underline{K}}$

Solve $\underline{\underline{K}}\underline{e} = \lambda \underline{e}$

$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$

*Note* : $\underline{\underline{K}}$ is a symmetric matrix- so eigenvalues are real, eigenvectors are orthogonal. $\vec{e}_{nu} \cdot \vec{e}_{\mu} = 1$ if $\mu = \nu$, and $= 0$ otherwise. Also, by construction, the matrix $\underline{\underline{K}}$ is positive semi-definite, so $\lambda_N \geq 0$ (i.e. no eigenvalues are negative).

PCA reduces the dimension by projection the data onto a space spanned by the eigenvectors $\underline{e}_i$ with $\lambda_i > T$, where $T$ is a threshold

Let $M$ eigenvectors be kept.

Then project data $\underline{x}$ onto the subspace spanned by the first $M$ eigenvectors, after subtracting out the mean.

3.

Formally:

Express

$\underline{x} - \underline{\mu} = \sum_{\nu=1}^{D} a_\nu \underline{e}_\nu$

where the coefficients $\{a_\nu\}$ are given by

$a_\nu = (\underline{x} - \underline{\mu}) \cdot \underline{e}_\nu$

Orthogonality, mean $\underline{e}_\nu \cdot \underline{e}_\mu = \delta_{\nu\mu} \rightarrow$ Kronecker delta

Hence:

$\underline{x} = \underline{\mu} + \sum_{\nu=1}^{D} \langle (\underline{x} - \underline{\mu}) \cdot \underline{e}_\nu \rangle \underline{e}_\nu$

and there is no dimension reduction (no compression)

Then, approximate

$\underline{x} \approx \underline{\mu} + \sum_{\nu=1}^{M} \langle (\underline{x} - \underline{\mu}) \cdot \underline{e}_\nu \rangle \underline{e}_\nu$

This Projects

the data into the M-dimension subspace of form:

$\underline{\mu} + \sum_{\nu=1}^{M} b_\nu \underline{e}_\nu$

4.

In 2-dimensions *Visually*

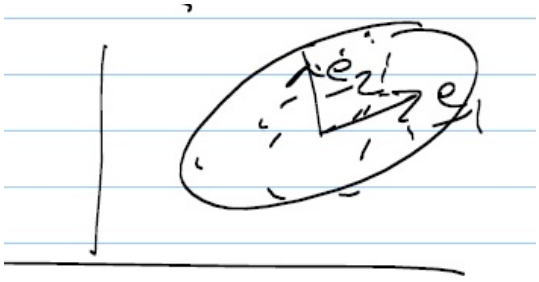

FIGURE 2. In two-dimensions, the eigenvectors give the principle axes of the data.

The eigenvector of $\underline{\underline{K}}$ corresponds to the second order movements of the data, see figure (2).

If the data lies (almost) on a straight line, then $\lambda_1 \gg 0, \lambda_2 \approx 0$, see figure (3).

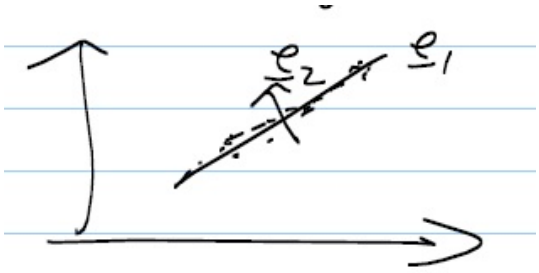

FIGURE 3. In two-dimensions, if the data lies along a line then $\lambda_1 > 0$ and $\lambda_2 \approx 0$.

## 5. PCA and Gaussian Distribution

PCA is equivalent to performing ML estimation of the parameters of a Gaussian

$p(\underline{x} \mid \underline{\mu}, \underline{\underline{\Sigma}}) = \frac{1}{(2\pi)^{D/2}\sqrt{\det \underline{\underline{\Sigma}}}} e^{-\frac{1}{z}(\underline{x}-\underline{\mu})^T \underline{\underline{\Sigma}}^{-}1(\underline{x}-\underline{\mu})}$

to get $\hat{\underline{\mu}}, \hat{\underline{\underline{\Sigma}}}$ by performing ML on $\prod_i p(\underline{x}_i \mid \underline{\mu}, \underline{\underline{\Sigma}})$.

And then throw away the directions where the standard deviation is small. ML gives $\vec{\mu} = \frac{1}{N}\sum_{i=1}^{N} \vec{x}_i$ and $\Sigma = \frac{1}{N}\sum_{i=1}^{N}(\vec{x}_i - \vec{m}u)(\vec{x}_i - \vec{\mu})^T$.
See Bishop's book for probabilistic PCA.

## 6. Cost Function for PCA

$J(\underline{M}, \{a\}, \{e\}) = \sum_{k=1}^{N} ||(\underline{\mu} + \sum_{i=1}^{M} a_{ki}\underline{e}_i) - \underline{x}_k||^2)$

Minimize $J$ w.r.t. $\underline{M}, \{a\}, \{e\}$ Data $\{\underline{x}_k : k = 1 \text{ to N}\}$

The $\{a_{ki}\}$ are projection coefficients

Intuition: find the M-dimensional subspace s.t. the projections of the data onto this subspace have minimal error, see figure (4).

Minimizing $J$, gives the $\{\hat{\underline{e}}_i\}$'s to be the eigenvectors of the covariance matrix
$\underline{K} = \frac{1}{N}\sum_{k=1}^{N}(\underline{x}_k - \vec{\mu})(\underline{x}_k - \vec{\mu})^T$
$\underline{\mu} = \frac{1}{N}\sum_{k=1^N} \underline{x}_k$
$\hat{a}_{ki} = (\underline{x}_k - \hat{\underline{\mu}}) \cdot \underline{e}_i$ the projection coefficients.

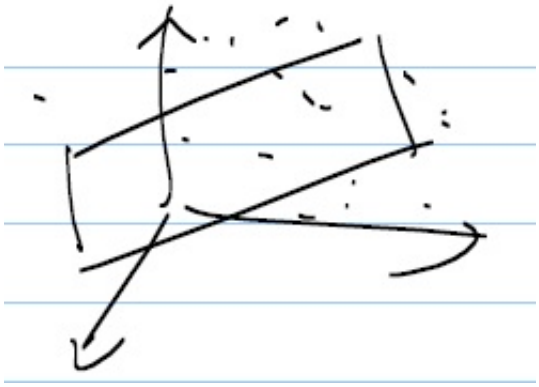

FIGURE 4. PCA can be obtained as the projection which minimizes the least square error of the residuals.

7.

To understand this fully, you must understand Singular Value Decomposition (SVD)

We can re-express the criteria as
$J[\underline{\mu}, \{a\}, \{e\}] = \sum_{k=1}^{N} \sum_{b=1}^{D} \{(\mu_b - x_{bk}) + \sum_{i=1}^{M} a_{ki} e_{ib})\}^2$
where $b$ denotes the vector component.

This is an example of a general class of problem.

Let $E[\Psi, e] = \sum_{a=1,k=1}^{a=D,k=N} (\tilde{x}_{ak} - \sum_{\nu=1}^{M} \Psi_{a\nu} \Phi_{\nu k})^2$

Goal: minimize $E[\Psi, e]$ w.r.t. $\Psi, e$

This is a bilinear problem, that can be solved by SVD.

Note: $\tilde{x}_{ak} = x_{ak} - \mu_a$

the position of the point, relative to the mean.

## 8. Singular Value Decomposition SVD

Note: $\underline{\underline{X}}$ is not a square matrix (unless $D = N$). So it has no eigenvalues or eigenvectors.

We can express any $N \times D$ matrix $\underline{\underline{X}}$, $x_{ak}$
in form

$\underline{\underline{X}} = \underline{\underline{E}}\ \underline{\underline{D}}\ \underline{\underline{F}}$
$x_{ak} = \sum_{\mu,\nu=1}^{M} e_{a\mu} d_{\mu\nu} f_{\nu k}$

where $\underline{\underline{D}} = \{d_{\mu\nu}\}$ is a diagonal matrix ($d_{\mu\nu} = 0, \mu \neq \nu$),

$\underline{\underline{D}} = \begin{pmatrix} \sqrt{\lambda_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sqrt{\lambda_N} \end{pmatrix}$, where the $\{\lambda_i\}$ are eigenvalues of $\underline{\underline{X}}\ \underline{\underline{X}}^T$(equivalently of $\underline{\underline{X}}^T\ \underline{\underline{X}}$)

$\underline{\underline{E}} = \{e_{a\mu}\}$ are eigenvectors of $(\underline{\underline{X}}\ \underline{\underline{X}}^T)_{ab}$

$\underline{\underline{F}} = \{f_{\nu k}\}$ are eigenvectors of $(\underline{\underline{X}}^T\ \underline{\underline{X}})_{kl}$,

* $\mu$&$\nu$ label the eigenvectors.

Note: For $\underline{\underline{\bar{X}}}$ defined on previous page, we get
that $(\underline{\underline{\tilde{X}}}\ \underline{\underline{\bar{X}}}^T) = \sum_{k=1}^{N} (\underline{x}_k - \underline{\mu})(\underline{x}_k - \underline{\mu})^T$

Note: if $(\underline{\underline{X}}\ \underline{\underline{X}}^T)\underline{e} = \lambda\underline{e}$
then $(\underline{\underline{X}}^T\ \underline{\underline{X}})(\underline{\underline{X}}^T\underline{e}) = \lambda(\underline{\underline{X}}^T\underline{e})$

This relates the eigenvectors of $\underline{\underline{X}}\ \underline{\underline{X}}^T$ and of $\underline{\underline{X}}^T\ \underline{\underline{X}}$
(calculate the eigenvectors for the smallest matrix, then deduce those of the bigger matrix - usually $D < N$)

9.

Minimize:

$E[\psi, e] = \sum_{a=1,k=1}^{a=D,k=N} (\tilde{x}_{ak} - \sum_{\nu=1}^{M} \psi_{a\nu}\phi_{\nu k})^2$

we set $\begin{cases} \psi_{a\nu} = \sqrt{\delta_{\nu\nu}}e_a^\nu \\ \phi_{\nu k} = \sqrt{\delta_{\nu\nu}}f_k^\nu \end{cases}$

Take M biggest terms in the SVD expansion of $\underline{\underline{x}}$

But there is an ambiguity.

$\sum_{\nu=1}^{M} \psi_{a\nu}\phi_{\nu k} = (\underline{\underline{\psi}}\,\underline{\underline{\phi}})_{ak} = (\underline{\underline{\psi}}\,\underline{\underline{A}}\,\underline{\underline{A}}^{-1}\underline{\underline{\phi}})_{ak}$
for any M × M invertible matrix $\underline{\underline{A}}$

$\underline{\underline{\psi}} \rightarrow \underline{\underline{\psi}}\,\underline{\underline{A}}$

$\underline{\underline{\phi}} \rightarrow \underline{\underline{A}}^{-1}\underline{\underline{\phi}}$

For the PCA problem - we have constants that the projection directions one orthogonal unit eigenvectors. This gets rid of the ambiguity.

# LECTURE NOTE #10

## 10. Relate SVD to PCA (Linear Algebra)

Start with an $n \times m$ matrix $\underline{\underline{X}}$

$\underline{\underline{X}}\,\underline{\underline{X}}^T$ is a symmetric $n \times n$ matrix

$\underline{\underline{X}}^T\underline{\underline{X}}$ is a symmetric $m \times m$ matrix

$((\underline{\underline{X}}\,\underline{\underline{X}}^T)^T = \underline{\underline{X}}\,\underline{\underline{X}}^T)$

By standard linear algebra

$\underline{\underline{X}}\,\underline{\underline{X}}^T\underline{e}^\mu = \lambda^\mu \underline{e}^\mu$

$n$ eigenvalues $\lambda^\mu$

eigenvectors $\underline{e}^\mu$

eigenvectors are orthogonal $\underline{e}^\mu \cdot \underline{e}^\nu = \delta^{\mu\nu}$ $(= 1$ if $\mu = \nu$, $= 0$ if $\mu \neq \nu)$.

Similarly,

$\underline{\underline{X}}\,\underline{\underline{X}}^T\underline{f}^\nu = \tau^\nu \underline{f}^\nu$

m eigenvalues $\tau^\nu$

eigenvectors $\underline{f}^\nu$

$\underline{f}^\mu \cdot \underline{f}^\nu = \delta^{\mu\nu}$

The $\{\underline{e}^\mu\}$ and $\{\underline{f}^\nu\}$ are related

because

$(\underline{\underline{X}}^T\underline{\underline{X}})(\underline{\underline{X}}^T\underline{e}^\mu) = \lambda^\mu(\underline{\underline{X}}^T\underline{e}^\mu)$

$(\underline{\underline{x}}\underline{\underline{x}}^T)(\underline{\underline{x}}\underline{f}^\mu) = \tau^\mu(\underline{\underline{x}}\underline{f}^\mu)$

Hence:

$\underline{\underline{X}}^T\underline{e}^\mu \propto \underline{f}^\mu, \;\; \underline{\underline{X}}\underline{f}^\mu \propto \underline{e}^\mu$

$\lambda^\mu = \tau^\mu$

If n >m , then there are n eigenvectors $\{\underline{e}_\mu\}$ and m eigenvectors $\{\underline{f}_\mu\}$.

So several $\{\underline{e}_\mu\}$ relate to the same $\underline{f}_\mu$.

11.

Claim:

we can express

$\underline{\underline{X}} = \sum_\mu \alpha^\mu \underline{e}^\mu \underline{f}^{\mu T}$

$\underline{\underline{X}}^T = \sum_\mu \alpha^\mu \underline{f}^\mu \underline{e}^{\mu T}$

For some $\alpha^\mu$. (we will solve for $\alpha^\mu$ later.)

$\underline{\underline{X}}\underline{f}^\nu = \sum_\mu \alpha^\mu \underline{e}^\mu \underline{f}^{\mu T}\underline{f}^\nu$

$= \sum_\mu \alpha^\mu \delta_{\mu\nu}\underline{e}^\mu = \alpha^\nu \underline{e}^\nu$

Verify the claim

$\underline{\underline{X}}\underline{\underline{X}}^T = \sum_{\mu,\nu} \alpha^\nu \underline{e}^\nu \underline{f}^{\nu T}\alpha^\mu \underline{f}^\mu \underline{e}^{\mu T}$

$= \sum_{\mu,\nu} \alpha^\nu \alpha^\mu \underline{e}^\nu \delta_{\mu\nu}\underline{e}^{\mu T} = \sum_\mu (\alpha^\mu)^2 \underline{e}^\mu \underline{e}^{\mu T}$

Similarly

$\underline{\underline{x}}^T \underline{\underline{x}} = \sum_\mu (\alpha^\mu)^2 \underline{f}^\mu \underline{f}^{\mu T}$

So $(\alpha^\mu)^2 = \lambda^\mu$

(Because we can express any symmetric matrix in form $\sum_\mu \lambda_\mu \underline{e}^\mu \underline{e}^{\mu T}$, where $\lambda^\mu$ are the eigenvalues and $\vec{e}^\mu$ are eigenvectors.)

$\underline{\underline{X}} = \sum_\mu \alpha^\mu \underline{e}^\mu \underline{f}^{\mu T}$ is the SVD of $\underline{\underline{X}}$

In coordinates:

$x_{ai} = \sum_\mu \alpha^\mu e_a^\mu f_i^\mu$

$x_{ai} = \sum_{\mu,\nu} e_a^\mu \alpha^\mu \delta_{\mu\nu} f_i^\nu$

$\underline{\underline{x}} = \underline{\underline{EDF}}$

$E_{a\mu} = e_a^\mu, \underline{\underline{D}}_{\mu\nu} = \alpha^\mu \delta_{\mu\nu}, F_{\nu i} = f_i^\nu$

## 12. Effectiveness of PCA

In practice, PCA often reduces the data dimension a lot.

But it will not be effective for some problems.

For example, if the data B a set of strings

$(1, 0, 0, 0, \dots) = \underline{x}_1$

$(0, 1, 0, 0, \dots) = \underline{x}_2$

$(0, 0, 0, 0, \dots, 0, 1) = \underline{x}_N$

then it can be computed that there is one zero eigenvalue of PCA. But all the other eigenvalues are not small. In general, PCA works best if there is a linear structure to the data. It works poorly if the data lies on a curved surface and not on a flat surface.

## 13. Fisher's Linear Discriminant

PCA may not be the best way to reduce the dimension if the goal is discrimination. Suppose you want to discriminate between two classes of data 1&2, shown in figure (5).

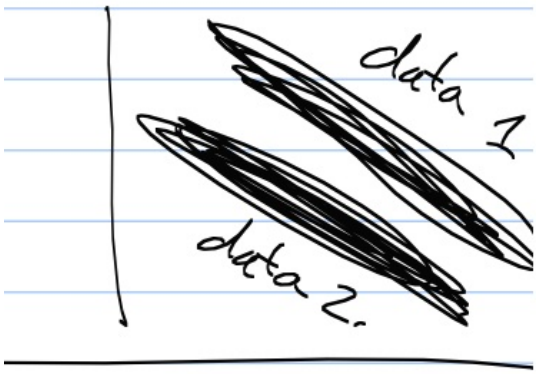


Figure 5. This type of data is bad for PCA. Fisher's Linear Discriminant does better of the goal is discrimination.

If you put both sets of data into PCA, you will get this, see figure (6). The eigenvectors are $\vec{e}_1, \vec{e}_2$ with eigenvalues $\lambda_1 > \lambda_2$. Because of the form of the data $\lambda_1 >> \lambda_2$.

The best axis, according to PCA is in the worst direction for discrimination (best axis is $\vec{e}_1$ because $\lambda_1 >> \lambda_2$).

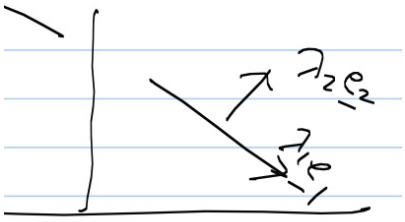

FIGURE 6. The PCA projections for the data in figure (5) The best axis, according to PCA, is the worst axis for projection if the goal is discrimination.

Projecting datasets onto $\vec{e}_1$ gives, see figure (13):

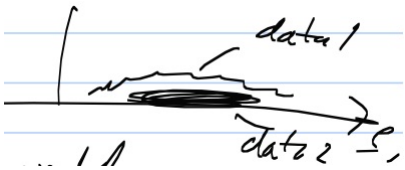


FIGURE 7. If we project the data onto $\vec{e}_1$, then data 1 and 2 gets all mixed together. Very bad for discrimination.

The second direction $\vec{e}_1$ would be for better. This would give, see figure (13):

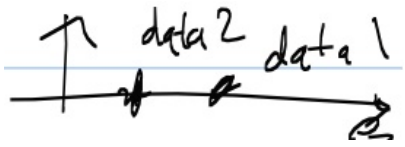


FIGURE 8. We get a better discrimination if we project the data onto $\vec{e}_2$, then it is easy to separate data 1 from data 2.

14.

Fisher's Linear Discriminant gives a way to find a better projection direction.

$n_1$ samples $\underline{x}_i$ from class $X_1$
$n_2$ samples $\underline{x}_i$ from class $X_2$

Goal: find a vector $\underline{w}$, project data onto this axis (i.e. $\underline{x}_i \cdot \vec{w}$) so that the data is well separated.

Define the sample mean
$\underline{m}_i = \frac{1}{N_i} \sum_{\underline{x} \in X_i} \underline{x}$ for $i = 1, 2$.

Define scatter matrices
$\underline{\underline{S}}_i = \sum_{\underline{x} \in X_i} (\underline{x} - \underline{m}_i)(\underline{x} - \underline{m}_i)^T$ for $i = 1, 2$.

Define the between-class scatter
$\underline{\underline{S}}_B = (\underline{x} - \underline{m}_i)(\underline{x} - \underline{m}_i)^T$ between class $X - 1$ and $X_2$.

Finally define the within-class scatter
$\underline{\underline{S}}_W = \underline{\underline{S}}_1 + \underline{\underline{S}}_2$

15.

Now project onto the (unknown) direction $\underline{w}$
$\hat{m}_i = \frac{1}{N_i} \sum_{\underline{x} \in X_i} \underline{w} \cdot \underline{x} = \underline{x} \cdot \underline{m}_i$, using the definitions of the sample means.

The means of the projections are the projections of the means.
The scatter of the projected points is
$\hat{S}_i^2 = \sum_{x \in X_i} (\underline{w} \cdot \underline{x} - \underline{w} \cdot \underline{m}_i)^2$
$= \underline{w}^T \underline{\underline{S}}_i \underline{w}$, by definition of the scatter matrices.

Fisher's criterion
choose the projection direction $\underline{w}$ to
minimize: $J(\underline{w}) = \frac{|\hat{\underline{m}}_1 - \hat{\underline{m}}_2|^2}{\hat{S}_1^2 + \hat{S}_2^2}$

This maximizes the ratio of the between-class distance ($|\hat{\underline{m}}_1 - \hat{\underline{m}}_2|$) to the within-class scatter.

16.

This is a good projection direction, see figure (9), while other projection directions are bad – see figure (10).

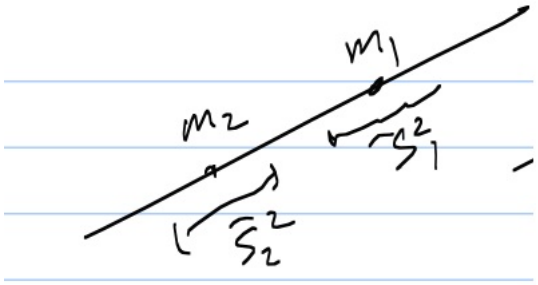


FIGURE 9. The projection of the data from figure (5) onto the best direction $\vec{w}$ (at roughly forty-five degrees). This separates the data well because the distance between the projected means $m_1, m_2$ is a lot bigger than the projected scatters $\hat{S}_1, \hat{S}_2$.

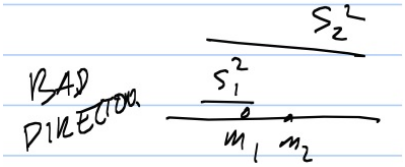


FIGURE 10. A bad projection direction, by comparison to figure (9). The distance between the projected means $m_1, m_2$ is smaller than the projected scatters $\hat{S}_1, \hat{S}_2$.

*Result* :

The projection direction that maximize $J(\underline{\omega})$ is $\underline{\omega} = \underline{\underline{S}}_{\omega}^{-1}(\underline{m}_1 - \underline{m}_2)$. Note that this is not normalized (i.e. $|\vec{\omega}| \neq 1$), so we must normalize the vector.

Proof

Note that the Fisher criterion is independent of the norm of $\vec{\omega}$. So we can maximize it by arbitrarily requiring that $\underline{\omega}^T \underline{\underline{S}}_{\omega} \underline{\omega} = \tau$, where $\tau$ is a constant. This can be formulated in term of maximization with constraints:
maximize $\underline{\omega}^T \underline{\underline{S}}_B \underline{\omega} - \lambda(\underline{\omega}^T \underline{\underline{S}}_{\omega} \underline{\omega} - \tau)$

* $\lambda$ : Lagrange multiplier

* $\tau$ : constant

$\frac{\delta}{\delta \underline{\omega}} \rightarrow \underline{\underline{S}}_B \underline{\omega} - \lambda \underline{\underline{S}}_{\omega} \underline{\omega} = 0$

Hence

$\underline{\underline{S}}_{\omega}^{-1} \underline{\underline{S}}_B \underline{\omega} = \lambda \underline{\omega}$

But

$\underline{\underline{S}}_B = (\underline{m}_1 - \underline{m}_2)^T(\underline{m}_1 - \underline{m}_2)$

$\underline{\underline{S}}_B \cdot \underline{\omega} = \rho(\underline{m}_1 - \underline{m}_2)$ for some $\rho$ $(= \vec{w} \cdot (\vec{m}_1 - \vec{m}_2))$.

Hence $\underline{\underline{S}}_B \hat{\underline{\omega}} \propto (\underline{m}_1 - \underline{m}_2)$. This implies that $\underline{\underline{S}}_w^{-1} \underline{\underline{S}}_B \hat{\underline{\omega}} \propto \underline{\underline{S}}_w^{-1}(\underline{m}_1 - \underline{m}_2)$, and the result follows from recalling that $\underline{\underline{S}}_\omega^{-1} \underline{\underline{S}}_B \underline{\omega} = \lambda \underline{\omega}$.

## 17. Fishev's Linear Discriminant

An alternative way to model this problem is to assign a Gaussian model to each dataset (i.e. learn the model parameters $\vec{\mu}, \underline{\underline{\Sigma}}$ for each dataset). Then if the covariance is the same for both datasets then the the Bayes classifier is a straight line whose normal is the direction $\underline{\omega}$

$\underline{\omega} \cdot \underline{x} + \omega_0 = 0, \underline{\omega} = \underline{\underline{\Sigma}}(\underline{\mu}_1 - \underline{\mu}_2)$. This is exactly the same as Fisher's method! See figure (11).

But

if the data comes from two Gaussian with different covariances, then Bayes classifier is a quadratic curve, so it differs from Fisher's linear discriminant.

## 18. Multiple Classes

For $c$ classes, compute $c-1$ discriminants project $D$-dimensional feature with $c-1$ space.

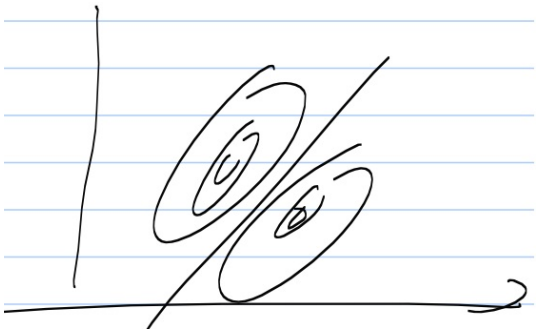

FIGURE 11. Try to model the datasets by learning Gaussian models for each dataset separately. Then we recover Fisher if both datasets have the same covariance. The decision plane will have a surface normal which points along the direction of Fisher's $\vec{\omega}$. But if the two datasets have different covariances, then the Bayes classifier will be a quadratic curve and differs from Fisher.

Without-class

$$\underline{\underline{S}}_{\omega} = \underline{\underline{S}}_{1} + \cdots + \underline{\underline{S}}_{c-1}$$

Between-class

$$\underline{\underline{S}}_{B} = \underline{\underline{S}}_{total} - \underline{\underline{S}}_{\omega} = \sum_{i=1}^{c} n_i \cdot (\underline{m}_i - \underline{m})(\underline{m}_i - \underline{m})^T$$

$\underline{\underline{S}}_{total}$ is the scatter matrix for all the classes.

Multiple Discriminant Analysis

Seek vectors $\omega_i : i = 1, \ldots, c-1$

Project samples to $c-1$ dim space: $(\omega_1 \cdot x, \ldots, \omega_{c-1} \cdot x) = \underline{\underline{\omega}}^T \underline{x}$.

Criteria is $J(\omega) = \frac{|\underline{\omega}^T \underline{\underline{S}}_B \underline{\omega}|}{|\underline{\omega}^T \underline{\underline{S}}_{\omega} \underline{\omega}|}$

$*|.|$ is the determinant

19.

The solution is given by the eigenvectors, where eigenvalues are the $c-1$ largest in $\underline{\underline{S}}_B \underline{W} = \lambda \underline{\underline{S}}_w \underline{w}$

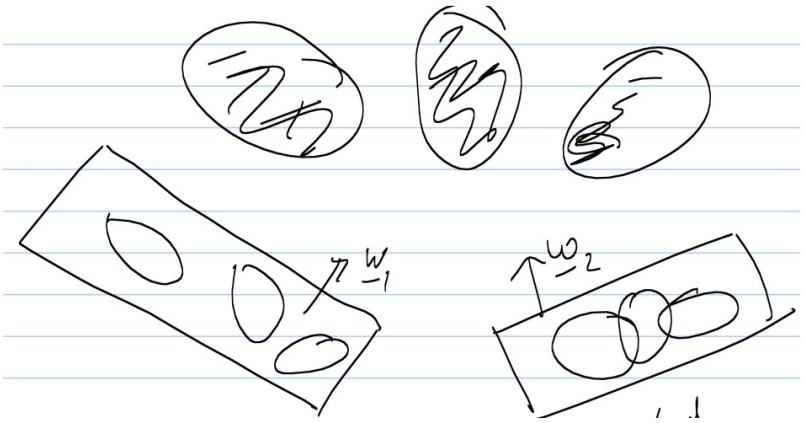


FIGURE 12. $w^1$ is a good project direction, $w^2$ is a bad projection direction.

$\underline{w}^1$ is a good projection of the data, $\underline{w}_2$ is a bad projection, see figure (12)

## LIMITATIONS OF PCA AND FISHER

It is important to realize the limitations of these methods and also why they are popular.

They are popular party because they are easy to implement. They both have optimization criteria which can be solved by linear algebra. This is because the optimization criteria are quadratic, so the solutions are linear. This restriction was necessary when computers did not exist.

But now it is possible to have other optimization criteria which can be solved by computers. For example, based on criteria like nearest neighbour classification. This will be discussed later.

Also PCA assumes that the data lies on a low-dimensional linear space. But what if it lies on a low-dimensional curved space? More advanced techniques can deal with this – e.g. ISOMAP.