① Regression + Multivariate Methods

- Regression: relating a series of measurements to one or more possible factors.

extensions * regularized regression, * logistic regression

- Principal Components Analysis: finding a reduced set of variables to summarize a multivariate quantity
  ~ Synonyms: Factor analysis, Karhunen - Loève decomposition, singular value decomposition

extensions "rotations" - varimax, ...
ICA (Independent components analysis)

- Discriminant Analysis (Fisher discriminant): finding a combination of variables that separate two sets of observations

extension GIFA Generalized indicator function analysis

Combination of the above:
Procrustes analysis: find a linear transformation from one set of multivariate quantities to another

Canonical correlation analysis: find a linear transformation that best relates (best correlates) one set of multivariate quantities and another.

Multidimensional Scaling Embed some points in a vector space to recover specified distances;

All of these ~~are~~ begin with a linear algebra setup, + some can be solved via matrix inversion, some as eigenvalue problems, + some only iteratively.

(2)

Basic setup for regression (extensible to PCA)

A "design matrix" $X = \{x_{mn}\}$, known

Observation $Y = \{y_m\}$, known (viewed as a column)

Find the best set of loadings $\{a_n\}$ for which

$$\sum_{n=1}^{N} x_{mn} a_n \approx y_m \qquad XA \approx Y$$

Convenient to write $y_m^{fit} = \sum_{n=1}^{N} x_{mn} a_n$,

"Best", by default, means that we want to minimize R:

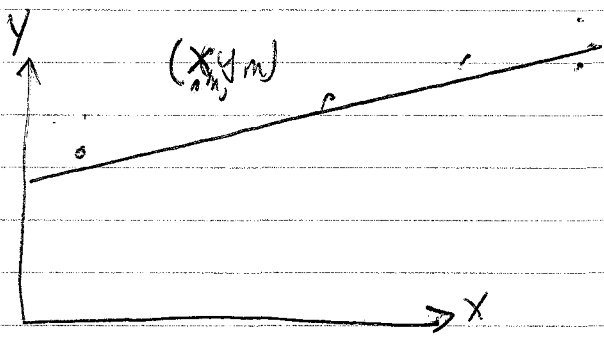
$$R = \sum_m |y_m^{fit} - y_m|^2 = \sum_m \left(\sum_n x_{mn} a_n - y_m \right)^2$$
$$= tr\left((Y - XA)^T (Y - XA) \right)$$

[note $tr\, M^T M = \sum_{ij} M^T_{ij} M_{ji} = \sum_{ij} M_{ij}^2$]

Could use some other R [logistic regression]

Could put priors on A [regularized regression]

Apply to curve-fitting



Find the best line through the data, or,

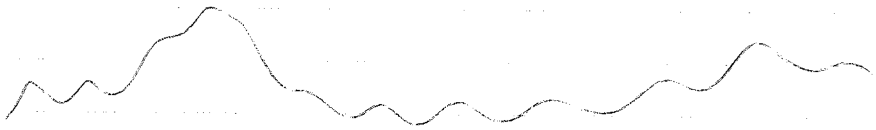
$y_m^{fit} = p x_m^2 + q x_m + r$ - How to put this in above form?

$x_{m1} = 1$, $x_{m2} = x_m$, $x_{m,3} = x_m^2$; $r = a_1$, $q = a_2$, $p = a_3$.

③

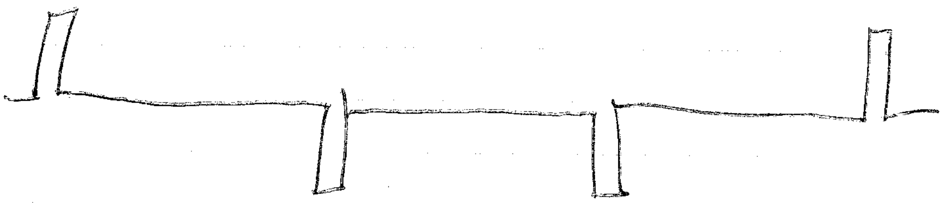
Applic to fMRI signal analysis (one pixel at a time)

Pixel signal is some $y(t)$, discretized as y_m



Each experimental variable is an $x_n(t)$, discretized as x_{mn}.

$x_1(t)$

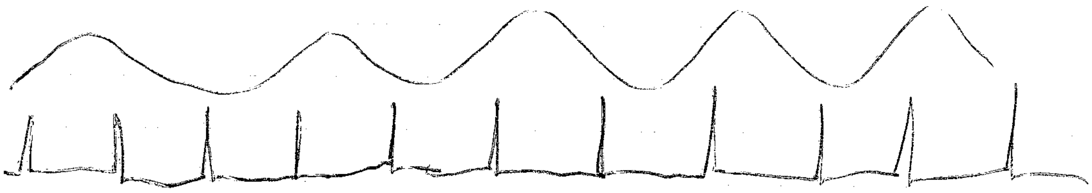


⋮

or nuisance variables (EKG, resps)

$x_2(t)$

$x_3(t)$



⋮

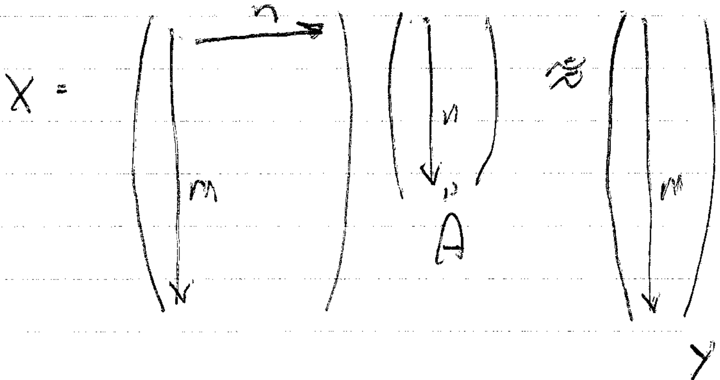
Want to write $y(t) \approx \sum a_n x_n(t)$.

(9)

Our basic regression problem is to minimize

$$R = tr[(Y - Y^{fit})^T (Y - Y^{fit})] = tr((Y - XA)^T (Y - XA))$$

for



$\#(m) \geq \#(n)$. to hope to find unique A, ideally $\#(m) >> \#(n)$.

Y^{fit} can be viewed as projection of Y into the space spanned by the column of X's.

$(Y^{fit} - Y) \perp Y^{fit}$, i.e., $tr((Y^{fit} - Y)^T Y^{fit}) = 0$

$$tr(Y^{fit\,T} Y^{fit}) = tr(Y^T Y^{fit})$$

$$R = tr((Y - Y^{fit})^T Y) = tr(Y^T Y) - tr(Y^{fit\,T} Y^{fit})$$

So minimizing R is the same as maximizing $tr(Y^{fit\,T} Y^{fit})$, i.e., maximizing the length of the projection of Y.

We'll minimize R by setting $\frac{\partial R}{\partial A}$ to 0.

(5)

$$R = tr(Y^T Y) - tr((XA)^T Y) - tr(Y^T XA) + tr((XA)^T XA))$$

What is $\frac{\partial}{\partial a_k}(tr\, QA)$ for A a column?

$tr(QA) = \sum q_{1k} a_{k1}$, so $\frac{\partial}{\partial a_k}(tr\, QA) = q_{1k}$

Thinking of the $\frac{\partial}{\partial a_k}(tr\, QA)$ as forming a column:

$$\begin{pmatrix} \frac{\partial}{\partial a_1} tr(QA) \\ \vdots \\ \frac{\partial}{\partial a_n} tr(QA) \end{pmatrix} = \begin{pmatrix} q_{11} \\ \vdots \\ q_{1n} \end{pmatrix}$$

$$\begin{bmatrix} \vdots \\ \frac{\partial}{\partial a_k} tr(QA) \\ \vdots \end{bmatrix} = Q^T \qquad (Q \text{ a row}).$$

Note $tr(Y^T XA) = tr(A^T X^T Y) = tr((XA)^T Y)$

Also, we can use the product rule to find that $\frac{\partial}{\partial a_k}(tr A^T G A)$

a column of $2(A^T G)^T = 2 G^T A$

$$\text{so } \frac{\partial R}{\partial a_k} = -2(Y^T X)^T + 2 \underbrace{X^T X}_{n \times n} A \qquad (G = X^T X)$$

$$\frac{\partial R}{\partial a_k} = 0 \Rightarrow X^T X A = (Y^T X)^T$$

$$A = (X^T X)^{-1} X^T Y$$

From this, $y^{fit} = XA = \underbrace{\left[X(X^TX)^{-1}X^T\right]}_{\text{the projection onto the space spanned by the columns of } X.} Y$

Simple extension: residual regression + related.

Why do we use $R = \Sigma (y^{fit} - y)^2$?

(A) it leads to a linear problem we can solve

(B) $e^{-R/2\sigma^2}$ can be interpreted as the probability of the observations Y, given that y^{fit} should have been observed (i.e., XA is the model), and each observation Y independently deviates from y^{fit}, with the measurement error drawn from a Gaussian of std dev σ.

Thus, minimizing R maximizes the a posteriori probability of the A's.

But say that we knew that the A's came from a distribution with covariance C_A, i.e.,

$$p(A) \sim e^{-A^T(C_A)^{-1}A/2}$$

[For example, $C_A = \sigma_A^2 I$ - A's not "too big"]

⑦

And the noise might not be independent:

$$p(Y^{fit}-Y) \sim e^{-(Y^{fit}-Y)C_Y^{-1}(Y^{fit}-Y)/2}$$

Now we want to maximize

$$e^{-A^T(C_A)^{-1}A/2} \; e^{-(Y^{fit}-Y)C_Y^{-1}(Y^{fit}-Y)/2}$$

i.e. minimize

$$(Y^{fit}-Y)^T C_Y^{-1}(Y^{fit}-Y) + A^T C_A^{-1} A$$

$$= (XA-Y)^T C_Y^{-1}(XA-Y) + A^T C_A^{-1} A$$

$\frac{\partial}{\partial a_k} = 0$ leads to

$$-(Y^T C_Y^{-1} X)^T + X^T C_Y^{-1} X A + C_A^{-1} A = 0$$

$$A = (X^T C_Y^{-1} X + C_A^{-1})^{-1}(X^T C_Y^{-1} Y)$$

C_A large $\Rightarrow$ its effect goes away

C_A small $\Rightarrow$ A forced to be small.

C_Y decorrelates the errors.

Note that if we have a series of regression problems with the same X's but diff. Y's, we can solve them in parallel:

$$X = \begin{array}{c} \xrightarrow{n} \\ \left(\begin{array}{c} \Big\downarrow m \end{array}\right) \end{array} \begin{array}{c} \xrightarrow{r} \\ \left(\begin{array}{c} \Big\downarrow n \end{array}\right) \\ A \end{array} = \begin{array}{c} \xrightarrow{r} \\ \left(\begin{array}{c} \Big\downarrow m \end{array}\right) \\ Y \end{array}$$

Each column of Y is a separate regression. R = sum of R's for each column.

Still have same solution, column by column, since the columns don't interact:

$$A = (X^T X)^{-1} X^T Y.$$

$\boxed{\text{Regression} \longrightarrow \text{PCA}}$

Say we want to deduce a good set of X's, i.e.

write $\quad Y^{fit}_{mr} = \sum x_{mn} a_{nr}$, with n small.

Can view columns of Y as time series, each col. is a pixel
or each col is an electrode

OR exchange rows + columns.
Each col is a "snapshot"

(9)

The solution will have to be ambiguous.

If $Y^{fit} = XA$, and Q is any $n\times n$ invertible matrix,

$$Y^{fit} = (XQ)(Q^{-1}A) = X'A', \text{ for}$$

$$X' = XQ, \quad A' = Q^{-1}A.$$

We could partially resolve this ambiguity by requiring X to be orthonormal (i.e., apply Gram-Schmidt if it wasn't).

But the above still is an ambiguity since $X' = XR$ is orthonormal for R any ~~ortho~~ unitary matrix.

[X has orthonormal cols $\Leftrightarrow X^T X = I_n$

$$(X')^T X' = (XR)^T XR = R^T X^T X R = R^T R = I.$$

So we really should think of this as a search for the subspace spanned by the columns of X

Same argument can be made for A - its rows can always be made orthogonal.

Leads to a more symmetric statement of the problem:

$$Y^{fit} = B^T \Lambda A \qquad \text{i.e.,} \qquad y^{fit}_{mr} = \sum_{n=1}^{r} \lambda_n b_{nm} a_{nr}$$

$$B = b_{nm}, \quad \Lambda = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}, \quad A = \begin{pmatrix} \xrightarrow{r} \\ \downarrow n \end{pmatrix}$$

$$\begin{pmatrix} \xrightarrow{m} \\ \downarrow n \end{pmatrix}$$

$$BB^T = I$$

$$AA^T = I \quad . \quad \text{Soln is unique other than ordering of the } \lambda\text{s.}$$

⑩

How to do it?

$$\underset{m \times n}{X}\;\underset{n \times r}{A} = \underset{m \times r}{Y}$$

Minimize $\operatorname{tr}\left[(Y - XA)^T (Y - XA) \right]$

over X <u>and</u> A. With X known, $A = (X^T X)^{-1} X^T Y$

We can keep X orthonormal (in columns) so $X^T X = I_{n \times n}$.

so $A = X^T Y$.

$$(Y - XA)^T (Y - XA) = Y^T Y - Y^T X A - A^T X^T Y + A^T X^T X A$$

$$= Y^T Y - Y^T X X^T Y - Y^T X X^T Y + (Y^T X)(X^T X)(X^T Y)$$

$$= Y^T Y - Y^T X X^T Y$$

$$\operatorname{tr}\left((Y - XA)^T (Y - XA)\right) = \operatorname{tr}\left(Y^T Y - Y^T X X^T Y\right)$$

minimizing this means maximizing $\operatorname{tr}(Y^T X X^T Y) = \operatorname{tr}(Y Y^T X X^T)$

$$= \operatorname{tr}(X^T Y Y^T X)$$

YY^T is $m \times m$, and symmetric. Let's write it via its eigenvectors & eigenvalues.

$$Y Y^T = \sum_{h=1}^{r} \lambda_h \phi_h \phi_h^T, \quad \phi\text{'s orthonormal, with}$$

$$(Y Y^T) \phi_h = \lambda_h \phi_h.$$

Orthonormality means $\phi_k^T \phi_l = \delta_{kl}$.

Let's order the λ's so $\lambda_1 \geq \lambda_2 \geq \lambda_3 \cdots$

(11)

Now, let's consider the $n=1$-case.

$$x = \sum_h z_h \phi_h.$$

$$x^T YY^T = \sum_h z_h \phi_h^T \sum_k \lambda_k \phi_k \phi_k^T$$

$$\boxed{\phi_h^T \phi_k = \delta_{hk}} \; !$$

$$= \sum_{h,k} z_h \lambda_k \phi_h^T \phi_k \phi_k^T$$

$$= \sum_k z_k \lambda_k \phi_k^T.$$

So $$x^T YY^T x = \left(\sum_k z_k \lambda_k \phi_k^T\right) \sum_h z_h \phi_h$$

$$= \sum_{k,h} z_k \lambda_k z_h \phi_k^T \phi_h$$

$$= \sum_k z_k^2 \lambda_k$$

How do we maximize $\sum_k z_k^2 \lambda_k$ subj to $\sum z_k^2 = 1$?

Take $z_1 = 1$ (largest $\lambda = \lambda_1$)

others $= 0$. So $x = \phi_1$.

12

$n \geq 2$

$$X = \left(\sum_h z_{h,1} \phi_h \;\middle|\; \sum_h z_{h,2} \phi_h \;\middle|\; \cdots \;\; \sum_h z_{h,n} \phi_h \right)$$

$$X^T Y Y^T = \begin{pmatrix} \sum_k z_{k,1} \lambda_k \phi_k^T \\ \sum_k z_{k,2} \lambda_k \phi_k^T \\ \vdots \\ \sum_k z_{k,n} \lambda_k \phi_k^T \end{pmatrix}$$

$$tr(X^T Y Y^T X) = \sum_k z_{k,1}^2 \lambda_k + \sum_k z_{k,2}^2 \lambda_k + \cdots + \sum_k z_{k,n}^2 \lambda_k$$

Coef of λ_1 is $z_{1,1}^2 + \cdots + z_{1,n}^2$, max poss. this 1.

λ_2 is $z_{2,1}^2 + \cdots + z_{2,n}^2$

So we can make the first n λ's have a coef of 1, by choosing

$$X = \left(\phi_1 \middle| \phi_2 \middle| \cdots \; \phi_n \right)$$, the first n eigenvectors of YY^T

YY^T is $m \times m$.

$$\boxed{YY^T X = X \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}}$$

$A = X^T Y$

Rows of A are Left eigenvecs of $Y^T Y$.

because cols of X are ei's of YY^T

$$A Y^T Y = (X^T Y) Y^T Y = X^T (YY^T) Y = \begin{pmatrix} \lambda_1 & & \\ & \lambda_2 & \\ & & \ddots \\ & & & \lambda_n \end{pmatrix} X^T Y = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix} A.$$

(13)

Symmetric form of soln:

$A: n \times r$
$B: n \times m$
$\Lambda: n \times n$, diagonal

$Y^{fit} = B^T \Lambda A$ where

n rows of A are left eigenvecs of $Y^T Y$ $(r \times r)$, $AA^T = I_r$

n cols of B^T are right eigenvecs of $Y Y^T$ $(m \times m)$, $BB^T = I_n$

($\Leftrightarrow$ n rows of B are left eigenvecs of YY^T
n cols of A^T are right eigenvecs of $Y^T Y$)

$$\Lambda = \begin{pmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_n} \end{pmatrix}$$, since

$$(B^T \Lambda A)(B^T \Lambda A)^T = YY^T$$
$$= B^T \Lambda A A^T \Lambda B$$
$$= B^T \Lambda^2 B$$

so eiv's of Λ^2 must be eivs of YY^T.

If m + r are very different, YY^T + $Y^T Y$ are very different in size, + time to diagonalize differs.
Always diagonalize the smaller one!

The Elegant Approach (avoiding coordinates) to PCA.

We'll need "Lagrange Multipliers" - a general approach for solving constrained minimization problems. Here, it will turn quadratic minimization with quadratic constraints into eigenvalue problems. [Also, big role in stat. physics + info theory]

<u>Recipe</u> Say you want to minimize (or maximize) $F(x_1, x_2, \cdots, x_k)$

subject to constraints $C_1(x_1, \cdots, x_k) = 0$

$\vdots$

$C_L(x_1, \cdots, x_k) = 0.$

Instead of trying to write $x_1 = G_1(x_{L+1}, \cdots, x_k)$

$\vdots$

$x_L = G_L(x_{L+1}, \cdots, x_k)$

[The "obvious approach - use constraints to eliminate variables]

+ minimizing

$$F(\underbrace{G_1(x_{L+1}, \cdots, x_k), \cdots, G_L(x_{L+1}, \cdots, x_k)}_{L \text{ variables eliminated via constraints}}, \underbrace{x_{L+1}, \cdots, x_k}_{K-L \text{ free variables}})$$

L.M. says: minimize $\mathcal{F}(x_1, \cdots, x_k) + \sum_{\ell=1}^{L} \lambda_\ell C_\ell(x_1, \cdots, x_k)$

and find the λ's which satisfy the constraints.

<u>Example</u> Maximize $\sum x_i a_i$ subject to $\sum b_i x_i^2 = 1$ [one constraint]

$\mathcal{F} = \sum x_i a_i + \lambda(\sum b_i x_i^2 - 1)$ $\quad \frac{\partial F}{\partial x_i} = a_i + 2x_i \lambda b_i.$

So $\frac{\partial F}{\partial x_i} = 0 \Rightarrow x_i = -a_i / 2\lambda b_i.$

Now find λ. $\sum b_i x_i^2 = 1 \Rightarrow \frac{1}{(2\lambda)^2} \sum \frac{a_i^2}{b_i} = 1 \Rightarrow \lambda = \frac{1}{2}\sqrt{\sum \frac{a_i^2}{b_i}}$

Obs: 1. Sometimes you don't even need to find λ. $x_i / x_j = \frac{a_i}{b_i} \cdot \frac{b_j}{a_j}$ [sometimes finding λ is hard]

2. Problem stays symmetric.

15

Why does it work? Toy example: 2 variables, one constraint.

Extremize $F(x,y)$ subj to $G(x,y)=0$. Say $G(x,y)=0 \Rightarrow x = H(y)$.

"Standard" approach: set $\frac{\partial F}{\partial y}=0$.

$$\frac{\partial F}{\partial y} = \frac{\partial}{\partial y}\left(F(H(y),y)\right) = \frac{\partial F}{\partial x}\frac{\partial H}{\partial y} + \frac{\partial F}{\partial y}.$$

$$G(x,y)=0 \Rightarrow G(H(y),y)=0 \Rightarrow \frac{\partial G}{\partial y}=0 \Rightarrow \frac{\partial G}{\partial x}\frac{\partial H}{\partial y} + \frac{\partial G}{\partial y}=0.$$

$$\Rightarrow \frac{\partial H}{\partial y} = -\frac{\partial G}{\partial y}\Big/\frac{\partial G}{\partial x}.$$

So we need to solve $\frac{\partial F}{\partial y}=0$, i.e. $\boxed{-\frac{\partial F}{\partial x}\frac{\partial G}{\partial y} + \frac{\partial F}{\partial y}\frac{\partial G}{\partial x}=0.}$

L.M. method: Solve $\frac{\partial}{\partial x}\left(F(x,y)+\lambda G(x,y)\right)=0$

$$\frac{\partial}{\partial y}\left(F(x,y)+\lambda G(x,y)\right)=0$$

$$\begin{cases} \frac{\partial F}{\partial x} + \lambda\frac{\partial G}{\partial x}=0 \\ \frac{\partial F}{\partial y} + \lambda\frac{\partial G}{\partial y}=0 \end{cases} \Rightarrow \lambda = -\frac{\partial F}{\partial y}\Big/\frac{\partial G}{\partial y}$$

$$\Rightarrow \frac{\partial F}{\partial x} + \left(-\frac{\partial F}{\partial y}\Big/\frac{\partial G}{\partial y}\right)\frac{\partial G}{\partial x}=0$$

(in the box).

Everything goes through with multiple variables/constraints, $\frac{\partial G}{\partial x} \to$ matrix of partials.

(16)

Applying LM's to the PCA problem:

Maximize $tr(YY^TXX^T)$ subject to $X^TX = I_{n\times n}$

View $X^TX = I_{n\times n}$ as a symmetric matrix of constraints, $\vec{x}_i \cdot \vec{x}_j = \delta_{ij}$

Each constraint is paired with a λ_{ij}, so Λ = matrix of λ_{ij}'s is symmetric

LM formulation is to maximize $tr(YY^TXX^T) - tr(\Lambda X^TX) = \mathcal{F}$

View $\frac{\partial}{\partial x_{uv}} \mathcal{F} = 0$ as a matrix of equations.

what's $\frac{\partial}{\partial x_{uv}} (tr M X^TX)$?

$$\frac{\partial}{\partial x_{uv}} tr(MX^TX) = \frac{\partial}{\partial x_{uv}} \left(\sum_{i,j} m_{ij} (X^TX)_{ji}\right) = \frac{\partial}{\partial x_{uv}} \sum_{i,j,k} m_{ij}\, x_{kj} x_{ki}$$

$$= \sum_{i,j,k} m_{ij} \left(\frac{\partial}{\partial x_{uv}} x_{kj}\right) x_{ki} + \sum_{i,j,k} m_{ij}\, x_{kj} \frac{\partial}{\partial x_{uv}} (x_{ki})$$

$$= \sum_{\substack{i,j,k \\ k=u,\, j=v}} m_{ij} x_{ki} + \sum_{\substack{i,j,k \\ k=u \\ i=v}} m_{ij} x_{kj} = \sum_i m_{iv} x_{ui} + \sum_j m_{vj} x_{uj}$$

$$= (XM + XM^T)_{uv} \quad \text{. Take } M = \Lambda\ (=M^T).$$

Similarly, $\frac{\partial}{\partial x_{uv}} (tr\, MXX^T) = (MX + M^TX)_{uv}$. Take $M = YY^T$ $(=M^T)$

So the LM formulation, $YY^TX = X\Lambda$, ~~subject to $X^TX = I$~~

simultaneous with $X^TX = I$. This solves for $\Lambda = diag(\text{eigs of } YY^T)$
and X = eigenvectors of YY^T.

("Guess" that Λ is diagonal).

(17)

Another example of quadratic quantity to extremize, $\bar{c}$ quadratic constraint.

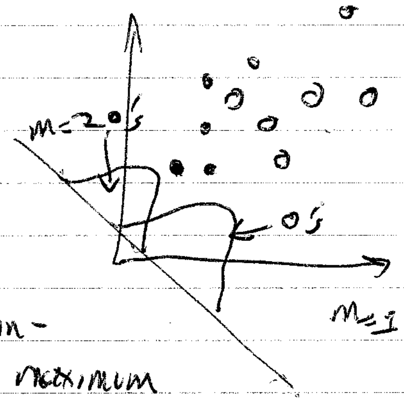
Fisher Discriminant & Canonical Variates.

Setup: multivariate quantities $Y = \begin{pmatrix} \vec y_1 & \vec y_2 & \cdots & \vec y_r \end{pmatrix}$ ($m \times r$), i.e., observations $\vec y_1, \dots, \vec y_r$

Say we know a priori that some of the $\vec y$'s are in category 1
⋮
etc. for c categories.

We want to find linear combination of the coordinates m that do the best job of segregating the $\vec y$'s.

E.g., $c = 2$ (Fisher case)



More formally, find $x_1, \dots, x_m$ s.t.

$x^T \vec y$'s have the minimum within-group variance & the maximum between-group variance. Equal-sized groups (for simplicity).

Assume $\sum \vec y = 0$.

Say $\vec y_1, \dots, \vec y_{r_1}$ in category 1, with mean $\vec\mu_1 = \begin{pmatrix} \mu_{11} \\ \vdots \\ \mu_{1m} \end{pmatrix}$

$\vec y_{r_1+1}, \dots, \vec y_{r_1+r_2}$ " " 2 " " $\vec\mu_2$

⋮

$\vec y_{r_1+r_2+\dots+r_{c-1}+1}, \dots, \vec y_{r_1+r_2+\dots+r_c}$ in cat c, with mean $\vec\mu_c = \begin{pmatrix} \mu_{c1} \\ \vdots \\ \mu_{cm} \end{pmatrix}$

Maximize $\sum_{j,k} [x_j (\vec\mu_k - \vec\mu)_j]^2$ subject to $\sum_{j,h} [x_j (\vec y_h - \vec\mu_{c(h)})_j]^2$

[$\vec\mu$ = global mean, may not be 0 if groups are unequal]
[$c(h)$ = category of h]

(18)

Previous strategy turns this into

$$S_g X = S_w X \Lambda$$ ["generalized eigenvalue problem"]

where S_g = covariance matrix of group means

S_w = covariance matrix within group

$$S_g = \sum_k (\vec{\mu}_k - \vec{\mu})(\vec{\mu}_k - \vec{\mu})^T, \quad S_w = \sum_h (\vec{y}_h - \vec{\mu}_{c(h)})(\vec{y}_h - \vec{\mu}_{c(h)})^T.$$

(Note that if $S_w = I$, then $X = \sum (\vec{\mu}_k - \vec{\mu}) a_k$ will solve.)

Two "flavors" of interest: (A) c categories, top $c-1$ eigenvectors X yield "best" ~~projection~~ linear map of data into a $c-1$-dimensional plane [in which categories separate best by between-group-variance]

$\boxed{r \gg m}$ This is "canonical variate". $c = 2$ is the "Fisher Discriminant".

$C = 2$: Further simplification, since $\vec{\mu}_1 = -\vec{\mu}_2$, so $S_g = 4(\mu_1^T \mu_1)$,

recognize above as $S_w^{-1} S_g X = X \Lambda$ [Many observations r, few cases m]

(B) Two categories, but consider more than just the leading eigenvector. [GIFA = generalized indicator function analysis, Yokoo et al.]

$\boxed{m \gg r}$ This yields all of the linear mappings that discriminate the two categories. Each explains successively less of the variance (as λ decreases). Choose some cutoff $\gamma \alpha$, select only the λ's $> \alpha$, and construct

$$\sum X(\lambda) f(\lambda - \alpha)$$

as the composite "discriminating image".

- "Generalized" - replace S_g by $S_g + \alpha S_w$

[This is used in the imaging context, where S_w is singular, so you can't calculate S_w^{-1}. $m \gg r$]