Lecture 17 EE 559 Mar ~~20~~ 25, 2024

Machine Learning I: Supervised Methods

B. Keith Jenkins

Announcements

- Slido event code: 4509123
- Midterm grading is almost completed; graded midterms will be handed out on Wednesday
- Homework 6 is due Friday
- Project assignment will be posted later this week.

Reading

- Bishop 12.1.1 (PCA), 12.1.3 (PCA applications - optional)
- Bishop, 4.1.4 (FLD), 4.1.6 (MDA)

Today's lecture

- Support vector regression (finish)
- Feature selection and dimensionality reduction (part 2)
  - Fisher's linear discriminant (FLD)
    - Criterion function and solution
    - Extensions to $d > 1$ and $C > 2$
      - Multiple classifiers
      - MDA
- Artificial neural networks (ANN) (part 1)
  - Single neuron unit
  - Activation functions

deferred

Review from Lecture 14:

Support Vector Regression (SVR) [Bishop 7.1.4] (non-augmented notation)

SVM adapted to regression problems

$\underline{x} \rightarrow$ [NL mapping $\underline{\phi}(\underline{x})$ or $k(\underline{x}, \underline{x}')$] $\rightarrow \underline{u} \rightarrow$ [Linear regresor] $\rightarrow \hat{\underline{y}}$

Motivation: to have a regression model suitable to sparse data (e.g., high D', relatively low N).

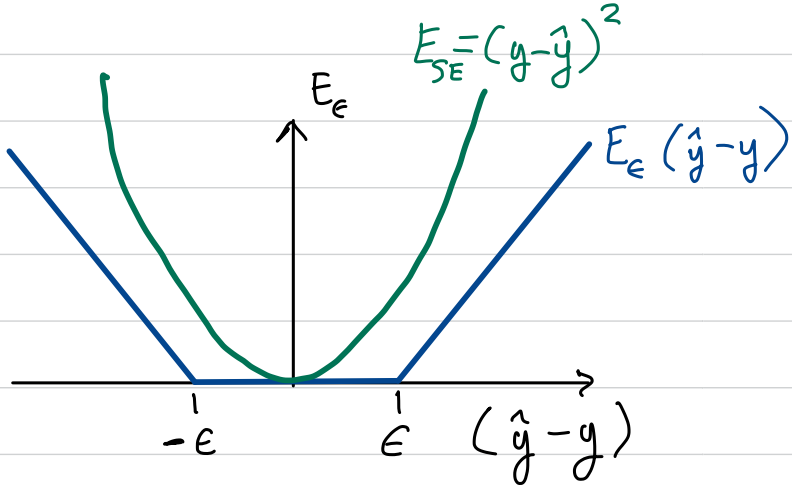
In Ridge Regression, we used a criterion fcn.: MSE + l_2^2 regularizer:

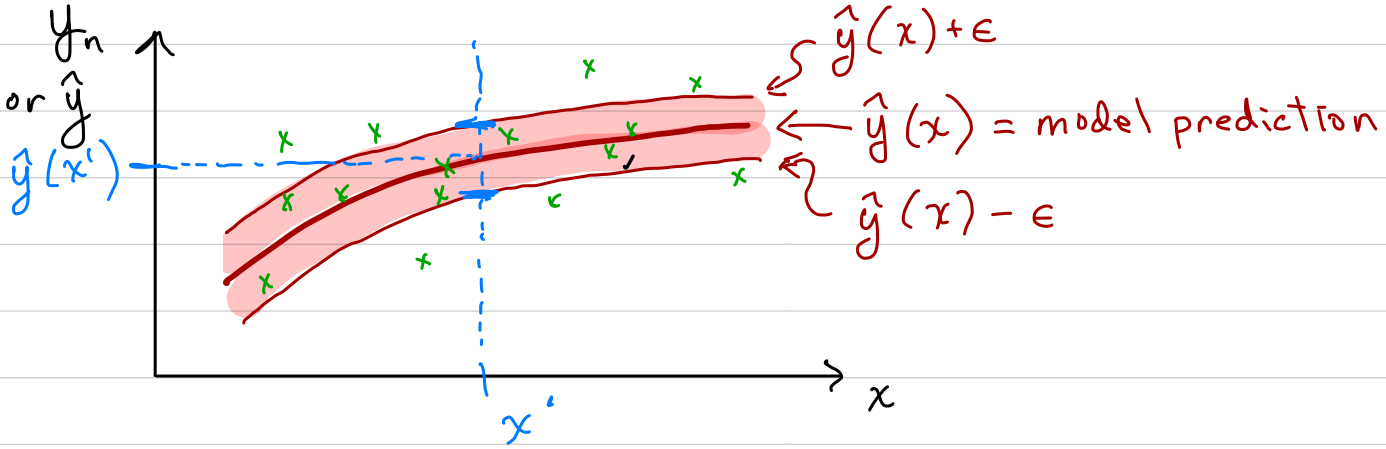
$$J_{RR}(\underline{w}) = \underbrace{\frac{1}{N}\sum_{n=1}^{N}\left[\hat{y}(\underline{x}_n) - y_n\right]^2}_{\text{MSE}} + \underbrace{\lambda \|\underline{w}\|_2^2}_{l_2^2 \text{ regularizer}}, \quad \lambda \geq 0$$

For SVR: use "ϵ-insensitive" loss instead of squared-error loss; keep l_2^2 regularizer

ϵ-insensitive loss E_ϵ:

$$E_\epsilon(\hat{y}-y) = \begin{cases} 0, & \text{if } |\hat{y}-y| < \epsilon \\ |\hat{y}-y| - \epsilon & \text{if } |\hat{y}-y| \geq \epsilon \end{cases}$$





Any data points in the shaded region have $E_\epsilon = 0$.

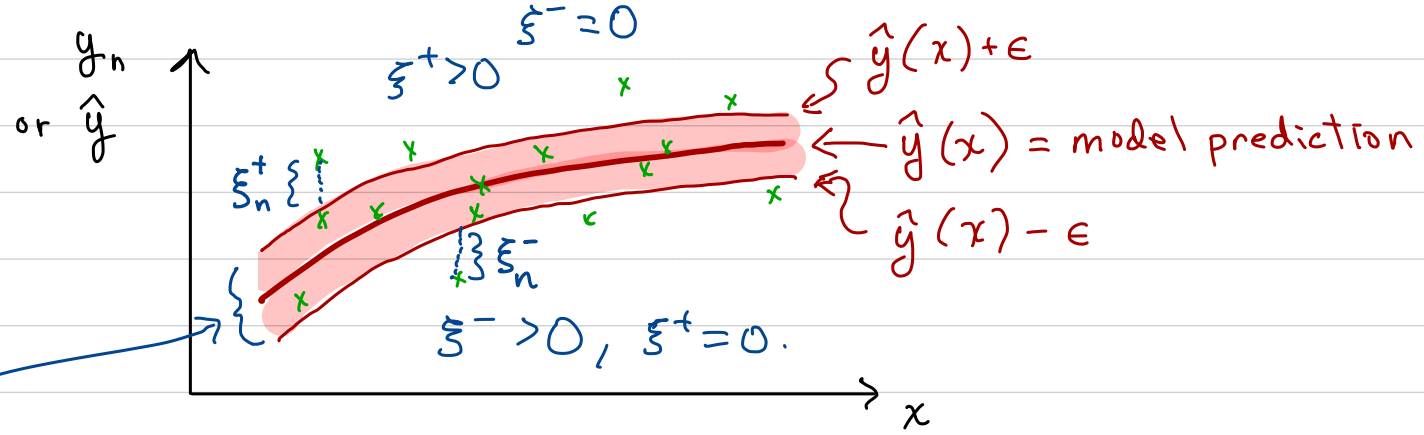
- 0 error for predictions $\hat{y}(\underline{x}_n)$ within ϵ of known output y_n, means learning algorithm is less likely to overfit to noise or small variations in data.

$\Rightarrow$ Use error function (before adding constraints):

$$\tilde{J}(\underline{w}) = C \sum_{n=1}^{N} E_\epsilon \left[\hat{y}(\underline{x}_n) - y_n\right] + \frac{1}{2}\|\underline{w}\|_2^2$$

↑ C here by convention.

To pose as a tractable constrained optimization problem, use slack variables and constraints in place of E_ϵ:



(1) If: $\{ y_n \le \hat{y}(x_n) + \epsilon \ \text{and} \ y_n \ge \hat{y}(x_n) - \epsilon \}$ then: y_n is inside the tube

and $\xi_n^+ = \xi_n^- = 0$ (where $t_\epsilon = 0$) ← Use as set of constraints ?

$\xi_n^+ > 0$ iff $y_n > \hat{y}(x_n) + \epsilon$; $\xi_n^- > 0$ iff $y_n < \hat{y}(x_n) - \epsilon$

Add slack variables:

⇒ we want $\{ y_n \le \hat{y}(x_n) + \epsilon + \xi_n^+ \ \text{and} \ y_n \ge \hat{y}(x_n) - \epsilon - \xi_n^- \}$ ↘ accomodates points that are outside the tube,

(S) we can write: $\tilde{J}(w) = C \sum_{n=1}^{N} (\xi_n^+ + \xi_n^-) + \frac{1}{2} \|w\|_2^2$

($\xi_n^+ + \xi_n^-$: prefers small ξ_n^+, ξ_n^- ; $\frac{1}{2}\|w\|_2^2$: regularizer term)

subject to constraints:

(2)
(i) $\xi_n^+ \ge 0, \ \xi_n^- \ge 0 \quad \forall n$
(ii) $y_n \le \hat{y}(x_n) + \epsilon + \xi_n^+ \quad \forall n$
(iii) $y_n \ge \hat{y}(x_n) - \epsilon - \xi_n^- \quad \forall n$

Yields a Lagrangian (primal form): ϵ insensitive loss; regularizer

$$L(\underline{w}, w_0, \underline{\xi}^+, \underline{\xi}^-, \underline{\mu}^+, \underline{\mu}^-, \underline{\lambda}^+, \underline{\lambda}^-) = C\sum_{n=1}^{N}(\xi_n^+ + \xi_n^-) + \frac{1}{2}\|\underline{w}\|_2^2$$

$$(3)\quad -\underbrace{\sum_{n=1}^{N}(\mu_n^+\xi_n^+ + \mu_n^-\xi_n^-)}_{(i)} - \underbrace{\sum_{n=1}^{N}\lambda_n^+(\epsilon + \xi_n^+ + \hat{y}_n - y_n)}_{(ii)} - \underbrace{\sum_{n=1}^{N}\lambda_n^-(\epsilon + \xi_n^- - \hat{y}_n + y_n)}_{(iii)}$$

in which $\hat{y}_n = \hat{y}(\underline{x}_n) = \underline{w}^T\underline{\phi}(\underline{x}_n) + w_0$

(i) $\xi_n^+ \geq 0,\ \xi_n^- \geq 0 \quad \forall n$

(ii) $y_n \leq y(\underline{x}_n) + \epsilon + \xi_n^+ \quad \forall n$

(iii) $y_n \geq y(\underline{x}_n) - \epsilon - \xi_n^- \quad \forall n$

(iv) $\mu_n^+, \mu_n^-, \lambda_n^+, \lambda_n^- \geq 0 \quad \forall n$

ϵ is user-specified (or obtained through model selection).

One can obtain the dual representation of L [see Bishop].

Support vectors are defined as the vectors that are outside the ϵ-tube or on the boundary of the ϵ-tube.

Feature Selection and Dimensionality Reduction (part 2)

Why?

(S) • Reduce overfitting • Reduce computation • Eliminate irrelevant features.

PCA - last lecture

From Lecture 16, p.18

How well does PCA work?

Ex: C=2 classes

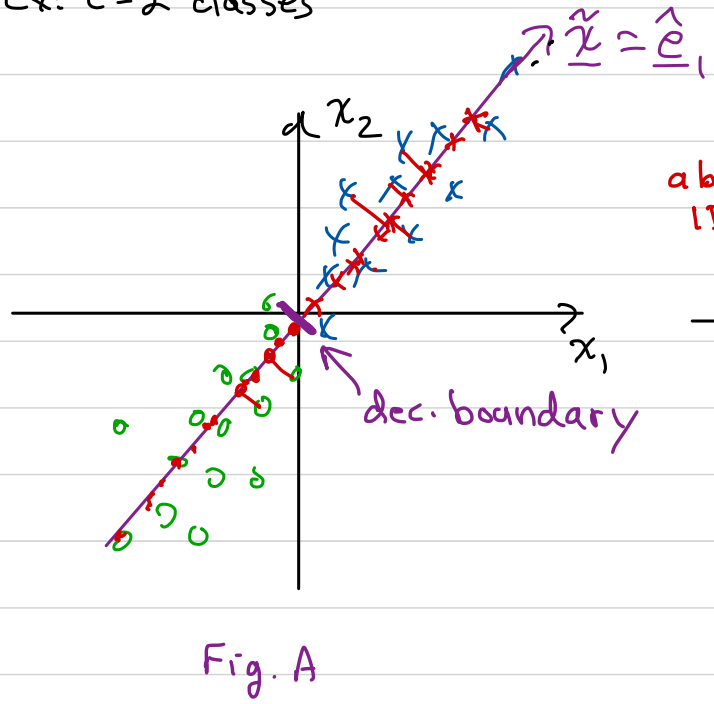


Fig. A

PCA should work well

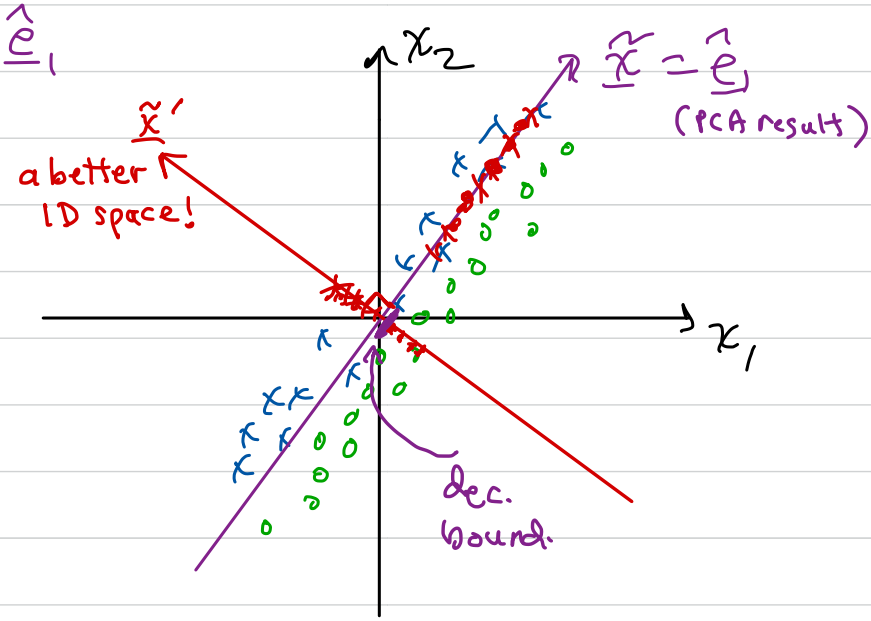


Fig. B

PCA works poorly.

but, using $\tilde{x}'$ as 1D feature space could work well.

Why can't PCA do better on Fig. B?

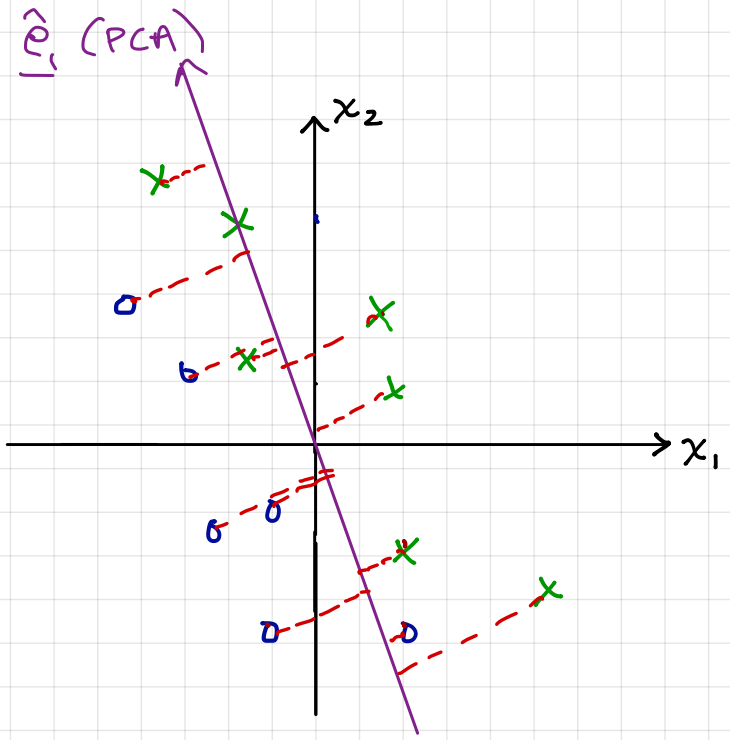
→ PCA ignores class labels

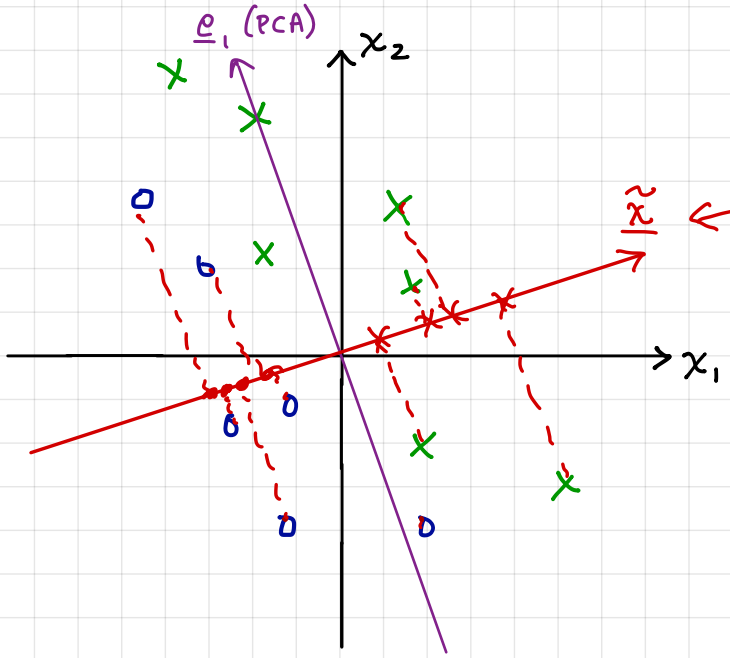
Fisher's Linear Discriminant (FLD) – Problem statement

[Bishop 4.1.4]

For classification problems.

$C = 2$ classes. Dimensionality $D \rightarrow 1 = D'$.





← optimal coord. (F.L.D.)

Find direction of a line (1-D space) that maximizes the separability of projected data pts. (in 1-D feature space).

Criterion:

$$J = \frac{[\text{distance between projected class means}]^2}{[\text{some measure of variance of each projected class, combined}]}$$

→ Then find max of J w.r.t. direction of line ($\underline{w}$).

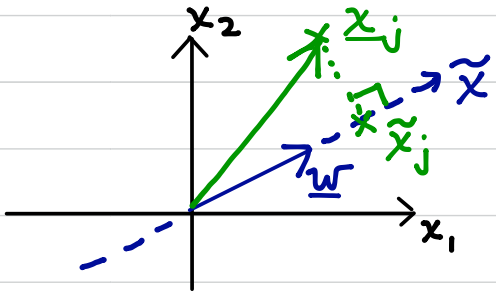
FISHER'S LINEAR DISCRIMINANT - Problem solution

$\tilde{x}$ is new 1D space

$\underline{x}_j \rightarrow$ Linear transformation $\rightarrow \tilde{x}_j$

$\tilde{x}_j = \underline{w}^T \underline{x}_j$

$\|\underline{w}\| = 1$ gives pure projection



Name	Original feat. space	Reduced (1D) feat. space
Data point	$\underline{x}_j$	$\tilde{x}_j$
Sample mean of S_i pts.	$\underline{m}_i$	$\tilde{m}_i = \underline{w}^T \underline{m}_i$
Sample (co)variance of S_i pts.	$\frac{1}{N_i} \underline{S}_i = \frac{1}{N_i} \sum_{\underline{x}_j \in S_i} (\underline{x}_j - \underline{m}_i)(\underline{x}_j - \underline{m}_i)^T$	$\frac{1}{N_i} \tilde{S}_i^2 = \frac{1}{N_i} \sum_{\tilde{x}_j \in S_i} (\tilde{x}_j - \tilde{m}_j)^2$
Within-class scatter	$\underline{S}_W = \underline{S}_1 + \underline{S}_2$	$\tilde{S}_1^2 + \tilde{S}_2^2$
Between-class scatter	$\underline{S}_B = (\underline{m}_2 - \underline{m}_1)(\underline{m}_2 - \underline{m}_1)^T$	$\lvert \tilde{m}_2 - \tilde{m}_1 \rvert^2$

Def Fisher's Linear Discriminant is the linear function $\underline{w}^T \underline{x}$ for which the criterion fcn.:

(1) $$J(\underline{w}) = \frac{|\tilde{m}_1 - \tilde{m}_2|^2}{\tilde{S}_1^2 + \tilde{S}_2^2}$$ is maximized.

$J(\underline{w})$ can be maximized by

$$\nabla_{\underline{w}} J(\underline{w}) = \underline{0}.$$

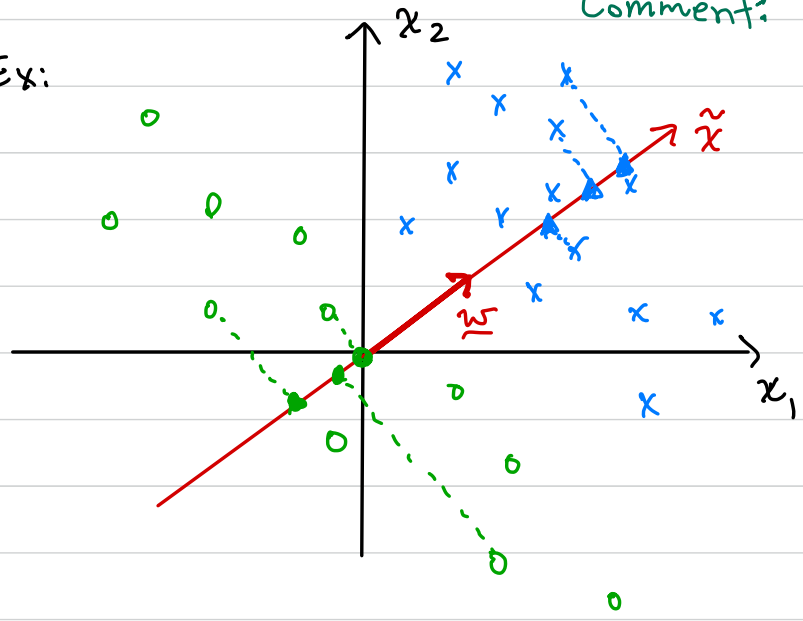
Can show:

$\Rightarrow$ if $\underline{\underline{S}}_w$ is nonsingular, then

(2) $$\boxed{\underline{w} = \underline{\underline{S}}_w^{-1}[\underline{m}_1 - \underline{m}_2]}$$ ← FLD solution.

Note: direction of $\underline{w}$ defines the new (1D) feature space. ∴ We can normalize $\underline{w}$ to $\|\underline{w}\| = 1$ for pure projection.

Ex:



Comment: Typically $\underline{\underline{S}}_w$ is nonsingular.

If $\underline{\underline{S}}_w$ is singular, we can:

$$\lim_{\epsilon \to 0} (\underline{\underline{S}}_w + \epsilon \underline{\underline{I}})^{-1}.$$

× S_1

o S_2

Above FLD has 2 limitations:

(i) Always $D \to 1$ dimension.

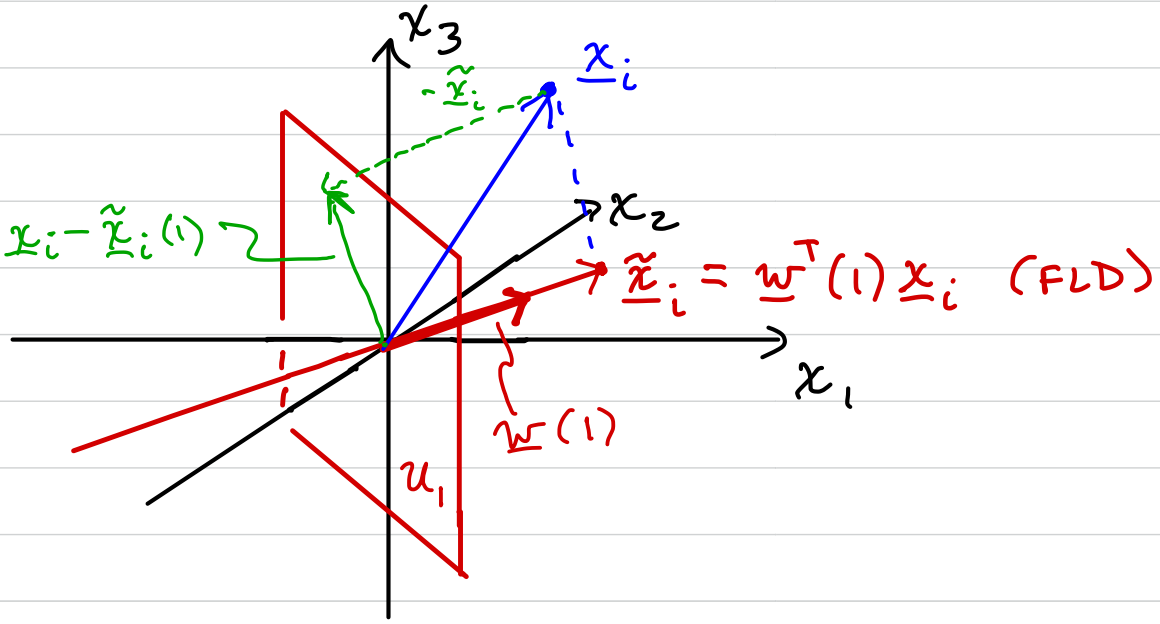
(ii) $C = 2$ only.

→ both limitations can be removed — see below.

FLD can be extended to d final dimensions $(1 < d < D)$

For example by:

1. Apply FLD for $D \to 1$; result is $\underline{w}(1)$
2. For $k = 1$ to $(d-1)$
3. Subtract $\tilde{\underline{x}}_i \triangleq \underline{w}^T(k)\,\underline{x}_i$ from each data point $\underline{x}_i$.



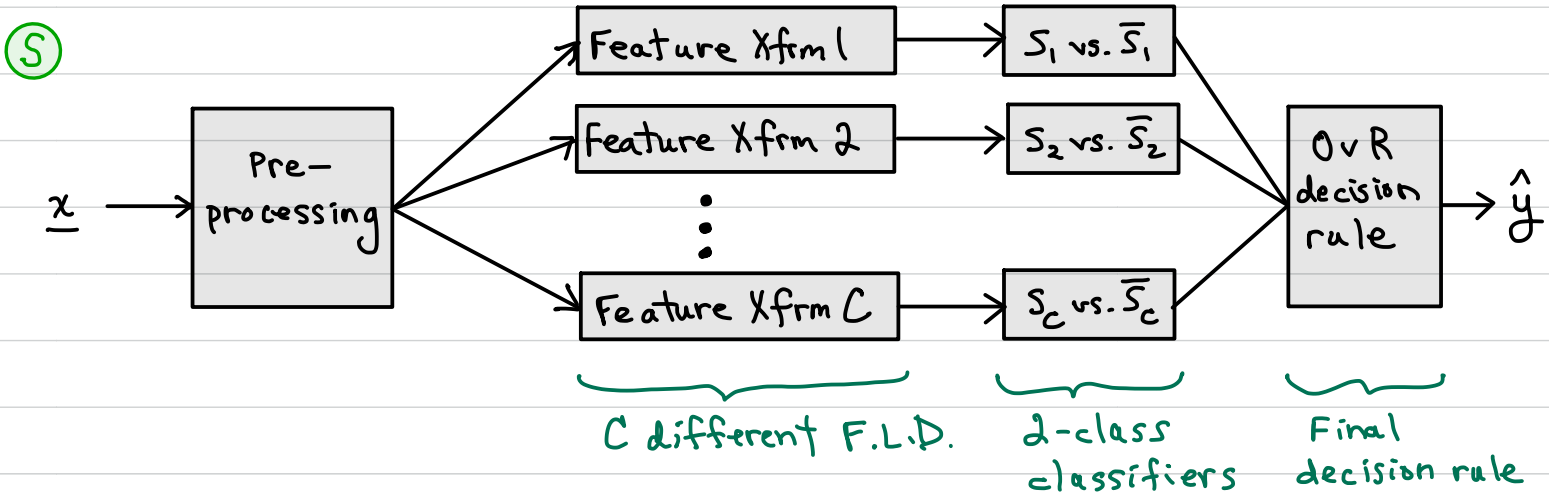
(3)

$\Rightarrow$ work next in subspace $\mathcal{U}_k$ orthogonal to $\underline{w}(k)$

4. Find FLD for $(D-k) \to 1$ dimension in new subspace $\mathcal{U}_k$; result is $\underline{w}(k+1)$. This $\underline{w}(k+1)$ becomes direction of $(k+1)^{\underline{th}}$ axis $\to$ $(k+1)^{\underline{th}}$ new feature.

For C>2

1. If using a set of 2-class classifiers (e.g., OvR, OvO), use FLD for each 2-class classifier, independently.
   Note that each 2-class problem could have a different feature set. For example:



OR

2. For any C>2 classifier, use:

Multiple Discriminant Analysis (MDA)

Generalization of FLD to C>2 classes.
for $D \geq C$
MDA will reduce dimensionality $D \rightarrow (C-1)$ (or sometimes less)
MDA optimizes a criterion J (generalization of $J(\tilde{w})$ for FLD).
Results are globally optimal (optimizes J globally)
For more information, see Bishop 4.1.6
(also Duda, Hart, & Stork 3.8.3)

Comments on PCA, FLD, MDA

1. PCA ignores class labels; FLD and MDA use class labels.

2. Algorithm (3) is a greedy algorithm.
   Each feature is chosen optimally from the remaining features, but end result isn't necessarily the best set of d features.

3. For FLD and MDA, if we add 2 assumptions:
   data from each class is normally distributed;
   all classes have the same covariance matrix;
   Then they are equivalent to linear discriminant analysis (LDA).
   (many references use these terms synonomously).

4. Any of these can be applied for nonlinear transformations, using a Φ basis-set expansion or kernel substitution (e.g., kernel PCA or kernel FLD).