Linear Methods for Classification

- Linear regression
- linear and quadatric discriminant functions
- example: gene expression arrays
- reduced rank LDA
- logistic regression
- separating hyperplanes

Linear classifiers

Some concepts:

- linear regression $f_k(x) = \beta_{k0} + \beta_k^T x$
- decision boundary between classes k and ℓ:

$$\{x : f_k(x) = f_\ell(x)\}$$

- linear discriminant analysis, logistic regression

$$\log \frac{P(G=1|X=x)}{P(G=2|X=x)} = \beta_0 + \beta^T x$$

- explicit approaches: separating hyperplanes
- discriminant functions:

$$\begin{aligned} \delta_k(x), &\qquad k = 1, 2, \ldots K \\ G(x) &\qquad = \text{argmin}\ \delta_k(x) \end{aligned}$$

Linear regression

Indicator response matrix

$$\mathbf{g} = \begin{pmatrix} 3 \\ 1 \\ 4 \\ \vdots \\ 2 \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \vdots & & & \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$\hat{\mathbf{F}} = \hat{\mathbf{Y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} = \mathbf{X}\hat{B}$$

$$\hat{f}(x) = \hat{B}^T x = \begin{pmatrix} \hat{f}_1(x) \\ \hat{f}_2(x) \\ \vdots \\ \hat{f}_K(x) \end{pmatrix} \quad \text{Note: } E(Y|X=x) \begin{pmatrix} p_1(x) \\ p_2(x) \\ \vdots \\ p_K(x) \end{pmatrix}$$

where $p_k(x) = P(G = k|X = x)$

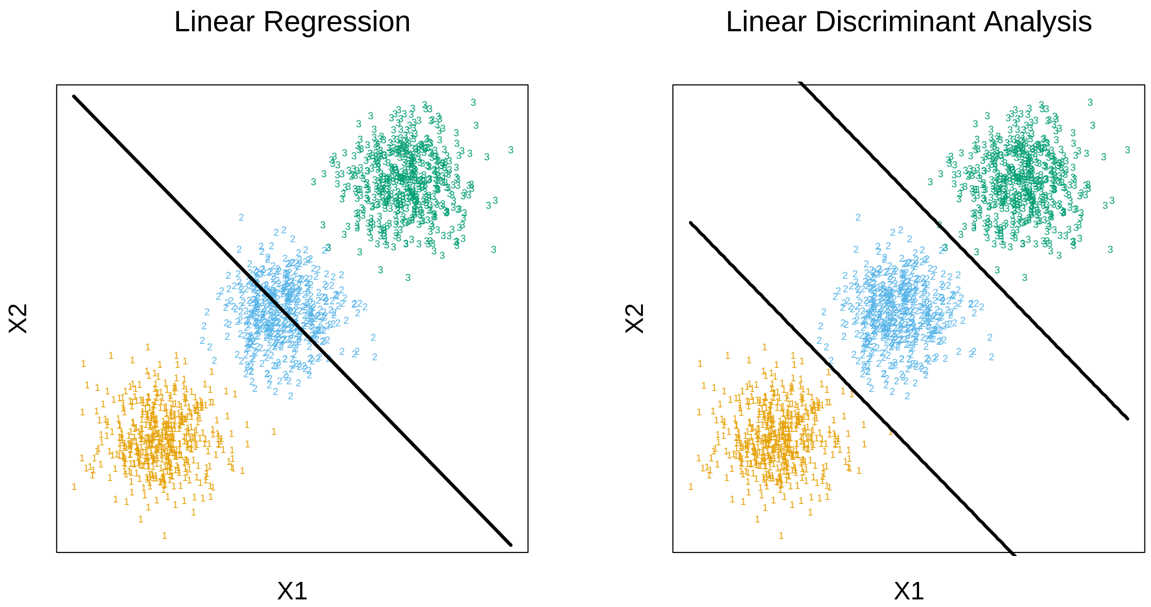
Targets:

$$\min_B \sum_{i=1}^{N} ||y_i - B^T x_i||^2,$$

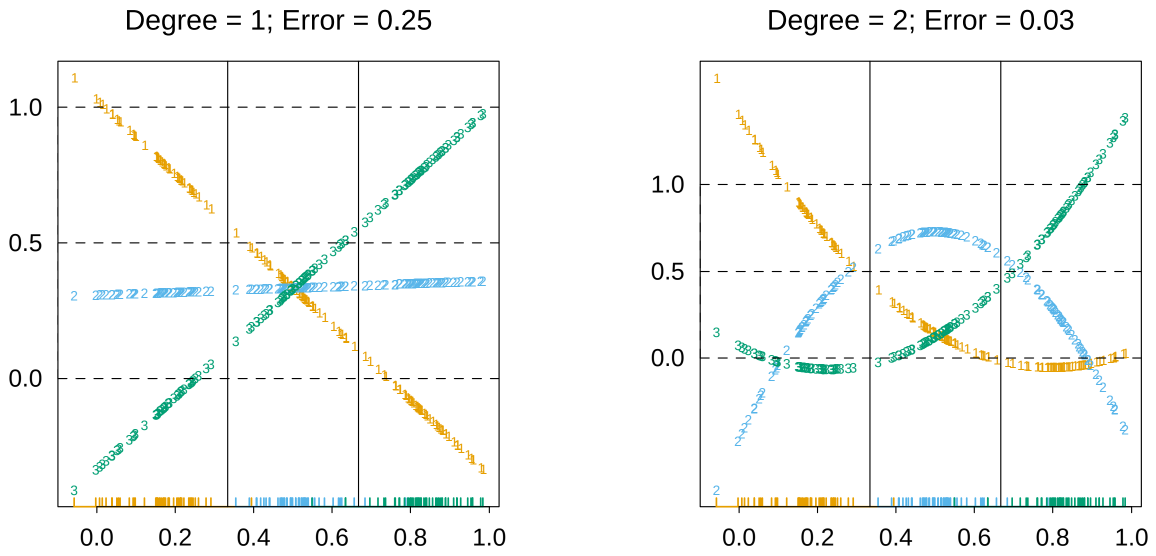
y_i, x_i are ith rows of $\mathbf{Y}$ and $\mathbf{X}$.

with $\hat{f}(x) = \hat{B}^T x$, $\hat{G}(x) = \text{argmin}_k ||\hat{f}(x) - t_k||^2$, where
$t_k = (0, 0, \ldots 0, 1, 0, \ldots)$ (1 in kth position).

Masking problems with linear regression



The data come from three classes in R^2 and are easily separated by linear decision boundaries. The right plot shows the boundaries found by linear discriminant analysis. The left plot shows the boundaries found by linear regression of the indicator response variables. The middle class is completely masked (never dominates).



The effects of masking on linear regression in R for a three-class problem. The *rug plot* at the base indicates the positions and class membership of each observation. The three curves in each panel are the fitted regressions to the three-class indicator variables; for example, for the green class, y_{green} is 1 for the green observations, and 0 for the orange and blue. The fits are linear and quadratic polynomials. Above each plot is the training error rate. The Bayes error rate is 0.025 for this problem, as is the LDA error rate.

Linear discriminant analysis

- $f_k(x)$ — density of X in class $G = k$
- π_k — class prior $Pr(G = k)$.
- Bayes theorem

$$Pr(G = k|X = x) = \frac{f_k(x)\pi_k}{\sum_{\ell=1}^{K} f_\ell(x)\pi_\ell}$$

- leads to LDA, QDA, MDA (mixture DA), Kernel DA, Naive Bayes
- LDA:

$$\begin{aligned} f_k(x) &= \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T\Sigma^{-1}(x-\mu_k)} \\ \log\frac{Pr(G = k|x)}{Pr(G = \ell|x)} &= \log\frac{\pi_k}{\pi_\ell} - \frac{1}{2}(\mu_k + \mu_\ell)^T\Sigma^{-1}(\mu_k - \mu_\ell) + x^T\Sigma^{-1}(\mu_k - \mu_\ell) \end{aligned}$$

More on LDA

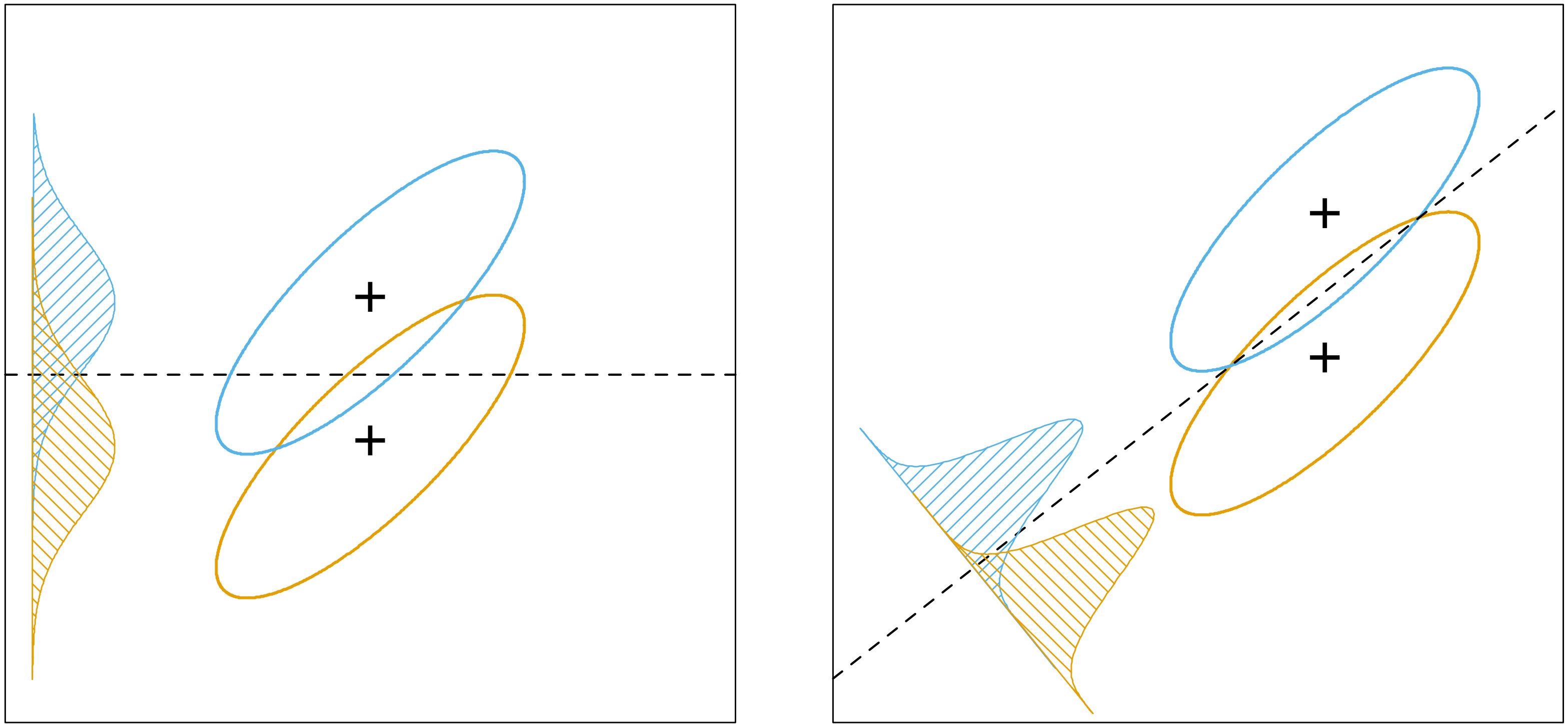
- estimate μ_k by centroid in class k, and Σ by pooled within class covariance matrix
- estimated Bayes rule: classify to class k that maximizes the discriminant function

$$\delta_k(x) = x^T\hat{\Sigma}^{-1}\hat{\mu}_k - \frac{1}{2}\hat{\mu}_k^T\hat{\Sigma}^{-1}\hat{\mu}_k + \log\hat{\pi}_k$$

- for two classes, we classify to class 2 if

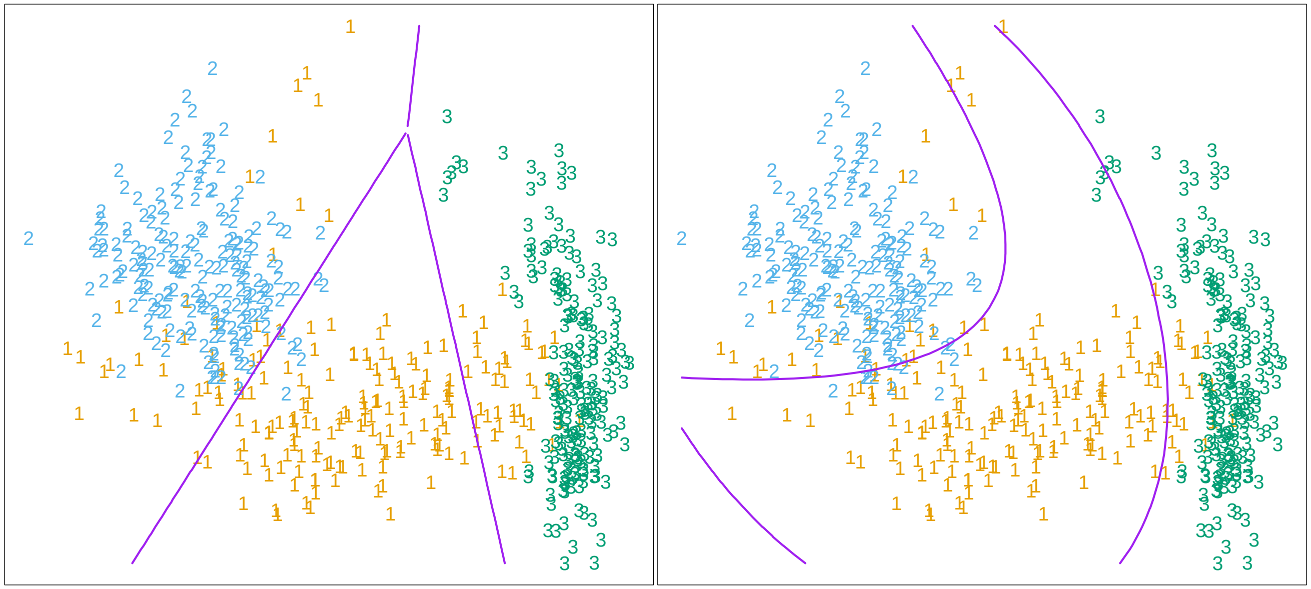
$$x^T\hat{\Sigma}^{-1}(\hat{\mu}_2 - \hat{\mu}_1) > \frac{1}{2}(\hat{\mu}_2 + \hat{\mu}_1)^T\hat{\Sigma}^{-1}(\hat{\mu}_2 - \hat{\mu}_1) - \log\frac{N_2}{N_1}$$

where N_1, N_2 are number of observations in each class.



Although the line joining the centroids defines the direction of greatest centroid spread, the projected data overlap because of the covariance (left panel). The discriminant direction minimizes this overlap for Gaussian data (right panel).

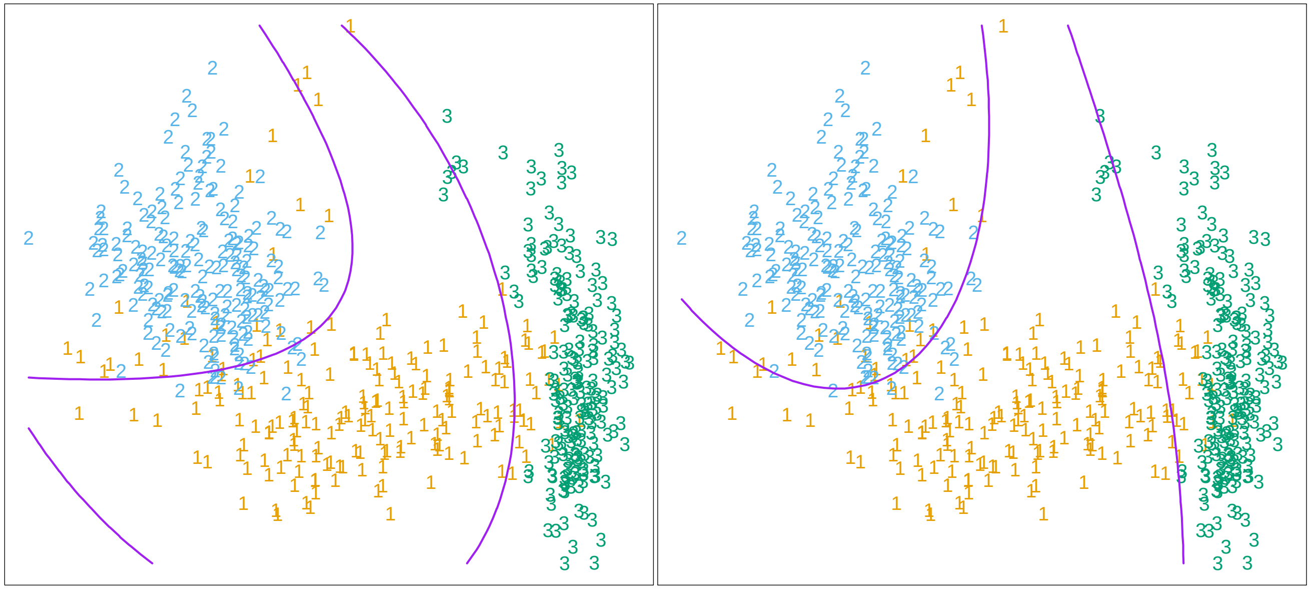
Linear boundaries and their projections



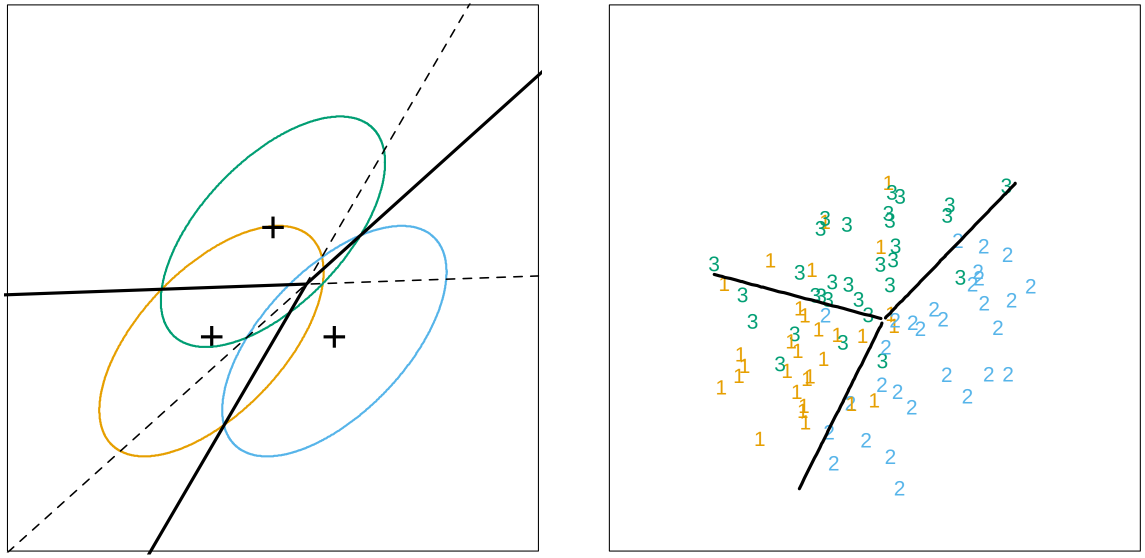
The left plot shows some data from three classes, with linear decision boundaries found by linear discriminant analysis. The right plot shows quadratic decision boundaries. These were obtained by finding linear boundaries in the five-dimensional space $X_1, X_2, X_1X_2, X_1^2, X_2^2$. Linear inequalities in this space are quadratic inequalities in the original space.

Quadratic discriminant analysis

$$\delta_k(x) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log\pi_k$$



Two methods for fitting quadratic boundaries. [Left] Quadratic decision boundaries, obtained using LDA in the five-dimensional “quadratic” space. [Right] Quadratic decision boundaries found by QDA. The differences are small, as is usually the case.



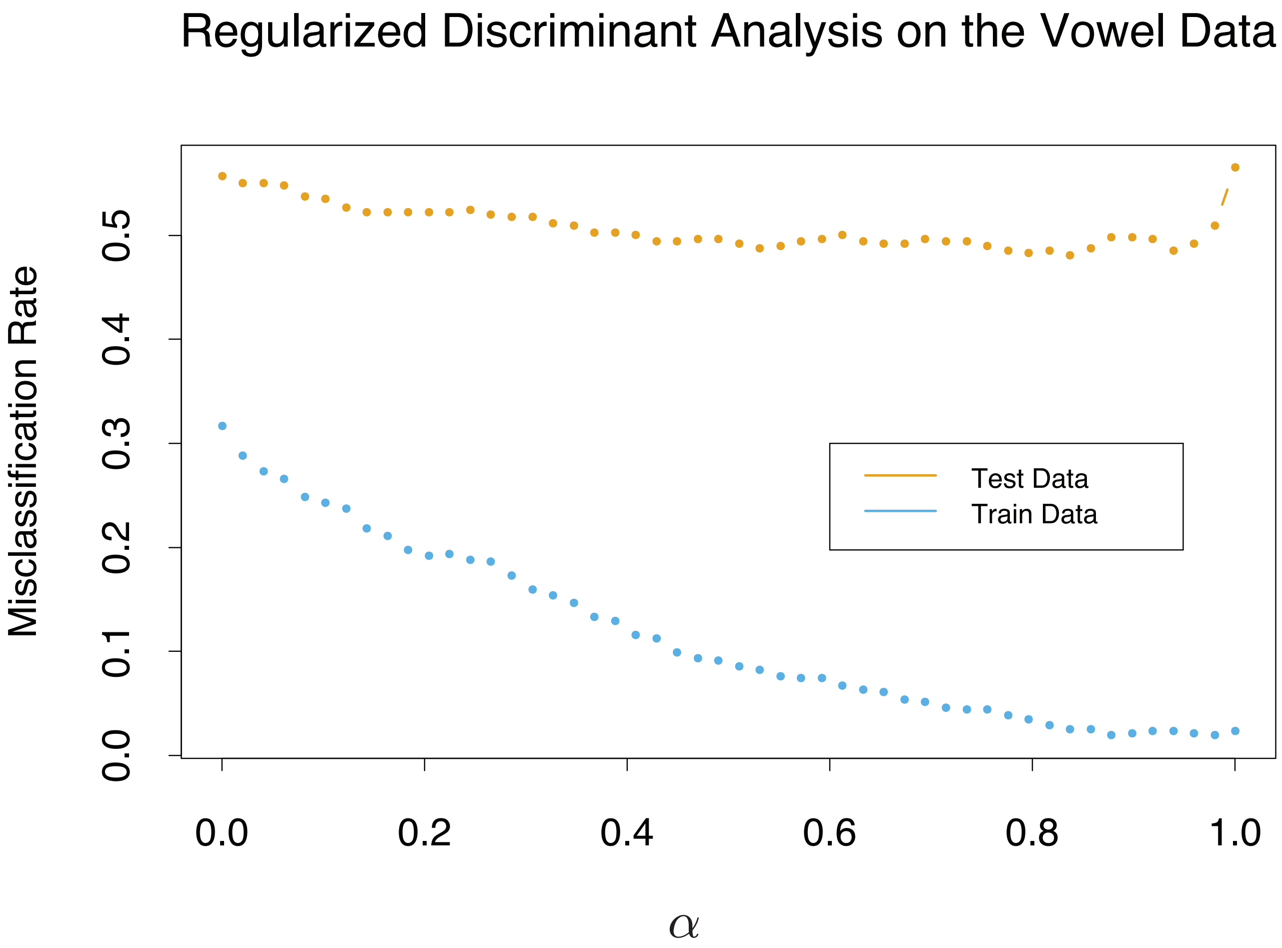
The left panel shows three Gaussian distributions, with the same covariance and different means. Included are the contours of constant density enclosing 95% of the probability in each case. The Bayes decision boundaries between each pair of classes are shown (broken straight lines), and the Bayes decision boundaries separating all three classes are the thicker solid lines (a subset of the former). On the right we see a sample of 30 drawn from each Gaussian distribution, and the fitted LDA decision boundaries.

Regularized discriminant analysis

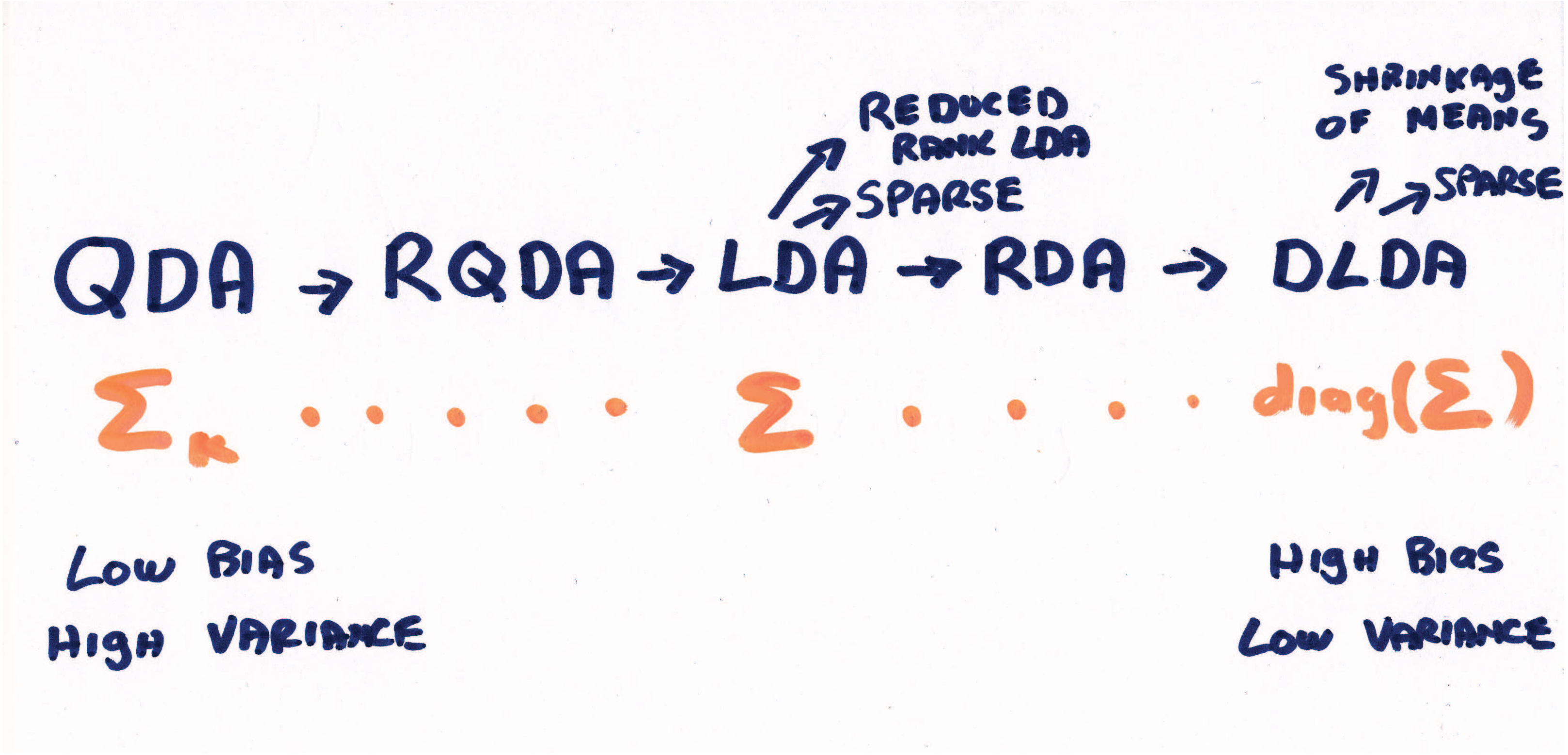
- Regularized QDA $\hat{\Sigma}_k(\alpha) = \alpha\hat{\Sigma}_k + (1-\alpha)\hat{\Sigma}$
- Regularized LDA $\hat{\Sigma}(\gamma) = \gamma\hat{\Sigma} + (1-\gamma)\hat{\sigma}^2 I$
- Together $\rightarrow \hat{\Sigma}(\alpha,\gamma)$
- could use $\hat{\Sigma}(\gamma) = \gamma\hat{\Sigma} + (1-\gamma)\mathrm{diag}(\hat{\Sigma})$
- in *“Nearest Shrunken Centroid”* work we use

$$\delta_k(x) = \sum_{j=1}^{p} \frac{(x_j - \hat{\mu}'_{jk})^2}{s_j^2} - \frac{1}{2}\log\pi_k$$

where $\hat{\mu}'_{jk}$ is a shrunken centroid. Details later.



Test and training errors for the vowel data, using regularized discriminant analysis with a series of values of $\alpha \in [0, 1]$. The optimum for the test data occurs around $\alpha = 0.9$, close to quadratic discriminant analysis.


REDUCED RANK LDA
SPARSE
SHRINKAGE OF MEANS
SPARSE
QDA → RQDA → LDA → RDA → DLDA
Σ_k . . . . . Σ . . . . . $\mathrm{diag}(\Sigma)$
LOW BIAS
HIGH VARIANCE
HIGH BIAS
LOW VARIANCE

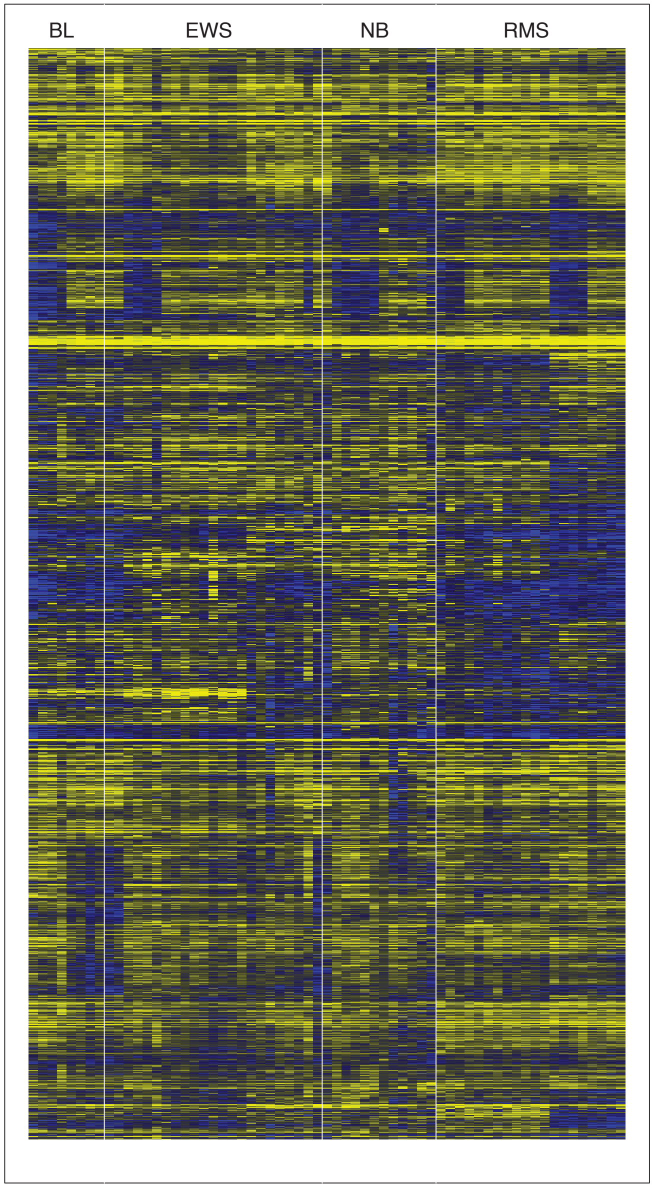
Classification in high dimensions

- important for gene expression microarray problems and other genomics problems
- Starting point: diagonal LDA which uses $\mathrm{diag}(\hat{\Sigma})$
- nearest centroid classification on standardized features is equivalent to diagonal LDA
- nearest shrunken centroids regularizes further, by discarding noisy features

Classification of microarray samples

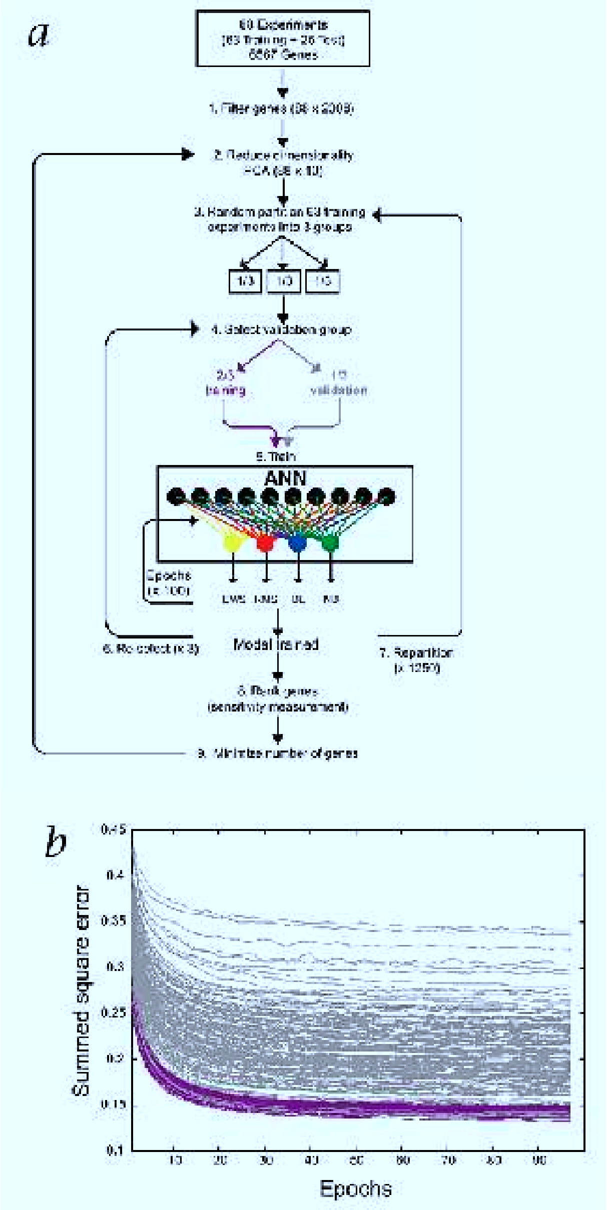
Example: small round blue cell tumors; Khan *et al*, Nature Medicine, 2001

- Tumors classified as BL (Burkitt lymphoma), EWS (Ewing), NB (neuroblastoma) and RMS (rhabdomyosarcoma).
- There are 63 training samples and 25 test samples, although five of the latter were not SRBCTs. 2308 genes
- Khan *et al* report zero training and test errors, using a complex neural network model. Decided that 96 genes were "important".
- Too complicated!

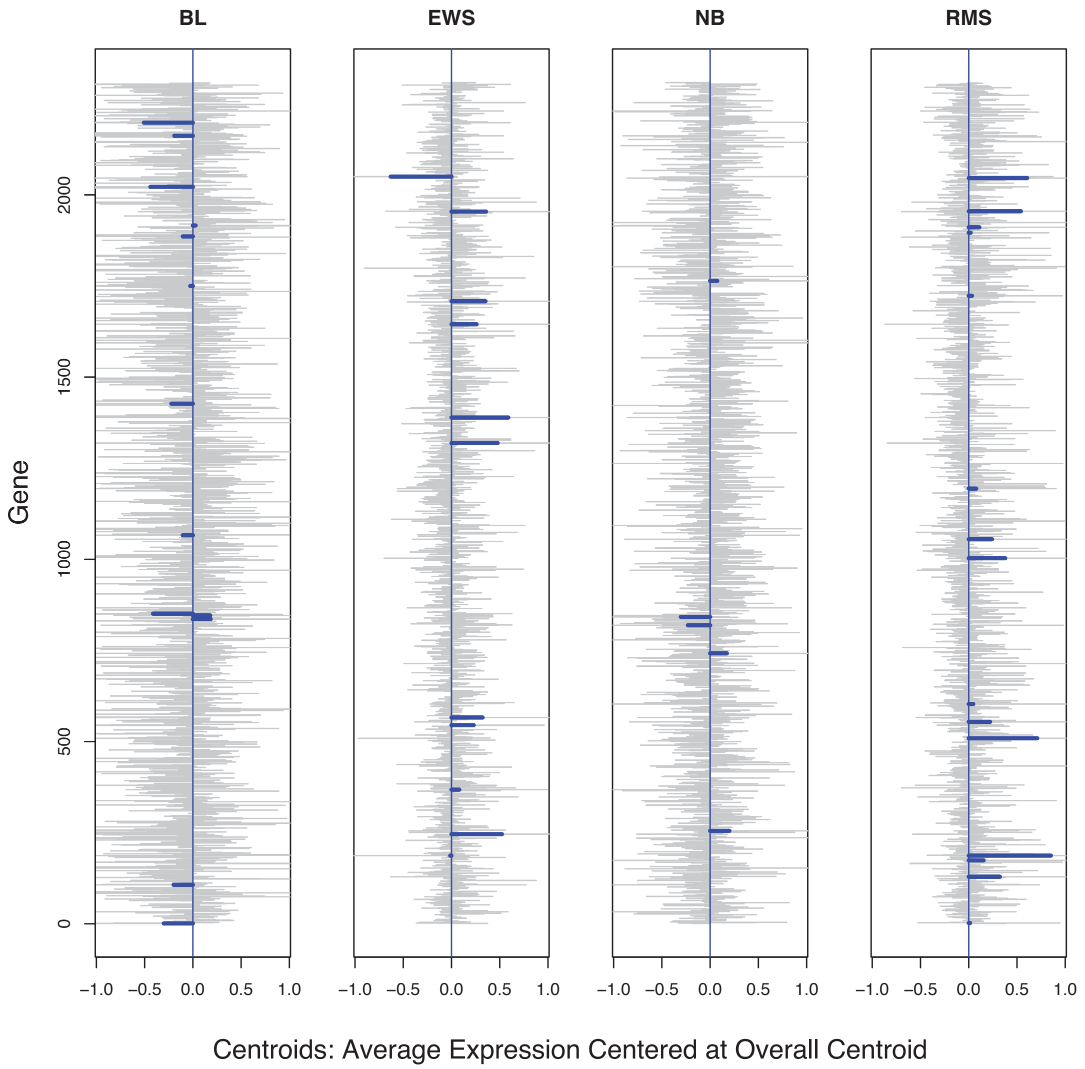


Khan data

Neural network approach



Class centroids



Shrunken centroids

- Idea: shrink each class centroid towards the overall centroid. First normalize by the within-class standard deviation for each gene.
- Let x_{ij} be the expression for samples $i = 1, 2, \ldots n$ and genes $j = 1, 2, \ldots p$.
- We have classes $1, 2, \ldots K$, and let C_k be indices of the n_k samples in class k.
- The jth component of the centroid for class k is $\bar{x}_{jk} = \sum_{i \in C_k} x_{ij}/n_k$, the mean expression value in class k for gene j; the jth component of the overall centroid is $\bar{x}_j = \sum_{i=1}^{n} x_{ij}/n$.

- Let

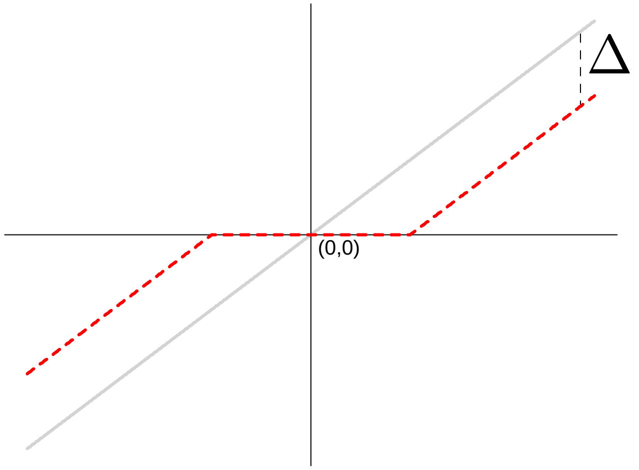
$$d_{jk} = (\bar{x}_{jk} - \bar{x}_j)/s_j, \tag{1}$$

where s_j is the pooled within class standard deviation for gene j:

$$s_j^2 = \frac{1}{n-K}\sum_k \sum_{i \in C_k} (x_{ij} - \bar{x}_{jk})^2. \tag{2}$$

- Shrink each d_{jk} towards zero, giving d'_{jk} and new shrunken centroids or prototypes

$$\bar{x}'_{jk} = \bar{x}_j + s_j d'_{jk} \tag{3}$$



- The shrinkage is *soft-thresholding*: each d_{jk} is reduced by an amount Δ in absolute value, and is set to zero if its absolute value is less than zero. Algebraically, this is expressed as

$$d'_{jk} = \text{sign}(d_{jk})(|d_{jk}| - \Delta)_+ \tag{4}$$

  where $+$ means *positive part* $(t_+ = t$ if $t > 0$, and zero otherwise).

- Choose Δ by cross-validation.

Advantages

- Simple, includes nearest centroid classifier as a special case.
- Thresholding denoises large effects, and sets small ones to zero- thereby selecting genes
- with more than two classes, method can select different genes, and different numbers of genes for each class.

Class probabilities

- For a test sample $x^* = (x_1^*, x_2^*, \ldots x_p^*)$, we define the *discriminant score* for class k

$$\delta_k(x^*) = \sum_{j=1}^{p} \frac{(x_j^* - \bar{x}'_{jk})^2}{s_j^2} - 2\log\pi_k \qquad (5)$$

- The classification rule is then

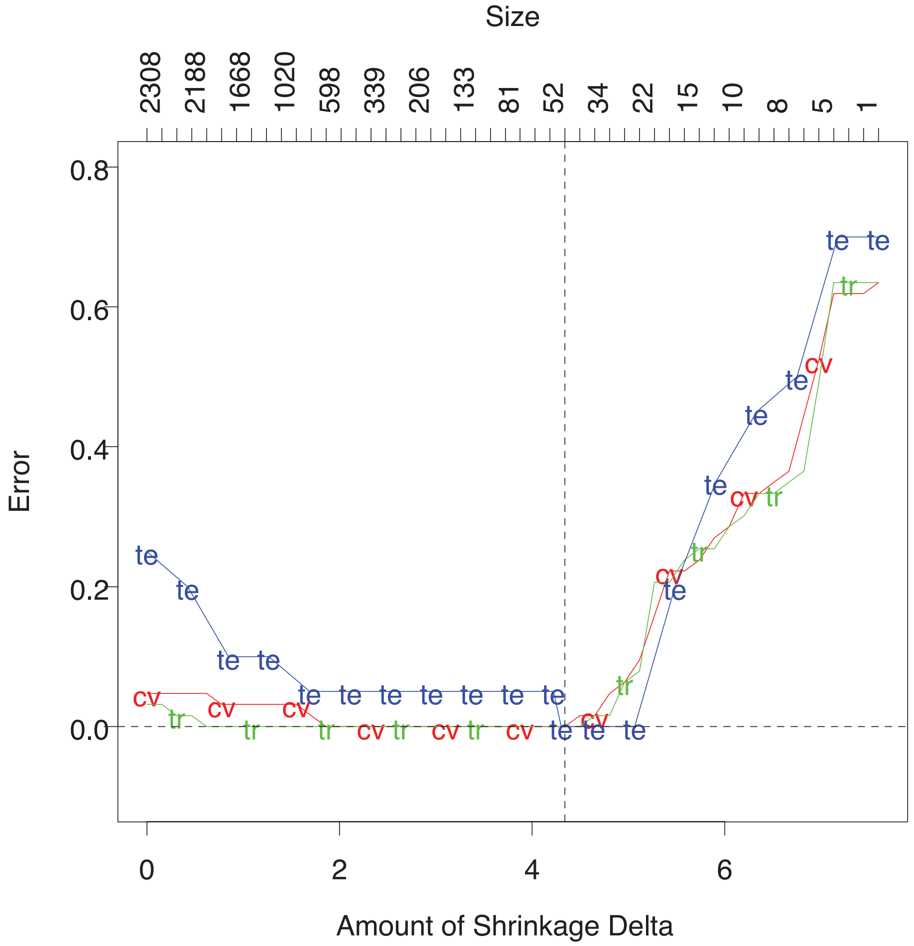
$$C(x^*) = \ell \text{ if } \delta_\ell(x^*) = \min_k \delta_k(x^*) \qquad (6)$$

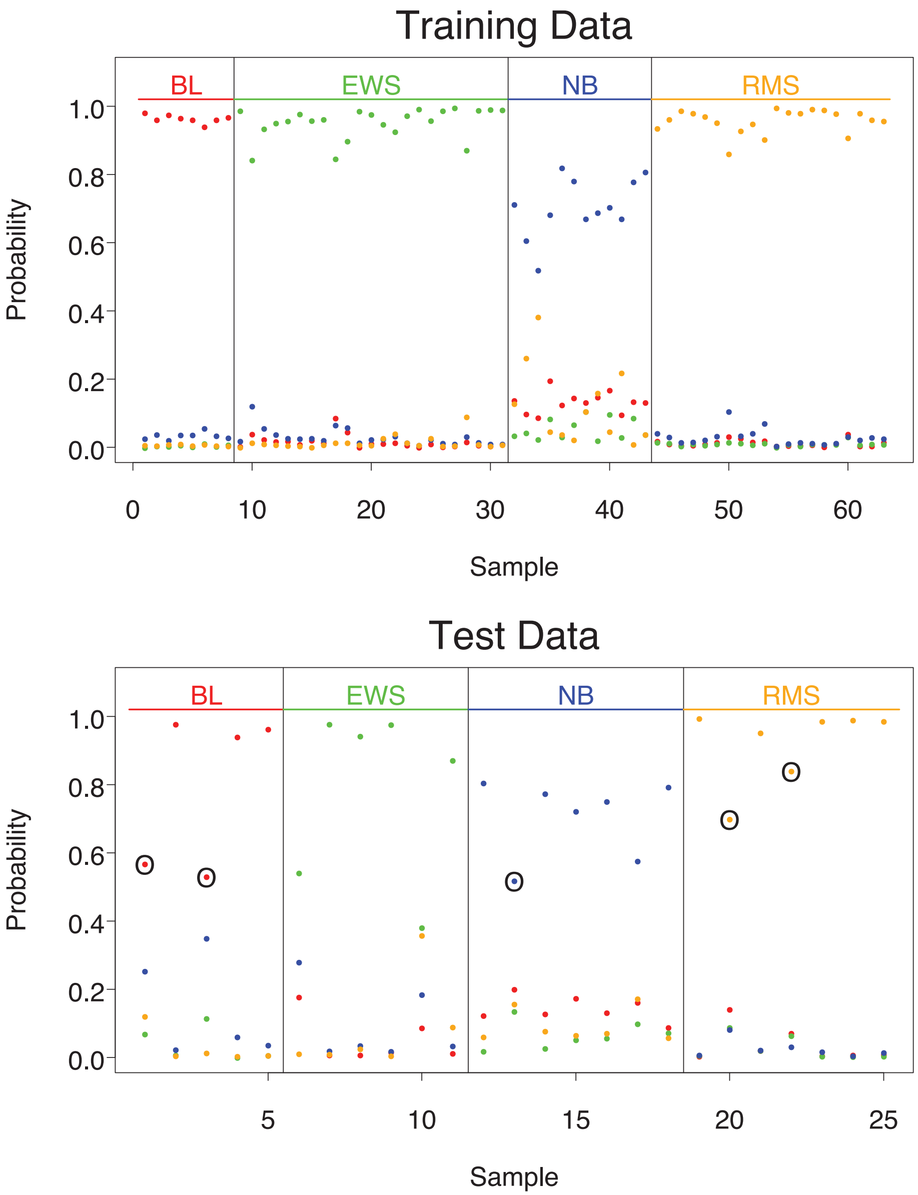
- estimates of the class probabilities, by analogy to Gaussian linear discriminant analysis, are

$$\hat{p}_k(x^*) = \frac{e^{-\frac{1}{2}\delta_k(x^*)}}{\sum_{\ell=1}^{K} e^{-\frac{1}{2}\delta_\ell(x^*)}} \qquad (7)$$

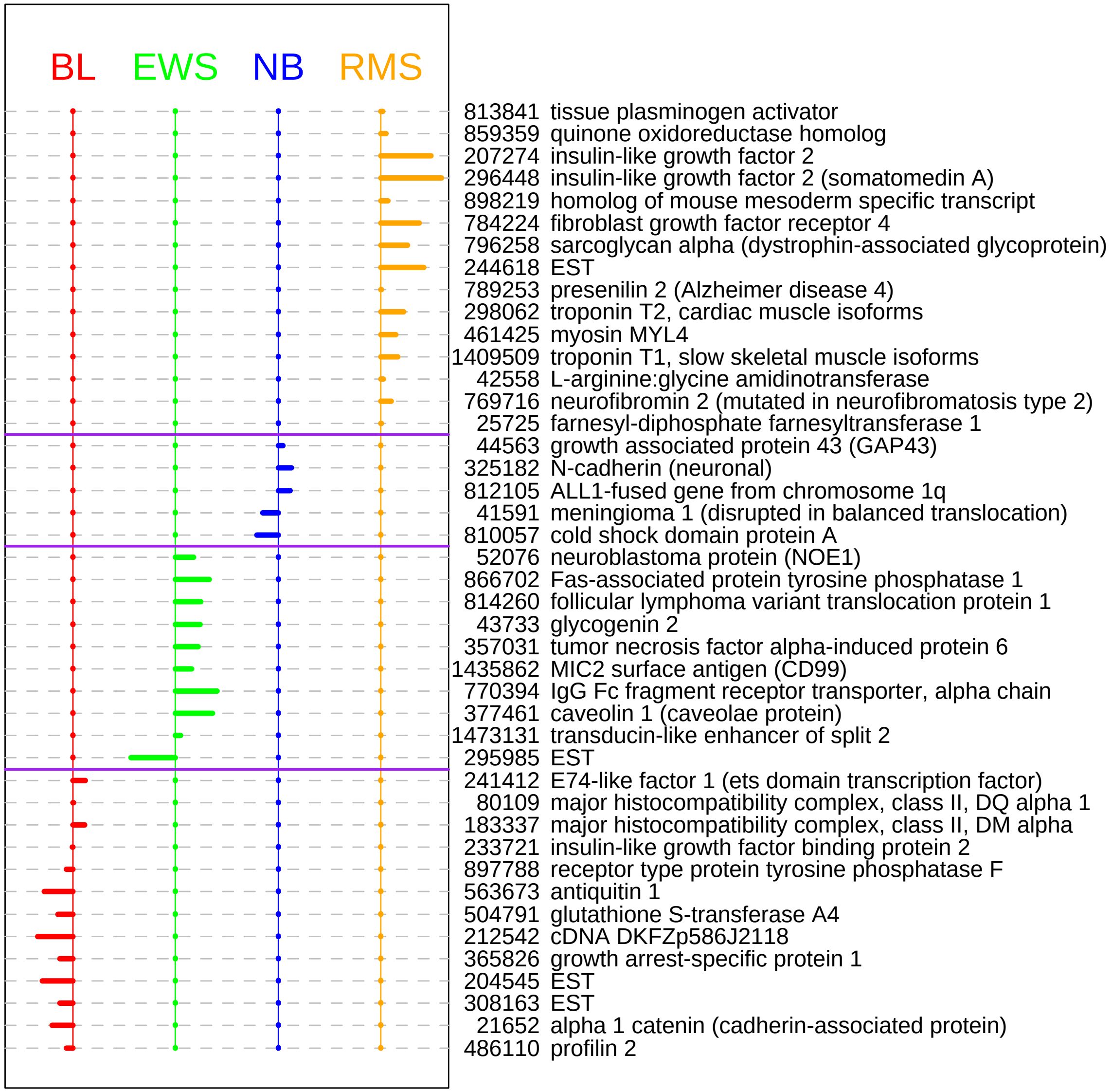
Results on Khan data

At optimal point, there are 43 active genes

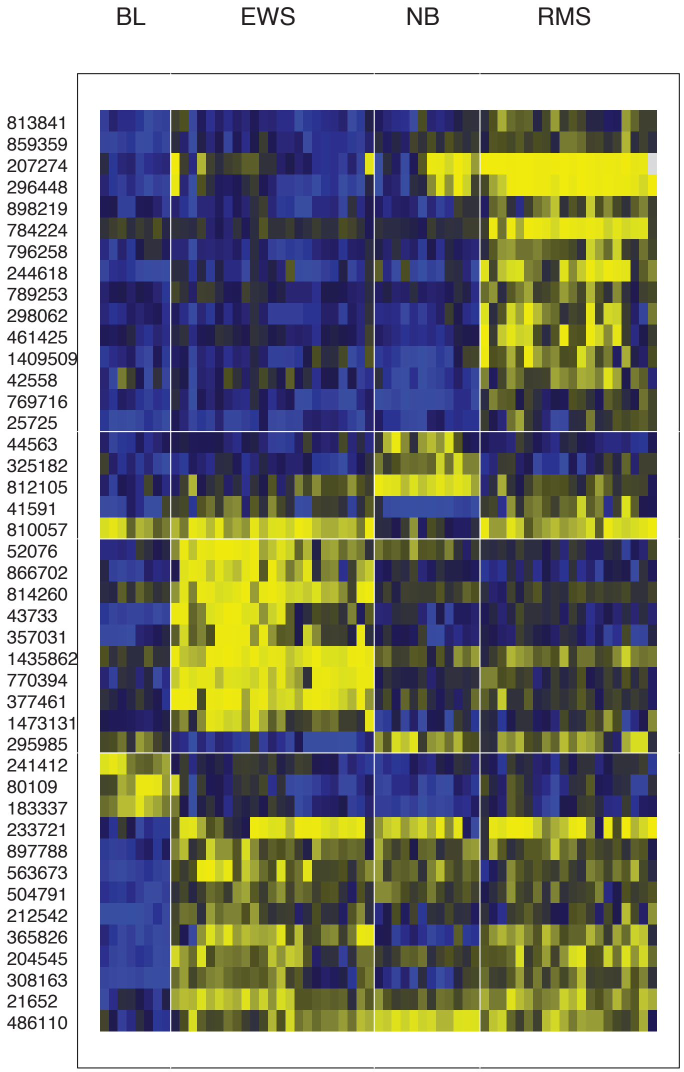



Training Data
BL
EWS
NB
RMS
Probability
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50
60
Sample
Test Data
BL
EWS
NB
RMS
Probability
0.0
0.2
0.4
0.6
0.8
1.0
5
10
15
20
25
Sample

The genes that matter



Heatmap of selected genes

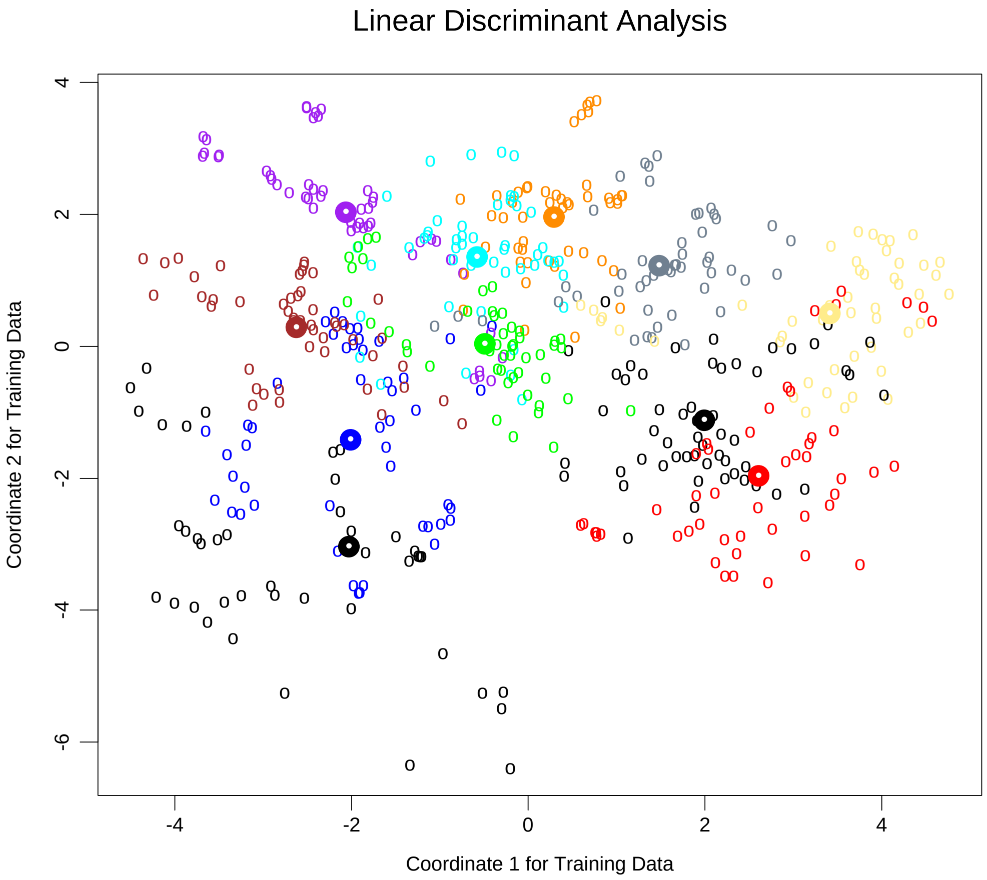


Reduced rank LDA

- let $\hat{\Sigma} = UDU^T$ (eigendecomposition)
- let $x^* = D^{-1/2}U^T x = \hat{\Sigma}^{-1/2}x$
- $\hat{\mu}_k^* = \hat{\Sigma}^{-1/2}\hat{\mu}_k$
- LDA: $\delta_k(x) = (1/2)||x^* - \hat{\mu}_k^*||^2 - \log\hat{\pi}_k$ (closest centroid in sphered space, with a correction for class size)
- hence if $p > K - 1$, can project data onto $K - 1$ dim space spanned by $\hat{\mu}_k^*$ and lose nothing

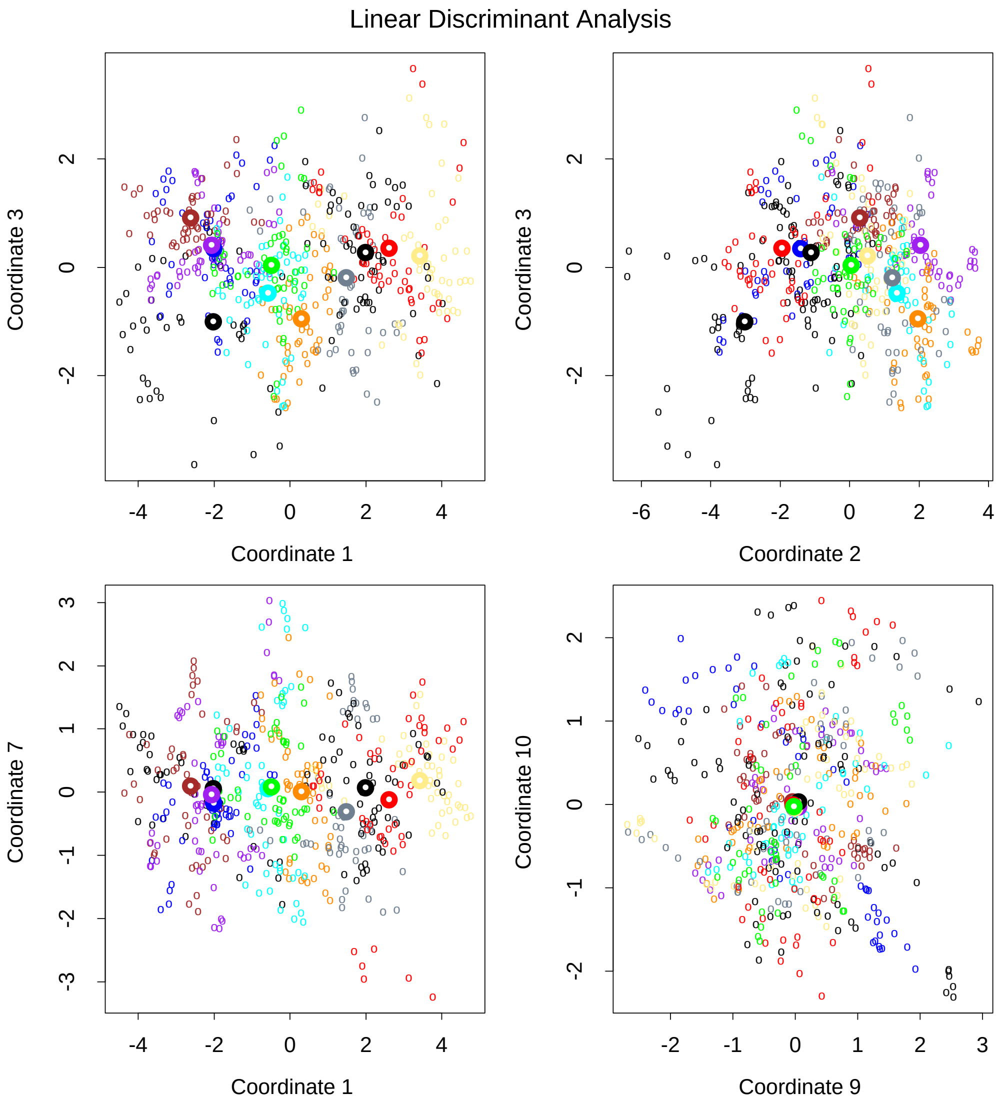
Can project onto even lower dimensions, using the principal components of $\hat{\mu}_k^*$:

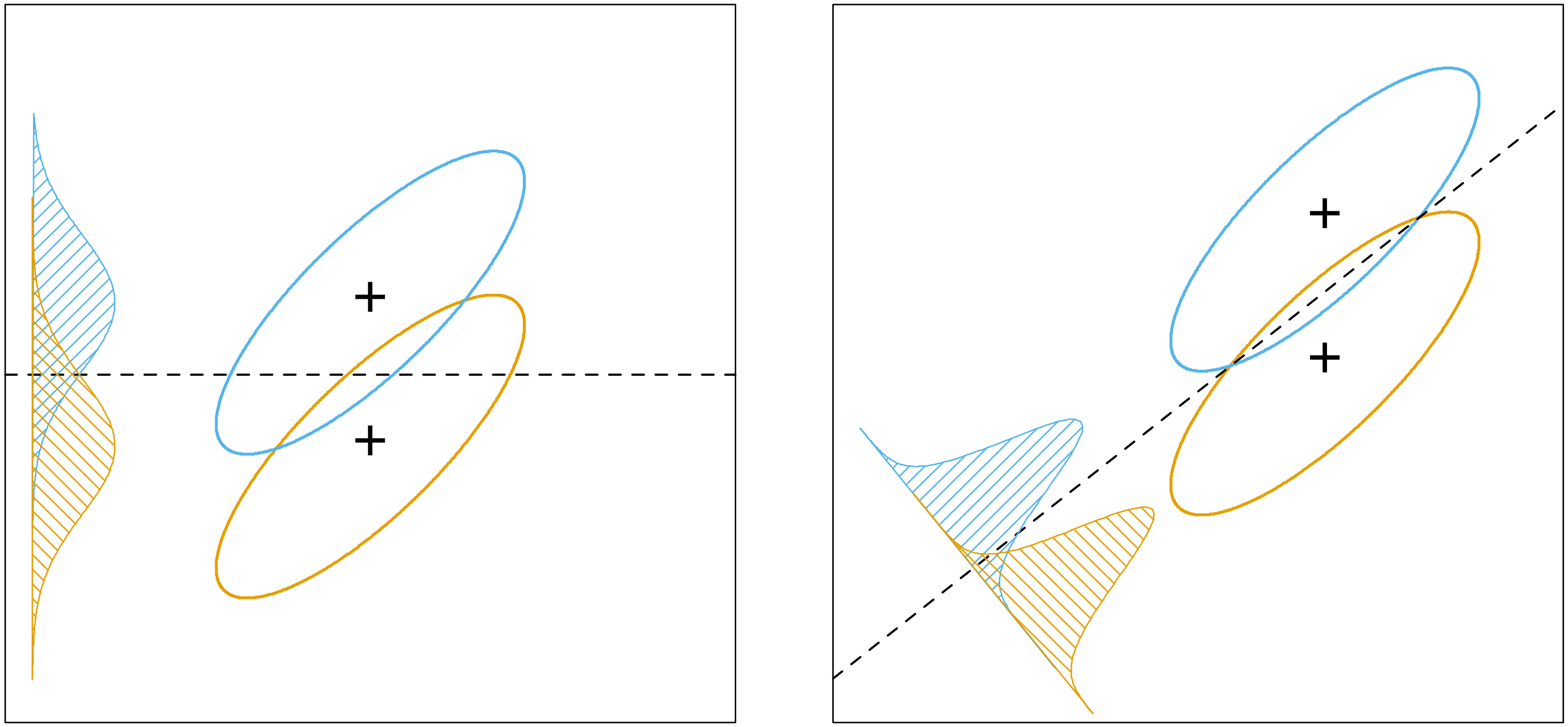
- compute $M = K \times p$ matrix of centroids, $\hat{\Sigma}$, $M^* = M\hat{\Sigma}^{-1/2}$
- Reduce M^* by principal components; i.e. compute B^* = covariance matrix of M^* (with mass π_k on each row), $B^* = V^* D_B V^{*T}$
- $z_\ell = v_\ell^T x$ is the ℓth discriminant (or canonical) variable, with $v_\ell = \hat{\Sigma}^{-1/2} v_\ell^*$



A two-dimensional plot of the vowel training data. There are eleven classes with $X \in R^{10}$, and this is the best view in terms of a LDA model. The heavy circles are the projected mean vectors for each class. The class overlap is considerable.

Projections onto pairs of discriminant variates





Although the line joining the centroids defines the direction of greatest centroid spread, the projected data overlap because of the covariance (left panel). The discriminant direction minimizes this overlap for Gaussian data (right panel).

Fisher’s formulation of discriminant analysis

- Find $z = a^T x$ such that the between-class variance is maximized relative to within-class variance $W = \hat{\Sigma}$:

$$\max_{a \in \mathbb{R}^p} \frac{a^T B a}{a^T W a}$$

or

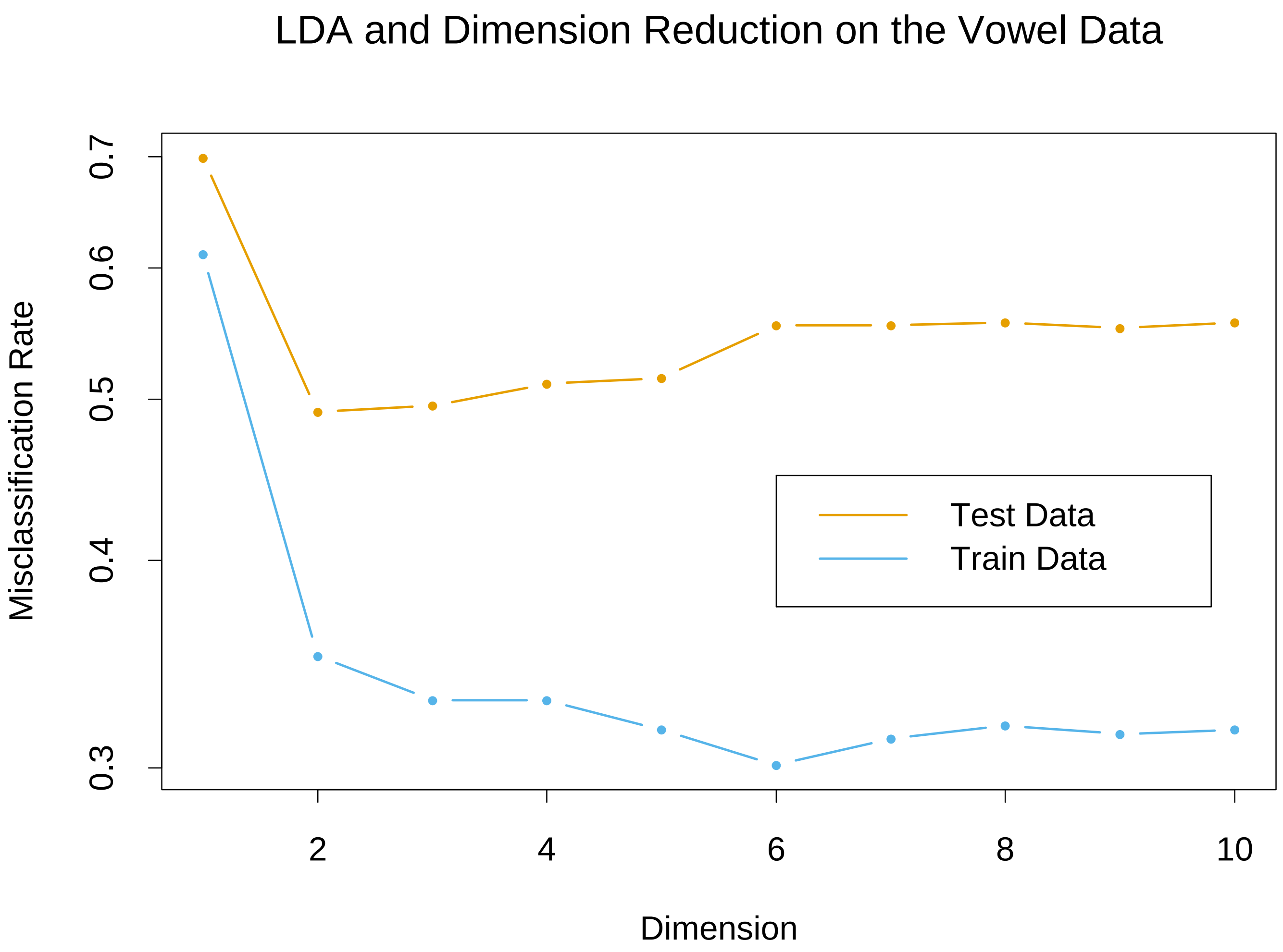
$$\max_{a \in \mathbb{R}^p} a^T B a \text{ subject to } a^T W a = 1$$

- gives $v_1 = a$; then find next direction orthogonal to first

$$\max_{a \in \mathbb{R}^p} a^T B a \text{ subject to } a^T W a = 1,\; a^T W v_1 = 0$$

- gives $v_2 = a$ etc

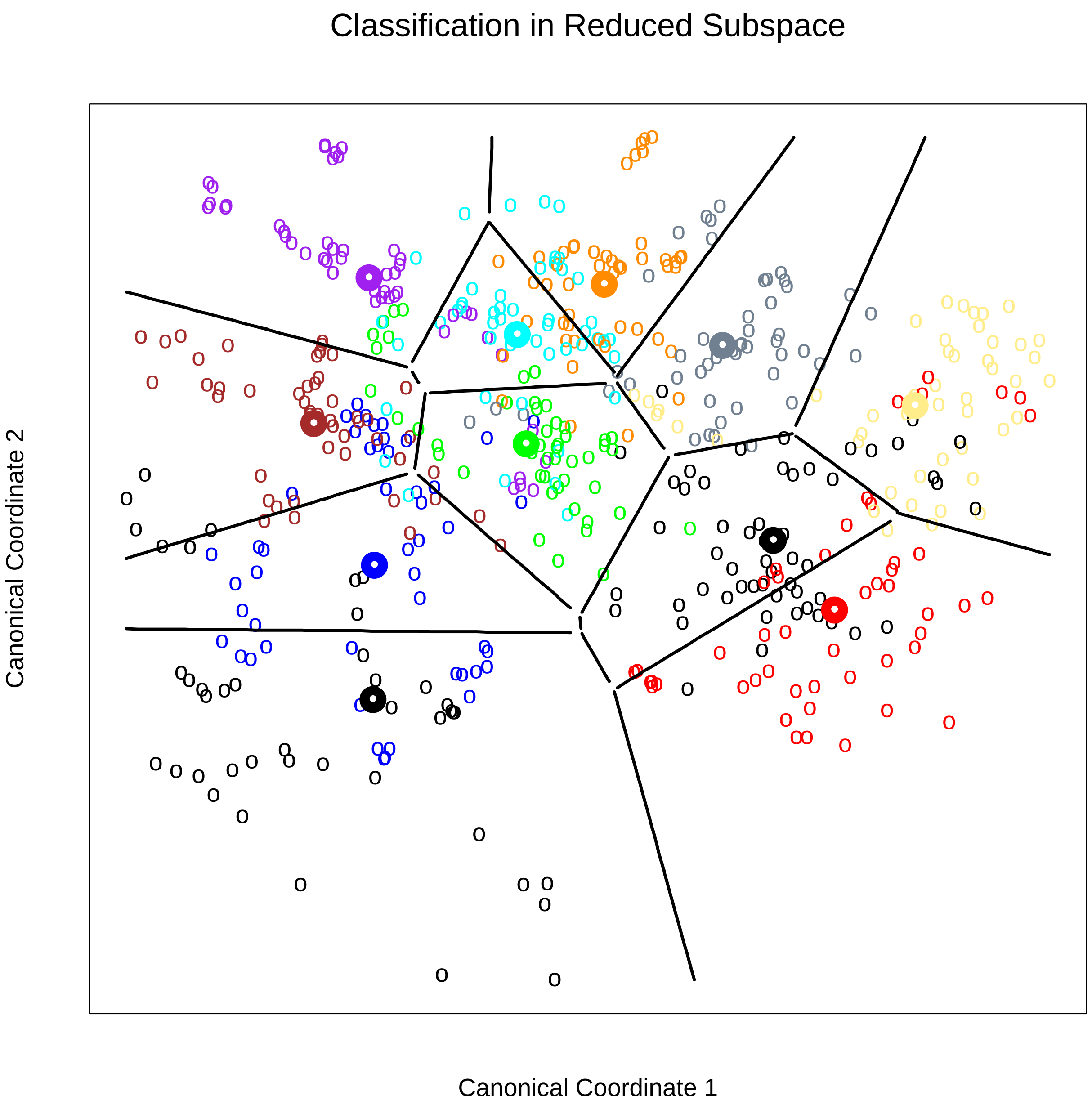
This equivalent to the PCA of standardized centroids of earlier slide.



Training and test error rates for the vowel data, as a function of the dimension of the discriminant subspace. In this case the best error rate is for dimension 2.

Performance on Vowel Data

	Train	Test
Linear Regression	0.48	0.67
LDA	0.32	0.56
Reduced Rank LDA	0.36	0.50
QDA	0.01	0.53
Logistic Regression	0.22	0.51


Classification in Reduced Subspace
Canonical Coordinate 2
Canonical Coordinate 1

Linear Logistic Regression

Two-class case. $Y = 0/1$ codes the classes. Model $p(x) = Pr(Y = 1|x)$.

$$\text{logit}p(x) \equiv \log \frac{p(x)}{1 - p(x)} = \beta^T x, \quad p(x) = \frac{e^{\beta^T x}}{1 + e^{\beta^T x}}$$

$$\text{Log-Likelihood} = \sum_{i=1}^{n} \{y_i \log p_i + (1 - y_i) \log(1 - p_i)\}$$

IRLS algorithm

1. Initialize β.
2. Form linearized responses $z_i = \beta^T x_i + (y_i - p_i)/\{p_i(1 - p_i)\}$
3. Form weights $w_i = p_i(1 - p_i)$
4. Update β by weighted LS of z_i on x_i with weights w_i.

Steps 2-4 are repeated until convergence.

Properties of logistic regression solutions

- Satisfy *score equations* $\mathbf{X}^T(\mathbf{y}-\hat{\mathbf{p}}) = 0$.
- If $\mathbf{W}$ is a diagonal matrix with weights $w_i = \hat{p}_i(1-\hat{p}_i)$, then the *asymptotic* covariance matrix of $\hat{\beta}$ is

$$\text{cov}(\hat{\beta}) = (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}$$

- If the two classes are *linearly separable*, then the solution is undefined! [MLE tries to achieve probabilities that are 0 and 1, and for this some of $\hat{\beta}$ must go to $\pm\infty$].
- Inference proceeds in a manner very similar to that for linear regression.

IRLS $\equiv$ Newton algorithm

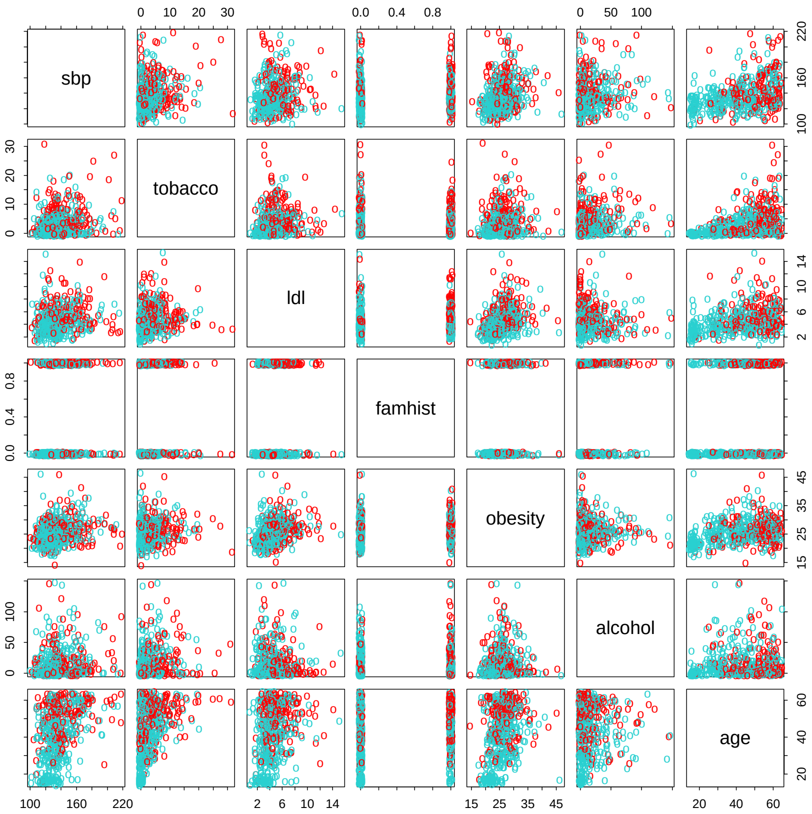
$$\begin{aligned}\frac{\partial \ell(\beta)}{\partial \beta} &= \mathbf{X}^T(\mathbf{y}-\mathbf{p})\\ \frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} &= -\mathbf{X}^T\mathbf{W}\mathbf{X}.\end{aligned}$$

A Newton step is thus

$$\begin{aligned}\beta^{new} &= \beta^{old} + (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{y}-\mathbf{p})\\ &= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\left(\mathbf{X}\beta^{old} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p})\right)\\ &= (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}.\end{aligned}$$

In the second and third line we have re-expressed the Newton-Raphson step as a weighted least squares step, with the response

$$\mathbf{z} = \mathbf{X}\beta^{old} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p}).$$



A scatterplot matrix of the **South African heart disease** data. Each plot shows a pair of risk factors, and the cases (160) and controls (302) are color coded (red is a case). The variable famhist *(family history of heart disease)* is binary (yes or no).

Results from a logistic regression fit to the South African heart disease data.

	Coefficient	Standard Error	Z Score
(Intercept)	-4.130	0.964	-4.285
sbp	0.006	0.006	1.023
tobacco	0.080	0.026	3.034
ldl	0.185	0.057	3.219
famhist	0.939	0.225	4.178
obesity	-0.035	0.029	-1.187
alcohol	0.001	0.004	0.136
age	0.043	0.010	4.184

Model building

- Deviance: $\mathrm{dev}(y, \hat{p}) = -2\ell(\hat{\beta})$
- H_0: First q components of β are non-zero
- H_1: β is unrestricted
- under H_0, $\mathrm{dev}(y, \hat{p}_0) - \mathrm{dev}(y, \hat{p}_1) \sim \chi^2_{p-q}$ asymptotically (as $N \to \infty$)
- *"Chi-square statistic"*—quadratic approximation to deviance:

$$\sum_{i=1}^{n} w_i(z_i - x_i^T\hat{\beta}) = \sum_{i=1}^{n} \frac{(y_i - \hat{p}_i)^2}{\hat{p}_i(1-\hat{p}_i)}$$

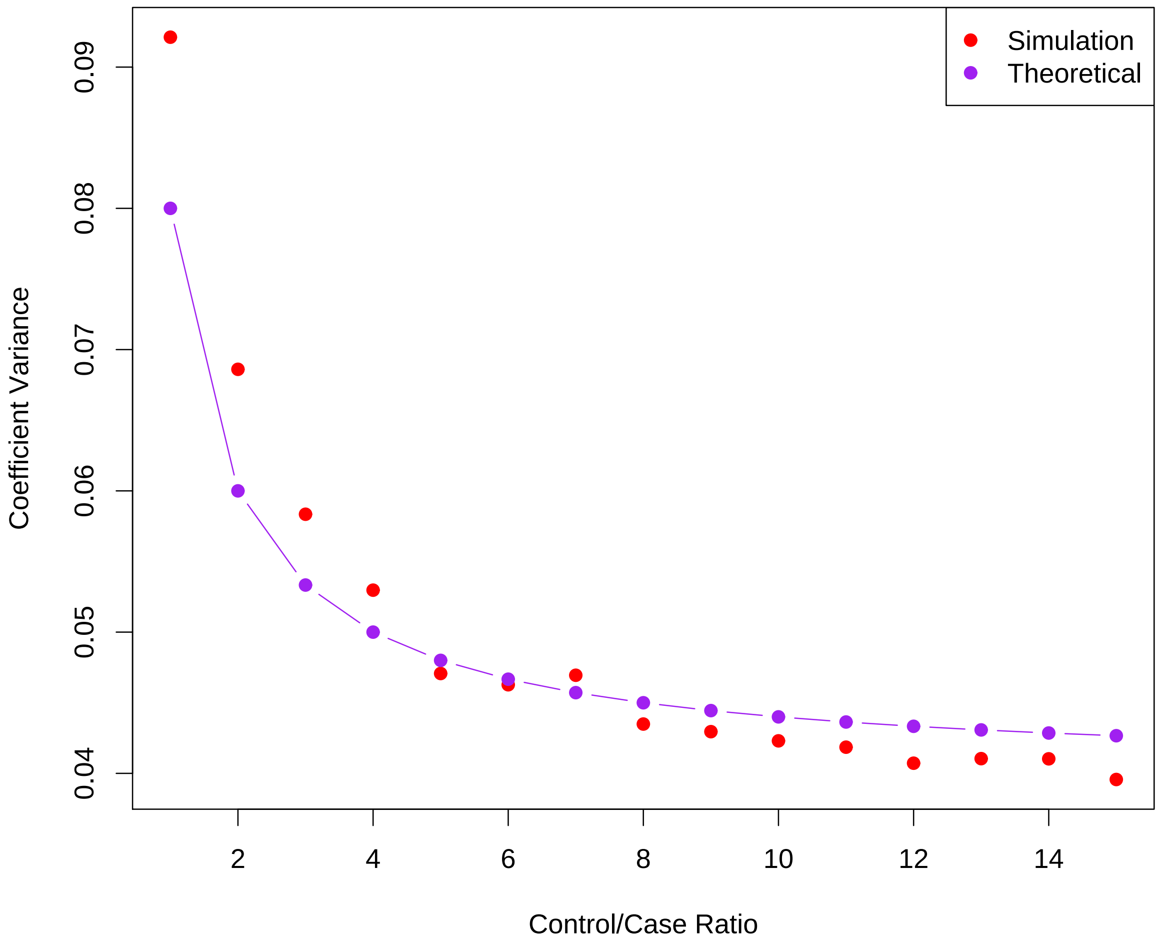
- $\hat{\beta} \sim N(\beta, (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1})$ asymptotically, if the model is correct.

Case-control sampling and logistic regression

- In South African data, there are 160 cases, 302 controls — $\tilde{\pi} = 0.35$ are cases. Yet the prevalence of MI in this region is $\pi = 0.05$.
- With case-control samples, we can estimate the regression parameters β_j accurately; the constant term β_0 is incorrect.
- We can correct the estimated intercept by a simple transformation

$$\hat{\beta}_0^* = \hat{\beta}_0 + \log \frac{\pi}{1-\pi} - \log \frac{\tilde{\pi}}{1-\tilde{\pi}}$$

- Often cases are rare and we take them all; up to five times that number of controls is sufficient. See next slide



Sampling more controls than cases reduces the variance of the parameter estimates. But after a ratio of about 5 to 1 the variance reduction flattens out.

Risk estimates and classification

- We can estimate the risk for a new observation x_0 via $\hat{\eta}(x_0) = x_0^T\hat{\beta}$ and $\hat{\mathrm{Pr}}(Y = 1|X = x_0) = e^{\hat{\eta}(x_0)}/(1 + e^{\hat{\eta}(x_0)})$.
- To obtain a 95% confidence interval for $\mathrm{Pr}(Y = 1|X = x_0)$, we first obtain one for $\eta(x_0)$ (using the estimated covariance of $\hat{\beta}$). We then apply the sigmoid transformation to the lower and upper values.
- To classify a new observation, we threshold $\hat{\mathrm{Pr}}(Y = 1|X = x_0)$ at 0.5. Other thresholds change the *sensitivity* and *specificity*, and are used to construct ROC curves.

Multiple Logistic Regression

Model is defined in terms of $J-1$ logits $\eta_j(X) = \beta_j^T X$:

$$\begin{aligned}
\log \frac{P(G=1|X)}{P(G=J|X)} &= \eta_1(X) \\
\log \frac{P(G=2|X)}{P(G=J|X)} &= \eta_2(X) \\
&\vdots \\
\log \frac{P(G=J-1|X)}{P(G=J|X)} &= \eta_{J-1}(X)
\end{aligned}$$

$$P(G=j|X) = \frac{e^{\eta_j(X)}}{1+\sum_{\ell=1}^{J-1} e^{\eta_\ell(X)}}$$

Fit by least squares or multinomial maximum likelihood.

Logistic Regression with p$\gg$N

- Typically linear models are sufficient — logit$(p_i) = \beta^T x_i$
- Models *have* to be regularized
  - *ridge penalty* — similar to SVM

$$\text{PLL} = \sum_{i=1}^{N}\{y_i \log p_i + (1-y_i)\log(1-p_i)\} - \lambda||\beta||^2$$

  - *lasso penalty* — selects variables

$$\text{PLL} = \sum_{i=1}^{N}\{y_i \log p_i + (1-y_i)\log(1-p_i)\} - \lambda\sum_{j=1}^{p}|\beta_j|$$

- IRLS algorithm for ridge, and LARS-like algorithm for Lasso

`Glmnet` software in R

`Glmnet` fits the GLM family of models by penalized maximum likelihood. This includes (multiple) logistic regression. `Glmnet` computes the entire *“regularization path”* for the *“elastic net”* penalty family:

$$\max_{\beta} \; l(\beta) - \lambda\left[\frac{1}{2}(1-\alpha)||\beta||_2^2 + \alpha||\beta||_1\right]$$

- The regularization path follows a complete grid of values for λ, with α fixed.
- α spans ridge to lasso
- For multiple logistic regression, the model is symmetric, with $\eta_j(x) \sim \log P(G=j|x) = x^T\beta_j$ and $P(G=j|x) = e^{\eta_j(x)}/\sum_{\ell=1}^{K} e^{\eta_\ell(x)}$.

Logistic regression or LDA?

- LDA:

$$\begin{aligned}\log\frac{\Pr(G=j|X=x)}{\Pr(G=K|X=x)} &= \log\frac{\pi_j}{\pi_K}-\frac{1}{2}(\mu_j+\mu_K)^T\Sigma^{-1}(\mu_j-\mu_K)\\ &\quad +x^T\Sigma^{-1}(\mu_j-\mu_K)\\ &= \alpha_{j0}+\alpha_j^T x.\end{aligned}$$

  This linearity is a consequence of the Gaussian assumption for the class densities, as well as the assumption of a common covariance matrix.

- Logistic model:

$$\log\frac{\Pr(G=j|X=x)}{\Pr(G=K|X=x)}=\beta_{j0}+\beta_j^T x.$$

  They use the same form for the logits

- Discriminative vs generative (informative) learning: logistic regression uses the conditional distribution of Y given x to estimate parameters, while LDA uses the full joint distribution (assuming normality).

$$\mathrm{Pr}(X, G = j) = \mathrm{Pr}(X)\mathrm{Pr}(G = j|X),$$

- If normality holds, LDA is up to 30% more efficient; o/w logistic regression can be more robust. But the methods are similar in practice.
- The additional efficiency is obtained from using observations far from the decision boundary to help estimate Σ (dubious!)

Naive Bayes Models

Suppose we estimate the class densities $f_1(X)$ and $f_2(X)$ for the features in class 1 and 2 respectively.

Bayes Formula tells us how to convert these to class posterior probabilities:

$$\Pr(Y = 1|X) = \frac{f_1(X)\pi_1}{f_1(X)\pi_1 + f_2(X)\pi_2},$$

where $\pi_1 = \Pr(Y = 1)$ and $\pi_2 = 1 - \pi_1$.

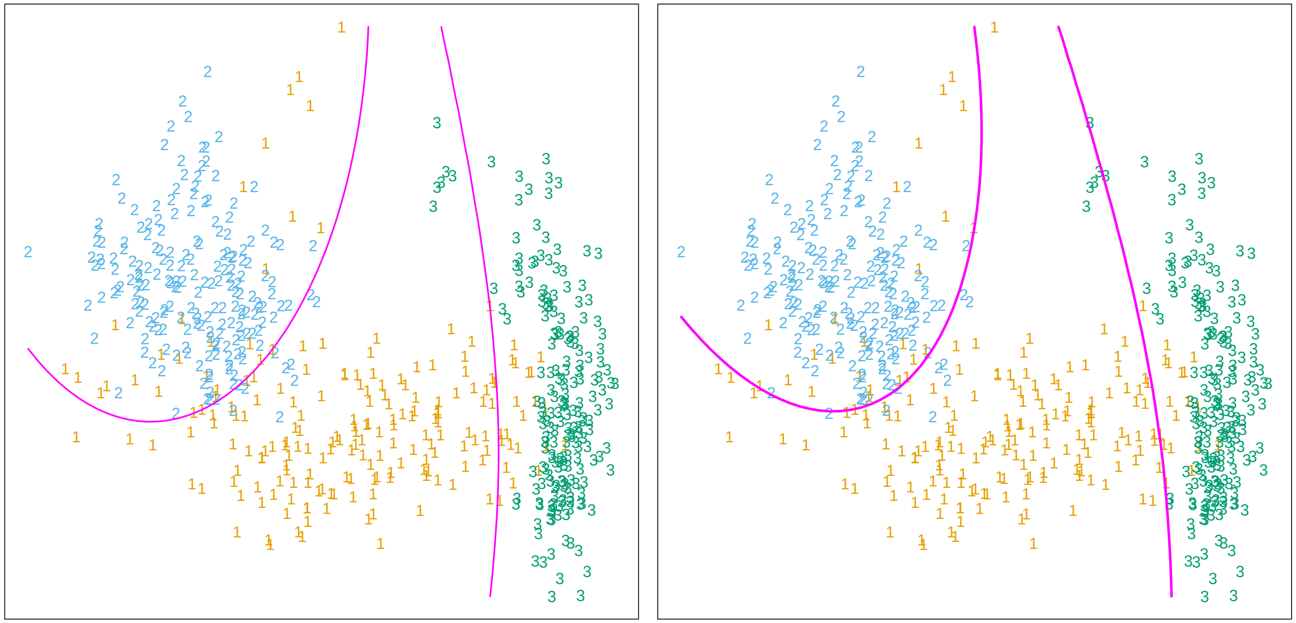
Since X is often high dimensional, the following *independence* model is convenient:

$$f_j(X) \approx \prod_{m=1}^{p} f_{jm}(X_m)$$

Works for more than two classes as well.

- Each of the component densities f_{jm} are estimated separately within each class:
  - Discrete components via histograms
  - quantitative components via Gaussians or smooth density estimates.
- The PAM model has this structure, and in addition
  - assumes the gaussian densities have the same variance in each class
  - shrinks the class centroids towards the overall mean in each class
- More general models have less bias but are typically hard to estimate in high dimensions, so the independence assumption may not hurt too much.

Naive Bayes vs Quadratic Discriminant Analysis



Two methods for fitting quadratic boundaries. [Left] Quadratic decision boundaries, obtained Naive Bayes. [Right] Quadratic decision boundaries found by QDA.

Naive Bayes and GAMs

Note that

$$\log \frac{f_1(X)\pi_1}{f_2(X)\pi_2} = \alpha + \sum_{m=1}^{p} g_m(X_m),$$

a generalized additive logistic regression model.

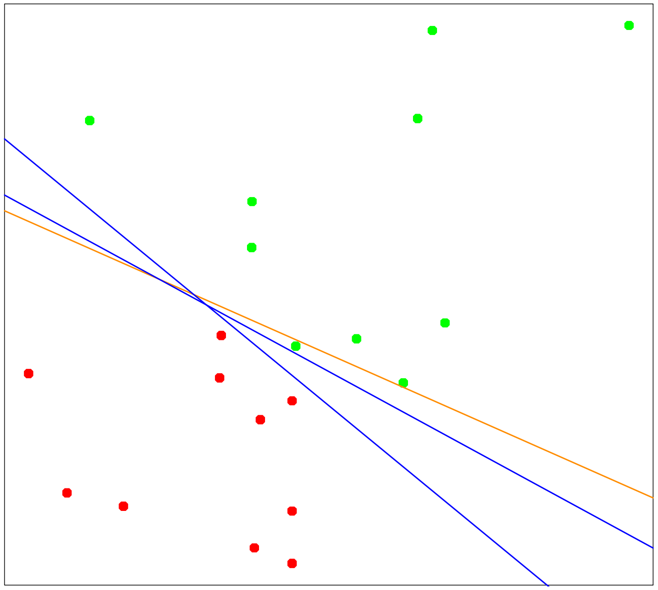
GAMs are fit by *binomial maximum likelihood.*

Naive Bayes models are fit using the *full likelihood*.

GAMs are discussed in chapters 5 and 6.

Separating hyperplanes

$$\{x : \beta_0 + \beta_1 x_1 + \beta_2 x_2 = 0\}$$



A toy example with two classes separable by a hyperplane. The orange line is the least squares solution, which misclassifies one of the training points. Also shown are two blue separating hyperplanes found by the *"perceptron learning algorithm"* with different random starts.

Rosenblatt's Perceptron Learning Algorithm

If a response $y_i = 1$ is misclassified, then $x_i^T\beta + \beta_0 < 0$, and the opposite for a misclassified response with $y_i = -1$. The goal is to minimize

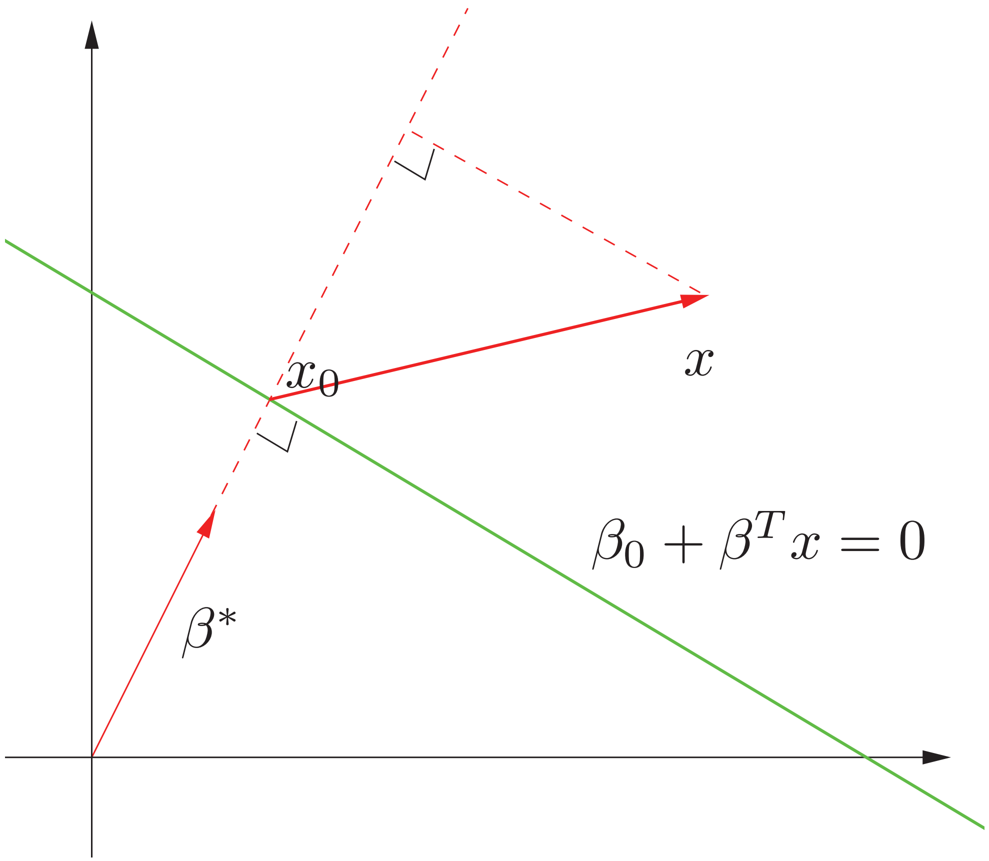
$$D(\beta, \beta_0) = -\sum_{i\in\mathcal{M}} y_i(x_i^T\beta + \beta_0),$$

over $||\beta|| = 1$, where $\mathcal{M}$ indexes the set of misclassified points.

$$\begin{aligned} \frac{\partial D(\beta,\beta_0)}{\partial\beta} &= -\sum_{i\in\mathcal{M}} y_i x_i, \\ \frac{\partial D(\beta,\beta_0)}{\partial\beta_0} &= -\sum_{i\in\mathcal{M}} y_i. \end{aligned}$$

Stochastic gradient descent converges if data are separable (Ex 4.6):

$$\begin{pmatrix}\beta\\ \beta_0\end{pmatrix} \leftarrow \begin{pmatrix}\beta\\ \beta_0\end{pmatrix} + \rho\begin{pmatrix}y_i x_i\\ y_i\end{pmatrix}.$$



Geometry

Consider the line L:
$f(x) = \beta_0 + \beta^T x = 0.$
For any x_1 and x_2 on the line, $\beta^T(x_1 - x_2) = 0$. Hence $\beta^* = \beta/||\beta||$ is the normal to the affine set $f(x) = 0$.
The signed distance of any point x to L is given by

$$\begin{aligned} \beta^{*T}(x - x_0) &= \frac{1}{||\beta||}(\beta^T x + \beta_0) \\ &= \frac{1}{||f'(x)||} f(x). \end{aligned}$$

Optimal Separating Hyperplanes

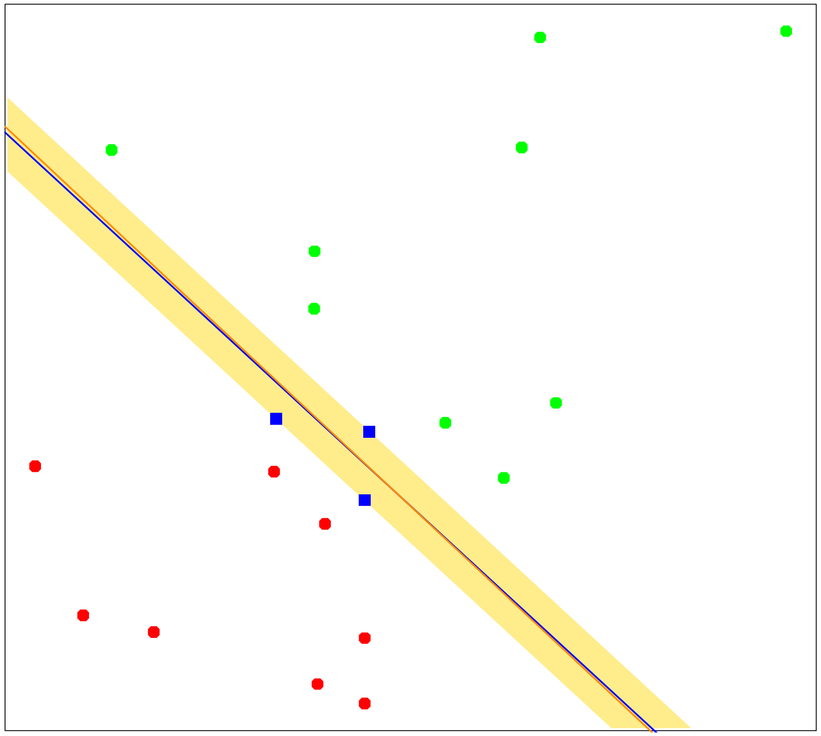
Problem

$$\max_{\beta,\beta_0,||\beta||=1} C$$
$$\text{subject to } y_i(x_i^T\beta + \beta_0) \geq C,\ i = 1, \ldots, N.$$

Convex optimization shows that the solution has the form

$$\hat{\beta} = \sum_{i=1}^{N} \hat{\alpha}_i y_i x_i$$

$\hat{\alpha}_i > 0$ if x_i is on boundary, else 0. Such boundary points are called support points (3 in toy example)



The same toy example. The shaded region delineates the maximum margin separating the two classes. There are three support points indicated, which lie on the boundary of the margin, and the optimal separating hyperplane (blue line) bisects the slab. Included in the figure is the boundary found using logistic regression (red line), which is very close to the optimal separating hyperplane (see Chapter 12 of ESL).