Classification 2: Linear discriminant analysis (continued); logistic regression

Ryan Tibshirani
Data Mining: 36-462/36-662

April 4 2013

Optional reading: ISL 4.4, ESL 4.3; ISL 4.3, ESL 4.4

Reminder: linear discriminant analysis

Last time we defined the Bayes classifier in the population, for the class label $C \in \{1, \ldots K\}$ and feature vector $X \in \mathbb{R}^p$, as

$$\begin{aligned} f(x) &= \underset{j=1,\ldots K}{\operatorname{argmax}} \ \mathrm{P}(C = j | X = x) \\ &= \underset{j=1,\ldots K}{\operatorname{argmax}} \ \mathrm{P}(X = x | C = j) \cdot \pi_j \end{aligned}$$

where $\pi_j = \mathrm{P}(C = j)$ is the prior probability of class j

Linear discriminant analysis approximates this rule by modeling the conditional class densities as multivariate normals:

$$h_j(x) = \mathrm{P}(X = x | C = j) = N(\mu_j, \Sigma) \text{ density}$$

i.e., each class j has its own mean $\mu_j \in \mathbb{R}^p$, but shares a common covariance matrix $\Sigma \in \mathbb{R}^{p \times p}$

We then replace π_j, μ_j, Σ by their sample estimates, based on labeled observations $y_i \in \{1, \dots K\}$, $x_i \in \mathbb{R}^p$, $i = 1, \dots n$,

$$\hat{\pi}_j = n_j/n, \quad \hat{\mu}_j = \frac{1}{n_j} \sum_{y_i = j} x_i,$$

$$\hat{\Sigma} = \frac{1}{n-K} \sum_{j=1}^{K} \sum_{y_i = j} (x_i - \hat{\mu}_j)(x_i - \hat{\mu}_j)^T$$

The rule then reduces to

$$\hat{f}^{\text{LDA}}(x) = \underset{j=1,\dots K}{\text{argmax}} \ \hat{\delta}_j(x)$$

where $\hat{\delta}_j(x)$ is the estimated discriminant function of class j,

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

LDA computations and sphering

Note that LDA equivalently minimizes over $j = 1, \ldots K$,

$$\frac{1}{2}(x - \hat{\mu}_j)^T \hat{\Sigma}^{-1}(x - \hat{\mu}_j) - \log \hat{\pi}_j$$

It helps to factorize $\hat{\Sigma}$ (i.e., compute its eigendecomposition):

$$\hat{\Sigma} = UDU^T$$

where $U \in \mathbb{R}^{p \times p}$ has orthonormal columns (and rows), and $D = \mathrm{diag}(d_1, \ldots d_p)$ with $d_j \geq 0$ for each j. Then we have $\hat{\Sigma}^{-1} = UD^{-1}U^T$, and

$$(x - \hat{\mu}_j)^T \hat{\Sigma}^{-1}(x - \hat{\mu}_j) = \| \underbrace{D^{-1/2}U^T x}_{\tilde{x}} - \underbrace{D^{-1/2}U^T \hat{\mu}_j}_{\tilde{\mu}_j} \|_2^2$$

This is just the squared distance between $\tilde{x}$ and $\tilde{\mu}_j$

Hence the LDA procedure can be described as:

1. Compute the sample estimates $\hat{\pi}_j, \hat{\mu}_j, \hat{\Sigma}$
2. Factor $\hat{\Sigma}$, as in $\hat{\Sigma} = UDU^T$
3. Transform the class centroids $\tilde{\mu}_j = D^{-1/2}U^T\hat{\mu}_j$
4. Given any point $x \in \mathbb{R}^p$, transform to $\tilde{x} = D^{-1/2}U^T x \in \mathbb{R}^p$, and then classify according to the nearest centroid in the transformed space, adjusting for class proportions—this is the class j for which $\frac{1}{2}\|\tilde{x} - \tilde{\mu}_j\|_2^2 - \log \hat{\pi}_j$ is smallest

What is this transformation doing? Think about applying it to the observations:

$$\tilde{x}_i = D^{-1/2}U^T x_i, \quad i = 1, \ldots n$$

This is basically sphering the data points, because if we think of $x \in \mathbb{R}^p$ were a random variable with covariance matrix $\hat{\Sigma}$, then

$$\mathrm{Cov}(D^{-1/2}U^T x) = D^{-1/2}U^T\hat{\Sigma}UD^{-1/2} = I$$

Linear subspace spanned by sphered centroids

LDA compares the quantity $\frac{1}{2}\|\tilde{x} - \tilde{\mu}_j\|_2^2 - \log \hat{\pi}_j$ across the classes $j = 1, \ldots K$. Consider the affine subspace $M \subseteq \mathbb{R}^p$ spanned by the transformed centroids $\tilde{\mu}_1, \ldots \tilde{\mu}_K$, which has dimension $K - 1$

For any $\tilde{x} \in \mathbb{R}^p$, we can decompose $\tilde{x} = P_M \tilde{x} + P_{M^\perp} \tilde{x}$, so

$$\|\tilde{x} - \tilde{\mu}_j\|_2^2 = \| \underbrace{P_M \tilde{x} - \tilde{\mu}_j}_{\in M} + \underbrace{P_{M^\perp} \tilde{x}}_{\in M^\perp} \|_2^2$$

$$= \|P_M \tilde{x} - \tilde{\mu}_j\|_2^2 + \|P_{M^\perp} \tilde{x}\|_2^2$$

The second term doesn't depend on j

What this is telling us: the LDA classification rule is unchanged if we project the points to be classified onto M, since the distances orthogonal to M don't matter

LDA procedure summarized

We can think of the LDA procedure as:

1. Compute the sample estimates $\hat{\pi}_j, \hat{\mu}_j, \hat{\Sigma}$
2. Make two transformations: first, sphere the data points, based on factoring $\hat{\Sigma}$; second, project down to the affine subspace spanned by the sphered centroids. This can all be summarized a single linear transformation $A \in \mathbb{R}^{(K-1)\times p}$
3. Given any point $x \in \mathbb{R}^p$, transform to $\tilde{x} = Ax \in \mathbb{R}^{K-1}$, and classify according to the class $j = 1, \ldots K$ for which

$$\frac{1}{2}\|\tilde{x} - \tilde{\mu}_j\|_2^2 - \log \hat{\pi}_j$$

is smallest, where $\tilde{\mu}_j = A\hat{\mu}_j$

This way of describing LDA may sound more complicated, but actually, it's much simpler! After applying A, we've reduced the problem from p to $K-1$ dimensions, and then it's basically nearest centroid classification:

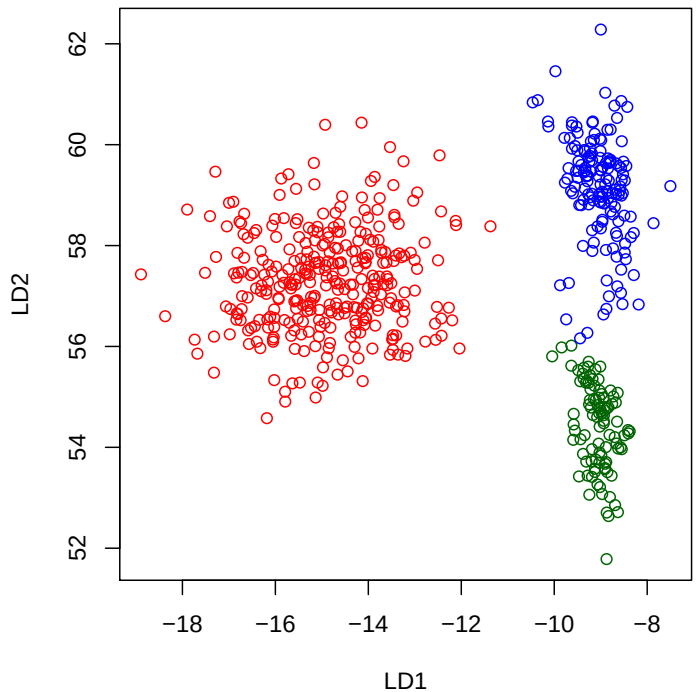
$$\hat{f}^{\mathrm{LDA}}(x) = \underset{j=1,\ldots K}{\operatorname{argmin}} \ \frac{1}{2}\|\tilde{x} - \tilde{\mu}_j\|_2^2 - \log \hat{\pi}_j$$

(The only distinction being that we adjust for class proportions)

In R, the matrix A^T is exactly what is returned by the `scaling` component from the `lda` function in the `MASS` package

Example: olive oil data

Recall our olive oil example: $n = 572$ olive oils, made in one of three regions of Italy. For each observation we have $p = 8$ features measuring the percentage composition of 8 different fatty acids



These are the transformed data points $Ax_i \in \mathbb{R}^2$, $i = 1, \ldots 572$. Note that here M is a 2-dimensional subspace, since there are $K = 3$ classes, and the transformation has dimension $A \in \mathbb{R}^{2\times 8}$

Decision boundaries revisited

Working in the transformed space makes it easier to draw decision boundaries. Now the decision boundary between classes j and k is the set of all $z \in \mathbb{R}^{K-1}$ such that

$$\frac{1}{2}\|z - \tilde{\mu}_j\|_2^2 - \log \hat{\pi}_j = \frac{1}{2}\|z - \tilde{\mu}_k\|_2^2 - \log \hat{\pi}_k$$

After some calculation, this is simply

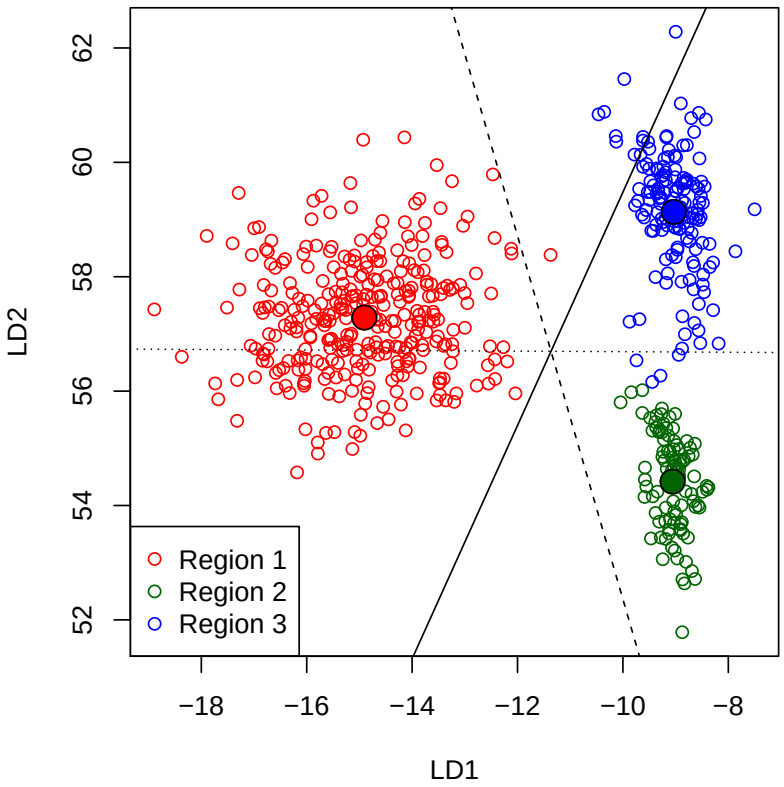
$$(\tilde{\mu}_j - \tilde{\mu}_k)^T z = \log \frac{\hat{\pi}_k}{\hat{\pi}_j} + \frac{1}{2}(\|\tilde{\mu}_j\|_2^2 - \|\tilde{\mu}_k\|_2^2)$$

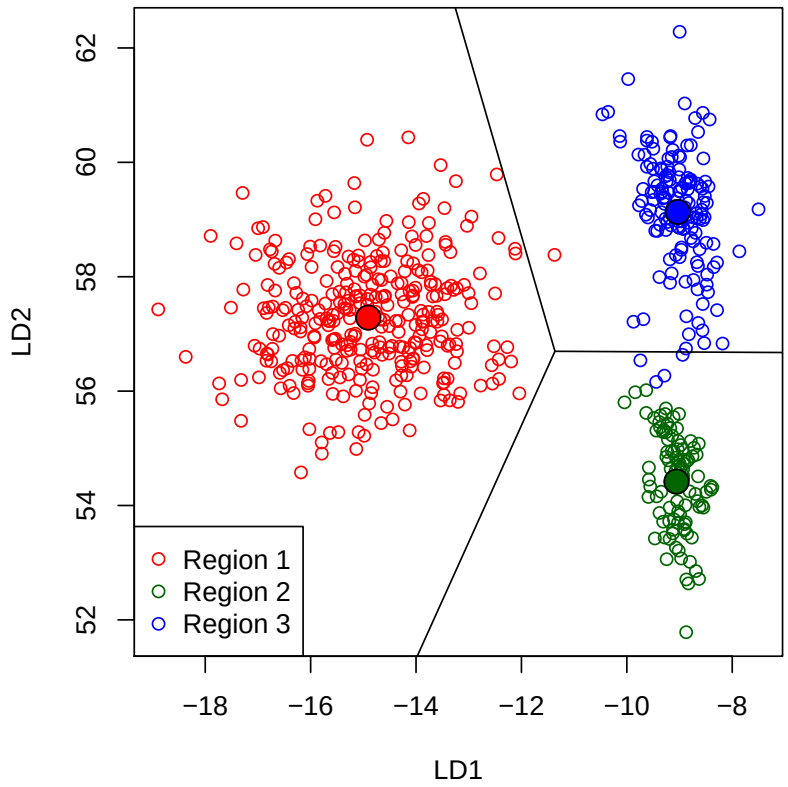
E.g., when $K = 3$, so that $z \in \mathbb{R}^2$, this is just the line given by $z_2 = a + bz_1$, where

$$a = \frac{\log \frac{\hat{\pi}_k}{\hat{\pi}_j} + \frac{1}{2}(\|\tilde{\mu}_j\|_2^2 - \|\tilde{\mu}_k\|_2^2)}{\tilde{\mu}_{j2} - \tilde{\mu}_{k2}}, \quad b = \frac{\tilde{\mu}_{k1} - \tilde{\mu}_{j1}}{\tilde{\mu}_{j2} - \tilde{\mu}_{k2}}$$

Example: olive oil data

Decision boundaries, using the formula that we derived:





Reduced-rank linear discriminant analysis

The dimension reduction from p to $K-1$ was exact, in that we didn't change the LDA rule at all. Why might we want to reduce further to a dimension $L < K-1$, if K is large?

- Visualization
- Regularization

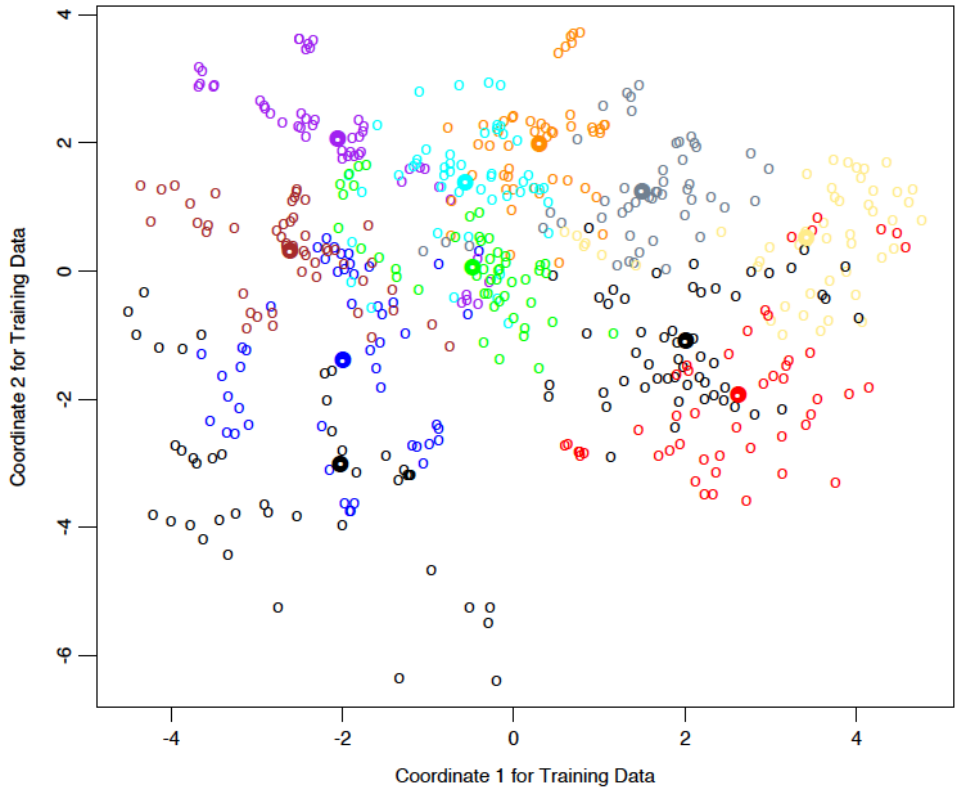
Reduced-rank linear discriminant analysis is a nice way to project down to lower than $K-1$ dimensions. It chooses the lower dimensional subspaces so as to spread out the centroids as much as possible

Does this sound like principal components analysis? It's not a coincidence! These dimensions are computed by looking at the principal components directions of the matrix of transformed centroids

Example: vowel data

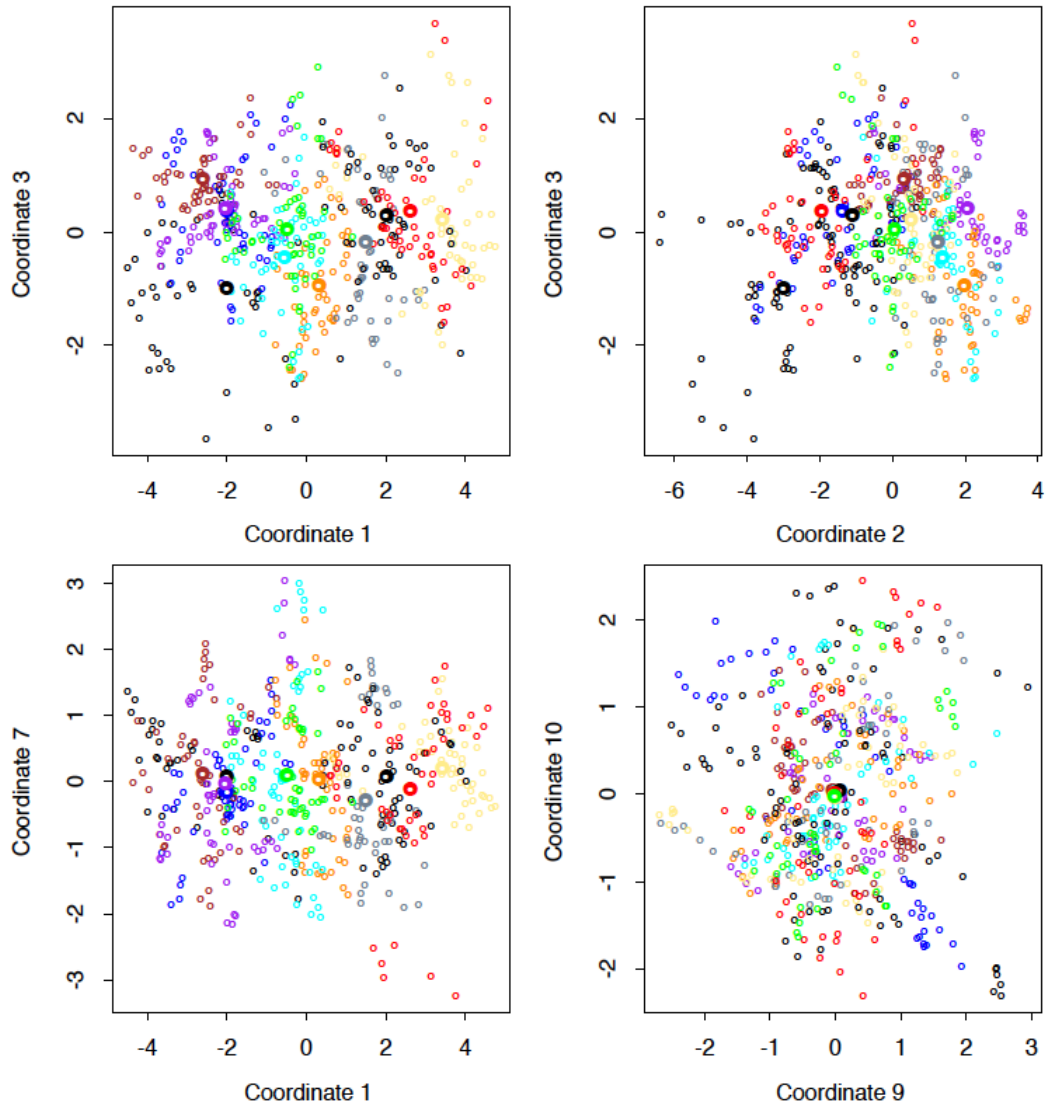
Example: this experiment recorded $n = 528$ instances of spoken words. The words fall into $K = 11$ classes ("vowels"), and there are $p = 10$ features measured on each instance

Reduced-rank linear discriminant analysis in 2 dimensions:



(From ESL page 107)

As the rank increases, the centroids become less spread out:



(From ESL page 115)

Reduced-rank LDA as regularization

If the number of classes K is large, then projecting down to a dimension $L < K - 1$ can actually be helpful in terms of applying regularization. This is because some dimensions may not providing a lot of separation between the classes, but just noise. (You should be thinking of the bias-variance tradeoff!)

Reduced-rank LDA in L dimensions delivers a matrix $A \in \mathbb{R}^{L \times p}$, and classification of a point $x \in \mathbb{R}^p$ proceeds as usual, by first transforming to $\tilde{x} = Ax$, and then choosing the class $j = 1, \ldots K$ that minimizes

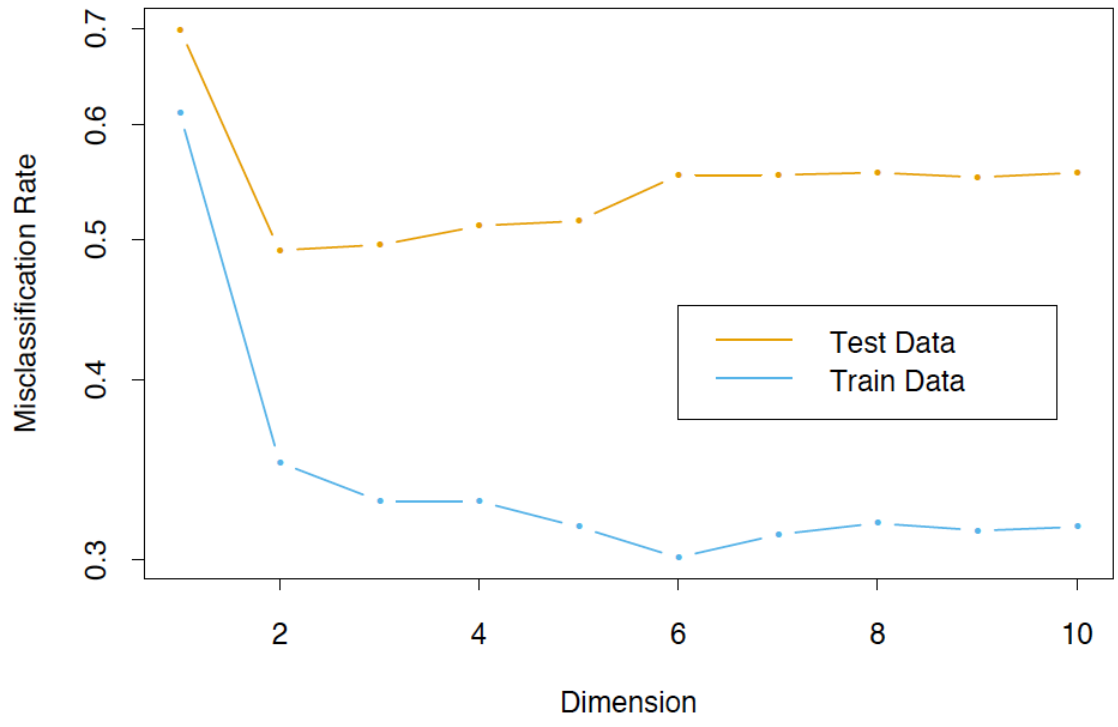
$$\frac{1}{2}\|\tilde{x} - \tilde{\mu}_j\|_2^2 + \log \hat{\pi}_j$$

where $\tilde{\mu}_j = A\hat{\mu}_j$. Smaller L means more regularization

For the `lda` function in R, the matrix $A \in \mathbb{R}^{L \times p}$ above corresponds to the (transpose) of the first L columns of the `scaling` matrix

Example: vowel data

For the vowel data set, there were actually $n = 462$ points held out as a test set. Here is the test error rate (and training error rate) of reduced-rank LDA as a function of L:



(From ESL page 117)

Original form of LDA for two classes

Let's go back to the original form of LDA, and consider just two classes, for simplicity. Recall that LDA assumes that the predictor variables are normally distributed within each class:

$$\mathrm{P}(X = x|C = j) = N(\mu_j, \Sigma) \text{ density}, \quad j = 1, 2$$

Bayes' rule then gives us $\mathrm{P}(C = j|X = x) = \frac{\mathrm{P}(X=x|C=j)\mathrm{P}(C=j)}{\mathrm{P}(X=x)}$. Now plugging in the normal density

$$\log\left\{\frac{\mathrm{P}(C = 1|X = x)}{\mathrm{P}(C = 2|X = x)}\right\} =$$

$$=$$

$$= \alpha_0 + \alpha^T x$$

That is, the log odds of class 1 versus 2 is a linear function of x. Estimating $\hat{\pi}_j, \hat{\mu}_j, \hat{\Sigma}$ amounts to estimating $\hat{\alpha_0}, \hat{\alpha}$. A question: why don't we estimate these coefficients directly?

Logistic regression

In logistic regression, we assume that

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

for some unknown $\beta_0 \in \mathbb{R}$, $\beta \in \mathbb{R}^p$, which we will estimate directly

Note that $P(C=2|X=x) = 1 - P(C=1|X=x)$, and

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta^T x \quad \Leftrightarrow$$

$$\Leftrightarrow$$

Therefore our assumption is that

$$\mathrm{P}(C=1|X=x) = \frac{\exp(\beta_0 + \beta^T x)}{1+\exp(\beta_0 + \beta^T x)}$$

Estimating logistic regression coefficients

Suppose that we are given a sample (x_i, y_i), $i = 1, \ldots n$. Here y_i denotes the class $\in \{1, 2\}$ of the ith observation. Assume that the class labels are conditionally independent given $x_1, \ldots x_n$. Then

$$\mathcal{L}(\beta_0, \beta) = \prod_{i=1}^{n} \mathrm{P}(C = y_i | X = x_i)$$

the likelihood of these n observations, so the log likelihood is

$$\ell(\beta_0, \beta) = \sum_{i=1}^{n} \log \mathrm{P}(C = y_i | X = x_i)$$

Now for convenience, we define the indicator

$$u_i = \begin{cases} 1 & \text{if } y_i = 1 \\ 0 & \text{if } y_i = 2 \end{cases}$$

The log likelihood can be written as

$$\ell(\beta_0, \beta) = \sum_{i=1}^{n} \log \mathrm{P}(C = y_i | X = x_i)$$

$$=$$

$$=$$

$$= \sum_{i=1}^{n} \left\{ u_i \cdot (\beta_0 + \beta^T x_i) - \log \left(1 + \exp(\beta_0 + \beta^T x_i)\right) \right\}$$

The coefficients are estimated by maximizing the likelihood,

$$\hat{\beta}_0, \hat{\beta} = \underset{\beta \in \mathbb{R}, \beta \in \mathbb{R}^p}{\operatorname{argmax}} \sum_{i=1}^{n} \left\{ u_i \cdot (\beta_0 + \beta^T x_i) - \log \left(1 + \exp(\beta_0 + \beta^T x_i)\right) \right\}$$

Note that logistic regression is somewhat more general than LDA in that we don't need to assume normality to estimate the linear coefficients

Recap: reduced-rank LDA and logistic regression

In this lecture we saw that the usual linear discriminant analysis for prediction in p dimensions can be transformed to a much simpler rule in $K-1$ dimensions, where K is the number of classes

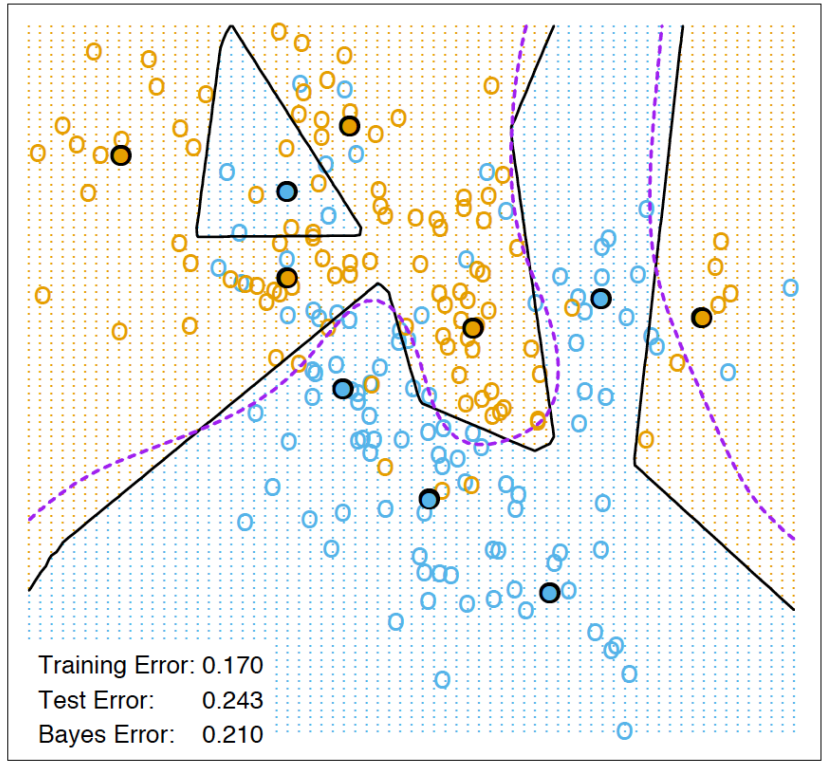
This transformation was achieved by first sphering the data points, and then projecting onto the affine subspace spanned by the sphered class centroids. The final prediction rule is basically nearest centroid classification, except for the factor adjusting for different class proportions

Further transformations, to dimensions $L < K-1$, can also be performed; this is called reduced-rank linear discriminant analysis. Doing so can be helpful for visualization or regularization purposes

Logistic regression models the log odds of being in class 1 versus 2 as a linear function of the features. It is fit by maximum likelihood

Next time: more logistic regression; nearest-neighbors and prototypes

Classification by K-means clustering: using an unsupervised tool for a supervised task



(From ESL page 464)