Lecture 9: Classification, LDA

Reading: Chapter 4

STATS 202: Data mining and analysis

Sergio Bacallado
October 26, 2013

Review: Main strategy in Chapter 4

Find an estimate $\hat{P}(Y \mid X)$. Then, given an input x_0, we predict the response as in a Bayes classifier:

$$\hat{y}_0 = \text{argmax}_{\,y}\ \hat{P}(Y = y \mid X = x_0).$$

Linear Discriminant Analysis (LDA)

Instead of estimating $P(Y \mid X)$, we will estimate:

1. $\hat{P}(X \mid Y)$: Given the response, what is the distribution of the inputs.
2. $\hat{P}(Y)$: How likely are each of the categories.

Then, we use *Bayes rule* to obtain the estimate:

$$\hat{P}(Y = k \mid X = x) = \frac{\hat{P}(X = x \mid Y = k)\hat{P}(Y = k)}{\hat{P}(X = x)}$$

Linear Discriminant Analysis (LDA)

Instead of estimating $P(Y \mid X)$, we will estimate:

1. $\hat{P}(X \mid Y)$: Given the response, what is the distribution of the inputs.
2. $\hat{P}(Y)$: How likely are each of the categories.

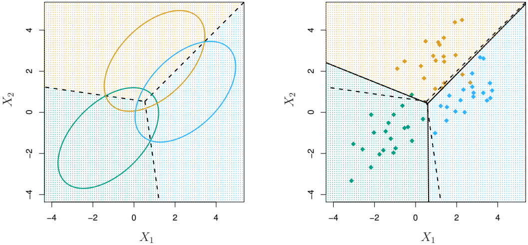
Then, we use *Bayes rule* to obtain the estimate:

$$\hat{P}(Y = k \mid X = x) = \frac{\hat{P}(X = x \mid Y = k)\hat{P}(Y = k)}{\sum_j \hat{P}(X = x \mid Y = j)\hat{P}(Y = j)}$$

Linear Discriminant Analysis (LDA)

Instead of estimating $P(Y \mid X)$, we will estimate:

1. We model $\hat{P}(X = x \mid Y = k) = \hat{f}_k(x)$ as a *Multivariate Normal Distribution*:



2. $\hat{P}(Y = k) = \hat{\pi}_k$ is estimated by the fraction of training samples of class k.

LDA has linear decision boundaries

Suppose that:

- We know $P(Y = k) = \pi_k$ exactly.
- $P(X = x|Y = k)$ is Mutivariate Normal with density:

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

μ_k : Mean of the inputs for category k.
$\mathbf{\Sigma}$: Covariance matrix (common to all categories).

Then, what is the Bayes classifier?

LDA has linear decision boundaries

By Bayes rule, the probability of category k, given the input x is:

$$P(Y = k \mid X = x) = \frac{f_k(x)\pi_k}{P(X = x)}$$

The denominator does not depend on the response k, so we can write it as a constant:

$$P(Y = k \mid X = x) = C \times f_k(x)\pi_k$$

Now, expanding $f_k(x)$:

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

LDA has linear decision boundaries

$$P(Y = k \mid X = x) = \frac{C\pi_k}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

Now, let us absorb everything that does not depend on k into a constant C':

$$P(Y = k \mid X = x) = C'\pi_k e^{-\frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k)}$$

and take the logarithm of both sides:

$$\log P(Y = k \mid X = x) = \log C' + \log \pi_k - \frac{1}{2}(x - \mu_k)^T \mathbf{\Sigma}^{-1}(x - \mu_k).$$

This is the same for every category, k.
So we want to find the maximum of this over k.

LDA has linear decision boundaries

Goal, maximize the following over k:

$$\begin{aligned}
&\log \pi_k - \frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k). \\
=&\log \pi_k - \frac{1}{2}\left[x^T \mathbf{\Sigma}^{-1} x + \mu_k^T \mathbf{\Sigma}^{-1}\mu_k\right] + x^T \mathbf{\Sigma}^{-1}\mu_k \\
=&C'' + \log \pi_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + x^T \mathbf{\Sigma}^{-1}\mu_k
\end{aligned}$$

We define the objective:

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

At an input x, we predict the response with the highest $\delta_k(x)$.

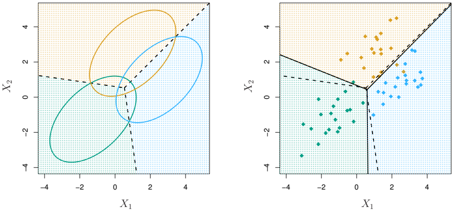
LDA has linear decision boundaries

What is the decision boundary? It is the set of points in which 2 classes do just as well:

$$\delta_k(x) = \delta_\ell(x)$$

$$\log \pi_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + x^T \mathbf{\Sigma}^{-1}\mu_k = \log \pi_\ell - \frac{1}{2}\mu_\ell^T \mathbf{\Sigma}^{-1}\mu_\ell + x^T \mathbf{\Sigma}^{-1}\mu_\ell$$

This is a linear equation in x.



Estimating π_k

$$\hat{\pi}_k = \frac{\#\{i\ ;\ y_i = k\}}{n}$$

In English, the fraction of training samples of class k.

Estimating the parameters of $f_k(x)$

Estimate the center of each class μ_k:

$$\hat{\mu}_k = \frac{1}{\#\{i\ ;\ y_i = k\}} \sum_{i\ ;\ y_i = k} x_i$$

Estimate the common covariance matrix $\mathbf{\Sigma}$:

- One predictor ($p = 1$):

$$\hat{\sigma}^2 = \frac{1}{n - K} \sum_{k=1}^{K} \sum_{i\ ;\ y_i = k} (x_i - \hat{\mu}_k)^2.$$

- Many predictors ($p > 1$): Compute the vectors of deviations $(x_1 - \hat{\mu}_{y_1}), (x_2 - \hat{\mu}_{y_2}), \ldots, (x_n - \hat{\mu}_{y_n})$ and use an unbiased estimate of its covariance matrix, $\mathbf{\Sigma}$.

LDA prediction

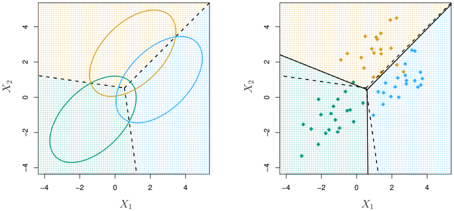
For an input x, predict the class with the largest:

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

The decision boundaries are defined by:

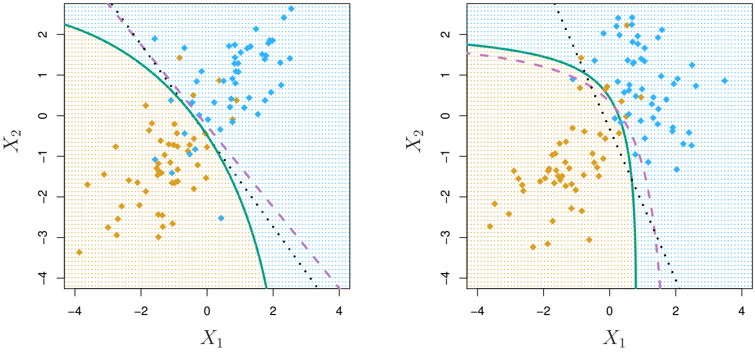
$$\log \hat{\pi}_k - \frac{1}{2}\hat{\mu}_k^T \hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_k + x^T \hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_k = \log \hat{\pi}_\ell - \frac{1}{2}\hat{\mu}_\ell^T \hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_\ell + x^T \hat{\mathbf{\Sigma}}^{-1}\hat{\mu}_\ell$$

Solid lines in:



Quadratic discriminant analysis (QDA)

The assumption that the inputs of every class have the same covariance $\mathbf{\Sigma}$ can be quite restrictive:



Quadratic discriminant analysis (QDA)

In **quadratic discriminant analysis** we estimate a mean $\hat{\mu}_k$ and a covariance matrix $\hat{\mathbf{\Sigma}}_k$ for each class separately.

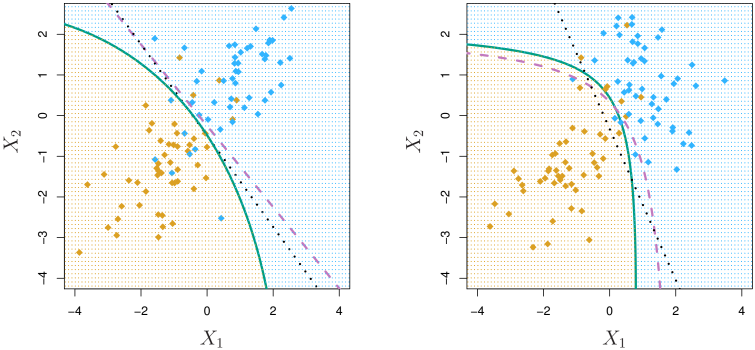
Given an input, it is easy to derive an objective function:

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

This objective is now quadratic in x and so are the decision boundaries.

Quadratic discriminant analysis (QDA)

- Bayes boundary (– – –)
- LDA (· · · · · ·)
- QDA (——).



Evaluating a classification method

We have talked about the 0-1 loss:

$$\frac{1}{m}\sum_{i=1}^{m}\mathbf{1}(y_i \neq \hat{y}_i).$$

It is possible to make the wrong prediction for some classes more often than others. The 0-1 loss doesn't tell you anything about this.

A much more informative summary of the error is a **confusion matrix**:

		Predicted class		
		− or Null	+ or Non-null	Total
True class	− or Null	True Neg. (TN)	False Pos. (FP)	N
	+ or Non-null	False Neg. (FN)	True Pos. (TP)	P
	Total	N*	P*	

Example. Predicting `default`

Used LDA to predict credit card default in a dataset of 10K people.

Predicted "yes" if $P(\texttt{default} = \text{yes}|X) > 0.5$.

		True default status		
		No	Yes	Total
Predicted default status	No	$9,644$	252	$9,896$
	Yes	23	81	104
	Total	$9,667$	333	$10,000$

- The error rate among people who do **not** default (false positive rate) is very low.
- However, the rate of false negatives is 76%.
- It is possible that false negatives are a bigger source of concern!
- One possible solution: Change the threshold.

Example. Predicting `default`

Changing the threshold to 0.2 makes it easier to classify to "yes".

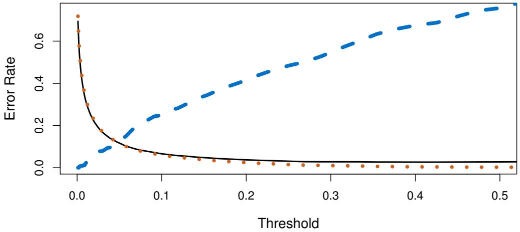
Predicted "yes" if $P(\texttt{default} = \text{yes}|X) > 0.2$.

		True default status		
		No	Yes	Total
Predicted default status	No	9,432	138	9,570
	Yes	235	195	430
	Total	9,667	333	10,000

Note that the rate of false positives became higher! That is the price to pay for fewer false negatives.

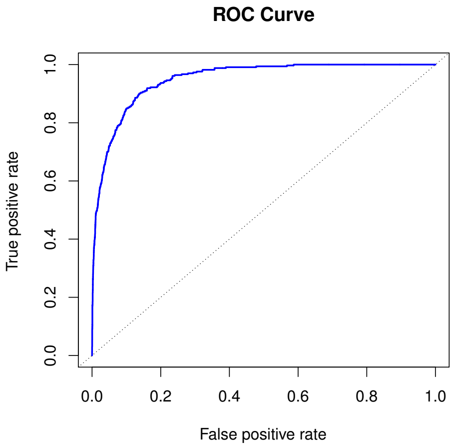
Example. Predicting `default`

Let's visualize the dependence of the error on the threshold:



- – – – False negative rate (error for defaulting customers)
- · · · · · False positive rate (error for non-defaulting customers)
- —— 0-1 loss or total error rate.

Example. The ROC curve



- Displays the performance of the method for any choice of threshold.
- The area under the curve (AUC) measures the quality of the classifier:
  - 0.5 is the AUC for a random classifier
  - The closer AUC is to 1, the better.

Next time

- Comparison of logistic regression, LDA, QDA, and KNN classification.
- Start Chapter 5: Resampling.