New York University: Machine Learning in Economics

Lecture 3: Linear Methods for Classification

Dr. George Lentzas

Classification

Error Rates

- The most common deviation from the classical linear regression setting is the problem of classification of observations into classes. Here we re-cast the concept of MSE as the *error rate*, that is the proportion of mistaken classifications.
- As before this can be either training error rate given by

$$ER_{train} = \frac{1}{n}\sum_{i=1}^{n} I\left(y_i \neq \hat{y}_i\right)$$

or the test error rate associated with a set of observations x_0, y_0 and given by

$$ER_{test} = Ave\left(I\left(y_0 \neq \hat{y}_0\right)\right)$$

where $\hat{y}_0$ is the predicted class label that results from applying the classifier to the test observation with predictor x_0. As before, a good classifier is the one for which the test error rate is small!

Classification

The Bayes Classifier

- We can show that the test error rate is minimized by a classifier that assigns each observation to the most likely class given its predictor values. In other words, one that assigns an observation with input values x_0 to the class j for which the conditional probability of being in a class given the input values

$$\Pr(Y = j|X = x_0)$$

  is the largest. This is called the *Bayes classifier*.
- The Bayes classifier results in the lowest possible test error rate which is as expected called the *Bayes error rate* and which is similar to the irreducible error rate discussed earlier.

Classification

The Bayes Classifier

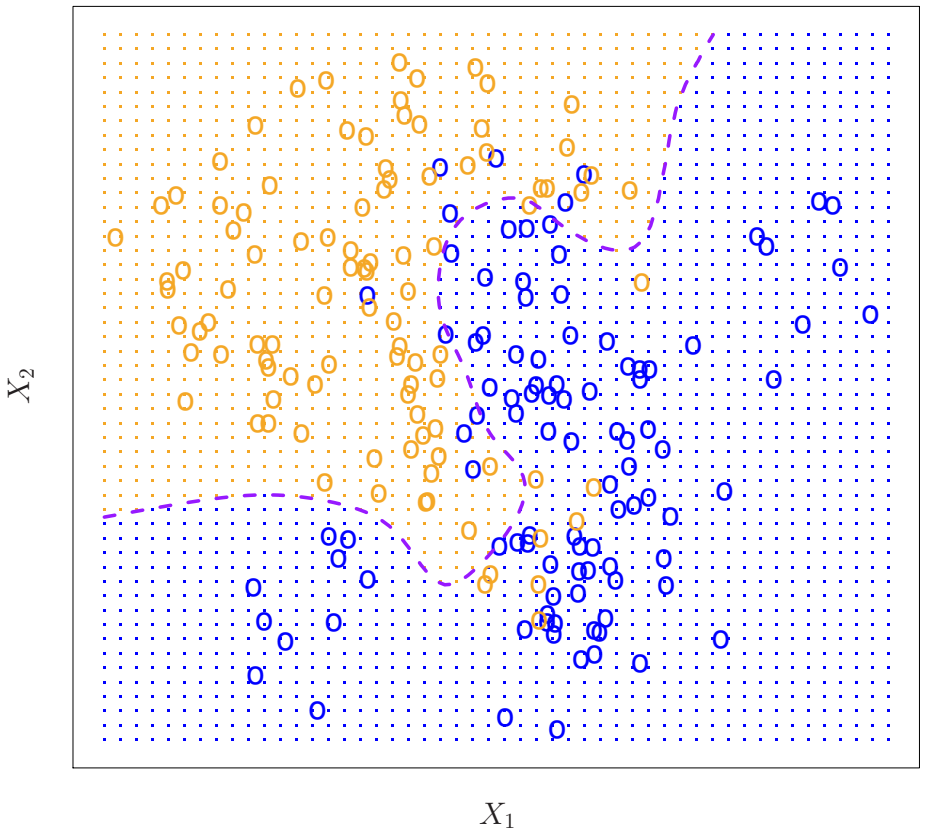

Figure: Simulated data for a two class problem. The dashed line illustrates the Bayes decision boundary.

Classification

k Nearest Neighbors

- In practice we do not know the conditional distribution of Y given X and so computing the Bayes classifier is not generally possible. Instead we use a number of ML techniques to estimate this conditional probability and then assign new observations to the class with the highest estimated probability.
- A first such ML method is the so-called k - Nearest Neighbors (kNN) classifier. The idea is simple: given a positive integer k and a test observation x_0 the kNN classifier finds the k points in the training sample that are closest to x_0 (lets call these N_0) and then estimates the conditional probability for class j as the fraction of points in N_0 whose output variable equals j. We can formalize this as:

$$\hat{\Pr}(Y = j|X = x_0) = \frac{1}{k} \sum_{i \in N_0} I(y_i = j)$$

- Then kNN applies Bayes rule and classifies each test observation x_0 to the class with the largest estimated probability.
- The word "closest" above implies a distance which is typically taken to be (standardized) Euclidean distance.

Classification

k Nearest Neighbors

- Even though this sounds simplistic, kNN often results in classifiers that are surprisingly good and close to the optimal Bayes classifier.

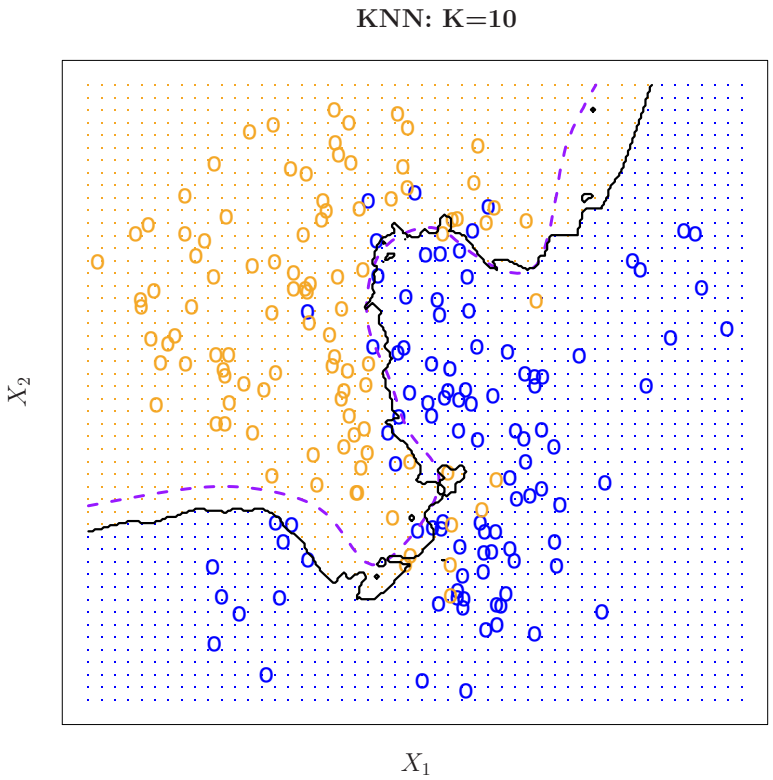

Figure: knn with k =10 on simulated data, black curve shows the kNN decision boundary and the purple line the Bayes decision boundary.

Classification

k Nearest Neighbors

- Obviously the choice of k is crucial: with $k = 1$ the decision boundary becomes noisy and overly flexible (low bias, high variance) while as k increases we reach the opposite point where the boundary becomes rigid and not flexible enough (low variance, high bias). We can see this below.

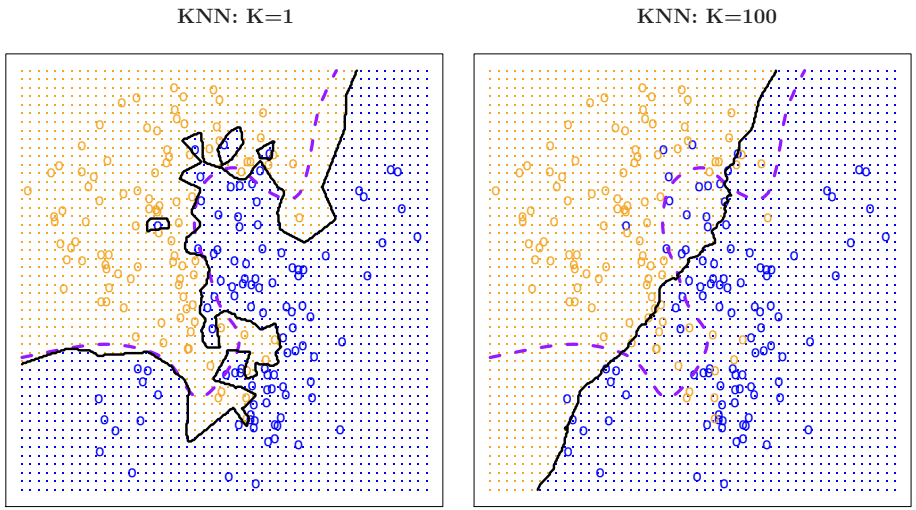

Figure: knn with k =1 (left) and k=100 (right)

Classification

k Nearest Neighbors

- In practice, k is usually chosen using a procedure called k-Fold Cross Validation. We will come back to this later. For now, lets look at an example of how the choice of k affects the training and test error rates.

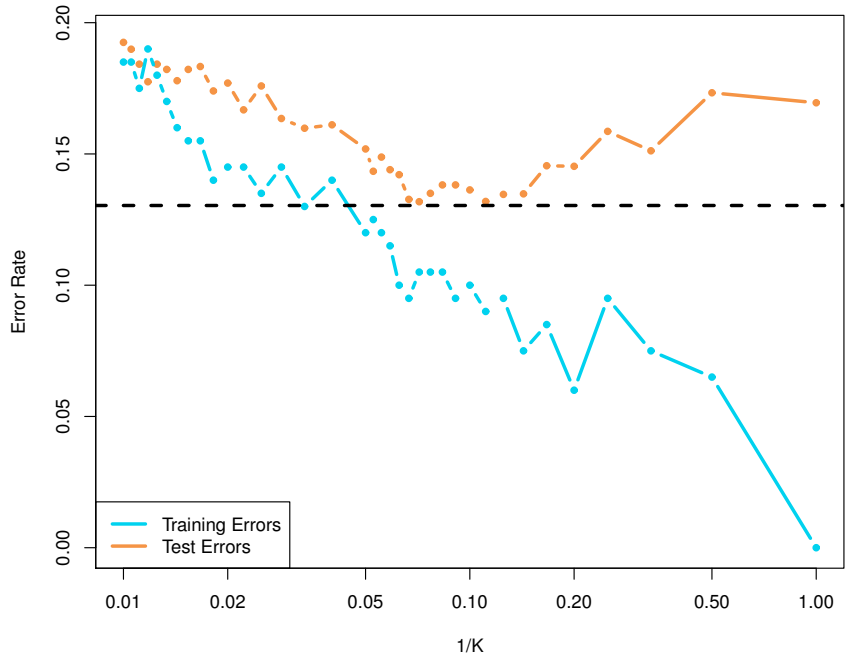

Figure: The kNN training error rate in blue and test error rate in orange.

Classification

Logistic Regression

- Logistic regression explicitly models $\Pr(Y = j|X = x_0)$, the conditional probability that Y belongs to a particular class. In order to ensure that results make sense as probabilities, the logistic function is used:

$$\Pr(\mathbf{X}) = \frac{\exp(\beta_0 + \beta_1 X_1 + \cdots + \beta_P X_P)}{1 + \exp(\beta_0 + \beta_1 X_1 + \cdots + \beta_P X_P)}$$

- This is typically estimated using maximum likelihood. Loosely speaking, maximum likelihood is a technique that estimates coefficients by maximizing the probability that we observe the training sample. In other words, we choose coefficients that make it the most likely that we would observe the sample that we have. This sounds very intuitive and it is!

Classification

Logistic Regression

- The details of maximum likelihood are beyond our scope but here is a brief overview of how it works.
- We express the probability of observing our sample as a function of the parameters we want to estimate, in this case the vector β. Then we choose such parameters (β that maximize the likelihood function L (intuitively the probability of observing the sample). This is usually done numerically although explicit solutions exist for some simple cases. The maximized likelihood function L can then be used to calculate the standard errors of the parameter estimates $\hat{\beta}$.
- In the logistic regression case, the likelihood of a sample is

$$L(\boldsymbol{\beta}|\mathbf{X}) = \prod_{i=1}^{n} \Big[\Pr(\mathbf{X}) \Big]^{y_i} \Big[1 - \Pr(\mathbf{X}) \Big]^{1-y_i}$$

where $\Pr(\mathbf{X})$ is given by the logistic function above.

Classification

Logistic Regression

- Lets look at an example; the right hand panel of the figure below shows the fit of a one variable logistic regression model on the Default data set from ISLR. Here we have only on X, the credit card balance.

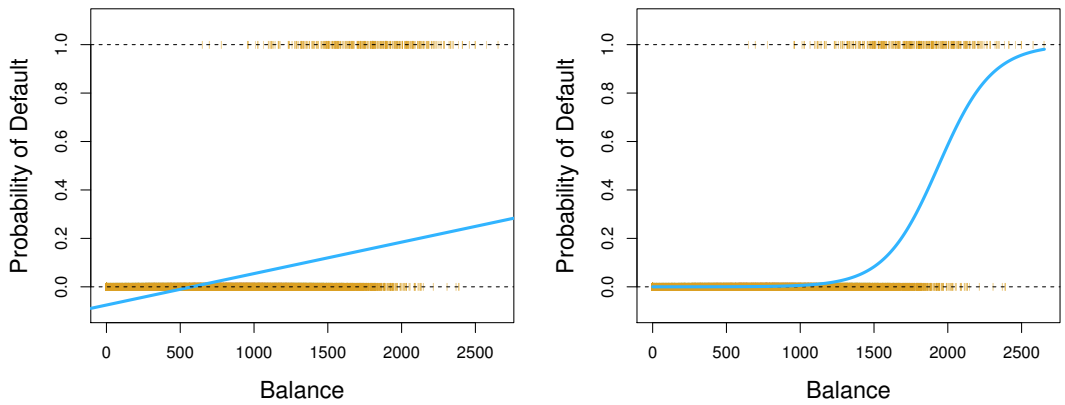

Figure: Left: estimated probabilities using a naive linear model. Right: estimated rpobabilities using logistic regression. The orange ticks indicate the classes (0/1) for various credit card balances.

Linear Discriminant Analysis

Bayes' Theorem & Classification

- An alternative approach is Linear Discriminant Analysis (LDA) where we model the distribution of the predictors X in each class (that is conditional on Y) and then use Bayes ' theorem to flip these and estimate the conditional distribution $\Pr(Y = k|X = x)$.
- Recall Bayes' Theorem

$$\Pr(A_i|B) = \frac{\Pr(B|A_i)\Pr(A_i)}{\Pr(B)} = \frac{\Pr(B|A_i)\Pr(A_i)}{\sum_j \Pr(B|A_j)\Pr(A_j)}$$

- Now suppose we have K possible classes for the dependent variable Y, let π_k represent the unconditional probability that a random observation comes from the k^{th} class and let

$$f_k(X) = \Pr(X = x|Y = y)$$

be the conditional density of x given that an observation comes from class k.

Linear Discriminant Analysis

LDA with one predictor

- We can use Bayes' theorem to express the conditional probability of interest as follows:

$$\Pr(Y = k|x = x) := p_k(x) = \frac{\pi_k f_k(x)}{\sum_{j=1}^{K} \pi_l f_l(x)}.$$

  The π_ks are easy to estimate; we simply need calculate the fraction of the training observations that belong to the k^{th} class. In order to estimate f_k however we will make some assumptions.

- First let us assume for simplicity that we only have one predictor which follows the normal distribution in each class but with common variance across classes:

$$f_k \sim N(\mu_k, \sigma),$$

Linear Discriminant Analysis

LDA with one predictor

- Then it follows that

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{1}{2\sigma^2}(x-\mu_k)^2\right)}{\sum_{l=1}^{K} \pi_l \frac{1}{\sqrt{2\pi}\sigma} exp\left(-\frac{1}{2\sigma^2}(x-\mu_l)^2\right)}$$

- The Bayes classifier implies assigning each observation to the class for which $p_k(x)$ is largest which we can show is equivalent to the class for which

$$\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log \pi_k$$

  is largest.

- Note that the word linear in this classifier comes from the fact that the discriminant functions δ_k are linear in x.

Linear Discriminant Analysis

LDA with one predictor

- We estimate the relevant quantities above using using

$$\hat{\mu}_k = \frac{1}{n_k} \sum_{i:y_i=k} x_i,$$

$$\hat{\pi}_k = \frac{n_k}{n}$$

and

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

the weighted (across classes) average of the sample variances of each class.

Linear Discriminant Analysis

LDA with multiple predictors

- In most applications of course we will have multiple predictors. It is relatively straightforward to show the above result generalizes to assigning each observation to the class for which

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

  is the largest.
- Again here we need to estimate the unknown parameters which is done in a similar fashion as before. Lets looks at an example with 3 classes and 2 predictors.
- A popular extension of LDA is Quadratic Discriminant Analysis (QDA). QDA assumes that each class has its own predictor covariance matrix ($\boldsymbol{\Sigma_k}$) and is very similar to LDA but allows for a non-linear decision boundary.

Linear Discriminant Analysis

LDA with multiple predictors

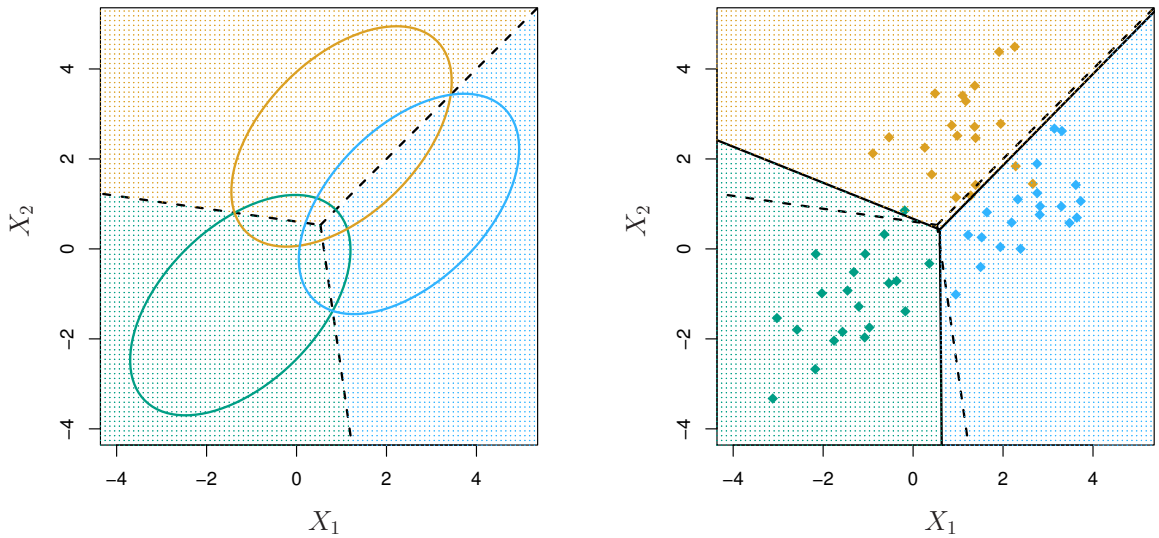

Figure: Left: Ellipses with 95% probability mass. Right: LDA decision boundaries (solid black) versus Bayes boundaries (dashed black) for 20 observations.

Linear Discriminant Analysis

Quadratic Discriminant Analysis

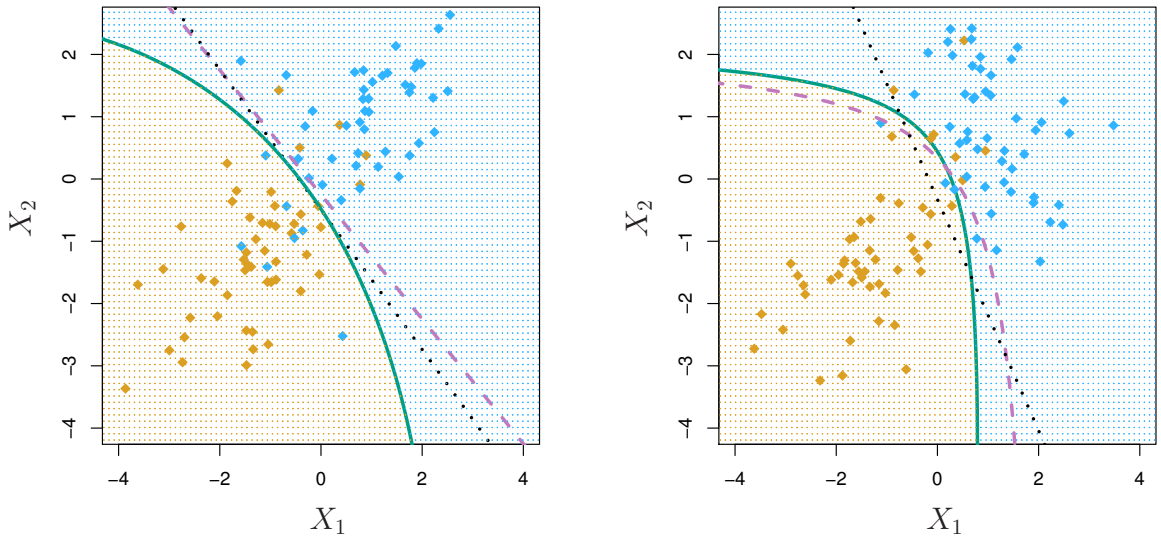

Figure: Bayes (purple dashed), LDA (black dotted) and QDA (green solid) decision boundariesfor a two case classification problem with $\mathbf{\Sigma_1} = \mathbf{\Sigma_2}$ (left) and $\mathbf{\Sigma_1} = \mathbf{\Sigma_2}$ (right).

Linear Discriminant Analysis

Error Analysis

- A binary classifier can generally make two types of errors: classify a yes response to the no category and vice versa. A *confusion matrix* is a common and informative way to display this information. Occasionally, however, we will be much more sensitive to one of the two types of errors!
- A classification technique that tries to approximate the Bayes classifier (which we know has the lowest *total* error rate of all classifiers treats both types of errors equally. However, in cases when we are concerned about one specific type of error (e.g. classifying a true Yes as a No) we can change the $\Pr \gtrless 0.5$ threshold of the Bayes classifier.
- An ROC Curve is a common way to display the two types of errors for all possible thresholds. The figure below shows the ROC curve for the LDA classifier on the 'default' data example from ISLR. The overall performance is given by the area under the ROC curve. A great classifier will have an ROC curve that hugs the top left corner.

Linear Discriminant Analysis

Error Analysis

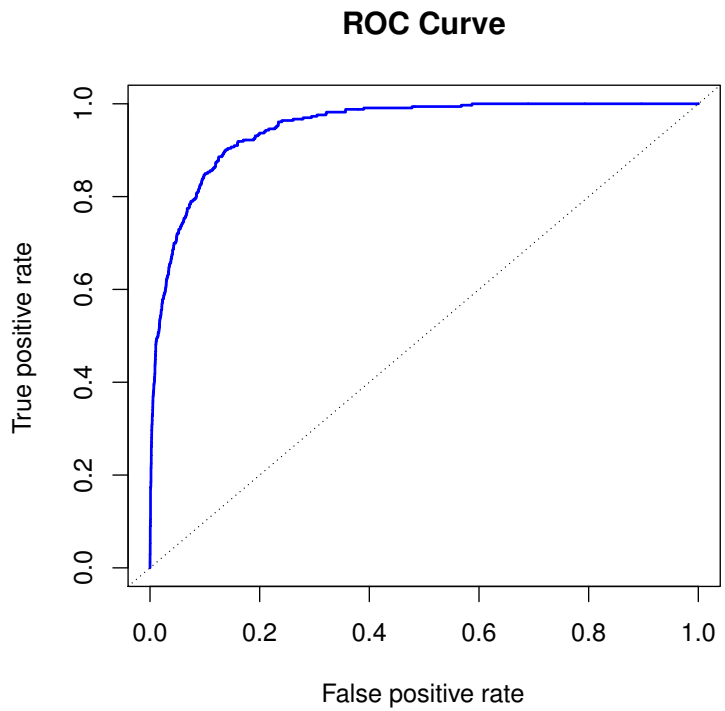

Figure: ROC curve for the default data example in ISLR. The dotted line is the "no-information" classifier. The actual thresholds are not shown in the figure.

Classification

A Brief Comparison of Methods

- When do we use kNN, Logistic Regression, LDA or QDA? There is no simple answer to this but here are some considerations:
- Typically, LDA is preferred to Logistic Regression when we have multiple classes ($k > 2$).
- When classes are well separated logistic regression can be unstable so LDA is preferable. Similarly, if n is small and the predictors are reasonable normally distributed, LDA is more stable than Logistic regression.
- If however the normality assumption is not realistic, Logistic regression is preferable.
- kNN, being a completely non-parametric approach can model even more highly non-linear decision boundaries. The downside is that exactly because it is non-parametric kNN gives no qualitative information. We do not know which predictors are important (or significant) and we do not get any coefficient estimates (e.g. sign of parameters can be relevant for policy). This is important for economists!
- QDA is useful when the training set is very large in which case the variance of the classifier is not a concern. Similarly LDA is preferable when we are confident that the common variance assumption is correct (bias not an issue) or when the sample is small (hence QDA would introduce high variance). Recall the variance/bias trade-off!

Notes

- These notes follow closely ISLR. Also, some of the figures in this presentation are taken from "An Introduction to Statistical Learning, with applications in R" (Springer, 2013) with permission from the authors: G. James, D. Witten, T. Hastie and R. Tibshirani
- Some suggested homework; you do not need to submit, it is here to help you understand the material:
- ISLR Chapter 2: Exercise 8
- ISLR Chapter 5: Exercise 8
- It is also **imperative** that you work through the lab sections of the covered chapters in ISLR. This means reproducing what you see in the book on your own R session!