Lecture 7: Classification Methods II

UNIVERSITY OF SAN FRANCISCO

James D. Wilson

MATH 373

Plan for this Lecture

- Linear Discriminant Analysis
- Quadratic Discriminant Analysis
- Logistic Regression
- Model Assessment via ROC and AUC
- A Comparison of Classification Methods

Reference: ISL Sections 4.3; 4.5

Recall: Bayes Classifiers

Training Data: n observations $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ with $y_i \in \{1, \ldots, m\}$

Test Data: Observations of the form $(\mathbf{x}_o, y_o)$.

Bayes Theorem gives the following relationship:

$$\mathbb{P}(Y = j \mid X = \mathbf{x}) = \frac{\pi_j f_j(\mathbf{x})}{f(\mathbf{x})} = \frac{\pi_j f_j(\mathbf{x})}{\sum_{j=1}^{m} \pi_j f_j(\mathbf{x})}$$

To calculate the Bayes classifier, we need

- Class conditional probabilities: $f_j(\mathbf{x})$ (*difficult!*)
- Prior probabilities π_j (*pretty easy*)

Discriminants

Definition

In many cases, we can view a classifier $\phi(\mathbf{x})$ as an optimization of some function of $\mathbf{x}$. Namely,

$$\phi(\mathbf{x}) = \text{argmax}_j \left(\delta_j(\mathbf{x})\right)$$

The function $\delta_j(\mathbf{x})$ is the discriminant of $\mathbf{x}$ as it is used to *discriminate* between classes of Y. Note that $\delta_j(\mathbf{x})$ is also the decision region for class $j \in \{1, \ldots, m\}$.

Example: For the Bayes classifier,

$$\delta_j(\mathbf{x}) = \mathbb{P}(Y = j \mid X = \mathbf{x})$$

Linear and Quadratic Discriminants

- The Bayes classifier discriminant need not take a simple form.
- We can talk about special (simple) cases of Bayes classifiers.
- We will talk about two classes of discriminants:
  - Linear Discriminants: $\delta_j(\mathbf{x})$ is a *linear* function of $\mathbf{x}$. For some matrices $\{A_j\}$ and vectors $\{\mathbf{b}_j\}$,

$$\delta_j(\mathbf{x}) = \mathbf{x}^T A_j + \mathbf{b}_j$$

  - Quadratic Discriminants: $\delta_j(\mathbf{x})$ is a *quadratic* function of $\mathbf{x}$. For some matrices $\{A_j\}$, $\{B_j\}$ and vectors $\{\mathbf{b}_j\}$,

$$\delta_j(\mathbf{x}) = \mathbf{x}^T A_j \mathbf{x} + \mathbf{x}^T B_j + \mathbf{b}_j$$

Linear and Quadratic Discriminants

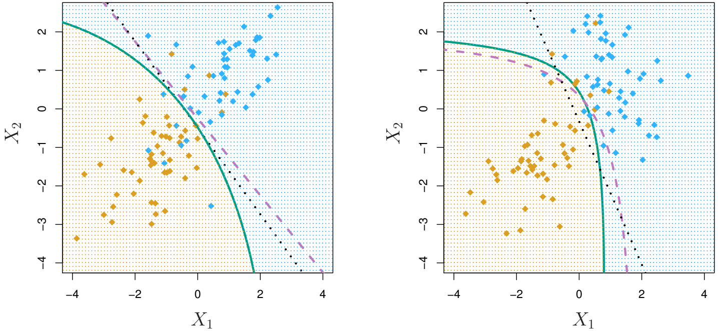


Figure: Linear vs. Quadratic Discriminants. The purple dashed line represents the true Bayes classifier.

Linear Discriminants: $p = 1$ case

- Suppose that there is only one predictor ($p = 1$) and Y takes on a class $j \in \{1, \ldots, m\}$
- **Distributional Assumption**: $X \mid Y = j$ is Guassian with mean μ_j and the *same* variance σ^2:

$$f_j(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}(x - \mu_j)^2\right)$$

- Then applying Bayes theorem and doing some algebra gives

$$\log\left(\mathbb{P}(Y = j \mid X = x)\right) = x * \frac{\mu_j}{\sigma^2} - \frac{\mu_j^2}{2\sigma^2} + \log(\pi_j)$$

Linear Discriminants: $p = 1$ case

Fact: Let $f(x) \geq 0$ for all x. Then maximizing $f(x)$ is equivalent to maximizing the function $g(x) = \log(f(x))$. (why?...)

Conclusion: If we assume that $X \mid Y = j$ as $N(\mu_j, \sigma^2)$, we can derive the discriminant function:

$$\delta_j(x) = x * \frac{\mu_j}{\sigma^2} - \frac{\mu_j^2}{2\sigma^2} + \log(\pi_j)$$

Question: We don't know μ_j and σ^2. How can we estimate them?

Linear Discriminants: $p = 1$ case

Let $Y \in \{1, \ldots, m\}$. Then we can estimate μ_j and σ^2 using

$$\hat{\mu}_j = \frac{1}{n_j} \sum_{i=1}^{n} x_i \, \mathbb{I}(y_i = j)$$

$$\hat{\sigma}^2 = \frac{1}{n - m} \sum_{j=1}^{m} \sum_{i=1}^{n} (x_i - \hat{\mu}_j)^2 \, \mathbb{I}(y_i = j)$$

As usual, we can estimate π_j using:

$$\hat{\pi}_j = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}(y_i = j)$$

Linear Discriminants: $p = 1$ case

The linear discriminants for Y are given by:

$$\hat{\delta}_j(x) = x * \frac{\hat{\mu}_j}{\hat{\sigma}^2} - \frac{\hat{\mu}_j^2}{2\hat{\sigma}^2} + \log(\hat{\pi}_j), \qquad j \in \{1, \ldots, m\}$$

In the simple case that $m = 2$, we can show that the Bayes decision boundary corresponds to the point where

$$x = \frac{\hat{\mu}_1^2 - \hat{\mu}_2^2}{2(\hat{\mu}_1 - \hat{\mu}_2)} = \frac{\hat{\mu}_1 + \hat{\mu}_2}{2}$$

Note: The Bayes decision boundary above is exactly the point where $\hat{\delta}_{-1}(x) = \hat{\delta}_{+1}(x)$

Linear Discriminants: Example

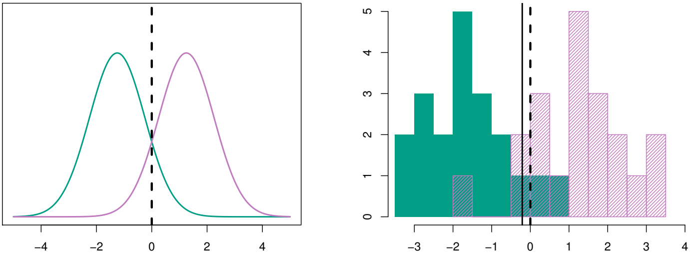


Figure: (Left): Two one-dimensional normal density functions. (Right): 20 observations were simulated from each of the two classes. The dashed black line represents the Bayes decision boundary; the black solid line on the right represents the LDA decision boundary.

Linear Discriminants: general $p > 1$ case

- Suppose there are $p > 1$ predictors and $Y \in \{1, \ldots, m\}$
- **Distributional Assumption**: $X \mid Y = j$ is multivariate Gaussian with mean μ_j and the *same* variance $\text{Cov}(X|Y = j) = \Sigma$:

$$f_j(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_j)^T \Sigma^{-1} (\mathbf{x} - \mu_j)\right)$$

- Then we can obtain the discriminant functions:

$$\delta_j(\mathbf{x}) := \log(\mathbb{P}(Y = j \mid X = \mathbf{x})) = \mathbf{x}^T \Sigma^{-1} \mu_j - \frac{1}{2}\mu_j^T \Sigma^{-1} \mu_j + \log(\pi_j)$$

Linear Discriminant Analysis (LDA) for general p

Suppose that $Y \in \{1, \ldots, m\}$ and $X \in \mathbb{R}^p$. The linear discriminant functions of $Y \mid X = \mathbf{x}$ are:

$$\hat{\delta}_j(\mathbf{x}) = \mathbf{x}^T \widehat{\Sigma}^{-1} \hat{\mu}_j - \frac{1}{2} \widehat{\Sigma}^{-1} \hat{\mu}_j + \log(\hat{\pi}_j), \qquad j = 1, \ldots, m$$

The Bayes decision boundaries are the values of $\mathbf{x}$ for which $\hat{\delta}_j(\mathbf{x}) = \hat{\delta}_\ell(\mathbf{x})$ for $j \neq \ell$, namely where

$$\mathbf{x}^T \widehat{\Sigma}^{-1} \hat{\mu}_j - \frac{1}{2} \hat{\mu}_j^T \widehat{\Sigma}^{-1} \hat{\mu}_j = \mathbf{x}^T \widehat{\Sigma}^{-1} \hat{\mu}_\ell - \frac{1}{2} \hat{\mu}_\ell^T \widehat{\Sigma}^{-1} \hat{\mu}_\ell$$

Summary of LDA

- Overall aim is to identify the Bayes classifier
- To acheive this, we assume that $X \mid Y = j \sim N(\mu_j, \Sigma)$
- We then estimate μ_j and Σ and calculate the discriminant functions $\delta_j(\mathbf{x}) = \log(\mathbb{P}(Y = j \mid X = \mathbf{x}))$
- $\delta_j(\mathbf{x})$ is a linear function of $\mathbf{x}$

Question: In many cases, we don't expect each observation $X \mid Y = j$ to have the same variance Σ. What if we allowed heteroscedacticity?

Quadratic Discriminants: general $p > 1$ case

- Suppose there are $p > 1$ predictors and $Y \in \{1, \ldots, m\}$
- **Distributional Assumption**: $X \mid Y = j$ is multivariate Gaussian with mean μ_j (potentially) *different* variances $\text{Cov}(X|Y = j) = \Sigma_j$:

$$f_j(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_j)^T \Sigma_j^{-1}(\mathbf{x} - \mu_j)\right)$$

- Then we can obtain the discriminant functions (via Bayes):

$$\begin{aligned}
\delta_j(\mathbf{x}) &:= \log(\mathbb{P}(Y = j \mid X = \mathbf{x})) \\
&= -\frac{1}{2}\mathbf{x}^T \Sigma_j^{-1}\mathbf{x} + \mathbf{x}^T \Sigma_j^{-1}\mu_j - \frac{1}{2}\mu_j^T \Sigma_j^{-1}\mu_j - \frac{1}{2}\log\left(|\Sigma_j|\right) + \log(\pi_j)
\end{aligned}$$

QDA for general p

Suppose that $Y \in \{1, \ldots, m\}$ and $X \in \mathbb{R}^p$. The quadratic discriminant functions of $Y \mid X = \mathbf{x}$ are:

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

Summary:

- We assume that $X \mid Y = j \sim N(\mu_j, \Sigma_j)$
- We then estimate μ_j and Σ and calculate the discriminant functions $\delta_j(\mathbf{x}) = \log(\mathbb{P}(Y = j \mid X = \mathbf{x}))$
- $\delta_j(\mathbf{x})$ is a quadratic function of $\mathbf{x}$

Linear vs. Quadratic Discriminant Analysis

Why should we ever use one method over another? The answer comes back to the bias / variance tradeoff.

- QDA is much more flexible, which often leads to high variance
  - QDA: $\frac{mp(p+1)}{2}$ parameters
  - LDA: mp parameters
- If the assumption of LDA that each $X \mid Y = j$ has the same covariance structure is wrong, then LDA will much higher bias than QDA.
- **Conclusion:** It again depends on the data and for the user to check assumptions.

Back to Regression

Setting: Y is binary, namely $Y \in \{-1, +1\}$ and fixed predictors $X \in \mathbb{R}^p$

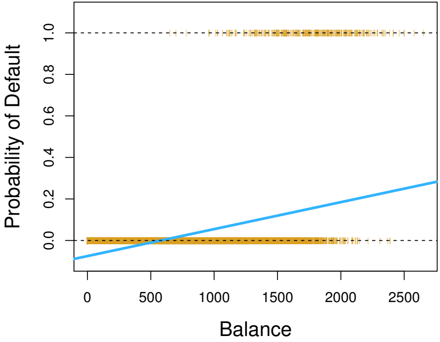
Question: How can we model $\mathbb{P}(Y = +1 \mid X = \mathbf{x})$ as a function of $\mathbf{x}$?

Standard Regression setting: We could use a linear model

$$\mathbb{P}(Y = +1 \mid X = \mathbf{x}) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

but...

Back to Regression



...using a linear model provides some values of $\mathbb{P}(Y = +1 \mid X = \mathbf{x})$ outside of 0 to 1!

The Logistic Function

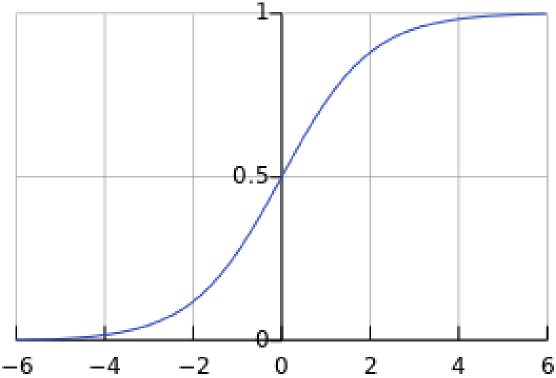
Instead, we can use a different model to ensure $\mathbb{P}(Y = +1 \mid X = \mathbf{x})$ is between 0 and 1:

$$\mathbb{P}(Y = +1 \mid X = \mathbf{x}) = \frac{\exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right)}{1 + \exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right)}$$

Here, $f(x) = \dfrac{e^x}{1 + e^x} \in (0, 1)$ is called the logistic function of x.

The Logistic Function

$$f(x) = \frac{e^x}{1 + e^x}$$



Logistic Regression

The model

$$\mathbb{P}(Y = +1 \mid X = \mathbf{x}) = \frac{\exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right)}{1 + \exp\left(\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right)}$$

can be rearranged and equivalently stated as:

$$\log\left(\frac{P(Y = +1 \mid X = \mathbf{x})}{1 - P(Y = +1 \mid X = \mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

The above model is called the logistic regression of Y on $X = \mathbf{x}$

Logistic Regression

Model: $$\log\left(\frac{P(Y=+1 \mid X=\mathbf{x})}{1-P(Y=+1 \mid X=\mathbf{x})}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

Features:

- $\dfrac{P(Y=+1 \mid X=\mathbf{x})}{1-P(Y=+1 \mid X=\mathbf{x})}$ is known as the odds of Y taking value $+1$
- $\log(\text{odds})$ is known as the logit or log-odds of Y taking value $+1$.
- The right hand side is linear in $\mathbf{x}$

Logistic Regression: Interpretation

Holding all other variables constant, increasing x_j by one unit changes the log odds of $Y = +1$ by β_j. Equivalently, increasing x_j by one unit multiplies the odds of $Y = +1$ by e^{β_j}.

Inference:

- $\beta_j < 0$: the odds of $Y = +1$ is decreased $\Rightarrow$ the probability of $Y = +1$ is decreased
- $\beta_j > 0$: the odds of $Y = +1$ is increased $\Rightarrow$ the probability of $Y = +1$ is increased
- $\beta_j = 0$: no effect on chances of $Y = +1$

Estimation via Maximum Likelihood

Goal:

- Estimate $\beta_0, \ldots, \beta_p$ via maximum likelihood
- Estimate $\mathbb{P}(Y = +1 \mid X = \mathbf{x})$ by plugging in the above estimates into the logistic function

Methodology: Identify $\hat{\beta}_0, \ldots, \hat{\beta}_p$ that maximizes the likelihood:

$$L(\beta \mid Y = y) = \prod_{i=1}^{n} \mathbb{P}(Y = +1 \mid X = \mathbf{x}_i)^{y_i} \mathbb{P}(Y = 0 \mid X = \mathbf{x}_i)^{1-y_i}$$

Important Fact: The maximum likelihood estimate (MLE) of $\hat{\beta}_j$ has an *approximate* Gaussian distribution with mean β_j. Therefore, statistical inference can be conducted the same as OLS.

Estimation via Maximum Likelihood

Methodology, continued... Equivalently, we can maximize the log-likelihood of $\beta_0, \beta \mid Y = y$, which we can simplify as follows:

$$
\begin{aligned}
\ell(\beta_0, \beta \mid Y = y) &= \log(L(\beta_0, \beta \mid Y = y)) \\
&= \sum_{i=1}^{n} [y_i \log(\mathbb{P}(Y = +1 \mid X = \mathbf{x}_i)) \\
&+ (1 - y_i) \log(\mathbb{P}(Y = 0 \mid X = \mathbf{x}_i))] \\
&= \sum_{i=1}^{n} \exp(\beta_0 + \mathbf{x}_i^T \beta) + \sum_{i=1}^{n} [y_i(\beta_0 + \mathbf{x}_i^T \beta)]
\end{aligned}
$$

Estimation via Maximum Likelihood

There is no analytical form for $\hat{\beta}$ that maximizes the log-likelihood (unlike OLS for standard regression). So, we must resort to a computational means using methods like:

- Gradient descent methods for each β_j or
- Fisher scoring algorithm

Once we obtain $\hat{\beta}$, we can calculate:

$$\hat{\mathbb{P}}(Y = +1 \mid X = \mathbf{x}) = \frac{\exp\left(\hat{\beta}_0 + \sum_{j=1}^{p} \hat{\beta}_j x_j\right)}{1 + \exp\left(\hat{\beta}_0 + \sum_{j=1}^{p} \hat{\beta}_j x_j\right)}$$

The binary classifier is defined as

$$\phi(x) = \text{argmax}_j \{\hat{\mathbb{P}}(Y = j \mid X = \mathbf{x})\}$$

Or, equivalently,

$$\phi(x) = \text{argmax}_j \{\text{logit}\left(\hat{\mathbb{P}}(Y = j \mid X = \mathbf{x})\right)\}$$

Hence, the discriminant function is given by

$$\delta_j(\mathbf{x}) = (-1)^{j-1}(\hat{\beta}_0 + \sum_{i=1}^{p} \hat{\beta}_i x_i)$$

Conclusion: Logistic regression gives a linear discriminant!

Summary of Logistic Regression

- Inference-based: β_j describes the multiplicative effect of x_j on the odds of $Y = +1$
- For binary classification only! (though there are multi-class extensions)
- Provides linear discriminants $\delta_j = \log(\mathbb{P}(Y = j \mid X = \mathbf{x}))$
- Estimates found via maximum likelihood + gradient descent / Fisher scoring algorithms

Classification via Thresholding

Issue: In binary classification settings, we often choose a class using a threshold τ. For example, we choose $Y = +1$ if

$$P(Y = +1 \mid X = \mathbf{x}) > \tau$$

So far, we've typically used $\tau = 0.5$.

Point: The error rate will change based on the threshold value τ that we choose.

Question: How can we assess the performance of a method based on τ?

Receiver Operating Characteristics (ROC)

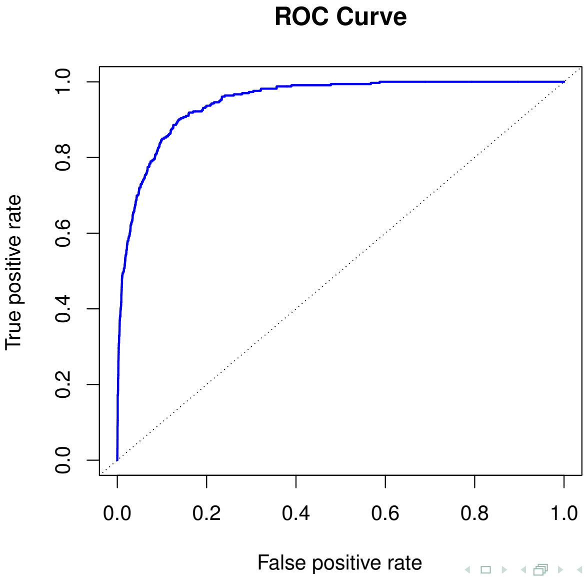
The receiver operating characteristics (ROC) of a binary classifier are the

- true positive rate (sensitivity)
- false positive rate (1 - specificity)

for the classifier across a grid of the threshold τ.

The ROC curve plots the comparison of these two quantities across τ.

ROC curve


ROC Curve
True positive rate
0.0
0.2
0.4
0.6
0.8
1.0
False positive rate
0.0
0.2
0.4
0.6
0.8
1.0

The Area Under the Curve (AUC)

The area under the curve (AUC) is the area under the ROC curve.

Features:

- AUC $\in [0, 1]$
- Measures the overall performance of a classifier
- The higher the better.
- We expect a classifier that performs no better than chance to have an AUC of 0.5 on an independent test set.

A Comparison of Classification Methods

Bayesian Methods: (X, Y) are jointly distributed

- Bayes classifier:
  - Classifier based on $\mathbb{P}(Y = j \mid X = \mathbf{x})$
  - Impossible to calculate probabilities, but if we could, the classifier would be the best.
- Naïve Bayes:
  - Assumes predictors $X_k \mid Y = j$ are conditionally independent

A Comparison of Classification Methods

Bayesian Methods: (X, Y) are jointly distributed

- LDA:
  - $X \mid Y = j \sim N(\mu_j, \Sigma)$
  - Discriminant functions are linear in **x**
- QDA:
  - $X \mid Y = j \sim N(\mu_j, \Sigma_j)$
  - Discriminant functions are quadratic in **x**

A Comparison of Classification Methods

Non-parametric Methods: No assumption on (X, Y)

- K-Nearest Neighbors
  - Classify test sample $\mathbf{x}_o$ based on the K points that are closest in the training set.

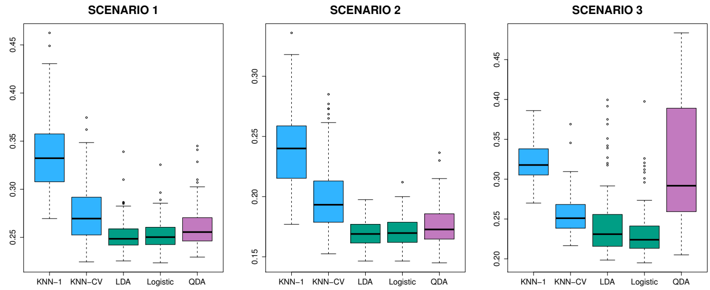
Frequentist Regression Methods: X is a fixed value

- Logistic Regression
  - Classify using the log - odds of Y:

$$\log\left(\frac{\mathbb{P}(Y = +1)}{\mathbb{P}(Y = -1)}\right) = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

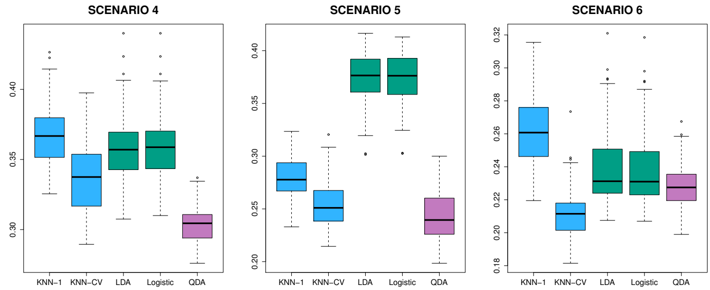
  - β parameters have meaning – inference-based

A Comparison of Classification Methods



- **Scenario 1**: $X \mid Y = j \sim N(\mu_j, \Sigma)$, $j = 1, 2$ and independent
- **Scenario 2**: $X \mid Y = j \sim N(\mu_j, \Sigma)$, $j = 1, 2$ with correlation $\rho = -0.5$
- **Scenario 3**: $X \mid Y = j \sim t(50)$ and independent

A Comparison of Classification Methods



- **Scenario 4**: $X \mid Y = j \sim N(\mu_j, \Sigma_j)$, $j = 1, 2$ and independent
- **Scenario 5**: Simulated so there is a quadratic boundary
- **Scenario 6**: Simulated so there is a complex non-linear boundary

A Comparison of Classification Methods

Summary:

- No method works best *all* the time. The choice depends on the problem, and each method varies in assumptions and peformance.
- Linear boundaries ⇒ LDA, Logistic
- Quadratic boundaries ⇒ QDA
- Non-linear boundaries ⇒ K-NN, Empirical Naïve Bayes

Take-away: You need to thoroughly explore the data, and try several methods to see which works best.

Next Up

- Implementing classification methods in R
- Decision trees
- Random Forests
- Bagging and Boosting