COMP6915 Machine Learning

Linear Methods for Classification

Dr. Lourdes Peña-Castillo
Departments of Computer Science and Biology
Memorial University of Newfoundland

Linear Models for Classification

- In many cases, the response variable is qualitative or categorical.
- In these cases, we need a method that models the posterior probabilities $\Pr(Y = k \mid X = x)$ where k is the class of instance x
  - Note that $\Pr(Y = k \mid X = x)$ must be modelled with a function that gives outputs between 0 and 1 for all values of X
- If $\Pr(Y = k \mid X = x)$ is linear in X then the decision boundaries will be linear and we can use a linear model.

Logistic Regression

- Suppose there are *K* classes labeled 1,2, ..., *K*. Logistic regression models the posterior probabilities Pr(Y=*k* | X = *x*) of the *K* classes via linear functions in *x*, while ensuring that they sum to one and remain in [0,1]

The model is specified in terms of $K-1$ logit transformations:

$$log\frac{Pr(Y=1|X=x)}{Pr(Y=K|X=x)} = \beta_{10} + \beta_1^T x$$
$$log\frac{Pr(Y=2|X=x)}{Pr(Y=K|X=x)} = \beta_{20} + \beta_2^T x$$
$$\vdots$$
$$log\frac{Pr(Y=K-1|X=x)}{Pr(Y=K|X=x)} = \beta_{(K-1)0} + \beta_{K-1}^T x$$

It can be shown that

$$Pr(Y = k|X = x) = \frac{e^{\beta_{k0}+\beta_k^T x}}{1+\sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_l^T x}}, \; k = 1, \ldots, K-1.$$

$$Pr(Y = K|X = x) = \frac{1}{1+\sum_{l=1}^{K-1} e^{\beta_{l0}+\beta_l^T x}}.$$

Logistic Regression

- However, logistic regression tends to be mostly used for binary classification
- When *K* = *2*, the logistic regression model is only a single linear function

Let $p(X) = Pr(Y = 1|X)$ then

$$log(\frac{p(X)}{1-p(X)}) = \beta_0 + \sum_{i=1}^{p} \beta_i X_i,$$

where $X = (X_1, \ldots, X_p)$ are p predictors. The left-hand side is called the *log-odds* or *logit*. This equation can be rewritten as

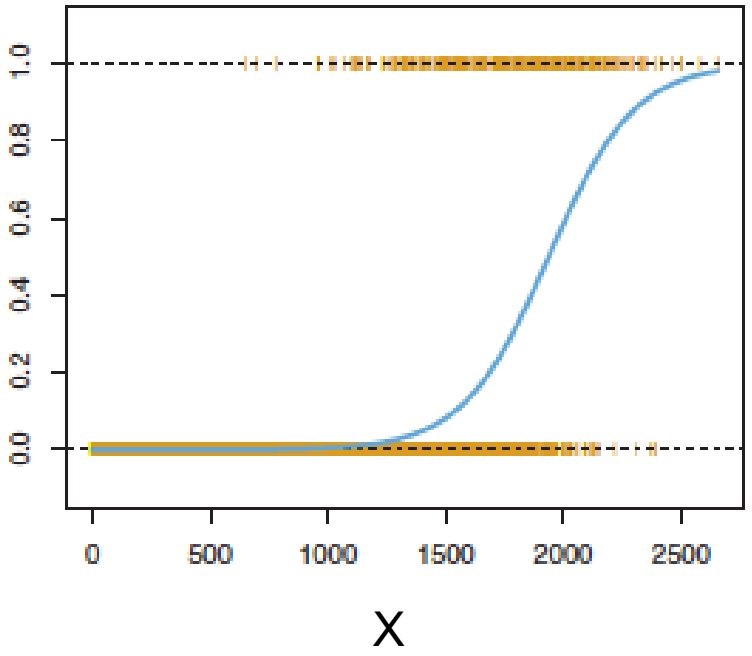
$$p(X) = \frac{e^{\beta_0 + \sum_{i=1}^{p} \beta_i X_i}}{1 + e^{\beta_0 + \sum_{i=1}^{p} \beta_i X_i}}.$$

The maximum likelihood method is used to estimate $\beta_0, \beta_1, \ldots, \beta_p$.

Logistic Regression

- The logistic function always produce an *S-shaped* curve regardless of the value of *X*

Probability of Y = 1



$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}$$

Fig. adapted from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Logistic Regression

- The decision boundary is the set of points for which the log-odds are zero.
- The logistic regression model has a logit that is linear in X
- If a coefficient β_i is positive then increasing X_i will increase $p(X)$, and if β_i is negative then increasing X_i will decrease $p(X)$
- One might predict Y = 1 for any instance for which p(X) > *threshold*

Finding the Coefficients

The likelihood function is:

$$L(\beta) = \prod_{i=1}^{n} p(y_i \mid x_i; \beta)$$

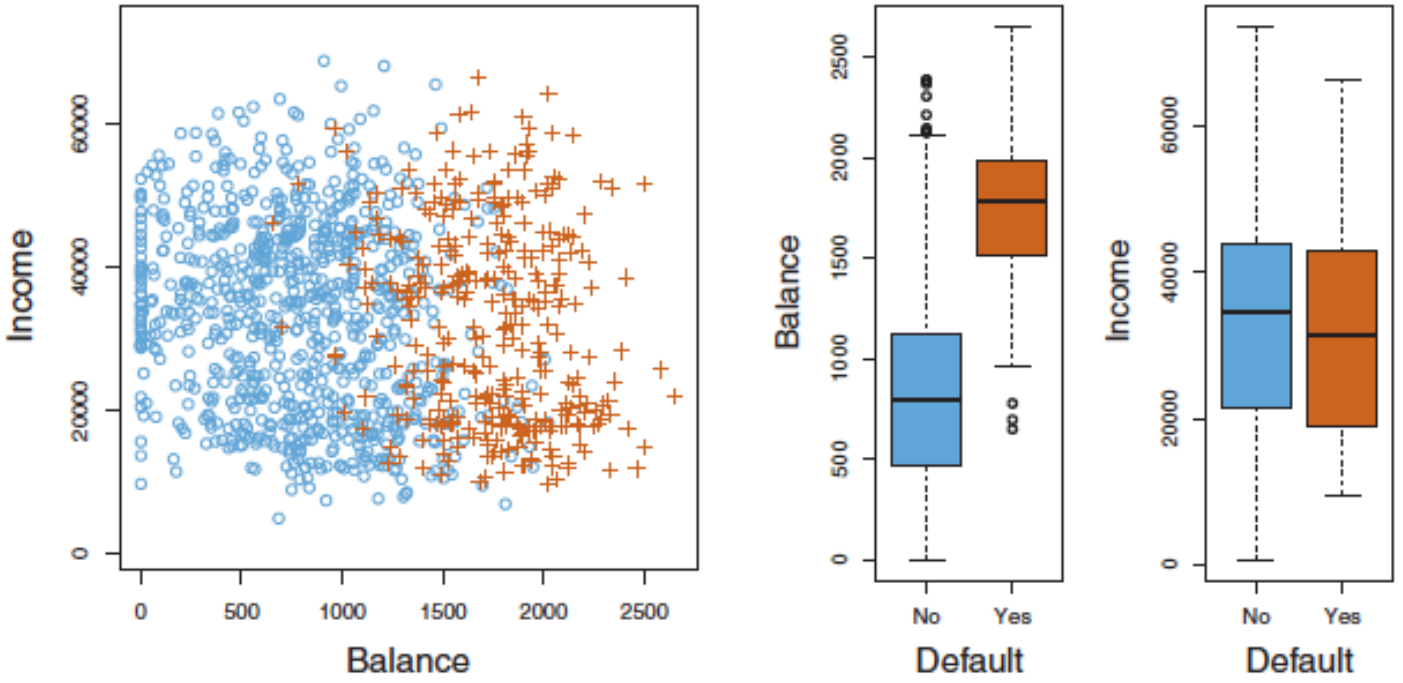
The maximum likelihood method choose β to maximize this function. It is easier to maximize the log likelihood function:

$l(\beta) = log(L(\beta))$.

To find β, SGD can be used or any statistical package.

Example

- Suppose we want to predict using logistic regression whether an individual will default on their credit card payment on the basis of annual income, monthly credit card balance and student status



Example and Figure from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Making Predictions

	Coefficient	Std. error	Z-statistic	P-value
Intercept	−10.6513	0.3612	−29.5	<0.0001
balance	0.0055	0.0002	24.9	<0.0001

TABLE 4.1. *For the* Default *data, estimated coefficients of the logistic regression model that predicts the probability of* default *using* balance*. A one-unit increase in* balance *is associated with an increase in the log odds of* default *by* 0.0055 *units.*

The default probability for an individual with a balance of $1,000 is:

$$\hat{p}(X) = \frac{e^{\hat{\beta}_0+\hat{\beta}_1 X}}{1+e^{\hat{\beta}_0+\hat{\beta}_1 X}} = \frac{e^{-10.6513+0.0055\times 1{,}000}}{1+e^{-10.6513+0.0055\times 1{,}000}} = 0.00576$$

Fig. from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Confounding

- Suppose that we construct a model of the probability of default using only the feature is_student and the coefficient associated with this feature is 0.4049. However, when we added the features balance and income to the model the coefficient associated with the feature is_student was negative. Why?

	Coefficient	Std. error	Z-statistic	P-value
Intercept	−3.5041	0.0707	−49.55	<0.0001
student[Yes]	0.4049	0.1150	3.52	0.0004

	Coefficient	Std. error	Z-statistic	P-value
Intercept	−10.8690	0.4923	−22.08	<0.0001
balance	0.0057	0.0002	24.74	<0.0001
income	0.0030	0.0082	0.37	0.7115
student[Yes]	−0.6468	0.2362	−2.74	0.0062

Confounding

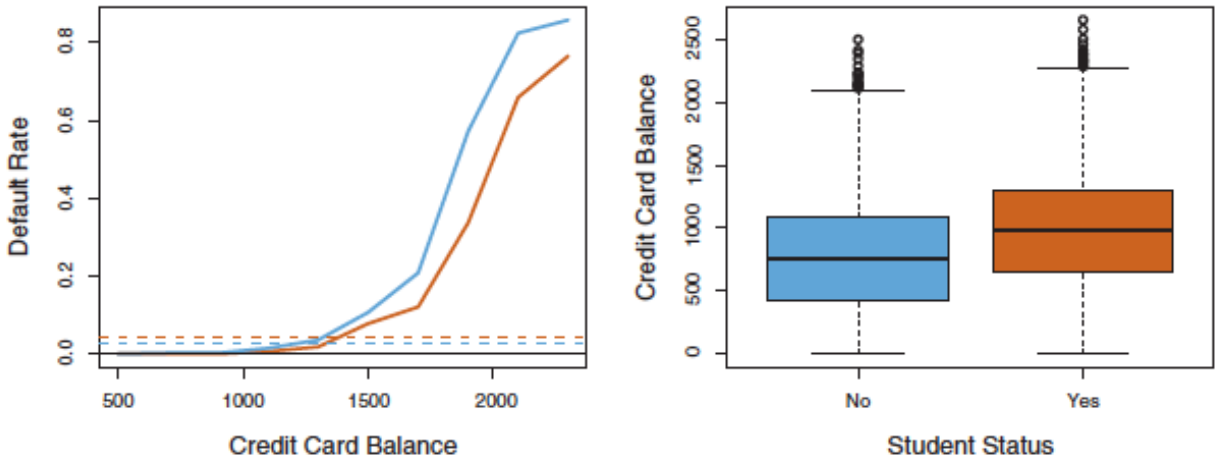


FIGURE 4.3. *Confounding in the* `Default` *data.* Left: *Default rates are shown for students (orange) and non-students (blue). The solid lines display default rate as a function of* `balance`*, while the horizontal broken lines display the overall default rates.* Right: *Boxplots of* `balance` *for students (orange) and non-students (blue) are shown.*

- Interpretation:
  - A student is less risky that a non-student with the same credit card balance
- Confounding occurs when features are correlated
- Results of linear models significantly change depending on the features included.
- It is important to include (all) relevant features

Example and Figure from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Linear Discriminant Analysis (LDA)

- Alternative model to estimate the probabilities $Pr(Y=k \mid X= x)$ when K > 2
- For the case of binary classes, linear discriminant analysis is more stable than logistic regression for small training sets and well-separated classes

LDA Basis

Suppose $f_k(x)$ is the class-conditional density of X in class $Y = k$, and π_k is the prior probability of class k, with $\sum_{k=1}^{K} \pi_k = 1$. By applying Bayes theorem, we get $Pr(Y = k|X = x) = \frac{f_k(x)\pi_k}{\sum_{l=1}^{K} f_l(x)\pi_l}$.

This suggests that instead of computing the posterior probability $p_k(X)$ as done by logistic regression, we can estimate π_k and $f_k(X)$ and insert them into the equation above.

If we have a random sample of Ys from the population, π_k can be estimated by computing the fraction of the training observations that belong to the kth class. Estimating $f_k(X)$ is more challenging unless one assumes some simple forms for these densities.

LDA Basis

Linear discriminant analysis (LDA) arises in the special case when each class density is modelled as a multivariate Gaussian and it is assumed that the classes have a common covariance matrix $\sum_k = \sum \forall k$.

LDA Basis

In the one-dimensional setting $p = 1$, the normal density takes the form

$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2\sigma_k^2}(x-\mu_k)^2}$ where μ_k and σ_k^2 are the mean and variance for the kth class.

LDA assumes that $\sigma_1^2 = \ldots = \sigma_k^2$: i.e., there is a shared variance term across all K classes, which for simplicity is denoted by σ^2. Then

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

An observation $X = x$ is assigned to the class k for which $p_k(x)$ is largest.

Taking the log of the equation above and rearranging the terms, it can be shown that this is equivalent to assigning the observation to the class for which the *discriminant function*

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

is largest.

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

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

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

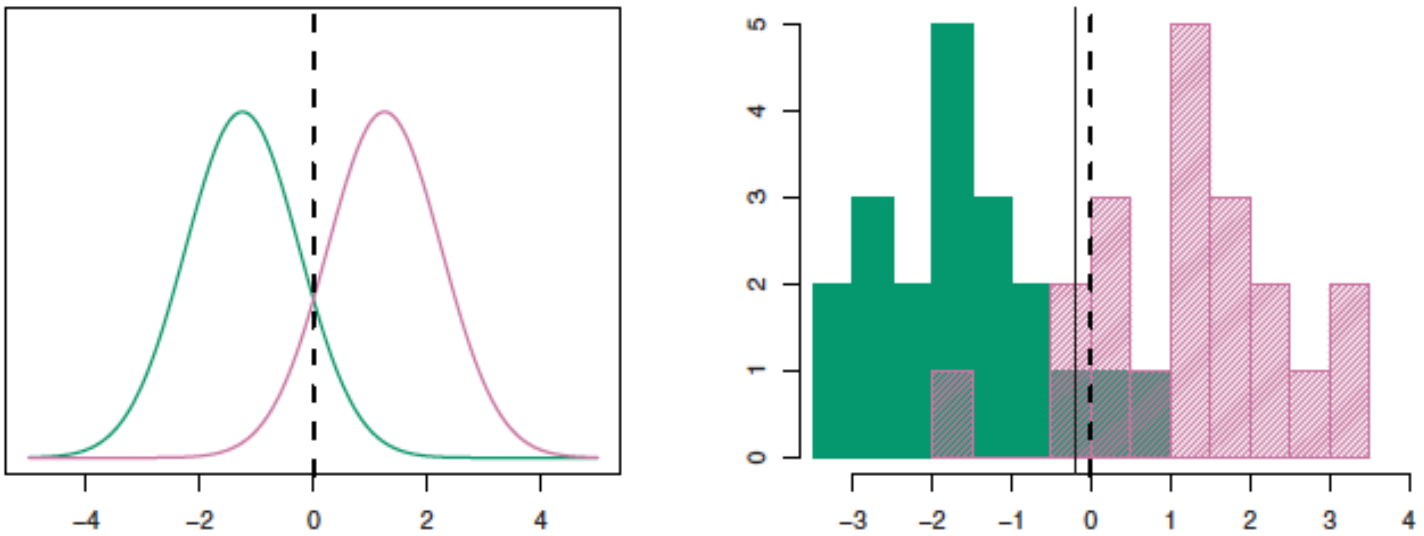


FIGURE 4.4. Left: *Two one-dimensional normal density functions are shown. The dashed vertical line represents the Bayes decision boundary.* Right: *20 observations were drawn from each of the two classes, and are shown as histograms. The Bayes decision boundary is again shown as a dashed vertical line. The solid vertical line represents the LDA decision boundary estimated from the training data.*

Fig. from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Multidimensional LDA

- Remember LDA assumes that $X = (X_1, X_2, ..., X_p)$ is drawn from a multivariate Gaussian (or normal) distribution with a class-specific mean vector and a common covariance matrix
- To indicate that a p-dimensional random variable X has a multivariate Gaussian distribution, we write $X \sim N(\mu, \Sigma)$ where μ is the mean of X (a vector with p components), and Σ is the $p \times p$ covariance matrix of X
- The multivariate Gaussian density is defined as:

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

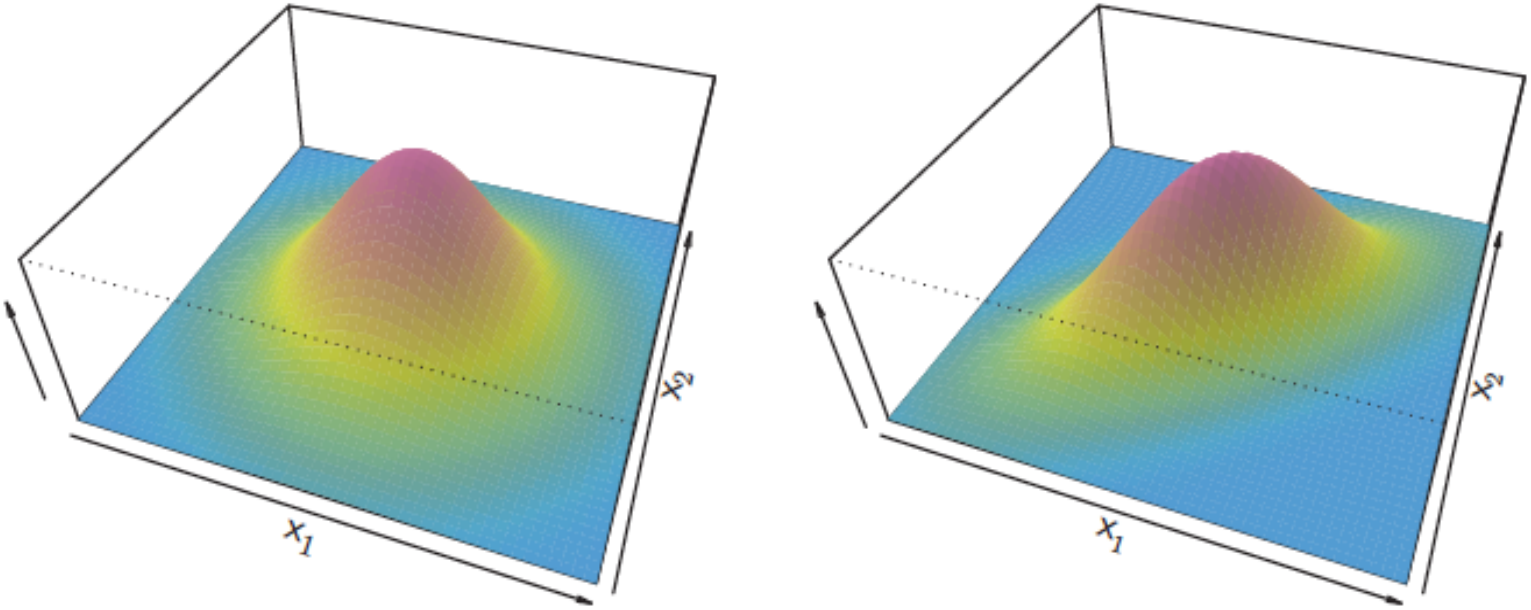


FIGURE 4.5. *Two multivariate Gaussian density functions are shown, with* $p = 2$. Left: *The two predictors are uncorrelated.* Right: *The two variables have a correlation of* 0.7.

Figure from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Estimating $f_k(X)$

- LDA classifier assumes that the observations in the *k* class are drawn from a multivariate Gaussian distribution $N(\mu_k, \Sigma)$, where μ_k is a class-specific mean vector and Σ is a covariance matrix that is common to all *K* classes
- We can then plug the density function for the *k*th class $f_k(X = x)$ into this equation:

$$\Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}$$

and doing some algebra we get that the Bayes classifier assigns an observation $X = x$ to the class for which the discriminant function below is the largest:

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

Then, ...

- We need to estimate the unknown parameters $\mu_1,..,\mu_k,\pi_1,...,\pi_k$ and Σ to plug them into the discriminant function
- We estimate these parameters from the data.
  - $\hat{\pi}_k = N_k/N;$
  - $\hat{\mu}_k = \sum_{y_i=k} x_i/N_k;$
  - $\hat{\Sigma} = \sum_{k=1}^{K} \sum_{y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T/(N-K)$

where N is the total number of training instances, N_k is the number of training examples in the k class, and K is the total number of classes.

Then, ...

- In simpler words:
  - The estimate for μ_k is the average of all the training examples from the *k* class.
  - The estimate for π_k is the proportion of the training examples that belong to the *k* class
  - The estimate for Σ is the weighted average of individual covariance matrices

LDA Example

- $\hat{\pi}_k = N_k/N;$
- $\hat{\mu}_k = \sum_{y_i=k} x_i/N_k;$
- $\hat{\sum} = \sum_{k=1}^{K} \sum_{y_i=k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T/(N-K)$

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

Visualizing LDA

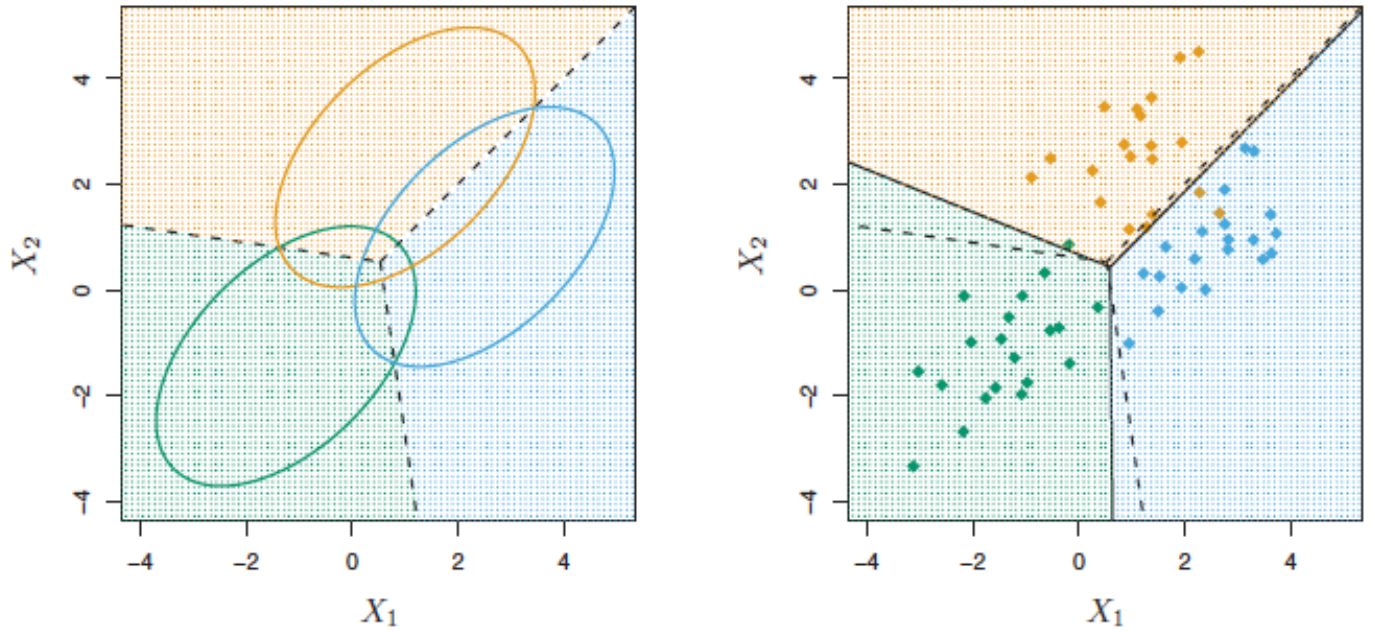


FIGURE 4.6. *An example with three classes. The observations from each class are drawn from a multivariate Gaussian distribution with $p = 2$, with a class-specific mean vector and a common covariance matrix.* Left: *Ellipses that contain 95 % of the probability for each of the three classes are shown. The dashed lines are the Bayes decision boundaries.* Right: 20 *observations were generated from each class, and the corresponding LDA decision boundaries are indicated using solid black lines. The Bayes decision boundaries are once again shown as dashed lines.*

LDA tries to approximate the Bayes classifier (if the Gaussian model is correct).

Figure from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Decision Boundaries

- In the previous slide, Bayes decision boundaries represent the set of values x for which $\delta_k(x) = \delta_l(x)$ for $k \neq l$. That is,

$$x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k = x^T \Sigma^{-1} \mu_l - \frac{1}{2} \mu_l^T \Sigma^{-1} \mu_l$$

The log π_k term has disappeared because is the same for each class.

- Note that there will be as many decision boundaries as pairs of classes

LDA or Logistic Regression?

- Consider the two-class setting with $p = 1$ feature, and let $p_1(x)$ and $p_2(x) = 1 - p_1(x)$ be the probabilities that the instance $X=x$ belongs to class 1 and class 2, respectively
- In LDA, the log odds are given by:

$$\log\left(\frac{p_1(x)}{1-p_1(x)}\right) = \log\left(\frac{p_1(x)}{p_2(x)}\right) = c_0 + c_1 x$$

where c_0 and c_1 are functions of μ_1, μ_2 and σ^2, and in logistic regression the log odds are given by:

$$\log\left(\frac{p_1}{1-p_1}\right) = \beta_0 + \beta_1 x$$

LDA or Logistic Regression?

- Both are linear functions of *x*
- Both produce linear decision boundaries
- Both tell us which features are important
- The only difference is their fitting procedure:
  - logistic regression estimates its coefficients by maximum likelihood while LDA's coefficients are computed using the estimated mean and variance from a normal distribution

LDA or Logistic Regression?

- Logistic regression can be seen as estimating the marginal density in a nonparametric fashion
- By relying on the additional model assumptions, LDA has more information about the parameters and hence can estimate them more efficiently (lower variance)
- In practice LDA assumptions are ofter incorrect and often some of the components of X are qualitative variables. Then, logistic regression (by relying on fewer assumptions) is safer and more robust than the LDA model
- Often both methods give very similar results for many tasks

Learning non-linear Boundaries

- By adding additional variables is possible to learn non-linear boundaries using LDA
  - For example, adding the squares of the features and cross-products
- These non-linear boundaries are linear in the augmented feature space

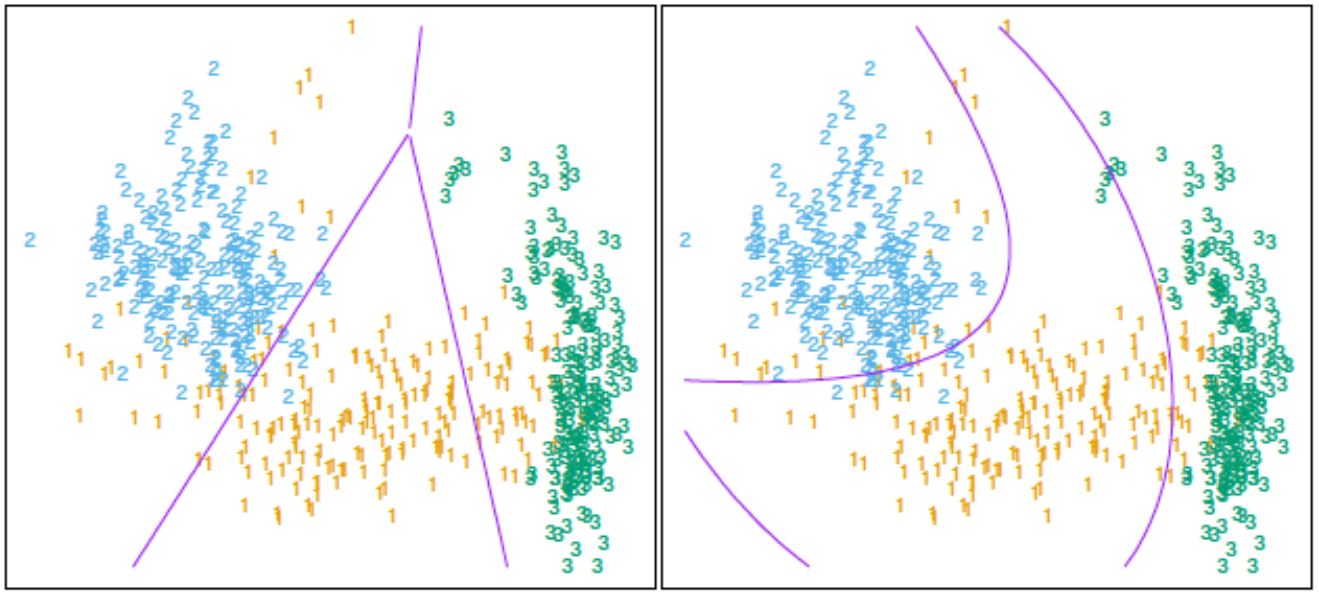


FIGURE 4.1. *The left plot shows some data from three classes, with linear decision boundaries found by linear discriminant analysis. The right plot shows quadratic decision boundaries. These were obtained by finding linear boundaries in the five-dimensional space* $X_1, X_2, X_1X_2, X_1^2, X_2^2$. *Linear inequalities in this space are quadratic inequalities in the original space.*

Figure from *The Elements of Statistical Learning* by Hastie, Tibshirani and Friedman, 2009.

Quadratic Discriminant Analysis (QDA)

- If the covariance matrices are not assumed to be equal, we get the discriminant quadratic function
- Like LDA, QDA assumes that the observations from each class are drawn from a Gaussian distribution. However, unlike LDA, QDA assumes that each class has its own covariance matrix
- That is, QDA assumes that an observation from the *k*th class is of the form $X \sim N(\mu_k, \Sigma_k)$, where Σ_k is a covariance matrix for the *k*th class
- QDA assigns an observation $X = x$ to the class for which the discriminant function below is the largest

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

LDA or QDA?

- When there are *p* predictors and *K* classes, LDA estimates *(K-1)*(p+1)* parameters; while, QDA estimates *(K-1)*(p(p+3)/2+1)* parameters
- So, it is a bias variance tradeoff – the bias of a linear decision boundary can be estimated with much lower variance
  - LDA is a less flexible classifier than QDA, and has lower variance
  - LDA tends to be better than QDA if there are relatively few training observations
  - QDA is recommended if the training set is large or if the assumption of a common covariance matrix for the *K* classes is clearly off

Learning non-linear Boundaries

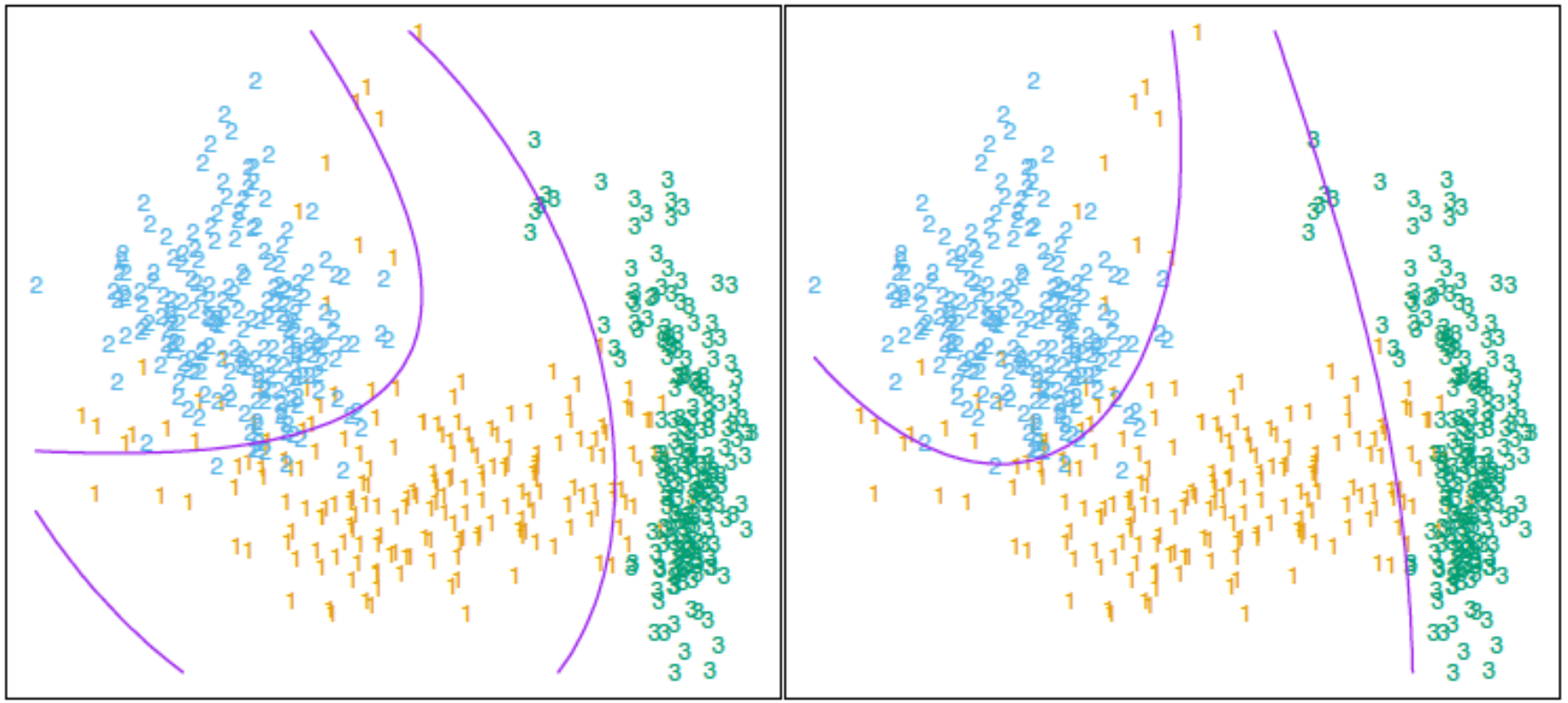


FIGURE 4.6. *Two methods for fitting quadratic boundaries. The left plot shows the quadratic decision boundaries for the data in Figure 4.1 (obtained using LDA in the five-dimensional space $X_1, X_2, X_1X_2, X_1^2, X_2^2$). The right plot shows the quadratic decision boundaries found by QDA. The differences are small, as is usually the case.*

Figure from *The Elements of Statistical Learning* by Hastie, Tibshirani and Friedman, 2009.

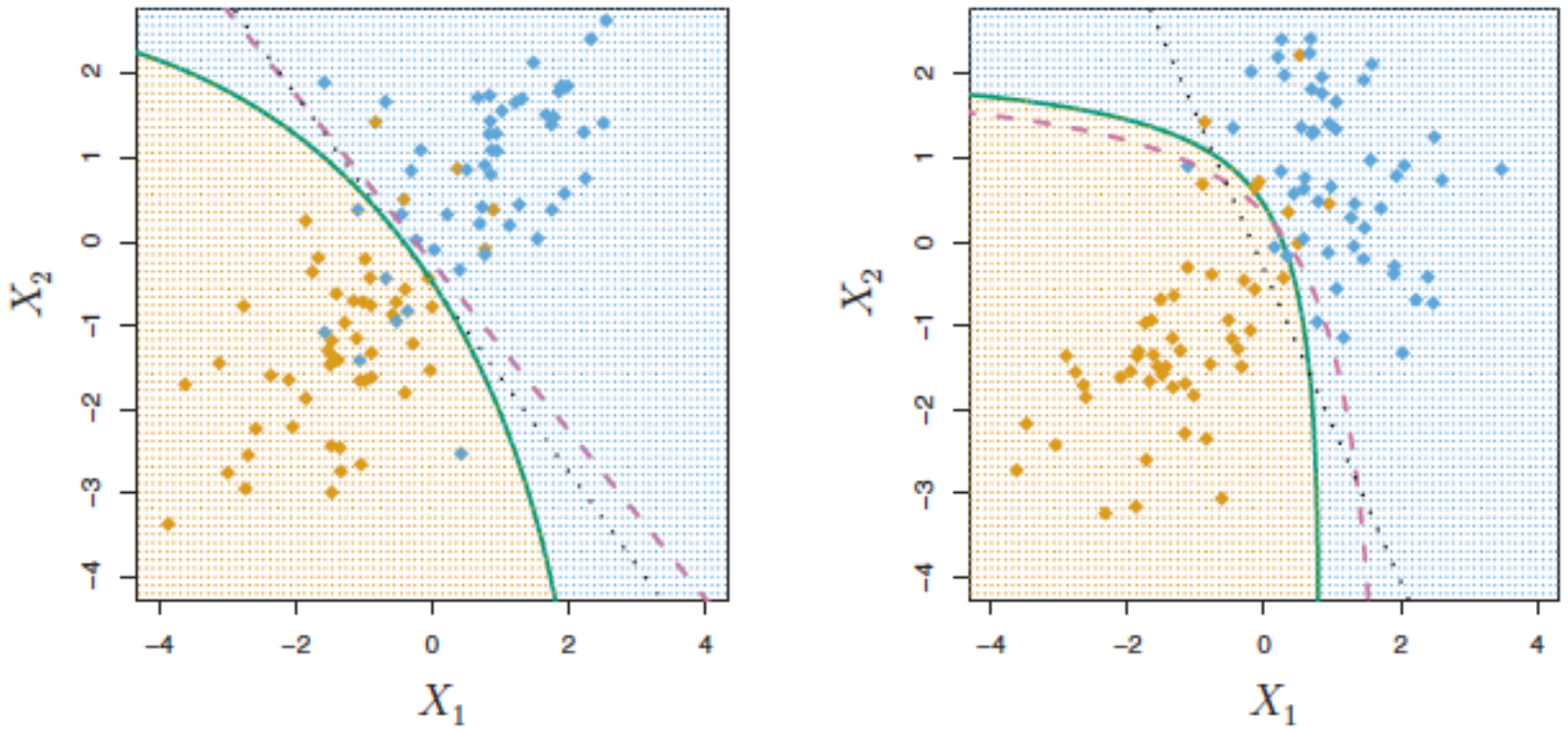


FIGURE 4.9. Left: *The Bayes (purple dashed), LDA (black dotted), and QDA (green solid) decision boundaries for a two-class problem with* $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2$. *The shading indicates the QDA decision rule. Since the Bayes decision boundary is linear, it is more accurately approximated by LDA than by QDA.* Right: *Details are as given in the left-hand panel, except that* $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$. *Since the Bayes decision boundary is non-linear, it is more accurately approximated by QDA than by LDA.*

Figure from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

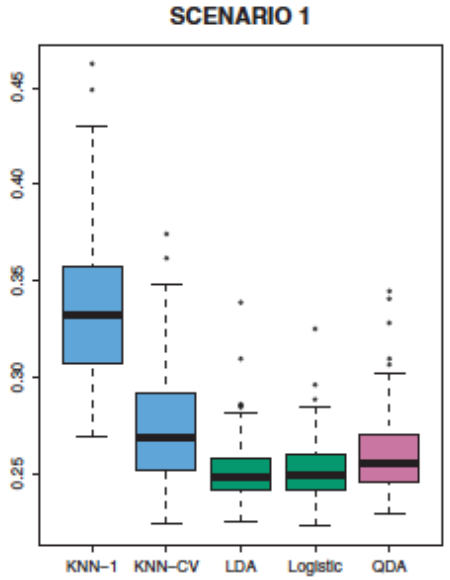
Comparing Classification Methods ($p = 2$, $n = 100$)

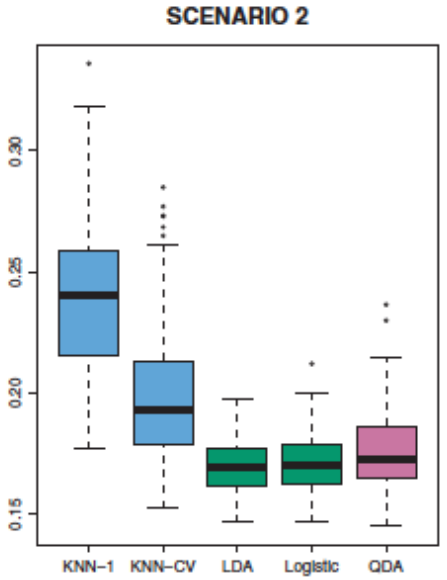
Uncorrelated normal variables with a different mean in each class

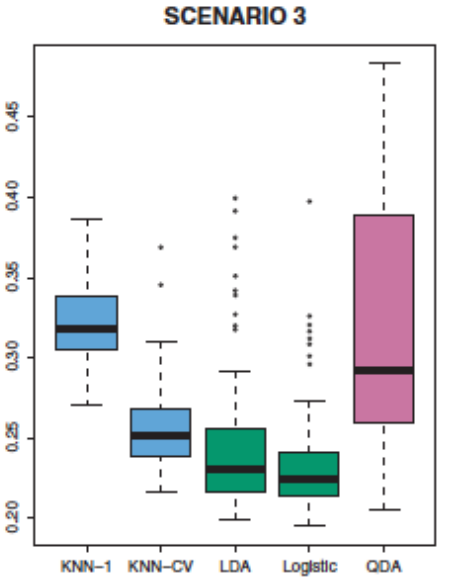
As 1, but predictors had a correlation of -0.5

Predictors generated from the *t-distribution*

Test error rates







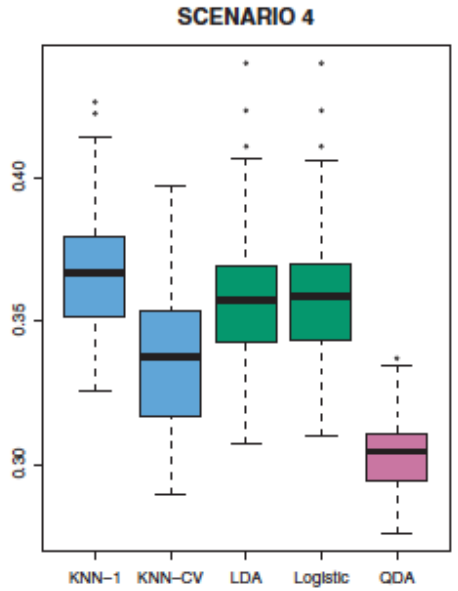
Linearly separable data

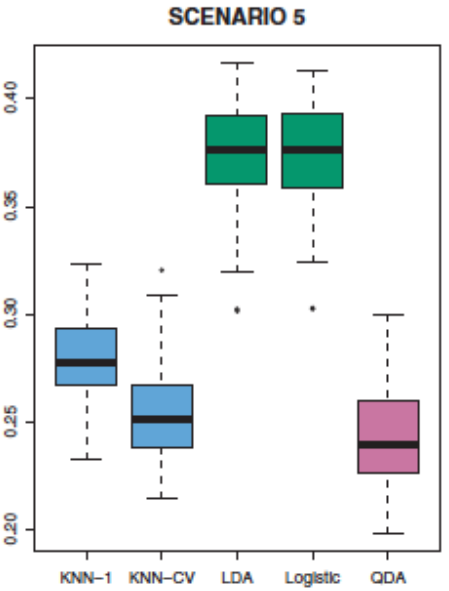
QDA assumptions

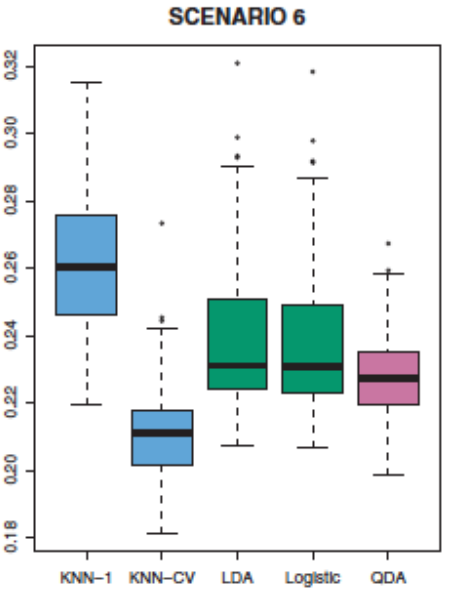
Uncorrelated predictors

Complicated non-linear function

Test error rates







Non-linearly separable data

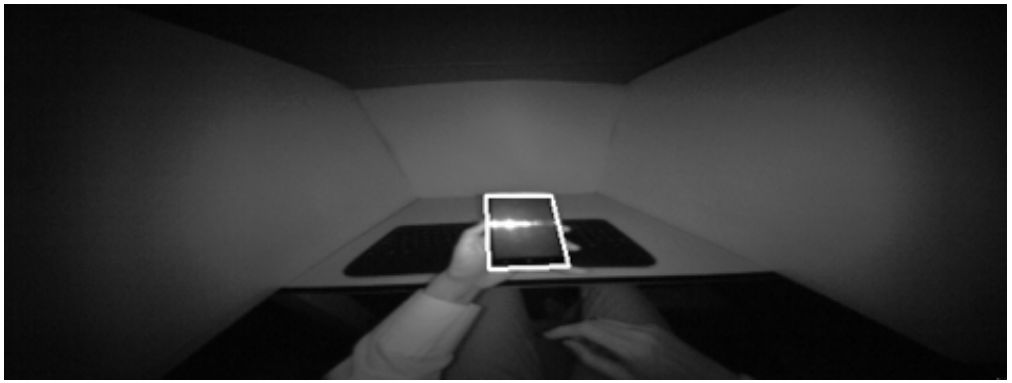
Figure from *An Introduction to Statistical Learning: with Applications in R* by James, Witten, Hastie, and Tibshirani, 2013.

Comparing Classifiers

- No one method dominates the others in every situation
  - For linear decision boundaries, LDA and logistic regression tend to perform well
  - For moderately non-linear boundaries, QDA may give better results
  - For more complicated decision boundaries, a nonparametric approach such as KNN can be superior (as long as the parameters are chosen correctly)

A sample application of logistic regression, LDA and QDA

- Research: Addressing Visual Isolation in Immersive Virtual Reality
  - VR Head Mounted Displays (VR HMD) are becoming popular in a wide variety of fields
    - Problem: The user is blinded from surrounding environment
  - Research goal:
    - To allow users to use smartphone while using HMD by displaying the smartphone screen inside VR HMD
  - ML task:
    - Detecting smartphone in real-world from image captured by leap motion



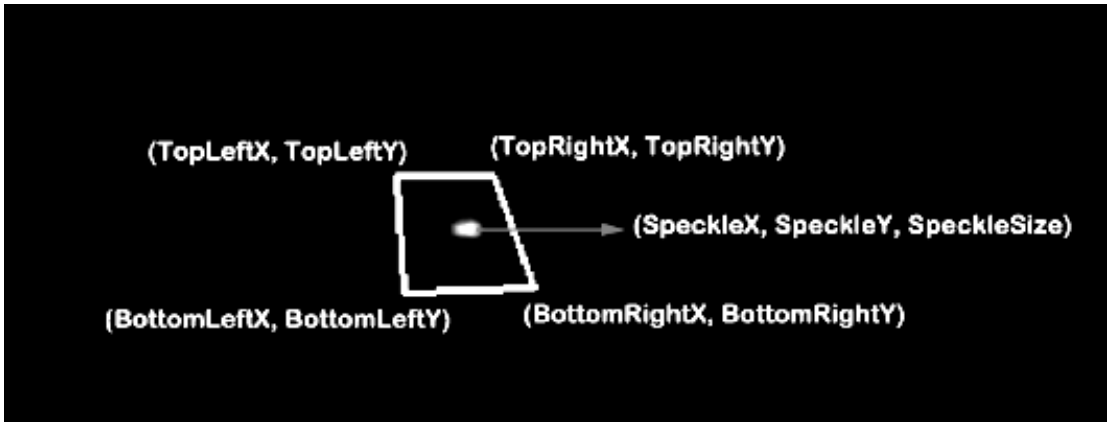
AP Desai et al, *A Window to Your Smartphone: Exploring Interaction and Communication in Immersive VR with Augmented Virtuality*. CRV 2017: 217-224

ML Methods

- Logistic regression, LDA and QDA were evaluated for this task
- Constraints considered while selecting the classifiers to evaluate:
  - Detection of smartphone in real-time
  - Ease of integration with the existing system
  - No additional process calls to external systems
  - No additional storage of instances vectors (as required by instance-based classifiers such as KNN)

Data & Methodology

- 312 observations – generated by a single user
- 2 classes:
  - Positives: edges correctly detected by an image-based system(29%)
  - Negative: edges incorrectly detected (71%)
- 11 features
- 10 fold CV

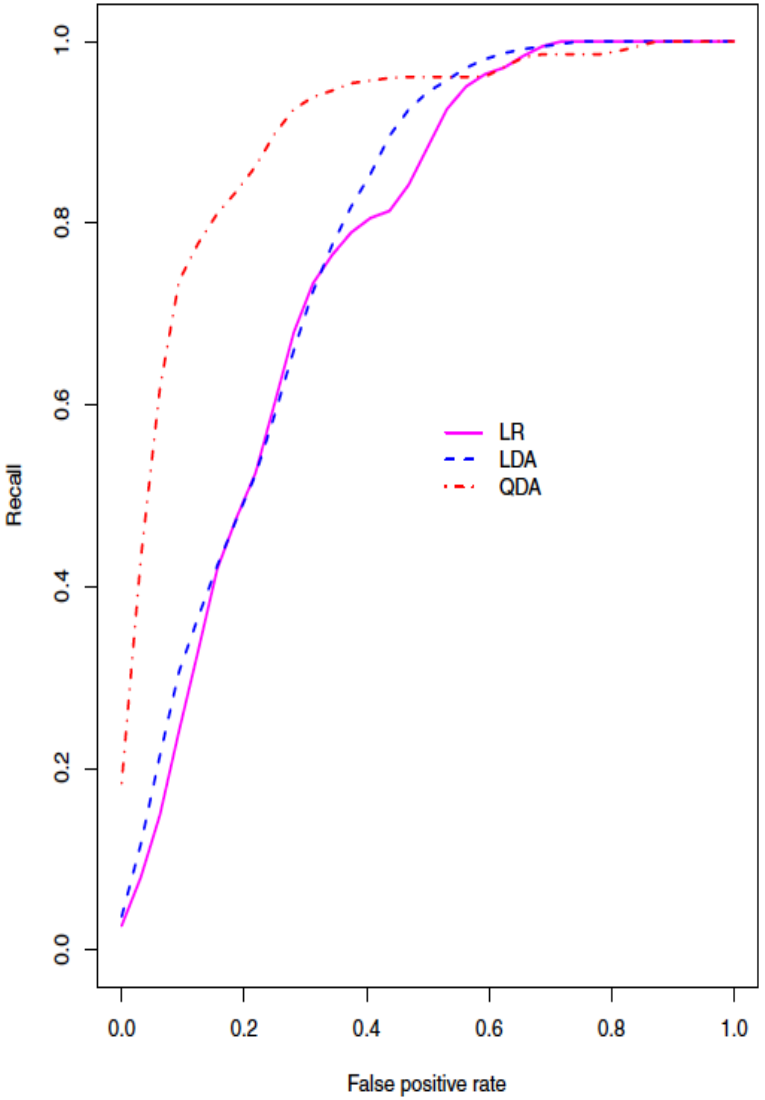


AP Desai et al, *A Window to Your Smartphone: Exploring Interaction and Communication in Immersive VR with Augmented Virtuality*. CRV 2017: 217-224

CV Results

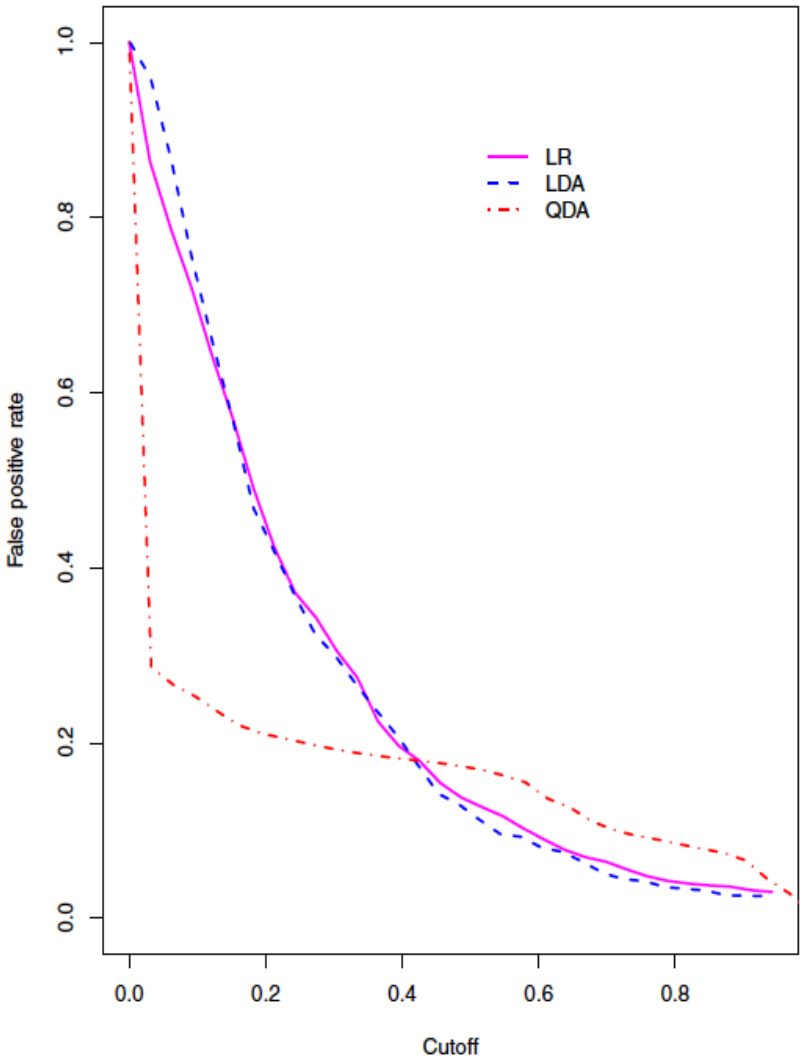
Table I
AVERAGE PERFORMANCE MEASURES AUC (± SD), SENSITIVITY (± SD), SPECIFICITY (± SD) AND ACCURACY (± SD) OF LR, LDA AND QDA OVER 10 FOLDS. SENSITIVITY, SPECIFICITY AND ACCURACY OF LR, LDA AND QDA FOR EACH OF THE 10 FOLDS WERE CALCULATED FOR THE OPTIMAL PROBABILITY CUTOFF. THE AVERAGE OPTIMAL CUTOFF FOR LR, LDA AND QDA WAS 0.295 ± 0.10, 0.280 ± 0.10 AND 0.642 ± 0.30 RESPECTIVELY

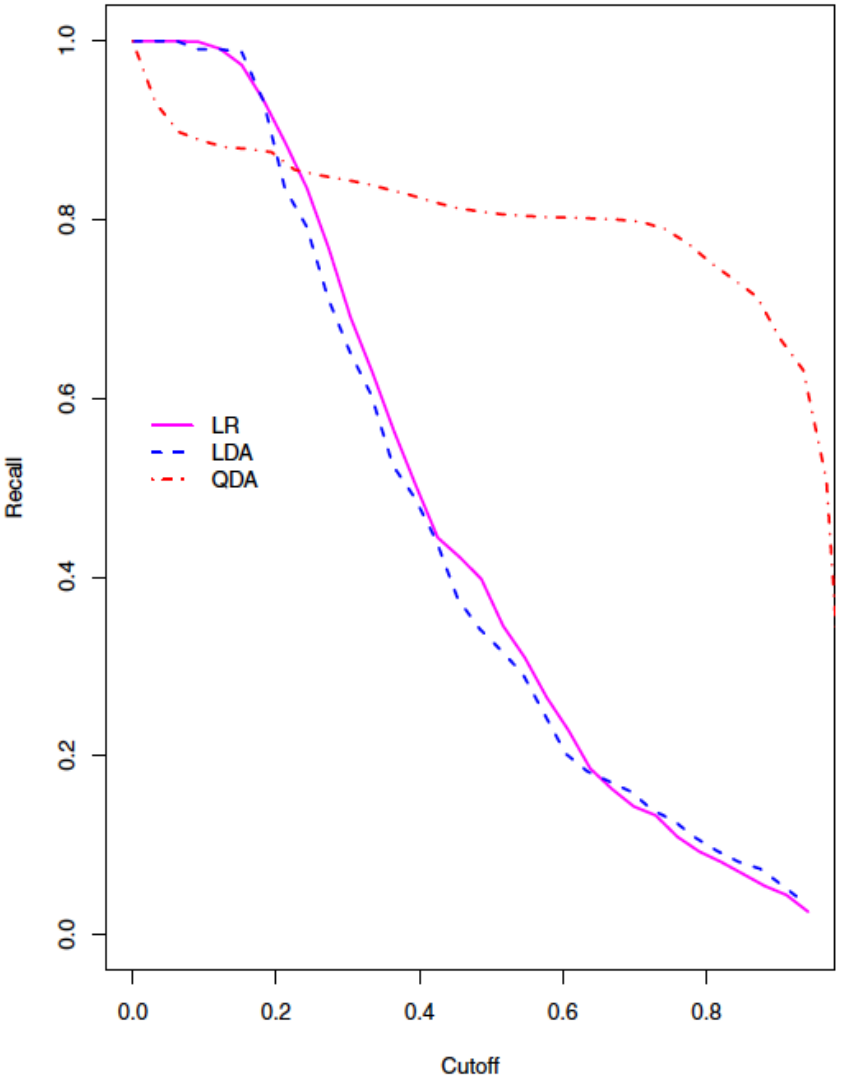
Performance Measure	LR	LDA	QDA
AUC	0.778 ± 0.13	0.777 ± 0.14	0.935 ± 0.04
Sensitivity	0.875 ± 0.12	0.873 ± 0.09	0.906 ± 0.09
Specificity	0.710 ± 0.17	0.701 ± 0.17	0.899 ± 0.07
Accuracy	0.758 ± 0.13	0.754 ± 0.12	0.907 ± 0.06



AP Desai et al, *A Window to Your Smartphone: Exploring Interaction and Communication in Immersive VR with Augmented Virtuality*. CRV 2017: 217-224

Model Selection

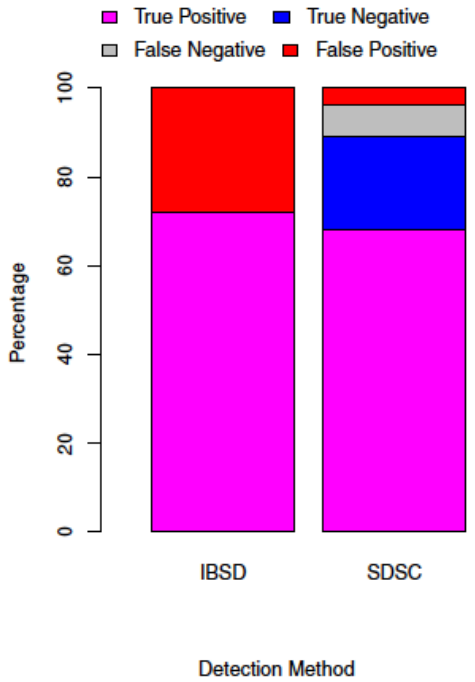




AP Desai et al, *A Window to Your Smartphone: Exploring Interaction and Communication in Immersive VR with Augmented Virtuality*. CRV 2017: 217-224

Model Assessment

- Integrating ML model into system
- Comparison with previous system using a test dataset (210 observations)



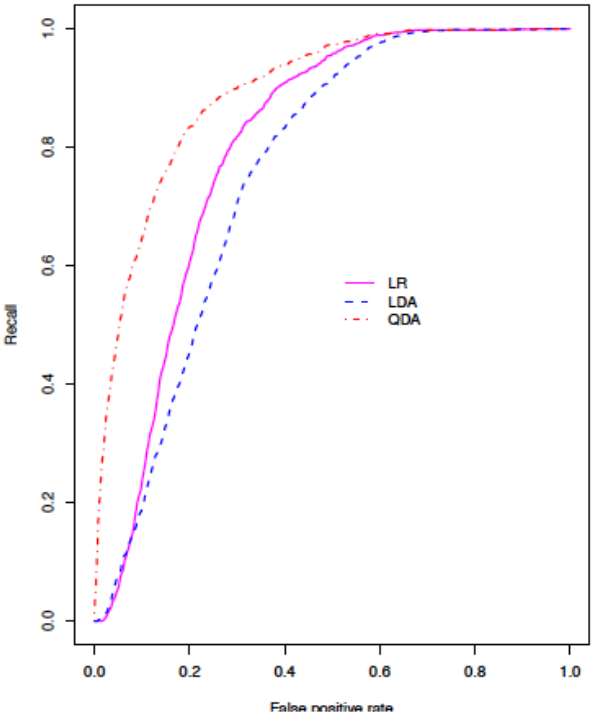
AP Desai et al, *A Window to Your Smartphone: Exploring Interaction and Communication in Immersive VR with Augmented Virtuality*. CRV 2017: 217-224

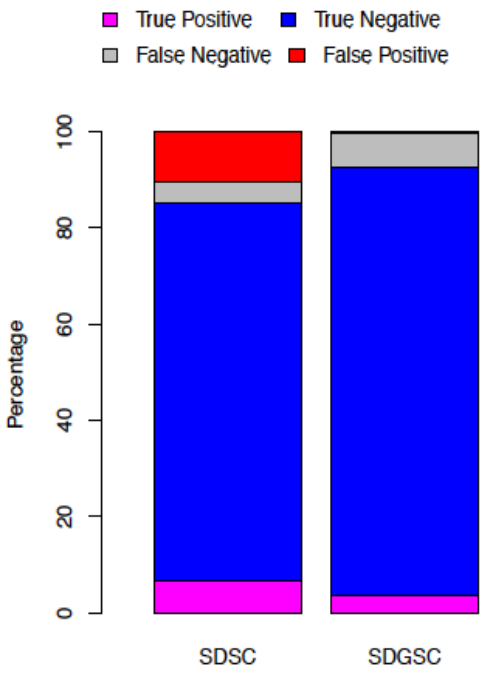
Improving Generalization

- Use QDA model to generate larger training data:
  - 8,600 observations generated by 10 users (15% positives)

Table 4.10: Average performance measures AUC (± SD), Sensitivity (± SD), Specificity (± SD) and Accuracy (± SD) of LR, LDA and QDA over 10 folds. Sensitivity, Specificity and Accuracy of LR, LDA and QDA for each of the 10 folds were calculated for the optimal probability cutoff. The average optimal cutoff for LR, LDA and QDA was 0.194 ± 0.01, 0.157 ± 0.0 and 0.959 ± 0.01 respectively

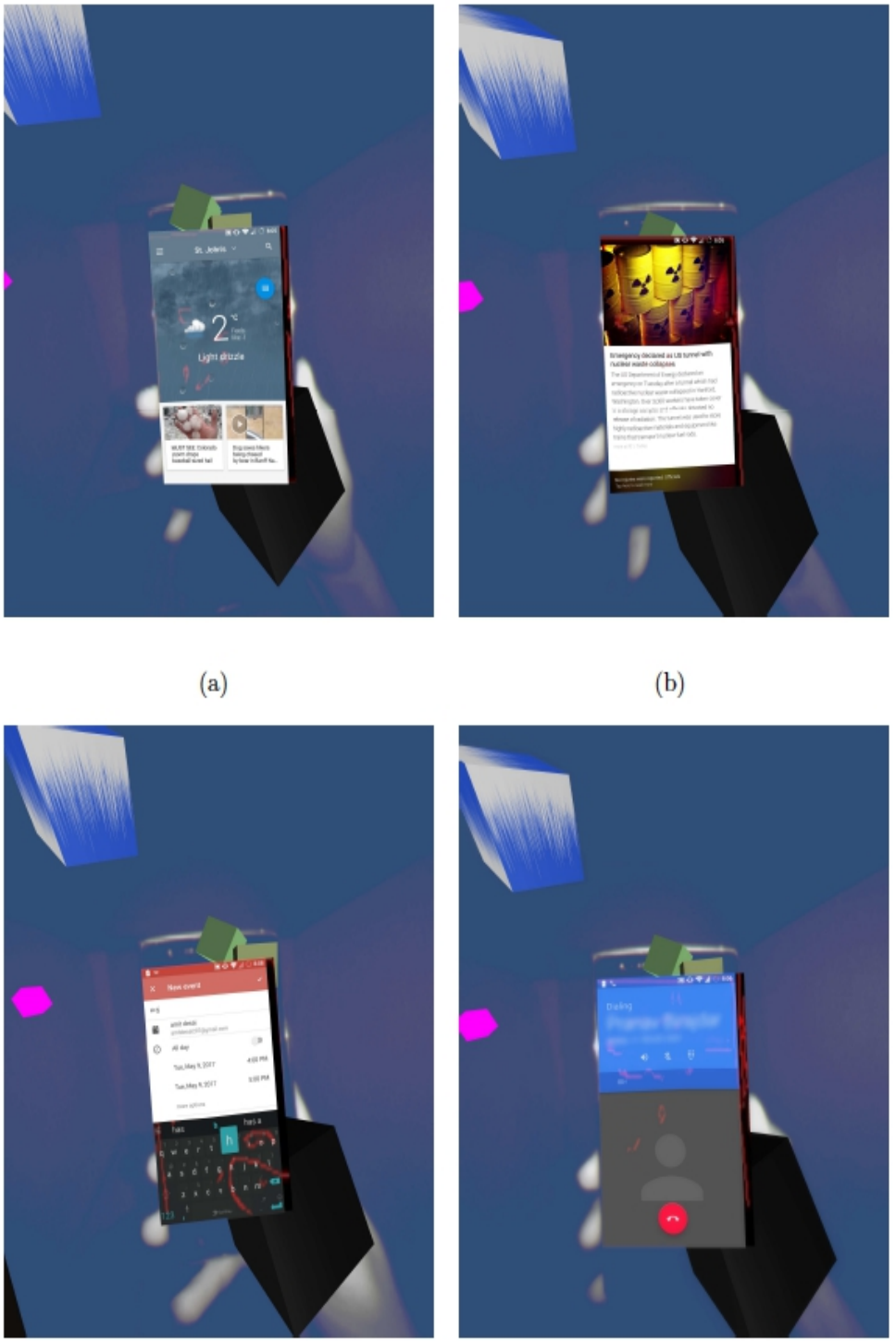
Performance Measure	LR	LDA	QDA
AUC	0.801 ± 0.01	0.758 ± 0.01	0.892 ± 0.01
Sensitivity	0.811 ± 0.03	0.782 ± 0.04	0.837 ± 0.02
Specificity	0.722 ± 0.02	0.667 ± 0.03	0.809 ± 0.02
Accuracy	0.736 ± 0.02	0.684 ± 0.03	0.814 ± 0.01





Error rate went from 14.7% to 8.8%

Desai, AP (2017) A window to your smartphone: exploring interaction and communication in immersive VR with augmented virtuality. Masters thesis, Memorial University of Newfoundland.



(a) (b)

Desai, AP (2017) A window to your smartphone: exploring interaction and communication in immersive VR with augmented virtuality. Masters thesis, Memorial University of Newfoundland.

By now, you should be able to...

- explain logistic regression (LR), LDA and QDA
- describe LR, LDA and QDA assumptions
- implement and use LR, LDA, and QDA
- describe how to obtain non-linear boundaries with LDA
- gauge for which tasks each of these ML methods: LR, LDA, QDA and KNN is more suitable than the others
- describe a sample application of LR, LDA and QDA
- understand the effect of the training data on classifier performance