Part I

Linear Discriminant Analysis

Discriminant analysis

**Nearest centroid rule**

- Suppose we break down our data matrix as by the labels yielding \((X^j)_{1 \leq j \leq k}\) with sizes \(\text{nrow}(X^j) = n_j\).
- A simple rule for classification is:
  
  Assign a new observation with features \(x\) to

  \[
  \hat{f}(x) = \arg\min_{1 \leq j \leq k} d(x, X^j)
  \]

- What do we mean by distance here?

If we can assign a central point or centroid \(\hat{\mu}_j\) to each \(X^j\),
then we can define the distance above as distance to the centroid \(\hat{\mu}_j\).

This yields the nearest centroid rule

\[
\hat{f}(x) = \arg\min_{1 \leq j \leq k} d(x, \hat{\mu}_j)
\]
Discriminant analysis

Nearest centroid rule

- This rule is described completely by the functions

\[ h_{ij}(x) = \frac{d(x, \hat{\mu}_j)}{d(x, \hat{\mu}_i)} \]

with \( \hat{f}(x) \) being any \( i \) such that

\[ h_{ij}(x) \geq 1 \ \forall j. \]

Why we should use covariance

Choice of distance

- Often, there is some background model for our data that is equivalent to a given procedure.

- For this nearest centroid rule, using the Euclidean distance effectively assumes that within the set of points \( X'_j \), the rows are multivariate Gaussian with covariance matrix proportional to \( I \).

- It also implicitly assumes that the \( n_j \)'s are roughly equal.

- For instance, if one \( n_j \) was very small just because a data point is close to that \( \hat{\mu}_j \) doesn't necessarily mean that we should conclude it has label \( j \) because there might be a huge number of points of label \( i \) near that \( \hat{\mu}_j \).
Gaussian discriminant functions

- Suppose each group with label \( j \) had its own mean \( \mu_j \) and covariance matrix \( \Sigma_j \), as well as proportion \( \pi_j \).
- The Gaussian discriminant functions are defined as

\[
G_j(x) = \log \pi_j - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (x - \mu_j)^T \Sigma_j^{-1} (x - \mu_j)
\]

- The classifier assigns \( x \) label \( i \) if \( G_i(x) \geq G_j(x) \) for all \( j \).
- Or,

\[
f(x) = \arg\max_{1 \leq i \leq k} G_i(x)
\]

- This is equivalent to a Bayesian rule. We’ll see more Bayesian rules when we talk about naïve Bayes . . .
- When all \( \Sigma_i \) and \( \pi_i \)’s are identical, the classifier is just nearest centroid using Mahalanobis distance instead of Euclidean distance.

Estimating discriminant functions

- In practice, we will have to estimate \( \pi_j, \mu_j, \Sigma_j \).
- Obvious estimates:

\[
\hat{\pi}_j = \frac{n_j}{\sum_{j=1}^{k} n_j}
\]

\[
\hat{\mu}_j = \frac{1}{n_j} \sum_{i=1}^{n_j} x_i^j
\]

\[
\hat{\Sigma}_j = \frac{1}{n_j - 1} (X^i - \hat{\mu}_j)(X^j - \hat{\mu}_j)^T
\]

- If we assume that the covariance matrix is the same within groups, then we might also form the pooled estimate

\[
\hat{\Sigma}_P = \frac{\sum_{j=1}^{k} (n_j - 1) \hat{\Sigma}_j}{\sum_{j=1}^{k} n_j - 1}
\]

- If we use the pooled estimate \( \Sigma_j = \hat{\Sigma}_P \) and plug these into the Gaussian discriminants, the functions \( G_j(x) \) are linear (or affine) functions of \( x \).
- This is called Linear Discriminant Analysis (LDA).
- Not to be confused with the other LDA (Latent Dirichlet Allocation) . . .
Linear Discriminant Analysis using (petal.width, petal.length)

Discriminant analysis

If we use don’t use pooled estimate $\Sigma_j = \hat{\Sigma}_j$ and plug these into the Gaussian discriminants, the functions $h_{ij}(\mathbf{x})$ are quadratic functions of $\mathbf{x}$.

This is called Quadratic Discriminant Analysis (QDA).
Quadratic Discriminant Analysis using 
(sepal.width, sepal.length)

Motivation for Fisher's rule

Discriminant analysis

Fisher’s discriminant function
- Fisher proposed to classify using a linear rule.
- He first decomposed
  \[(X - \hat{\mu}1)^T(X - \hat{\mu}1) = \hat{SS}_B + \hat{SS}_W\]
- Then, he proposed,
  \[
  \hat{v} = \arg\max_{v: v^T\hat{SS}_Wv = 1} v^T\hat{SS}_B v
  \]
- Having found \(\hat{v}\), form \(X^i\hat{v}\) and the centroid \(\eta_j = \text{mean}(X^i\hat{v})\)
- In the two-class problem \(k = 2\), this is the same as LDA.

Discriminant analysis

Fisher’s discriminant functions
- The direction \(\hat{v}_1\) is an eigenvector of some matrix. There are others, up to \(k - 2\) more.
- Suppose we find all \(k - 1\) vectors and form \(X_j V^T\), each one an \(n_j \times (k - 1)\) matrix with centroid \(\eta_j \in \mathbb{R}^{k-1}\).
- The matrix \(V\) determines a map from \(\mathbb{R}^p\) to \(\mathbb{R}^{k-1}\), so given a new data point we can compute \(Vx \in \mathbb{R}^{k-1}\).
- This gives rise to a classifier
  \[\hat{f}(x) = \text{nearest centroid}(Vx)\]
- This is LDA (assuming \(\pi_j = \frac{1}{k}\)) . . .
Discriminant analysis

Discriminant models in general

- A discriminant model is generally a model that estimates
  \[ P(Y = j | x), 1 \leq j \leq k \]
- That is, given that the features I observe are \( x \), the
  probability I think this label is \( j \) ...
- LDA and QDA are actually generative models since they
  specify
  \[ P(X = x | Y = j). \]
- There are lots of discriminant models . . .

Logistic regression

- The logistic regression model is ubiquitous in binary
  classification (two-class) problems
- Model:
  \[ P(Y = 1 | x) = \frac{e^{x^T \beta}}{1 + e^{x^T \beta}} = \pi(\beta, x) \]

Software that fits a logistic regression model produces an
estimate of \( \beta \) based on a data matrix \( X_{n \times p} \) and binary
labels \( Y_{n \times 1} \in \{0, 1\}^n \)
- It fits the model minimizing what we call the deviance
  \[ \text{DEV}(\beta) = -2 \sum_{i=1}^{n} (Y_i \log \pi(\beta, X_i) + (1 - Y_i) \log(1 - \pi(\beta, X_i))) \]
- While not immediately obvious, this is a convex
  minimization problem, hence is fairly easy to solve.
- Unlike trees, the convexity yields a globally optimal
  solution.
Logistic regression, *virginica* vs *setosa*, *versicolor* using \((\text{sepal.width}, \text{sepal.length})\)

$\begin{align*}
3.5 & 4.0 & 4.5 & 5.0 & 5.5 & 6.0 & 6.5 & 7.0 & 7.5 & 8.0 \\
2.0 & 2.5 & 3.0 & 3.5 & 4.0 & 4.5 & 5.0
\end{align*}$

**Discriminant analysis**

**Logistic regression**

- Logistic regression produces an estimate of $P(y = 1|x) = \hat{\pi}(x)$

- Typically, we classify as 1 if $\hat{\pi}(x) > 0.5$.

- This yields a $2 \times 2$ confusion matrix

<table>
<thead>
<tr>
<th></th>
<th>Predicted: 0</th>
<th>Predicted: 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual: 0</td>
<td>$TN$</td>
<td>$FP$</td>
</tr>
<tr>
<td>Actual: 1</td>
<td>$FN$</td>
<td>$TP$</td>
</tr>
</tbody>
</table>

- From the $2 \times 2$ confusion matrix, we can compute Sensitivity, Specificity, etc.

**Logistic regression**

- However, we could choose the threshold differently, perhaps related to estimates of the prior probabilities of 0’s and 1’s.

- Now, each threshold $0 \leq t \leq 1$ yields a new confusion matrix

<table>
<thead>
<tr>
<th></th>
<th>Predicted: 0</th>
<th>Predicted: 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual: 0</td>
<td>$TN(t)$</td>
<td>$FP(t)$</td>
</tr>
<tr>
<td>Actual: 1</td>
<td>$FN(t)$</td>
<td>$TP(t)$</td>
</tr>
</tbody>
</table>
Discriminant analysis

ROC curve
- Generally speaking, we prefer classifiers that are both highly sensitive and highly specific.
- These confusion matrices can be summarized using an ROC (Receiver Operating Characteristic) curve.
- This is a plot of the curve
  \[(1 - \text{Specificity}(t), \text{Sensitivity}(t))_{0 \leq t \leq 1}\]
- Often, Specificity is referred to as TNR (True Negative Rate), and \((1 - \text{Specificity}(t))\) as FPR.
- Often, Sensitivity is referred to as TPR (True Positive Rate), and \((1 - \text{Sensitivity}(t))\) as FNR.

AUC: Area under ROC curve
- A common numeric summary of the ROC curve is
  \[\text{AUC}(\text{ROC curve}) = \text{Area under ROC curve}.\]
- Can be interpreted as an estimate of the probability that the classifier will give a random positive instance a higher score than a random negative instance.
- Maximum value is 1.
- For a random guesser, AUC is 0.5.
ROC curve: logistic, rpart, lda

Part II

Midterm review

Overview
- General goals of data mining.
- Datatypes.
- Preprocessing & dimension reduction.
- Distances.
- Multidimensional scaling.
- Multidimensional arrays.
- Decision trees.
- Performance measures for classifiers.
- Discriminant analysis.
Midterm review

General goals

- Definition: what is and isn’t data mining.
- Different types of problems:
  - Unsupervised problems.
  - Supervised problems.
  - Semi-supervised problems.

Datatypes

- Continuous, discrete, etc.
- How data is represented in R.
- Descriptive statistics for different datatypes.
- General characteristics:
  - Is it observational or experimental?
  - Is it very noisy?
  - Is there spatial or temporal structure?

Preprocessing

- General tasks: aggregation, transformation, discretization.
- Feature extraction: wavelet / FFT transform.
- Another type of feature extraction: dimension reduction.

Midterm review

Dimension reduction

- PCA as a dimension reduction tool.
- PCA in terms of the SVD.
- PCA loadings, scores.

Graphical summaries

- Various plots of univariate continuous data:
  - stem-leaf;
  - histogram;
  - density (using kernel estimate);
  - ECDF;
  - quantile.
  - Boxplot.
  - Pairs plot.
  - Correlation / similarity matrix (cases or features)

Multidimensional scaling

- Relation between distances and similarities.
- Graphical (Euclidean) representation of data that is “closest” to original dissimilarity.
- Relation to PCA when similarity is Euclidean.

Multidimensional arrays

- Data cubes.
- Standard types of operations on data cubes.
- A few examples in R.
**Decision trees**
- Definition of a classifier.
- Specific form of a decision tree classifier.
- Applying the decision tree model.
- Fitting a decision tree using Hunt’s algorithm.
- Various measures of impurity: Gini, entropy, Misclassification Rate.
- Pre and post-pruning to simplify tree.

**Performance measures**
- Sensitivity, specificity, true/false negatives, true/false positives.
- Confusion matrix.
- Using a cost matrix to evaluate a classifier.

**Discriminant analysis**
- Linear / Quadratic Discriminant Analysis;
- Logistic regression.
- ROC curve.