Statistical Learning

Logistic Regression and Discriminant Analysis

Spring 2024

Outline

- Classification problems
- Logistic Regression
- Evaluating Classification Models
- LDA and QDA
- Remarks

Classification Problems

- Training data $\mathcal{D}_n = \{x_i, y_i\}_{i=1}^n$:
  - $x_i \in \mathbf{R}^p$, and $y_i \in \{0, 1\}$ (sometimes we use $\{-1, +1\}$).
- The goal is to find a classifier

$$f : \mathbf{R}^p \longrightarrow \{0, 1\}$$

- At any target point x with outcome y, the performance of a classifier is usually measured by 0–1 loss

$$L(f(x), y) = \begin{cases} 0 & \text{if} \quad y = f(x) \\ 1 & \text{if} \quad \text{o.w.} \end{cases}$$

Soft vs. Hard Classification

- There are two popular approaches when modeling binary data
- Soft classifiers
  - Estimate the conditional probabilities $P(Y|X)$
  - Use $\mathbf{1}\{P(Y|X) > c\}$ for classification
  - e.g., logistic regression
- Hard classifiers
  - Directly estimate the classification decision boundary
  - e.g., SVM
- Many methods are capable of doing or have been extended to perform both

Logistic Regression

Motivation

- Directly model the probability

$$\eta(x) = \mathsf{P}(Y = 1 | X = x)$$

- $\eta(x)$ should be bounded within $[0, 1]$
- Consider a link function g that transform $\eta(x)$ into $(-\infty, \infty)$, then

$$g\big(\eta(x)\big) = x^{\mathsf{T}}\beta$$

- Generalized linear model (GLM)

Motivation

- Response Y follows a Bernoulli distribution conditioning on x:

$$p(Y = y_i | X = x_i) = \eta(x_i)^{y_i} [1 - \eta(x_i)]^{1-y_i}$$

- For Logistic regression, we use the logit link function

$$\log \frac{\eta(x)}{1 - \eta(x)} = x^\mathsf{T} \beta, \quad \eta(x) = \frac{\exp(x^\mathsf{T} \beta)}{1 + \exp(x^\mathsf{T} \beta)}$$

- $\log \frac{p}{1-p}$ is called log-odds, and we are modeling it as a linear function of x.

Fitting Logistic Models

- Maximize the log-likelihood function, using the conditional likelihood of Y given X:

$$\begin{aligned}
\ell(\boldsymbol{\beta}) &= \sum_{i=1}^{n} \log\, p(y_i|x_i, \boldsymbol{\beta}) \\
&= \sum_{i=1}^{n} \log \left\{ \eta(x_i)^{y_i} [1 - \eta(x_i)]^{1-y_i} \right\} \\
&= \sum_{i=1}^{n} y_i \log \frac{\eta(x_i)}{1 - \eta(x_i)} + \log[1 - \eta(x_i)] \\
&= \sum_{i=1}^{n} y_i x_i^{\mathsf{T}} \boldsymbol{\beta} - \log[1 + \exp(x_i^{\mathsf{T}} \boldsymbol{\beta})]
\end{aligned}$$

Newton-Raphson

- Derive the first and second derivatives
- Use Newton's method to update $\boldsymbol{\beta}$ by

$$\boldsymbol{\beta}^{\text{new}} = \boldsymbol{\beta}^{\text{old}} - \left[\frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathsf{T}}}\right]^{-1} \frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}\bigg|_{\boldsymbol{\beta}^{\text{old}}}$$

where

$$\text{(gradient)} \quad \frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = \sum_{i=1}^{n} y_i x_i^{\mathsf{T}} - \sum_{i=1}^{n} \frac{\exp(x_i^{\mathsf{T}}\boldsymbol{\beta}) x_i^{\mathsf{T}}}{1 + \exp(x_i^{\mathsf{T}}\boldsymbol{\beta})}$$

$$\text{(Hessian)} \quad \frac{\partial^2 \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^{\mathsf{T}}} = -\sum_{i=1}^{n} x_i x_i^{\mathsf{T}} \eta(x_i)[1 - \eta(x_i)]$$

Interpreting Parameters

- What is the effect of β?
- Wrong:
  - Each unit increases of X_j increases the probability of Y by β_j
- Correct:
  - Each unit increases of X_j increases the log-odds of Y by β_j

Evaluating Classification Models

Correct or Wrong decision?

- In hypothesis testing problems, we have the following 2×2 table:

	Accept H_0	Reject H_0
H_0 true	✓	Type I Error
H_0 false	Type II Error	✓

- For classification problems, we face the same decision problems
- Instead of using the α-level (to tune), we can use different thresholds on $P(Y|X)$ to make the decision.

A Motivating Example

- A new lab test is developed for detecting Covid-19 infection based on the score obtained from a device
  - If the test returns positive (the score is larger than a threshold c), then we conclude infection.
  - If the test returns negative (the score is lower than c), we conclude no infection.
- We collect the following data from 1000 tests

	Infection	No Infection	
Test Positive	20	70	90
Test Negative	10	900	910
	30	970	1000

Confusion Matrix

	Infection	No Infection
Test Positive	True Positive (TP)	False Positive (FP, Type I Error)
Test Negative	False Negative (FN, Type II Error)	True Negative (TN)

- One way to evaluate this model (test) is the overall accuracy

$$\begin{aligned}\text{Overall Accuracy} &= \frac{\text{All Correct Decisions}}{\text{All Decisions}} \\ &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}\end{aligned}$$

- However, this is not always good, especially when we have unbalanced data

Sensitivity and Specificity

- Sensitivity (also called “Recall”) is defined as the true positive rate (among the infected population, what proportion are correctly identified by the test)

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

	Infection	No Infection
Test Positive	TP	FP
Test Negative	FN	TN

- In our data, the sensitivity is $\frac{20}{20+10} \approx 66.7\%$

Sensitivity and Specificity

- Specificity is defined as the true negative rate (among the non-infected population, what proportion are correctly identified by the test)

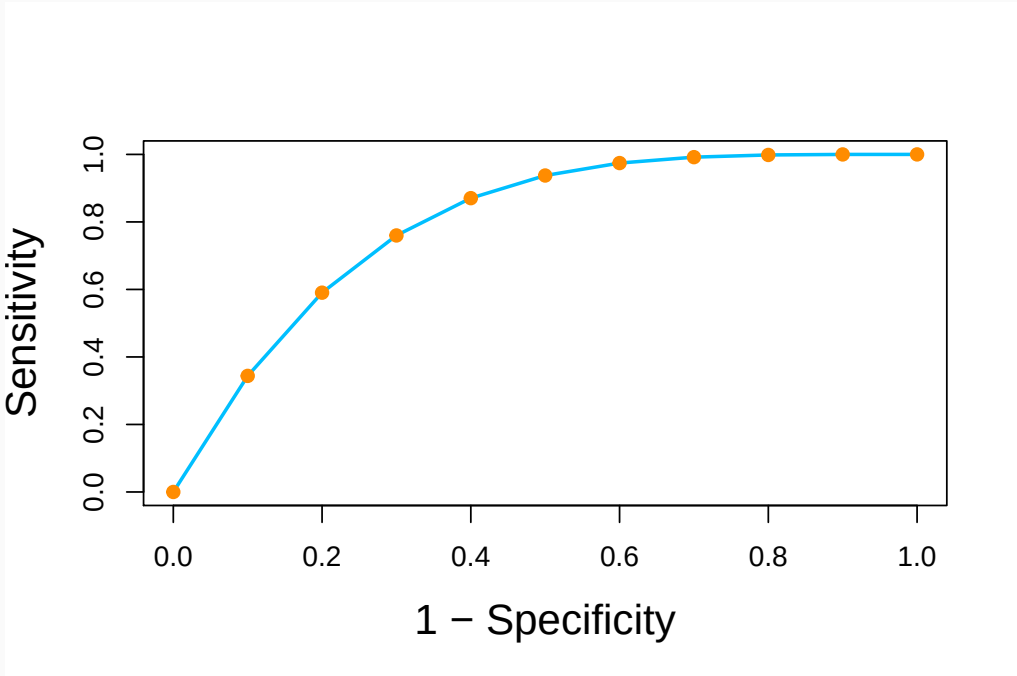
$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

	Infection	No Infection
Test Positive	TP	FP
Test Negative	FN	TN

- In our data, the specificity is $\frac{900}{900+70} \approx 92.8\%$

Receiver Operating Characteristic Curve

- However, we can also alter the decision threshold c, which leads to a different sensitivity and specificity combination
- As we alter the threshold, the two measures form a receiver operating characteristic curve
  - x-axis: 1 − Specificity: False Positive Rate
  - y-axis: Sensitivity: True Positive Rate



Motivation of LDA and QDA

Motivation: Bayes Rule

- Recall that if we use the 0–1 loss as the criterion

$$L(f(x), y) = \begin{cases} 0 & \text{if} \quad y = f(x) \\ 1 & \text{if} \quad \text{o.w.} \end{cases}$$

- Then, the best rule we can get is

$$f_B(x) = \arg\min_f \mathsf{R}(f) = \begin{cases} 1 & \text{if} \quad \eta(x) > 1/2 \\ 0 & \text{if} \quad \eta(x) < 1/2. \end{cases}$$

- This rule f_B is called the Bayes rule, and the corresponding risk (expected loss) is the Bayes risk or Bayes error.

Bayes Rule

- The name of “Bayes rule” comes from understanding the optimal rule from the Bayes perspective:

$$\mathsf{P}(Y = 1|X = x) = \frac{\mathsf{P}(X = x|Y = 1)\mathsf{P}(Y = 1)}{\mathsf{P}(X = x)}$$

$$\mathsf{P}(Y = 0|X = x) = \frac{\mathsf{P}(X = x|Y = 0)\mathsf{P}(Y = 0)}{\mathsf{P}(X = x)}$$

Bayes Rule

- Treating $\pi = \mathsf{P}(Y = 1)$ and $(1 - \pi) = \mathsf{P}(Y = 0)$ as prior probabilities, and define the conditional densities of X as

$$f_1 = \mathsf{P}(X = x|Y = 1) \text{ and } f_0 = \mathsf{P}(X = x|Y = 0).$$

- The Bayes rule can also be written as

$$f_B(x) = \arg\min_f \mathsf{R}(f) = \begin{cases} 1 & \text{if} \quad \pi f_1(x) > (1 - \pi) f_0(x) \\ 0 & \text{if} \quad \pi f_1(x) < (1 - \pi) f_0(x). \end{cases}$$

- Note that the marginal density of X can also be written as

$$\mathsf{P}(X = x) = \pi f_1(x) + (1 - \pi) f_0(x)$$

although it does not play a role in the optimal decision rule.

Multi-Class Problems

- In multi-class problems, $y \in \{1, \ldots K\}$. We want to construct classifier

$$f : \mathbf{R}^p \longrightarrow \{1, \ldots, K\}$$

- The optimal rule is

$$f_B(x) = \arg\max_k \mathsf{P}(Y = k | X = x) = \arg\max_k \pi_k f_k(x)$$

  where π_k is prior probability and $f_k(x)$ is the conditional density for class k.

- Classify x to the most probable class by comparing $\mathsf{P}(Y|X = x)$.

Masking Problems in Linear Models

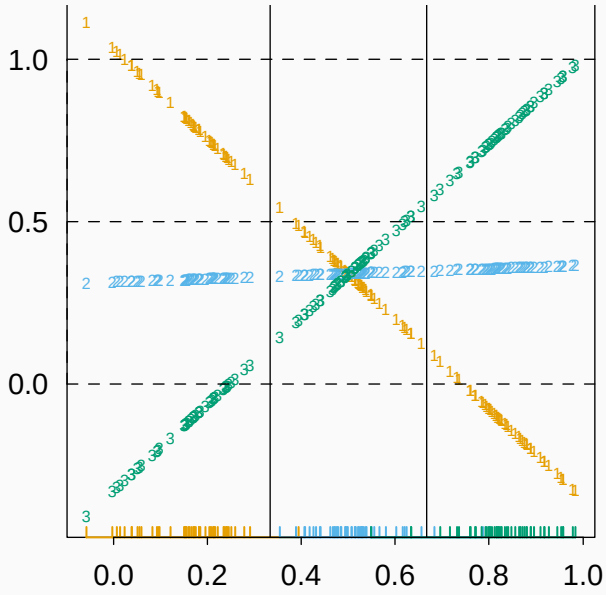
- For outcome Y, which may fall into categories $1, \ldots, K$, define a vector of indicators $(Y_1, \ldots, Y_K)$

$$Y_k = 1 \quad \text{if} \quad Y = k$$

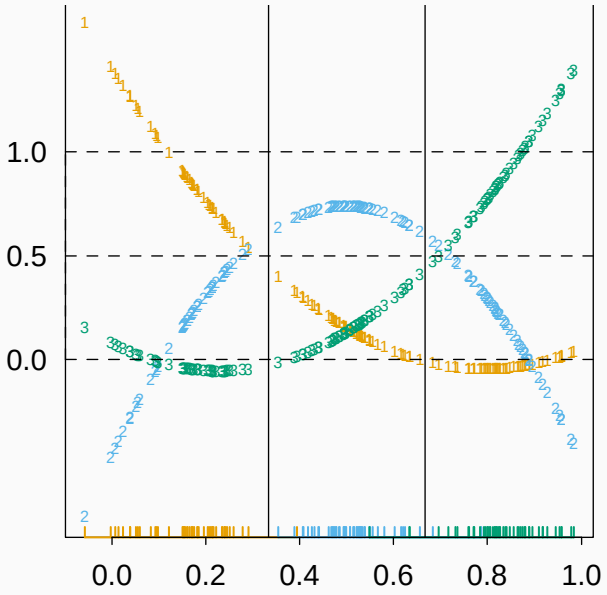
  - Each vector $(Y_1, \ldots, Y_K)$ has a single 1.
  - The n training samples form an $n \times K$ indicator response matrix $\mathbf{Y}$, where each row is such an indicator vector.
- If we model each Y_k separately, then this is essentially one-vs-other
- However, we may face serious masking problems

Masking Problems

Fitting the three-class problem using polynomials



Degree = 1; Error = 0.33



Degree = 2, Error = 0.04

Note: LDA can avoid this problem

Binary vs. Multi-Class

- We will focus on binary classifiers. Some binary classifiers can also handel multi-classes, such as discriminate analysis (LDA, QDA, NB), logistic regression, kNN and random forests. But for some others, the extension is non-trivial (SVM).
- There are some naive (although may not be optimal) ways to apply a binary classifier on a classification problem with $K > 2$ categories.
  - Train K one-vs-other classifiers
  - Train $K(K-1)/2$ pairwise classifiers

  Then we can combine the results to get a consensus prediction.

Discriminant Analysis

Linear Discriminant Analysis

- The idea is to model the distribution of X in each of the classes separately, and then use Bayes theorem to flip things around and obtain P$(Y|X = x)$.
- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- Naive Bayes (NB)

Bayes Theorem for Classification

- As we demonstrated earlier (Bayes rules), the conditional probability can be formulated using Bayes Theorem:

$$\begin{aligned}\mathsf{P}(Y = k|X = x) &= \frac{\mathsf{P}(X = x|Y = k)\mathsf{P}(Y = k)}{\mathsf{P}(X = x)} \\ &= \frac{\mathsf{P}(X = x|Y = k)\mathsf{P}(Y = k)}{\sum_{l=1}^{K} \mathsf{P}(X = x|Y = l)\mathsf{P}(Y = l)} \\ &= \frac{\pi_k f_k(x)}{\sum_{l=1}^{K} \pi_l f_l(x)}\end{aligned}$$

  where $f_k(x)$ is the conditional density function of $X|Y = k$, and $\pi_k = \mathsf{P}(Y = k)$ is the prior probability.

Bayes Theorem for Classification

- The best prediction is picking the one that maximizes the posterior

$$\arg\max_k \pi_k f_k(x)$$

- LDA and QDA model $f_k(x)$ as a normal distribution

Bayes Theorem for Classification

- Suppose we model each class density as multivariate Gaussian $\mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k)$, and assume that the covariance matrices are the same across all k, i.e., $\Sigma_k = \Sigma$. Then the

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-\boldsymbol{\mu}_k)^{\mathsf{T}}\Sigma^{-1}(x-\boldsymbol{\mu}_k)\right]$$

- The log-likelihood function for the conditional distribution is

$$\begin{aligned}\log f_k(x) = & -\log((2\pi)^{p/2}|\Sigma|^{1/2}) - \frac{1}{2}(x-\boldsymbol{\mu}_k)^{\mathsf{T}}\Sigma^{-1}(x-\boldsymbol{\mu}_k) \\ = & -\frac{1}{2}(x-\boldsymbol{\mu}_k)^{\mathsf{T}}\Sigma^{-1}(x-\boldsymbol{\mu}_k) + \text{constant}\end{aligned}$$

Bayes Theorem for Classification

- Hence we just need to select the category that attains the highest posterior density (MAP: maximum a posteriori):

$$\begin{aligned}\widehat{f}(x) &= \arg\max_k \ \log\left(\pi_k f_k(x)\right) \\ &= \arg\max_k \ -\frac{1}{2}(x-\boldsymbol{\mu}_k)^{\mathsf{T}}\Sigma^{-1}(x-\boldsymbol{\mu}_k) + \log(\pi_k)\end{aligned}$$

Interpretations of LDA

- The term $(x - \boldsymbol{\mu}_k)^\mathsf{T} \Sigma^{-1} (x - \boldsymbol{\mu}_k)$ is simply the Mahalanobis distance between x and the centroid $\boldsymbol{\mu}_k$ for class k
- Classify x to the class with the closest centroid (after adjusting for the prior)
- Special case: $\Sigma = \mathbf{I}$ (only Euclidean distance is needed)

$$\arg\max_k \; -\frac{1}{2}\|x - \boldsymbol{\mu}_k\|^2 + \log(\pi_k)$$

Decision Boundary

- Noticing that that quadratic term can be simplified to

$$
\begin{aligned}
&-\frac{1}{2}(x-\boldsymbol{\mu}_k)^\mathsf{T}\Sigma^{-1}(x-\boldsymbol{\mu}_k) \\
&= x^\mathsf{T}\Sigma^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\mu}_k^\mathsf{T}\Sigma^{-1}\boldsymbol{\mu}_k + \text{irrelevant things}
\end{aligned}
$$

- Then the discriminant function is defined as

$$
\begin{aligned}
\delta_k(x) &= x^\mathsf{T}\Sigma^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\mu}_k^\mathsf{T}\Sigma^{-1}\boldsymbol{\mu}_k + \log\pi_k \\
&= \mathbf{w}_k^\mathsf{T}x + b_k,
\end{aligned}
$$

- We can calculate $\mathbf{w}_k$'s and b_k's for each class k from the data.

Decision Boundary

- The decision boundary function between class k and l is

$$\begin{aligned} & \mathbf{w}_k^\mathsf{T} x + b_k = \mathbf{w}_l^\mathsf{T} x + b_l \\ \Leftrightarrow \quad & (\mathbf{w}_k - \mathbf{w}_l)^\mathsf{T} x + (b_k - b_l) = 0 \\ \Leftrightarrow \quad & \widetilde{\mathbf{w}}^\mathsf{T} x + \widetilde{b} = 0 \end{aligned}$$

- Since $\mathbf{w}_k = \Sigma^{-1}\boldsymbol{\mu}_k$ and $\mathbf{w}_l = \Sigma^{-1}\boldsymbol{\mu}_l$, the decision boundary has the directional vector

$$\widetilde{\mathbf{w}} = \Sigma^{-1}(\boldsymbol{\mu}_k - \boldsymbol{\mu}_l)$$

Parameter Estimations in LDA

- We estimate the LDA parameters from the training data
  - Prior probabilities: $\widehat{\pi}_k = n_k/n = n^{-1} \sum_k \mathbf{1}\{y_i = k\}$, where n_k is the number of observations in class k.
  - Centroid: $\widehat{\boldsymbol{\mu}}_k = n_k^{-1} \sum_{i:\, y_i = k} x_i$
  - Pooled covariance:

$$\widehat{\Sigma} = \frac{1}{n-K} \sum_{k=1}^{K} \sum_{i:\, y_i = k} (x_i - \widehat{\boldsymbol{\mu}}_k)(x_i - \widehat{\boldsymbol{\mu}}_k)^{\mathsf{T}}$$

Quadratic Discriminant Analysis

- Quadratic Discriminant Analysis simply abandons the assumption of the common covariance matrix. Hence, Σ_k's are not equal.
- In this case, the determinant $|\Sigma_k|$ of each covariance matrix will be different. The MAP decision becomes

$$\begin{aligned}
&\max_k \ \log\left(\pi_k f_k(x)\right) \\
&= \max_k \ -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\boldsymbol{\mu}_k)^{\mathsf{T}}\Sigma_k^{-1}(x-\boldsymbol{\mu}_k) + \log(\pi_k) + \text{constant} \\
&\delta_k(x) = x^{\mathsf{T}}\mathbf{W}_k x + \mathbf{w}_k^{\mathsf{T}} x + b_k
\end{aligned}$$

- This leads to quadratic decision boundary between class k and l

$$\{x : x^{\mathsf{T}}(\mathbf{W}_k - \mathbf{W}_l)x + (\mathbf{w}_k - \mathbf{w}_l)^{\mathsf{T}}x + (b_k - b_l) = 0\}$$

Estimations in QDA

- We estimate the QDA parameters from the training data
  - Prior probabilities: $\widehat{\pi}_k = n_k/n = n^{-1} \sum_k \mathbf{1}\{y_i = k\}$, where n_k is the number of observations in class k.
  - Centroid: $\widehat{\boldsymbol{\mu}}_k = n_k^{-1} \sum_{i:\, y_i = k} x_i$
  - Sample covariance matrix for each class:

$$\widehat{\Sigma}_k = \frac{1}{n_k - 1} \sum_{i:\, y_i = k} (x_i - \widehat{\boldsymbol{\mu}}_k)(x_i - \widehat{\boldsymbol{\mu}}_k)^\mathsf{T}$$

LDA vs. QDA

- More parameters in QDA than LDA, especially when p is large
- Both are extremely simple to implement
- Both LDA and QDA can perform well on real classification problems
- We can include selected quadratic terms of the covariates, such as X_1X_2 or X_1^2, and still perform LDA

LDA vs. QDA

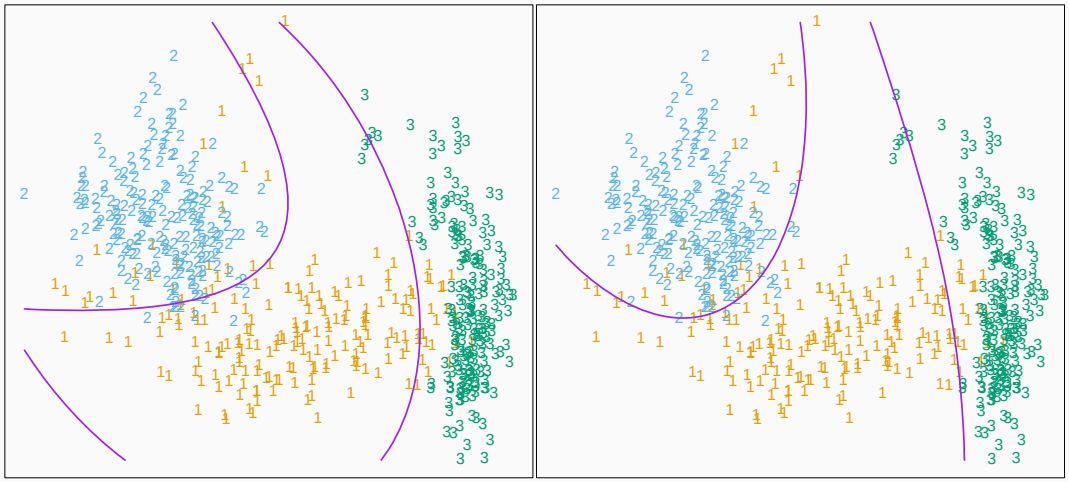


FIGURE 4.6. *Two methods for fitting quadratic boundaries. The left plot shows the quadratic decision boundaries for the data in Figure 4.1 (obtained using LDA in the five-dimensional space $X_1, X_2, X_1X_2, X_1^2, X_2^2$). The right plot shows the quadratic decision boundaries found by QDA. The differences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

Alternative Formulations

Fisher's Criterion

- The between-class variation on those K centroid $(\boldsymbol{\mu}_1, \ldots \boldsymbol{\mu}_K)$ is

$$\mathbf{B} = \sum_{k=1}^{K} \pi_k (\boldsymbol{\mu}_k - \bar{\boldsymbol{\mu}})(\boldsymbol{\mu}_k - \bar{\boldsymbol{\mu}})^{\mathsf{T}}$$

$$\text{where} \quad \bar{\boldsymbol{\mu}} = \sum_{k=1}^{K} \pi_k \boldsymbol{\mu}_k$$

- The within-class variation is just the common covariance matrix Σ that we calculated in LDA, denote it as $\mathbf{W}$.
- If we define a linear combination $Z = a^{\mathsf{T}} X$ such that we want the between-class variance is maximized relative to the within-class variance, we maximize the *Rayleigh quotient*,

$$\underset{a}{\text{maximize}} \ \frac{a^{\mathsf{T}} \mathbf{B} a}{a^{\mathsf{T}} \mathbf{W} a}$$

Fisher's Criterion

- This is a generalized eigenvalue problem, with a given by the largest eigenvalue of $\mathbf{W}^{-1}\mathbf{B}$.
- Similarly, one can find the next direction by extracting the second eigenvector.

Discriminant Coordinates

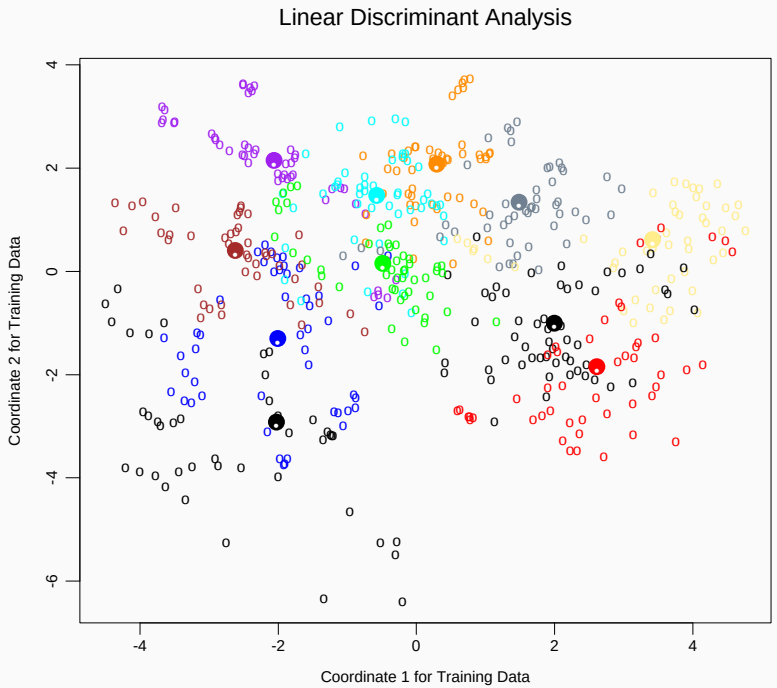


FIGURE 4.4. *A two-dimensional plot of the vowel training data. There are eleven classes with* $X \in \mathbb{R}^{10}$*, and this is the best view in terms of a LDA model (Section 4.3.3). The heavy circles are the projected mean vectors for each class. The class overlap is considerable.*

Discriminant Coordinates

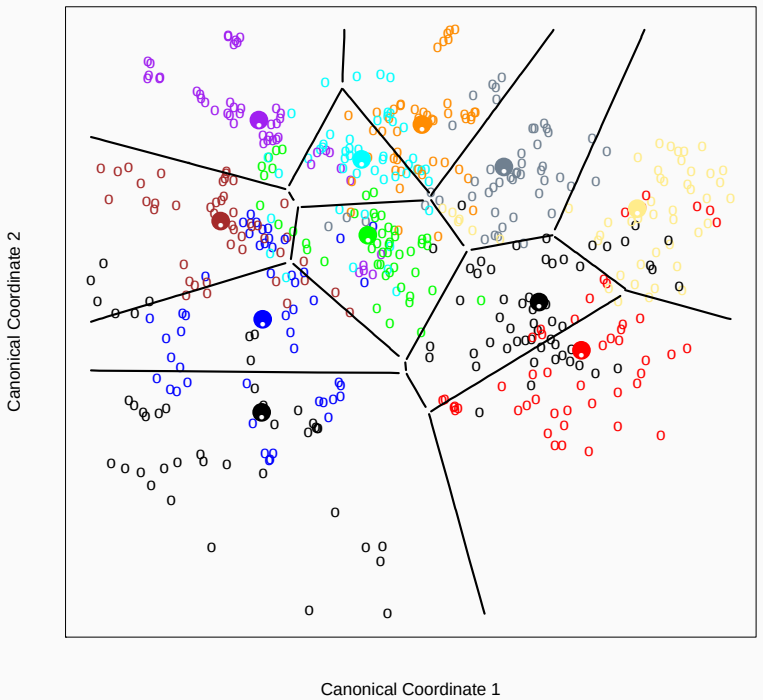


FIGURE 4.11. *Decision boundaries for the vowel training data, in the two-dimensional subspace spanned by the first two canonical variates. Note that in any higher-dimensional subspace, the decision boundaries are higher-dimensional affine planes, and could not be represented as lines.*

Reduced Rank LDA

- Low-dimensional structure of the data may help reduce the noise
- We may consider many different ways to reduce the rank of the data, and perform discriminant analysis on the dimensionality reduced space.
- Example 1 (a simple reduced-rank LDA):
  - The K centroids $(\boldsymbol{\mu}_1, \ldots \boldsymbol{\mu}_K)$ in p-dimensional input space span a subspace of rank $K-1$, denote this subspace as H
  - For any point x, we can project it onto H, and perform LDA on this reduced space
- Example 2 (PCA)
  - Perform PCA on the entire data, and take the first several eigen-vectors as the subspace H

Discriminant Analysis in Large p problems

- When p is large, QDA/LDA may not be applicable, because $\widehat{\Sigma}^{-1}$ does not exist
- Using generalized inverse can easily overfit the data
- A warning sign: Classes are well-separated on the training data could be meaningless for high-dimensional data
- Regularization: sparse LDA, Naive Bayes, RDA

Sparse LDA

- Witten and Tibshirani (2011): penalized LDA

$$\underset{a}{\text{maximize}} \{a^{\mathsf{T}}\mathbf{B}a - P(|a|)\} \quad \text{subject to} \quad a^{\mathsf{T}}\tilde{\mathbf{W}}a \leq 1$$

  where $\tilde{\mathbf{W}}$ is a positive definite estimate of $\mathbf{W}$, and $P(|a|)$ is a penalty function over the vector $|a|$.

- Another approach Clemmensen et al. (2011): similar idea with a different objective function that makes the optimization problem easier.

Regularized Discriminant Analysis (RDA)

- Friedman (1989): shrink the separate covariances of QDA toward a common covariance in LDA. Regularized covariance matrices are

$$\widehat{\Sigma}_k(\alpha) = \alpha\widehat{\Sigma}_k + (1-\alpha)\widehat{\Sigma}$$

- $\alpha \in [0, 1]$, a continuum of models between LDA and QDA, if $\widehat{\Sigma}$ is the pooled covariance matrix used in LDA
- In practice, chose α using CV.
- We can further shrink towards the diagonal covariance, with $\gamma \in [0, 1]$

$$\widehat{\Sigma}_k(\alpha, \gamma) = \alpha\widehat{\Sigma}_k + (1-\alpha)\gamma\widehat{\Sigma} + (1-\alpha)(1-\gamma)\widehat{\sigma}^2\mathbf{I}$$

Naive Bayes

- Recall that the optimal decision rule is

$$\arg\max_k \mathsf{P}(Y = k|X = x) = \arg\max_k \pi_k f_k(x)$$

- We can approximate $f_k(x)$ by

$$f_k(x) \approx \prod_{j=1}^{p} f_{kj}(x_j),$$

  meaning that each dimension of x is approximately independently

- $f_{kj}(x_j)$ can be estimated using histograms (discrete), or kernel densities (continuous)

Remarks

Logistic Regression vs. LDA

- For LDA, the log-posterior odds between class 1 and 0 are linear in x

$$\begin{aligned}\log\frac{\mathsf{P}(Y=1|X=x)}{\mathsf{P}(Y=0|X=x)} &= \log\frac{\pi_1}{\pi_0} - \frac{1}{2}\boldsymbol{\mu}_1^\mathsf{T}\Sigma^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_0^\mathsf{T}\Sigma^{-1}\boldsymbol{\mu}_0 \\ &\quad + x^\mathsf{T}\Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0) \\ &= \alpha_0 + x^\mathsf{T}\boldsymbol{\alpha}\end{aligned}$$

- Logistic model has linear logit by construction

$$\log\frac{\mathsf{P}(Y=1|X=x)}{\mathsf{P}(Y=0|X=x)} = \beta_0 + x^\mathsf{T}\boldsymbol{\beta}$$

- Are they the same estimators?

Logistic Regression vs. LDA

- For LDA, the linearity is a consequence of the Gaussian assumption for the class densities, and the assumption of a common covariance matrix.
- For logistic regression, the linearity comes by construction.
- The difference lies in how the coefficients are estimated.
- Which is more general?
  - LDA assumes Gaussian distribution of X; while logistic leaves the density of X arbitrary

R Functions

- LDA and QDA: R package MASS, functions lda, qda.
- Logistic: R function glm
- General optimization: R function optim