# Linear Statistical Modeling Methods with SAS

## Classification - Part II

Xuemao Zhang
East Stroudsburg University

April 12, 2024

# Outline

- Introduction
- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- Naive Bayes
- DA vs Logistic Regression

# Introduction

- Recall that in classification problem, we try to calculate

$$p_k(x) = P(Y = k|X = x), k = 1, \ldots, K.$$

  where $K$ is the number of elements in $\mathcal{C}$, the set of collection of responses of $Y$.

- Suppose we now have information on $f_k(x) = Pr(X = x|Y = k)$, feature distribution within each class,
- How do we use this to make predictions?
- We apply the Bayes Rule in probability:

**Bayes' Rule**

Let $S_1, S_2, \cdots, S_K$ be a partition of the sample space $S$ with **prior probabilities** $P(S_1), P(S_2), \cdots, P(S_K)$. Suppose an event $A$ occurs and $P(A|S_i)$ is known for each $i = 1, \ldots, K$. Then the **posterior probability** of $S_i$, given that $A$ occurred is

$$P(S_i|A) = \frac{P(A \cap S_i)}{P(A)} = \frac{P(S_i)P(A|S_i)}{\sum_{j=1}^{K} P(S_j)P(A|S_j)}, i = 1, \ldots, K.$$

# Introduction

- Bayes Theorem in our context is:

$$p_k(x) = Pr(Y = k|X = x) = \frac{Pr(X = x|Y = k) \cdot Pr(Y = k)}{Pr(X = x)}$$

or

$$p_k(x) = Pr(Y = k|X = x) = \frac{\pi_k f_k(x)}{\sum_{i=1}^{K} \pi_i f_i(x)},$$

where in the formula, we need

  - $f_k(x) = Pr(X = x|Y = k)$ which is the density for $X$ in class $k$, $k = 1, \ldots, K$
  - $\pi_k = Pr(Y = k), k = 1, \ldots, K$ which is the marginal or prior probability of $Y$ for class $k$.

- We refer to $p_k(x)$ as the posterior probability that an observation posterior $X = x$ belongs to the *kth* class.

# Linear Discriminant Analysis

$\pi_k$ is generally simple to estimate:

- If our data are a random sample of size $n$, then we can use the sample proportion

$$\hat{\pi}_k = \frac{\#\{Y = k\}}{n},$$

  which is the fraction of the training observations that belong to the $k$th class.
- Otherwise can use outside information (eg. historical data)

# Linear Discriminant Analysis

- Technically the notation $f_k(x) = Pr(X = x|Y = k)$ is only correct if $X$ is a discrete random variable. If $X$ is continuous, $f_k(x)dx$ would correspond to the probability of $X$ falling in in a small region $dx$ around $x$.
- Estimate of $f_k(x) = Pr(X = x|Y = k)$ is more difficult. This is a **density estimation** problem.
- In LDA (Linear Discriminant Analysis), we will use Gaussian/normal densities for these, separately in each class.

# Linear Discriminant Analysis when $p = 1$

- There is only one feature $X$.
- The Gaussian density has the form

$$\boxed{f_k(x) = \frac{1}{\sigma_k\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}, -\infty < x < \infty.}$$

  Here $\mu_k$ is the mean, and $\sigma_k^2$ is the variance in class $k, k = 1, \ldots, K$.
- We will assume that all the $\sigma_k = \sigma$ are the same.
- Plugging this into Bayes formula,

$$p_k(x) = \frac{\pi_k \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma}\right)^2}}{\sum_{i=1}^{K} \pi_i \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu_i}{\sigma}\right)^2}}, k = 1, \ldots, K.$$

  Happily, there are simplifications and cancellations.

# Linear Discriminant Analysis when $p = 1$

- To classify at the value $X = x$, we need to see which of the $p_k(x)$ is largest.
- Taking logs, and discarding terms that do not depend on $k$, we see that this is equivalent to assigning $x$ to the class with the largest **discriminant score**:

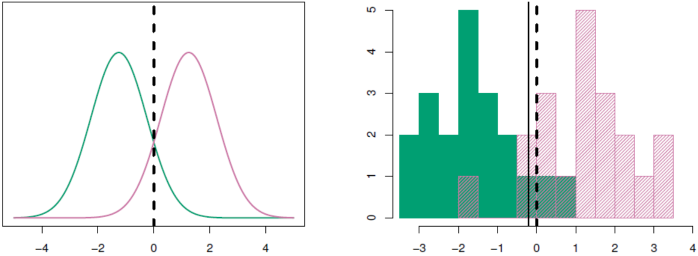
$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

  which is a linear function of $x$.
- If there are $K = 2$ classes and $\pi_1 = \pi_2 = 0.5$, then one can show that the decision boundary is at

$$x = \frac{\mu_1 + \mu_2}{2}.$$

# Linear Discriminant Analysis when $p = 1$

- Example with $\mu_1 = -1.5, \mu_2 = 1.5$, $\pi_1 = \pi_2 = 0.5$, and $\sigma = 1$.




- Typically we don't know these parameters; we just have the training data. In that case we simply estimate the parameters and plug them into the rule.

# Linear Discriminant Analysis when $p = 1$

$$
\begin{aligned}
\hat{\pi}_k =& \frac{n_k}{n}, n_k \text{ is the number of observations in class } k \\
\hat{\mu}_k =& \frac{1}{n_k} \sum_{i:y_i=k} x_i \\
\hat{\sigma}^2 =& \frac{1}{n-k} \sum_{k=1}^{K} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2 \\
=& \frac{1}{n_k} \sum_{k=1}^{K} (n_k - 1)\hat{\sigma}_k^2,
\end{aligned}
$$

where $\hat{\sigma}_k^2 = \dfrac{1}{n_k} \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2$ is the sample variance for the $k$th class. That is, $\hat{\sigma}^2$ is the pooled estimate of the common variance $\sigma^2$.

# Linear Discriminant Analysis when $p > 1$

- When $p > 1$, we consider multivariate normal distribution for $f_k(x) = Pr(X = x|Y = k), k = 1, \ldots, K$.
- To use matrix notation, we define the following matrices:

$$\boxed{\mathbf{x} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix}, \qquad \boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix}}$$

# Linear Discriminant Analysis when $p > 1$

**Definition.** A random vector

$$\mathbf{X} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix}$$

is said to have a $MVN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ distribution if its pdf is given by

$$f(\mathbf{x}) = f(x_1, \ldots, x_p) = \left(\frac{1}{2\pi}\right)^{p/2} \left[\frac{1}{\det \boldsymbol{\Sigma}}\right]^{1/2} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\right].$$

If $n = 2$, the distribution is called Bivariate Normal Distribution. Let $X_1$ and $X_2$ have a bivariate normal distribution, then

$$\boxed{\boldsymbol{\mu} = (\mu_1, \mu_2)', \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}}$$

# Linear Discriminant Analysis when $p > 1$

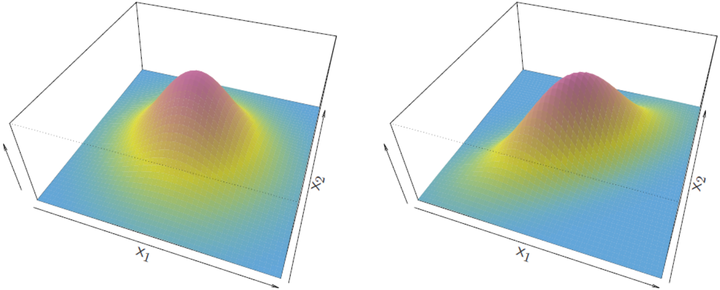
- linear correlation coefficient
  - The linear correlation coefficient of $X_1$ and $X_2$ is defined to be,

$$\boxed{\rho = \frac{Cov(X_1, X_2)}{\sigma_1 \sigma_2}}$$

    where $\sigma_1$ and $\sigma_2$ are the standard deviations of $X_1$ and $X_2$, respectively.
  - $\boxed{-1 \leq \rho \leq 1}$
- Marginal distributions
  - Let $X \sim MVN(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The marginal distribution of any set of component $X$ is multivariate normal with means, variance and covariance obtained by taking the corresponding components of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ respectively.
  - Let $X_1$ and $X_2$ have a bivariate normal distribution. Then
    **(a).** The marginal distribution of $X_1$ is normal with mean $\mu_1$ and variance $\sigma_1^2$.
    **(b).** The marginal distribution of $X_2$ is normal with mean $\mu_2$ and variance $\sigma_2^2$.

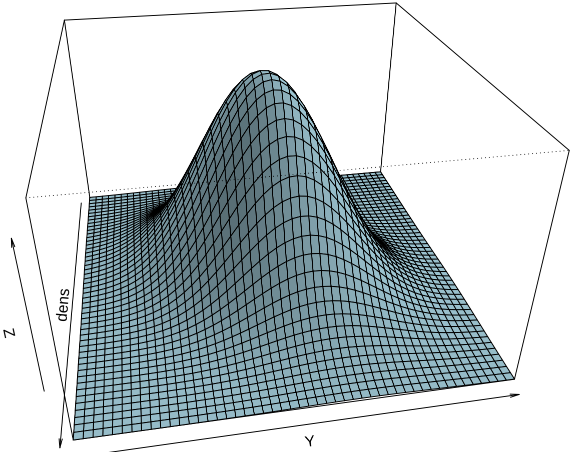
# Linear Discriminant Analysis when $p > 1$

- Bivariate normal density



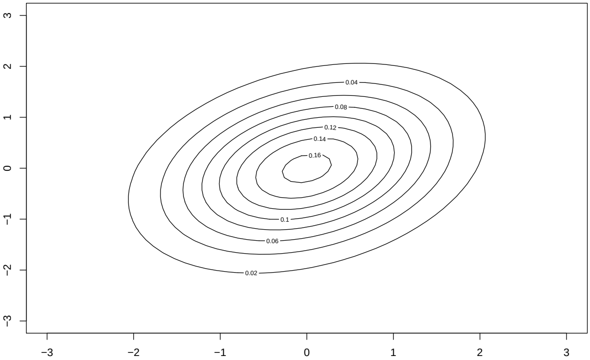
# Linear Discriminant Analysis when $p > 1$

- Bivariate normal density with $\rho = 0.3, \mu_1 = \mu_2 = 0, \sigma_1 = \sigma_2 = 1, cov(X_1, X_2) = 0$:



# Linear Discriminant Analysis when $p > 1$

- BContour plot with $\rho = 0.3, \mu_1 = \mu_2 = 0, \sigma_1 = \sigma_2 = 1, cov(X_1, X_2) = 0$:

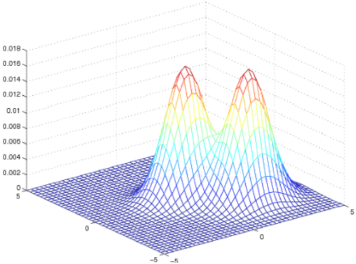


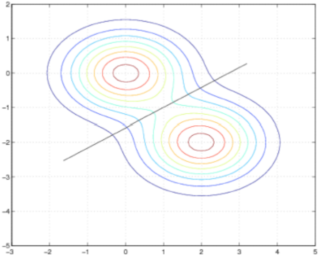
# Linear Discriminant Analysis when $p > 1$

- Discriminant function:

$\delta_k(\mathbf{x}) = \mathbf{x}'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k - \frac{1}{2}\boldsymbol{\mu}_k'\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}_k + \log \pi_k$. Despite its complex form, it is a linear function of $X$:

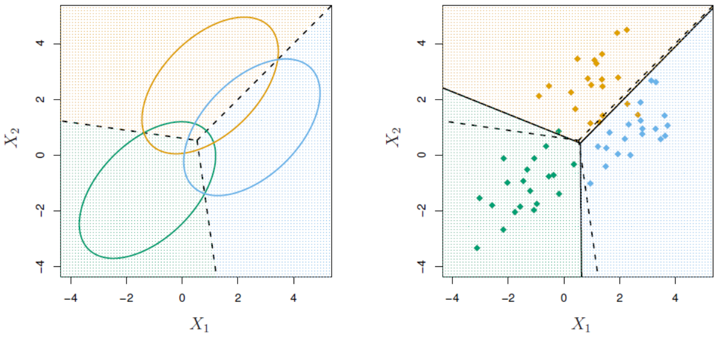
$$\delta_k(\mathbf{x}) = c_{k0} + c_{k1}x_1 + \cdots + c_{kp}x_p$$





# Linear Discriminant Analysis when $p > 1$

- Illustration: $p = 2$ and $K = 3$ classes
  - Here $\pi_1 = \pi_2 = \pi_3 = 1/3$
  - The dashed lines are the exact/true Bayes decision boundaries
  - The solid lines are LDA decision boundaries



# Linear Discriminant Analysis

- From $\delta_k(x)$ back to probabilities:

Once we have estimates $\delta_k(x)$, we can turn these into estimates for class probabilities:

$$\widehat{Pr}(Y = k|X = x) = \frac{e^{\delta_k(x)}}{\sum_{i=1}^{K} e^{\delta_i(x)}}$$

- So classifying to the largest $\delta_k(x)$ amounts to classifying to the class for which $\widehat{Pr}(Y = k|X = x)$ is largest.
- When $K = 2$, we classify to class 2 if $\widehat{Pr}(Y = 2|X = x) > 0.5$, else to class 1.

# Linear Discriminant Analysis

- Example: Consider the Stock Market Data again in the last lecture

```
PROC IMPORT
DATAFILE='/home/u5235839/my_shared_file_links/u5235839/Smarket.csv'
DBMS=CSV
OUT=Smarket;
GETNAMES=YES;
RUN;

data Smarket1;
set Smarket;
/* Create binary indicators */
Up = (Direction = 'Up');
drop Direction;
run;
```

# Linear Discriminant Analysis

- We split the data as training data and test data in the following way because the data is time-series data
  - Suppose the test data are future data

```
data train test;
    set Smarket1;
    if Year < 2005 then output train;
    else output test;
run;
```

# Linear Discriminant Analysis

- Again, we fit an LDA model using the PROC DISCRIM.
  - We have seen that PROC DISCRIM cannot do variable selection.

```
proc discrim data=train testdata=test METHOD=NORMAL
testout=tout TESTLIST TESTLISTERR;
   class Up;
   var Lag1 Lag2 Lag3 Lag4 Lag5 Volume Today;
  run;
```

- METHOD=NORMAL is the default method

# Linear Discriminant Analysis

```
proc print data=tout;
run;
```

# Linear Discriminant Analysis

- If you want to manually check the rate of correct predictions,

```
proc freq data=tout;
tables  Up*_INTO_;
run;
```

# Linear Discriminant Analysis

- The prediction of `Up` or `Down` is based on a 50% threshold to the posterior probabilities. We can change the threshold.
- If the largest posterior probability of group membership is less than the THRESHOLD value, the observation is labeled as **Other**

```
 proc discrim data=train testdata=test METHOD=NORMAL
testout=tout TESTLIST TESTLISTERR THRESHOLD=0.7;
   class Up;
   var Lag1 Lag2 Lag3 Lag4 Lag5 Volume Today;
    run;
```

# Linear Discriminant Analysis

- Let's convert `Other` to `0`
  - Here the function `ifn` is used
    - ★ https://documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/lefunctionsref/n0l3n5z2h31h7wn1fmnqd33ibhap.htm

```
data tout2;
set tout;
_INTO_ = ifn(missing(_INTO_), 0, _INTO_);
/*replace missing values with 0*/
run;

proc print data=tout2;
run;

proc freq data=tout2;
tables  Up*_INTO_;
run;
```

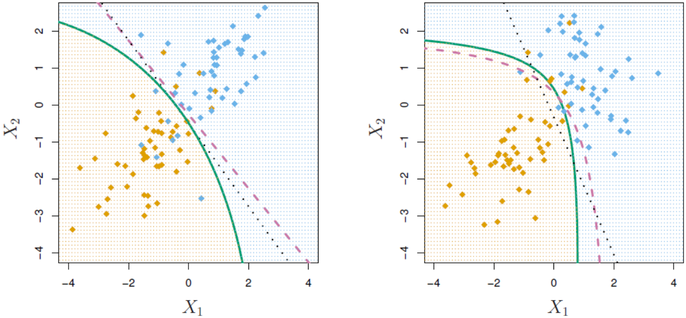
# Quadratic Discriminant Analysis

- LDA assumed that every class has the same variance/covariance
- However, LDA may perform poorly if this assumption is far from true
- QDA (Quadratic Discriminant Analysis) works identically as LDA except that it estimates separate variances/covariance for each class
- That is, $f_k(x) = Pr(Y = k|X = x)$ are Gaussian densities but with different variance-covariance matrix $\mathbf{\Sigma}_k$ in each class $k$, $k = 1, \ldots, K$.

# Quadratic Discriminant Analysis

- QDA results in non-linear decision boundaries (quadratic in fact)
- Discriminant function:

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



# Quadratic Discriminant Analysis

- QDA is implemented using `proc discrim` as well. We need to specify `pool=no` in the `proc discrim` statement.
- By default, `pool=yes` which performs Linear Discriminant Analysis (LDA)

```
    proc discrim data=train testdata=test
    METHOD=NORMAL  pool=no
testout=tout TESTLIST TESTLISTERR;
   class Up;
   var Lag1 Lag2 Lag3 Lag4 Lag5 Volume Today;
  run;
```

- The QDA predictions are not better in this example.

# Naive Bayes

- The method assumes **features are independent** in each class $k$, $k = 1, \ldots, K$.
- It is useful when $p$ is large, and so multivariate methods like QDA and even LDA break down.
- Gaussian naive Bayes assumes each $\boldsymbol{\Sigma}_k$ is diagonal (correlation is 0)

$$
\begin{aligned}
\delta_k(\mathbf{x}) &\propto \log \left[ \pi_k \prod_{j=1}^{p} f_{kj}(x_j) \right] \\
&= -\frac{1}{2} \sum_{j=1}^{p} \left[ \frac{(x_j - \mu_{kj})^2}{\sigma_{kj}^2} + \log \sigma_{kj}^2 \right] + \log \pi_k
\end{aligned}
$$

- can use for mixed feature vectors (qualitative and quantitative). If $X_j$ is qualitative, replace $f_{kj}(x_j)$ with probability mass function (histogram) over discrete categories.
- Despite strong assumptions, naive Bayes often produces good classification results.

# Naive Bayes

```
proc hpbnet data=Smarket1 structure=Naive;
   target Up;
   input Lag1 Lag2 Lag3 Lag4 Lag5 Volume Today/level=int;
   output pred=predicted;
run;

proc print data=predicted;
run;
```

# DA vs Logistic Regression

- Discriminant Analysis model can actually be rewritten as multinomial logistic models:

Beginning with

$$p_k(\mathbf{x}) = Pr(Y = k|\mathbf{X} = \mathbf{x}) = \frac{\pi_k f_k(\mathbf{x})}{\sum_{i=1}^{K} \pi_i f_i(\mathbf{x})},$$

and

$$f_k(\mathbf{x}) \propto \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})_k^{'} \boldsymbol{\Sigma}_k^{-1}(\mathbf{x} - \boldsymbol{\mu}_k)\right]$$

substituting and simplifying we get

$$P(Y = k|\mathbf{x}) = \frac{e^{\eta_k}}{\sum_i e^{\eta_i}}$$

where $\eta_k = \beta_0 + \mathbf{x}^{'}\boldsymbol{\beta} + \mathbf{x}^{'}\boldsymbol{\Sigma}_k^{-1}\mathbf{x}$.

# DA vs Logistic Regression

- This is just a multinomial logistic model with quadratic terms and interactions.
- In particular for LDA (where $\mathbf{\Sigma}_k = \mathbf{\Sigma}$ is pooled) we have cancellation and get

$$\eta_k = \beta_0 + \mathbf{x}^{'}\boldsymbol{\beta} = \beta_0 + \beta_1 x_1 + \ldots + \beta_p x_p$$

  which is simply a linear logistic model.
- The difference is in how the parameters are estimated.
- Logistic regression uses the conditional likelihood based on $Pr(Y|\mathbf{x})$ (known as *discriminative learning).*
- *LDA uses the full likelihood based on* $Pr(\mathbf{X}, Y)$ *(known as generative learning).*
- *Despite these differences, in practice the results are often very similar.*

# Summary

- Logistic regression is very popular for classification, especially when $K = 2$.
- LDA is useful when $n$ is small, or the classes are well separated, and Gaussian assumptions are reasonable. Also when $K > 2$.
- Both Logistic Regression and LDA produce linear boundaries. LDA would do better than Logistic Regression if the assumption of normality hold, otherwise logistic regression can outperform LDA
- KNN is completely non-parametric: No assumptions are made about the shape of the decision boundary.
- We can expect KNN to dominate both LDA and Logistic Regression when the decision boundary is highly non-linear. But KNN does not tell us which features/predictors are important (no table of coefficients)

# Summary

- Naive Bayes is useful when $p$ is very large.
- QDA is a compromise between non-parametric KNN method and the linear LDA and logistic regression
- If the true decision boundary is:
  - Linear: LDA and Logistic outperforms
  - Moderately Non-linear: QDA outperforms
  - More complicated: KNN is superior

# Stock Market Data by Logistic Regression

- Fit a logistic regression model on the training data, and output the model

```
proc logistic data=train outmodel=Model_train;
   model Up(event='1') = Lag1 Lag2;
run;
```

# Stock Market Data by Logistic Regression

- Apply the above model to the test data

```
proc logistic inmodel=Model_train;
   score data=test OUT=predicted_test;
  run;
```

# Stock Market Data by Logistic Regression

- Check the performance of the model to the test data

```
data tout3;
set predicted_test;
score_test = ifn(P_1>0.5, 1, 0);
run;

proc freq data=tout3;
tables  Up*score_test;
run;
```

# License