DATA 622 Assignment 2

CUNY: Spring 2021

Philip Tanofsky

17 March 2021

Introduction

The purpose of this project is to apply generative model approaches to the Palmer Penguin data set available at https://allisonhorst.github.io/palmerpenguins/articles/intro.html. The first approach performs linear discriminant analysis (LDA) on the dataset in order to predict the species of the penguin subjects. The second approach performs a quadratic discriminant analysis in order to predict the species of penguin subjects. The final approach uses the Naive Bayes modeling approach to also predict the species of the penguin subjects. The exploratory data analysis informs the decisions of the predictor variables included for each generative model. A final comparison of the models are presented in order to compare the accuracy of each.

```
# Import required R libraries
library(palmerpenguins)
library(tidyverse)
library(caret)
library(MASS)
library(ggplot2)
library(mvtnorm)
theme_set(theme_classic())
```

Initial Data Inspection

```
ds <- penguins

head(ds)
```

```
## # A tibble: 6 x 8
##   species island bill_length_mm bill_depth_mm flipper_length_~ body_mass_g sex
##   <fct>   <fct>           <dbl>         <dbl>            <int>       <int> <fct>
## 1 Adelie  Torge~           39.1          18.7              181        3750 male
## 2 Adelie  Torge~           39.5          17.4              186        3800 fema~
## 3 Adelie  Torge~           40.3          18                195        3250 fema~
## 4 Adelie  Torge~           NA            NA                 NA          NA <NA>
## 5 Adelie  Torge~           36.7          19.3              193        3450 fema~
## 6 Adelie  Torge~           39.3          20.6              190        3650 male
## # ... with 1 more variable: year <int>
```

```
summary(ds)
```

```
##       species          island    bill_length_mm  bill_depth_mm
##  Adelie   :152   Biscoe   :168   Min.   :32.10   Min.   :13.10
##  Chinstrap: 68   Dream    :124   1st Qu.:39.23   1st Qu.:15.60
##  Gentoo   :124   Torgersen: 52   Median :44.45   Median :17.30
##                                  Mean   :43.92   Mean   :17.15
##                                  3rd Qu.:48.50   3rd Qu.:18.70
##                                  Max.   :59.60   Max.   :21.50
##                                  NA's   :2       NA's   :2
##  flipper_length_mm  body_mass_g       sex           year
##  Min.   :172.0     Min.   :2700   female:165   Min.   :2007
##  1st Qu.:190.0     1st Qu.:3550   male  :168   1st Qu.:2007
##  Median :197.0     Median :4050   NA's  : 11   Median :2008
##  Mean   :200.9     Mean   :4202                Mean   :2008
##  3rd Qu.:213.0     3rd Qu.:4750                3rd Qu.:2009
##  Max.   :231.0     Max.   :6300                Max.   :2009
##  NA's   :2         NA's   :2
```

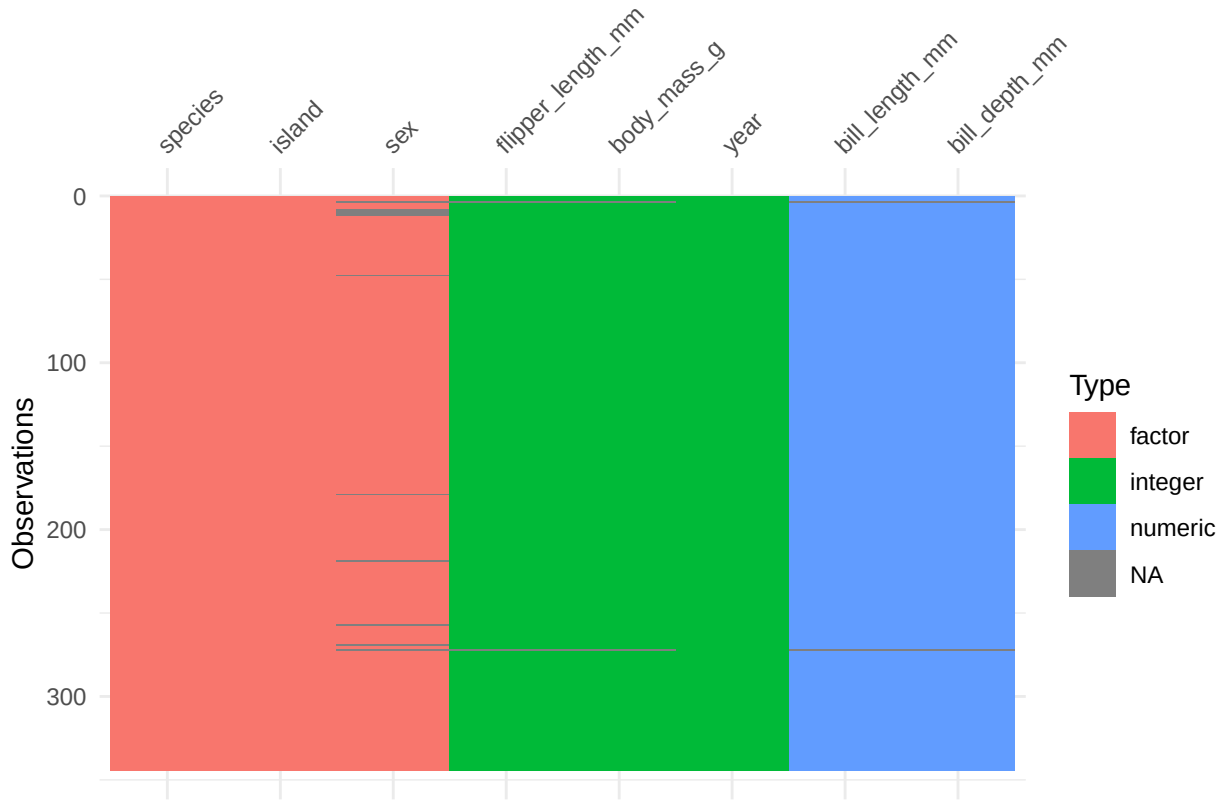
```
dim(ds)
```

```
## [1] 344   8
```

```
glimpse(ds)
```

```
## Rows: 344
## Columns: 8
## $ species           <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, A...
## $ island            <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torge...
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34....
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18....
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, ...
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 347...
## $ sex               <fct> male, female, female, NA, female, male, female, m...
## $ year              <int> 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2007, 2...
```

```
visdat::vis_dat(ds)
```

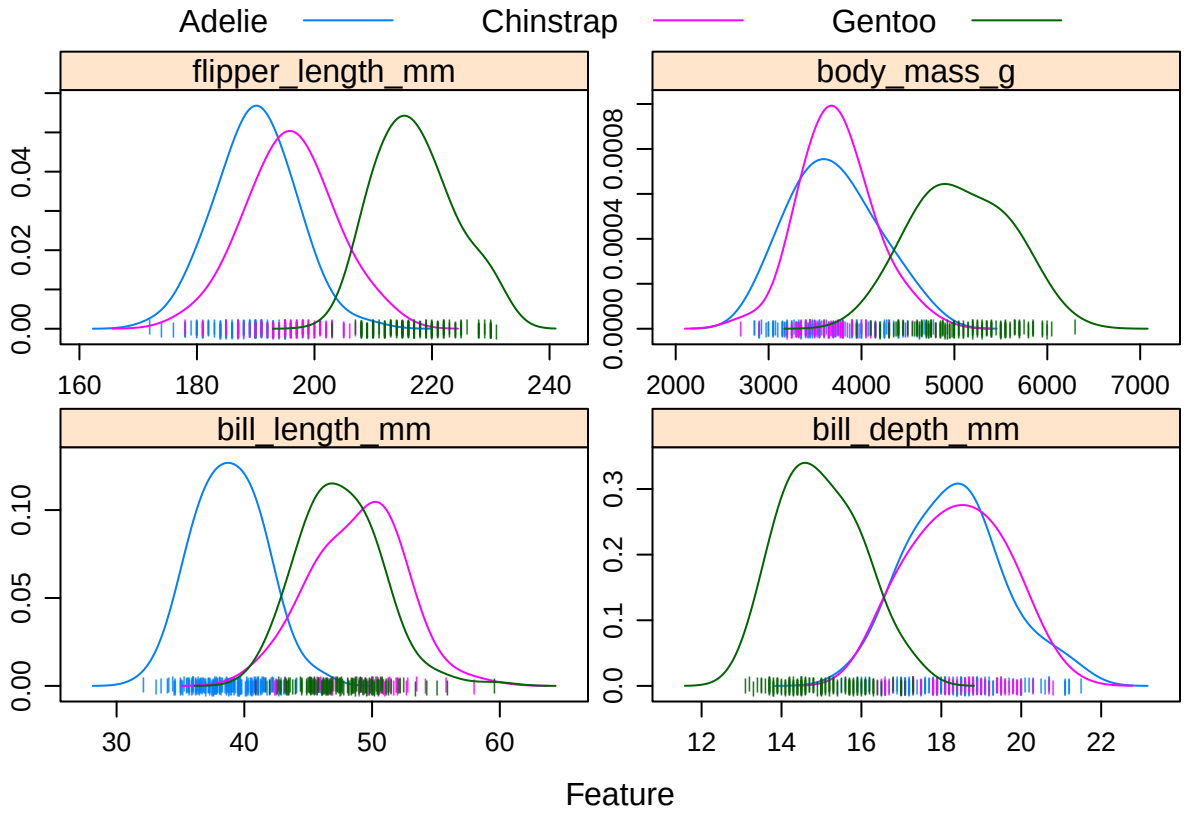


Exploratory Data Analysis

For linear discriminant analysis, two assumptions are made about the data. One, the predictors are normally distributed, which means the data follows a Gaussian distribution for each class. Two, the classes have class-specific means but also have equal variance and covariance

First step, confirm multivariate normal distribution. A density plot of the four continuous variables by species indicates the normal distribution for the class-specific plots.

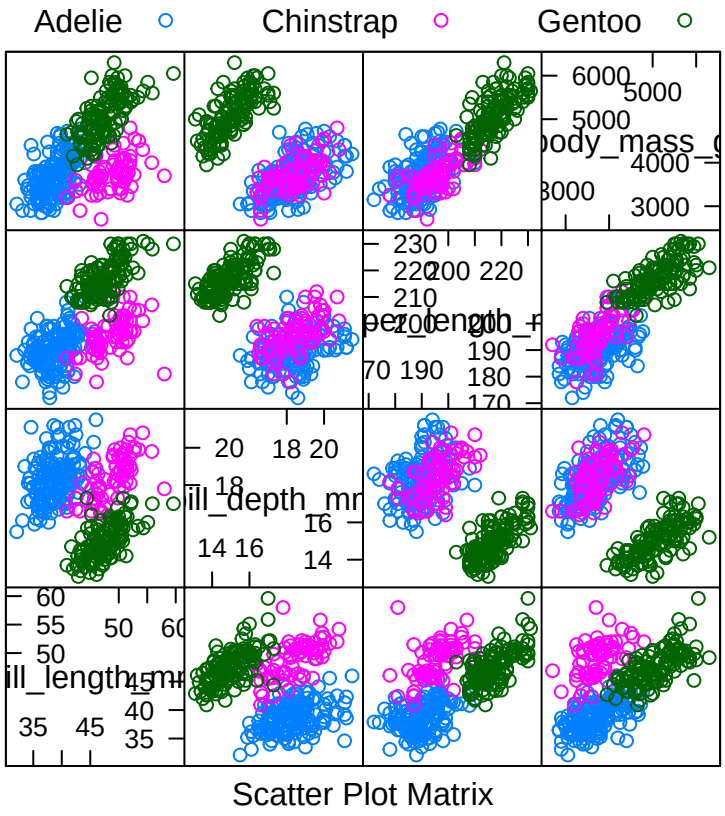
```
# Overlayed density plots
featurePlot(x = penguins[, 3:6],
            y = penguins$species,
            plot = "density",
            # Pass in options to xyplot() to
            # make it prettier
            scales = list(x = list(relation="free"),
                          y = list(relation="free")),
            adjust = 1.5,
            pch = "|",
            layout = c(2, 2),
            auto.key = list(columns = 3))
```



The variable $flipper_length_mm$ shows the clearest example of normal distribution by species class. The four density plots also indicate common evaluations between the Adelie and Chinstrap species for the independent variables flipper_length_mm, body_mass_g, and bill_depth_mm. The remaining continuous variable, bill_length_mm, shows an overlap between the Chinstrap and Gentoo species.

The scatterplot matrix of the continuous variables indicates the relationships between the independent variables for each of the three penguin species.

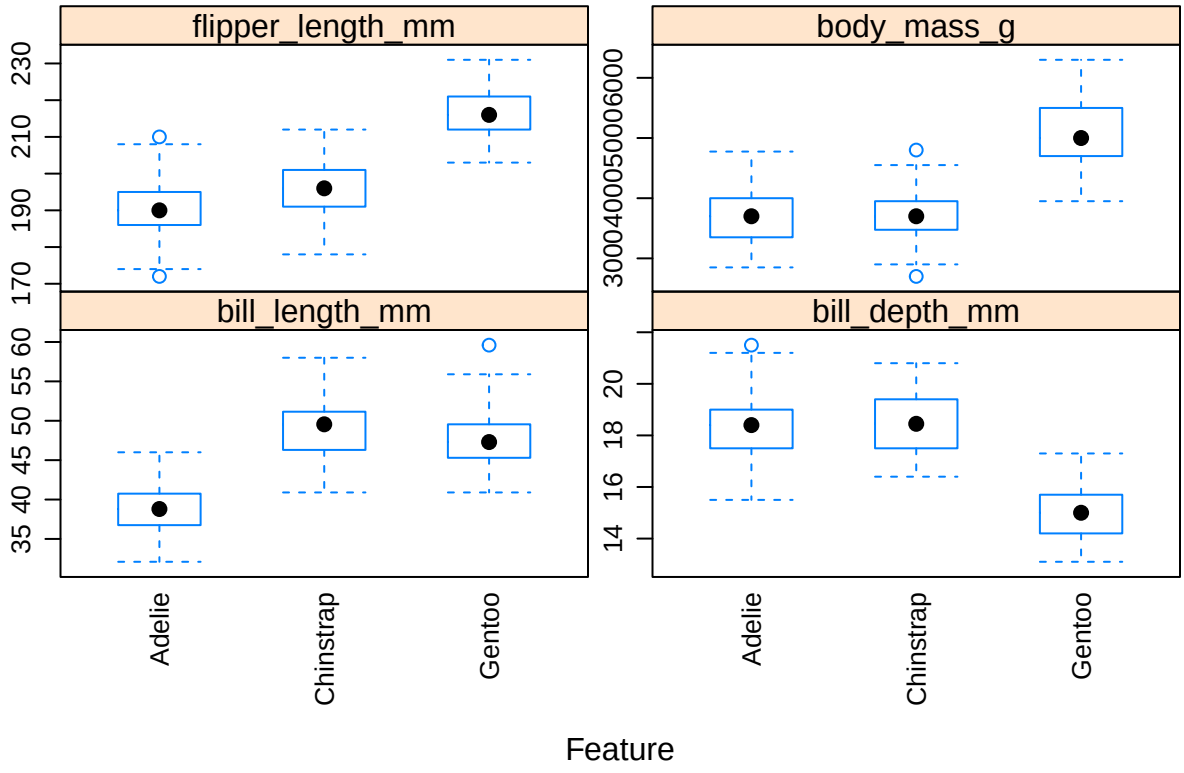
```
# Use featurePlot
# https://topepo.github.io/caret/visualizations.html

# Scatterplot
featurePlot(x = penguins[, 3:6],
            y = penguins$species,
            plot = "pairs",
            # Add a key at the top
            auto.key = list(columns = 3))
```



The independent variable $bill_length_mm$ stands out by providing a clear distinction between the three penguin species when compared with any of the other three independent variables. As seen in the above scatterplot, when using any two of the other three independent variables, the plot results in a clear overlap of the Adelie and Chinstrap species, an expected result given the prior density plots.

So far, the independent variable $bill_length_mm$ is a prime candidate to be included in the generative models. The key will be identifying which of the other variables will provide additional significance to the model.

```
featurePlot(x = penguins[, 3:6],
            y = penguins$species,
            plot = "box",
            ## Pass in options to bwplot()
            scales = list(y = list(relation="free"),
                          x = list(rot = 90)),
            layout = c(2,2),
            auto.key = list(columns = 2))
```



With the first assumption confirmed, the second assumption is to confirm the similar covariance across the species classes. The following Covariance checks

```
# Compute covariance matrix
#cov_mat <- cov(penguins[,3:6],  use = "complete.obs")
#round(cov_mat, 2)

# Covariance matrix
p_g <- penguins %>% filter(species == 'Gentoo')
cov_mat <- cov(p_g[,3:6],  use = "complete.obs")
round(cov_mat, 2)
```

```
##                    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
## bill_length_mm               9.50          1.95             13.21     1039.63
## bill_depth_mm                1.95          0.96              4.50      355.69
## flipper_length_mm           13.21          4.50             42.05     2297.14
## body_mass_g               1039.63        355.69           2297.14   254133.18
```

```
p_a <- penguins %>% filter(species == 'Adelie')
cov_mat <- cov(p_a[,3:6],  use = "complete.obs")
round(cov_mat, 2)
```

```
##                    bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
## bill_length_mm               7.09          1.27              5.67      670.36
## bill_depth_mm                1.27          1.48              2.45      321.44
```

```
## flipper_length_mm           5.67          2.45            42.76     1404.03
## body_mass_g                670.36        321.44          1404.03   210282.89
```

```
p_c <- penguins %>% filter(species == 'Chinstrap')
cov_mat <- cov(p_c[,3:6],  use = "complete.obs")
round(cov_mat, 2)
```

```
##                   bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
## bill_length_mm             11.15          2.48             11.23      659.20
## bill_depth_mm               2.48          1.29              4.70      263.79
## flipper_length_mm          11.23          4.70             50.86     1758.54
## body_mass_g                659.20        263.79           1758.54   147713.45
```

```
# Means
p_a[,3:6] %>% summarise_each(funs( mean( .,na.rm = TRUE)))
```

```
## # A tibble: 1 x 4
##   bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
##            <dbl>         <dbl>             <dbl>       <dbl>
## 1           38.8          18.3              190.       3701.
```

```
p_g[,3:6] %>% summarise_each(funs( mean( .,na.rm = TRUE)))
```

```
## # A tibble: 1 x 4
##   bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
##            <dbl>         <dbl>             <dbl>       <dbl>
## 1           47.5          15.0              217.       5076.
```

```
p_c[,3:6] %>% summarise_each(funs( mean( .,na.rm = TRUE)))
```

```
## # A tibble: 1 x 4
##   bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
##            <dbl>         <dbl>             <dbl>       <dbl>
## 1           48.8          18.4              196.       3733.
```

Assessing the mean of each continuous independent variable for each species class, the variable *bill_depth_mm* highlights the least difference in mean values of four variables. In fact, the difference in mean for Adelie and Chinstrap is one tenth of the measurement.

Comparing the variance of each continuous independent variable across the three species classes, *body_mass_g* shows the divergence in the variance evaluation, particularly for the Gentoo species. Then again, this variable is a weight measured in grams, so perhaps dividing each value by 1000 to measure in kilograms would make the variance seems less divergent. The prior density plot does show the wider distribution among Gentoo species for *body_mass_g*.

Highlighting the covariance, in which a low number indicates a weak relationship, *bill_length_mm* and *bill_depth_mm*. Two variables, *bill_depth_mm* and *flipper_length_mm*, also show a comparatively low covariance. *bill_length_mm* and *flipper_length_mm* indicates a covariance higher than the aforementioned pairs, but the resulting values across the species classes are much better than the not listed variable pairs.

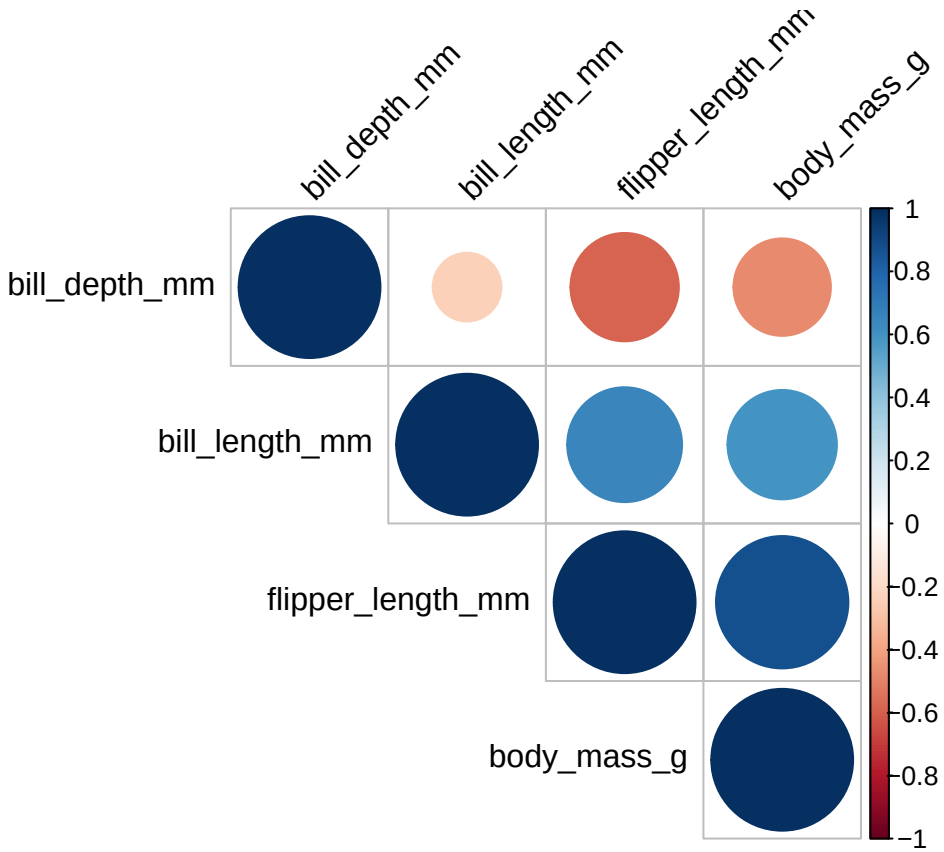
Correlation matrix

Based on the covariance matrix, the correlation matrix should confirm the variable relationships as previously noted. Again, *bill_length_mm* and *bill_depth_mm* show the least correlation. *body_mass_g* and *flipper_length_mm* have a high correlation.

```
# Compute correlation matrix
cor_mat <- cor(penguins[,3:6],  use = "complete.obs")
round(cor_mat, 2)
```

```
##                   bill_length_mm bill_depth_mm flipper_length_mm body_mass_g
## bill_length_mm              1.00         -0.24              0.66        0.60
## bill_depth_mm              -0.24          1.00             -0.58       -0.47
## flipper_length_mm           0.66         -0.58              1.00        0.87
## body_mass_g                 0.60         -0.47              0.87        1.00
```

```
library(corrplot)
corrplot(cor_mat, type = "upper", order = "hclust",
         tl.col = "black", tl.srt = 45)
```




```
# Plots of individual variables
#http://www.sthda.com/english/articles/32-r-graphics-essentials/133-plot-one-variable-frequency-graph-d


# Change density plot fill colors by groups
```

```r
p <- ggplot(penguins, aes(x=bill_length_mm, fill=species)) +
  geom_density(alpha=0.4)

p

a <- penguins %>%
      filter(species == 'Gentoo') %>%
      ggplot(aes(x = bill_length_mm))

#a <- ggplot(penguins, aes(x = bill_length_mm))

a + geom_histogram(bins = 30, color = "black", fill = "gray") +
  geom_vline(aes(xintercept = mean(bill_length_mm)),
             linetype = "dashed", size = 0.6)

b <- ggplot(penguins, aes(x = bill_depth_mm))

b + geom_histogram(bins = 30, color = "black", fill = "gray") +
  geom_vline(aes(xintercept = mean(bill_depth_mm)),
             linetype = "dashed", size = 0.6)

c <- ggplot(penguins, aes(x = flipper_length_mm))

c + geom_histogram(bins = 30, color = "black", fill = "gray") +
  geom_vline(aes(xintercept = mean(flipper_length_mm)),
             linetype = "dashed", size = 0.6)

d <- ggplot(penguins, aes(x = body_mass_g))

d + geom_histogram(bins = 30, color = "black", fill = "gray") +
  geom_vline(aes(xintercept = mean(body_mass_g)),
             linetype = "dashed", size = 0.6)
```

LDA: Linear Discriminant Analysis

LDA does not handle categorical data well.

LDA assumes the feature variables come from multivariate normal distribution, all of them continuous

http://www.sthda.com/english/articles/36-classification-methods-essentials/146-discriminant-analysis-essentials-in-r/

LDA assumes the predictors are normally distributed (Gaussian distribution) and that the different classes have class-specific means and equal variance/covariance

Make sure each variable is normally distributed

https://web.stanford.edu/class/stats202/notes/Classification/LDA.html That is, within each class the features have multivariate normal distribution with center depending on the class and common covariance

In the covariance matrix in the output, the off-diagonal elements contain the covariances of each pair of variables. The diagonal elements of the covariance matrix contain the variances of each variable. The variance measures how much the data are scattered about the mean.

bill_depth, bill_length, body_mass

```r
# Load the data
data("penguins")

# Only complete entries
penguins <- na.omit(penguins)

# Remove 'year' and 'sex feature
# Apparently leaving 'island' in for LDA improves the model
drops <- c("year", "sex")
penguins <- penguins[ , !(names(penguins) %in% drops)]

#Split the data into training (75%) and test set (25%)
set.seed(123)
training.samples <- penguins$species %>%
  createDataPartition(p = 0.75, list=FALSE)
train.data <- penguins[training.samples, ]
test.data <- penguins[-training.samples, ]

#2. Normalize the data. Categorial variables are automatically ignored from normalizing
# Estimate preprocessing parameters
preproc.param <- train.data %>%
  preProcess(method = c("center", "scale"))

# Transform the data using the estimated parameters
train.transformed <- preproc.param %>% predict(train.data)
test.transformed <- preproc.param %>% predict(test.data)
```

Focus on normal distributions

If n is small and the distribution of the predictors X is approximately normal in each of the classes, the LD model is more stable than logistic regression

Correlation will cause a line in the Gaussian density plot

When there are K classes, linear discriminant analysis can be viewed exactly in a K-1 dimensional plot. Measuring which centroid is the closest. Distance in the subspace

```r
# Fit the model
model <- lda(species~bill_length_mm + flipper_length_mm, data = train.transformed)
# Make predictions
predictions <- model %>% predict(test.transformed)

# Confusion matrix
table(predictions$class, test.transformed$species)
```

```
##
##              Adelie Chinstrap Gentoo
##   Adelie         35         0      0
##   Chinstrap       1        15      0
##   Gentoo          0         2     29
```

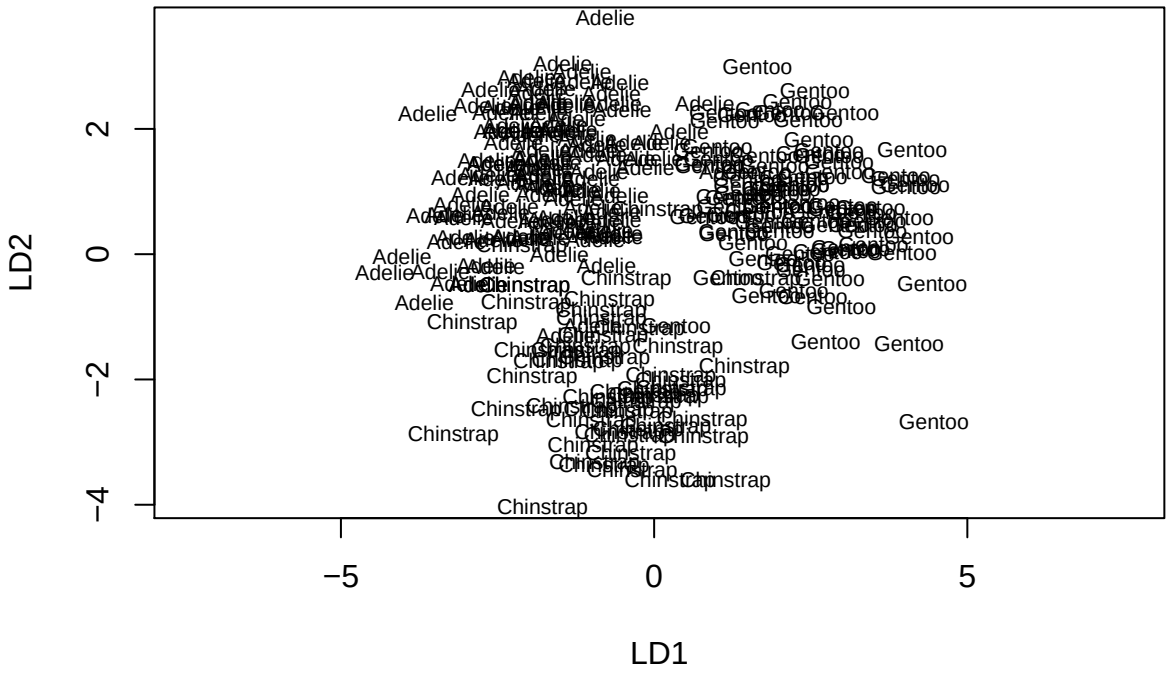
```r
# Model accuracy
mean(predictions$class == test.transformed$species)
```

```
## [1] 0.9634146
```

```
# Output Model
model
```

```
## Call:
## lda(species ~ bill_length_mm + flipper_length_mm, data = train.transformed)
##
## Prior probabilities of groups:
##    Adelie Chinstrap    Gentoo
## 0.4382470 0.2031873 0.3585657
##
## Group means:
##           bill_length_mm flipper_length_mm
## Adelie        -0.9366488        -0.7688194
## Chinstrap      0.8659650        -0.3862079
## Gentoo         0.6540795         1.1585193
##
## Coefficients of linear discriminants:
##                          LD1       LD2
## bill_length_mm    0.1696402 -2.133231
## flipper_length_mm 1.9798393  1.513973
##
## Proportion of trace:
##    LD1    LD2
## 0.6858 0.3142
```

```
# Display model
plot(model)
```




```
names(predictions)
```

```
## [1] "class"     "posterior" "x"
```

```
# Predicted classes
head(predictions$class, 6)
```

```
## [1] Adelie Adelie Adelie Adelie Adelie Adelie
## Levels: Adelie Chinstrap Gentoo
```

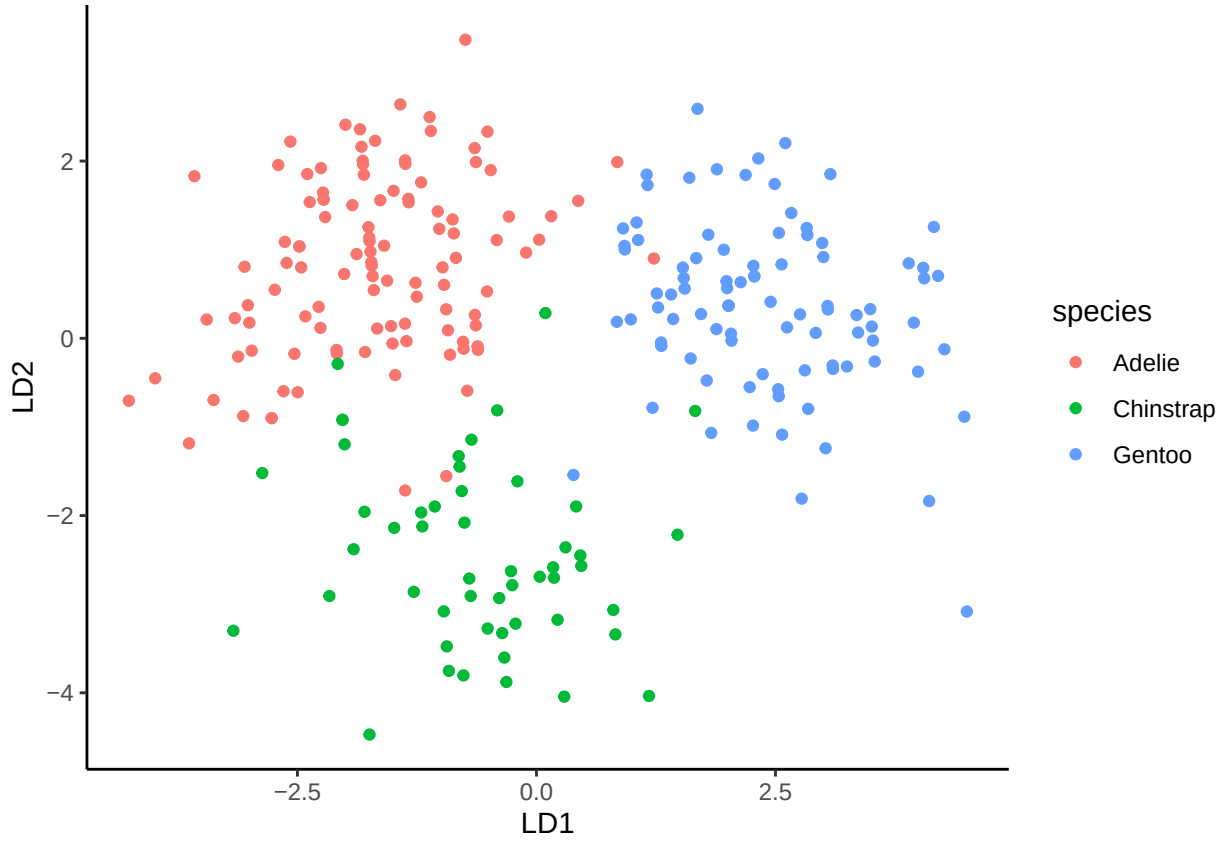
```
# Predicted probabilities of class membership
head(predictions$posterior, 6)
```

```
##      Adelie    Chinstrap       Gentoo
## 1 0.9900706 9.927921e-03 1.454851e-06
## 2 0.9937551 6.222693e-03 2.217512e-05
## 3 0.9976779 2.155645e-03 1.665020e-04
## 4 0.9998292 1.654396e-04 5.315367e-06
## 5 0.9999366 2.948069e-05 3.395157e-05
## 6 0.9986215 1.369715e-03 8.798491e-06
```

```
# Linear discriminants
head(predictions$x, 3)
```

```
##         LD1        LD2
## 1 -2.971129 -0.2182306
## 2 -2.261646  0.1569107
## 3 -1.710373  0.6621646
```

```
# Plot
lda.data <- cbind(train.transformed, predict(model)$x)
ggplot(lda.data, aes(LD1, LD2)) +
  geom_point(aes(color = species))
```




```
# Model accuracy
mean(predictions$class==test.transformed$species)
```

```
## [1] 0.9634146
```

```
sum(predictions$posterior[ ,1] >= .5)
```

```
## [1] 35
```

```
# QDA

# Remove 'island' feature as it was causing rank deficiency in group Chinstrap
#drops <- c("flipper_length_mm", "body_mass_g", "island")
drops <- c("island")
```

```r
train.transformed <- train.transformed[ , !(names(train.transformed) %in% drops)]

# Fit the model
model <- qda(species~., data = train.transformed)

# Output model results
model

# Make predictions
predictions <- model %>% predict(test.transformed)

# Model accuracy
mean(predictions$class == test.transformed$species)
```

https://www.geeksforgeeks.org/linear-discriminant-analysis-in-r-programming/

QDA: Quadratic Discriminant Analysis

Same link as above

QDA: works well with fewer features, that's when NB works well, works with higher number of features

mixed features can be used for NB

NB: Naive Bayes

https://www.r-bloggers.com/2018/01/understanding-naive-bayes-classifier-using-r/

```r
library(e1071)

# Next load the Titantic dataset
data("Titanic")

# Save into a data frame and view it
t_df <- as.data.frame(Titanic)

# Creating data from table
repeating_sequence <- rep.int(seq_len(nrow(t_df)), t_df$Freq)

# Create the dataset by row repetition created
t_ds <- t_df[repeating_sequence, ]

# We no longer need the frequency, drop the feature
t_ds$Freq = NULL

# Fitting the Naive Bayes model
nbm <- naiveBayes(Survived~., data=t_ds)
# Output the model
nbm

# Prediction on the dataset
```

```r
nb_predictions <- predict(nbm, t_ds)
# Confusion matrix to check accuracy
table(nb_predictions, t_ds$Survived)

# Getting started with Naive Bayes in mlr
library(mlr)

# Create a classification task for learning on Titantic Dataset and specify the target feature
task <- makeClassifTask(data = t_ds, target="Survived")

# Initialize the Naive Bayes classifier
selected_model <- makeLearner("classif.naiveBayes")

# Train the model
nb_mlr <- train(selected_model, task)

# Read the model learned
nb_mlr$learner.model

# Predict on the dataset without passing the target feature
predictions_mlr <- as.data.frame(predict(nb_mlr, newdata = t_ds[,1:3]))

# Confusion matrix to check accuracy
table(predictions_mlr[,1], t_ds$Survived)
```

https://www.geeksforgeeks.org/naive-bayes-classifier-in-r-programming/

Palmer Penguins citation

==== Prompt =====

```
Homework # 2 (Generative Models) (100 points) Due on March 12, 11:59pm EST
We will be working with the Penguin dataset again as we did for Homework #1. Please use "Species" as yo
Using the target variable, Species, please conduct:
a. LinearDiscriminantAnalysis(30points):
a. Youwanttoevaluateallthe'features'ordependentvariablesandsee what should be in your model. Please comm
b. Justasuggestion:YoumightwanttoconsiderexploringfeaturePlot on the caret package. Basically, you look
c. Fit your LDA model using whatever predictor variables you deem appropriate. Feel free to split the da
d. Lookatthefitstatistics/accuracyrates.
b. QuadraticDiscriminantAnalysis(30points)
a. Samestepsasabovetoconsider
c. Naive Bayes (30 points)
a. Samestepsasabovetoconsider
d. Commentonthemodelsfits/strength/weakness/accuracyforallthesethree models that you worked with. (10 po
```