DATA 622 Assignment 1

CUNY: Spring 2021

Philip Tanofsky

19 February 2021

Introduction

The purpose of this project is to apply logistic regression approaches to the Palmer Penguin data set available at https://allisonhorst.github.io/palmerpenguins/articles/intro.html. The first approach performs binary logistic regression on the dataset in order to predict a species or not of the penguin subjects. The second approach utilizes all three species to perform a multinomial logistic regression in order to predict the species of penguin subjects. While the primary goal of the logistic regression is to predict the penguin species, statistical interpretation also presents the context for the prediction models.

```
# Import required R libraries
library(palmerpenguins)
library(dplyr)
library(ggplot2)
library(tidyr)
library(caret)
library(MASS)
library(pROC)
library(nnet) # Used for multinomial logistic regression
library(mlogit)
library(stargazer)
library(popbio)
# Set theme, based on the Penguin vignettes
theme_set(theme_minimal())
```

The palmer penguins dataset consists of 8 variables, 7 independent variables and 1 dependent variable (species).

Variables

- species: species of the penguin observed (dependent variable)
- island: island of penguin's observation
- bill_length_mm: penguin bill length in millimeters
- bill_depth_mm: penguin bill depth in millimeters
- flipper_length_mm: penguin flipper length in millimeters
- body_mass_g: penguin body mass in grams

- sex: penguin sex
- year: year of observation

EDA

Initial data summary and exploratory data analysis.

```
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)
```

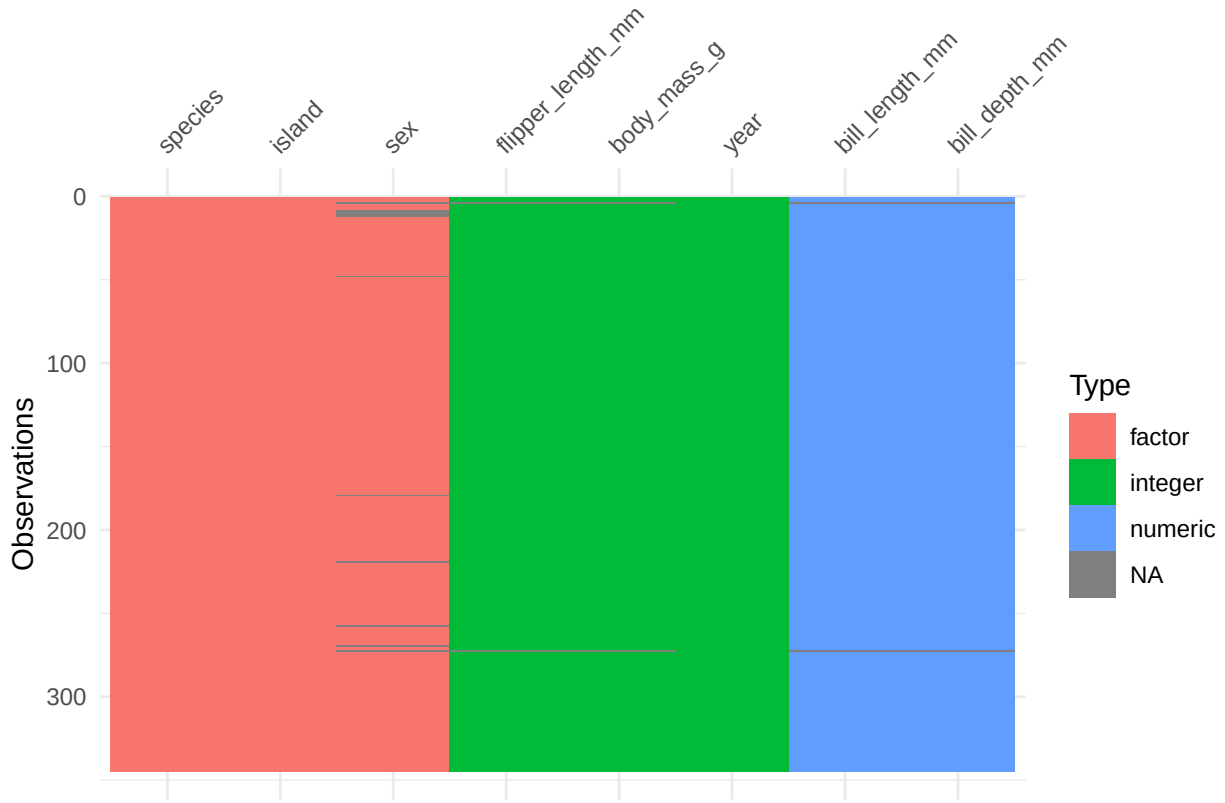
```
## [1] 344   8
```

```
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)
```



Initial summary outputs show 344 instances of the 8 variables. The final graph indicates the missing values among the 8 variables. Variables with missing values include *sex*, *bill_length_mm*, *bill_depth_mm*, *flipper_length_mm* and *body_mass_g*.

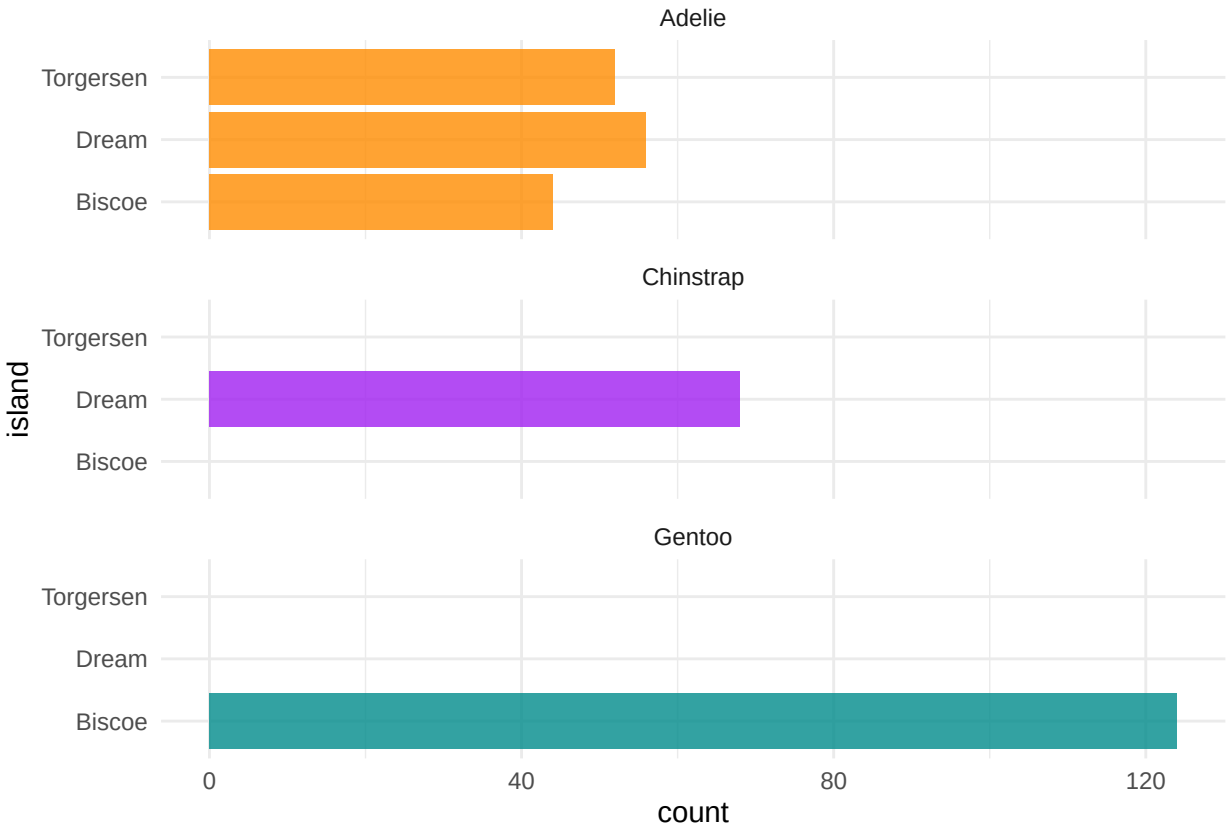
```
# Penguins data has three factor variables
ds %>%
  dplyr::select(where(is.factor)) %>%
  glimpse()
```

```
## Rows: 344
## Columns: 3
## $ species <fct> Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Adelie, Ade...
## $ island  <fct> Torgersen, Torgersen, Torgersen, Torgersen, Torgersen, Torg...
## $ sex     <fct> male, female, female, NA, female, male, female, male, NA, N...
```

```r
# Count penguins for each species / island
ds %>%
  count(species, island, .drop=F)
```

```
## # A tibble: 9 x 3
##   species   island        n
##   <fct>     <fct>     <int>
## 1 Adelie    Biscoe       44
## 2 Adelie    Dream        56
## 3 Adelie    Torgersen    52
## 4 Chinstrap Biscoe        0
## 5 Chinstrap Dream        68
## 6 Chinstrap Torgersen     0
## 7 Gentoo    Biscoe      124
## 8 Gentoo    Dream         0
## 9 Gentoo    Torgersen     0
```

```r
ggplot(ds, aes(x = island, fill = species)) +
  geom_bar(alpha = 0.8) +
  scale_fill_manual(values = c("darkorange", "purple", "cyan4"),
                    guide = F) +
  theme_minimal() +
  facet_wrap(~species, ncol = 1) +
  coord_flip()
```

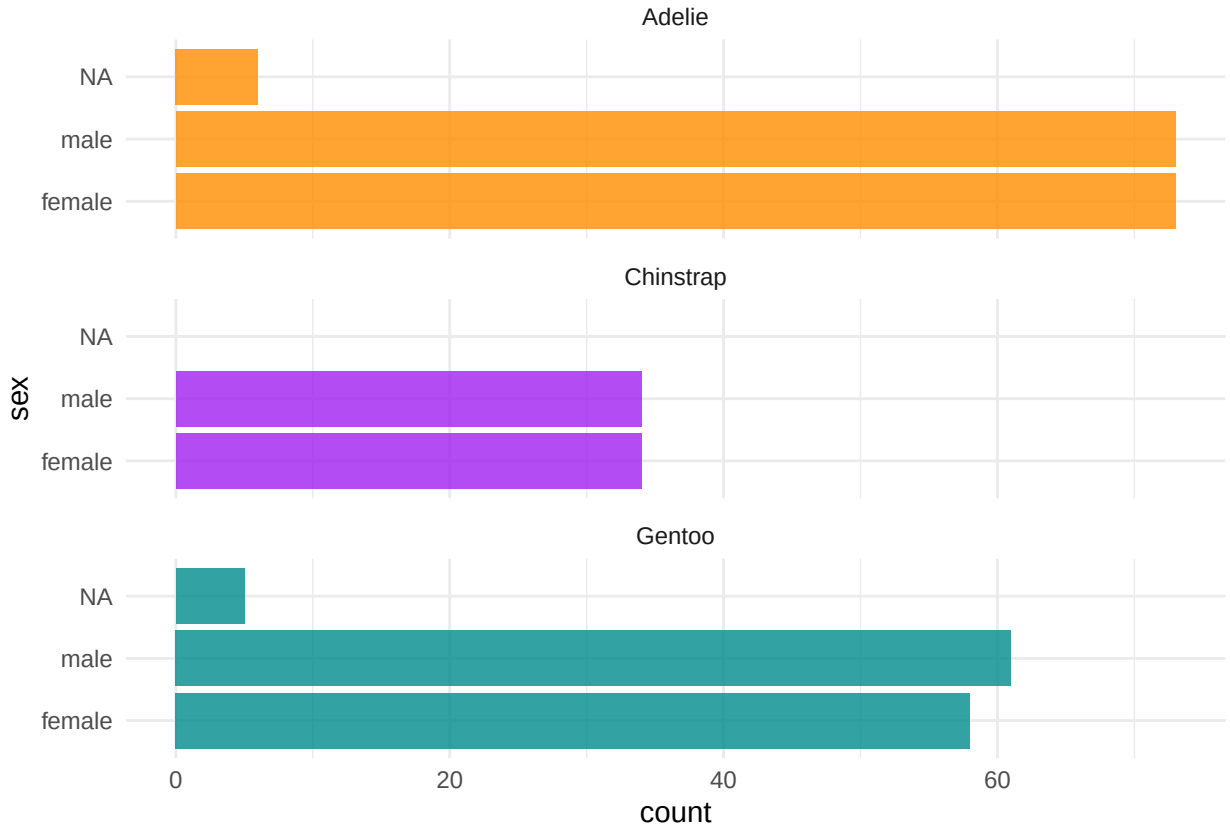


The above plot indicates the population of penguin species based on island. Interesting finding that Chinstrap are only observed on Dream island and Gentoo are only observed on Biscoe island, while the Adelie species are observed on all three islands in the study.

```
# Count penguins for each species / sex
ds %>%
  count(species, sex, .drop = F)
```

```
## # A tibble: 8 x 3
##   species   sex        n
##   <fct>     <fct>  <int>
## 1 Adelie    female    73
## 2 Adelie    male      73
## 3 Adelie    <NA>       6
## 4 Chinstrap female    34
## 5 Chinstrap male      34
## 6 Gentoo    female    58
## 7 Gentoo    male      61
## 8 Gentoo    <NA>       5
```

```
ggplot(ds, aes(x = sex, fill = species)) +
  geom_bar(alpha = 0.8) +
  scale_fill_manual(values = c("darkorange", "purple", "cyan4"),
                    guide = F) +
  theme_minimal() +
  facet_wrap(~species, ncol = 1) +
  coord_flip()
```

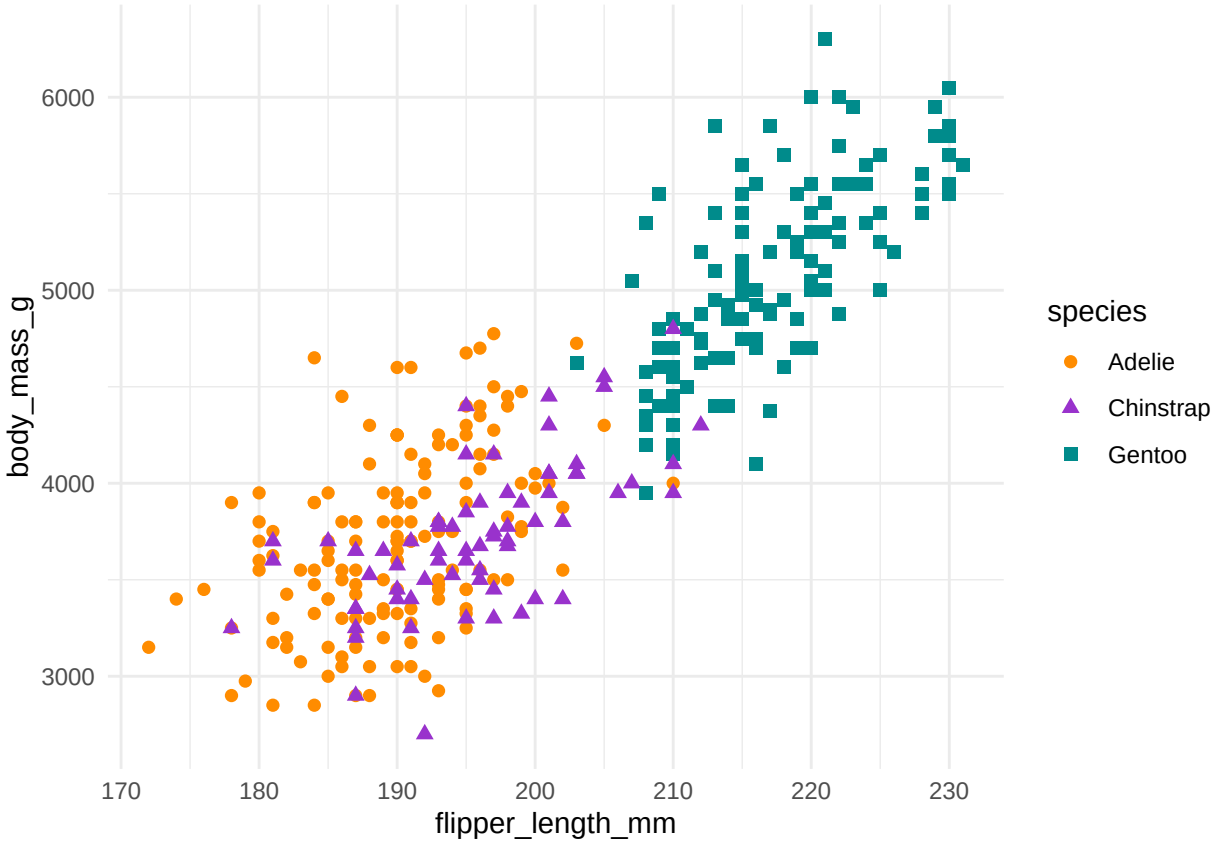


The above breakdown of penguin species by sex shows an expected ratio of near 50-50 for each species.

```
# Penguins data also has four continuous variables, making six unique scatterplots
ds %>%
  dplyr::select(body_mass_g, ends_with("_mm")) %>%
  glimpse()
```

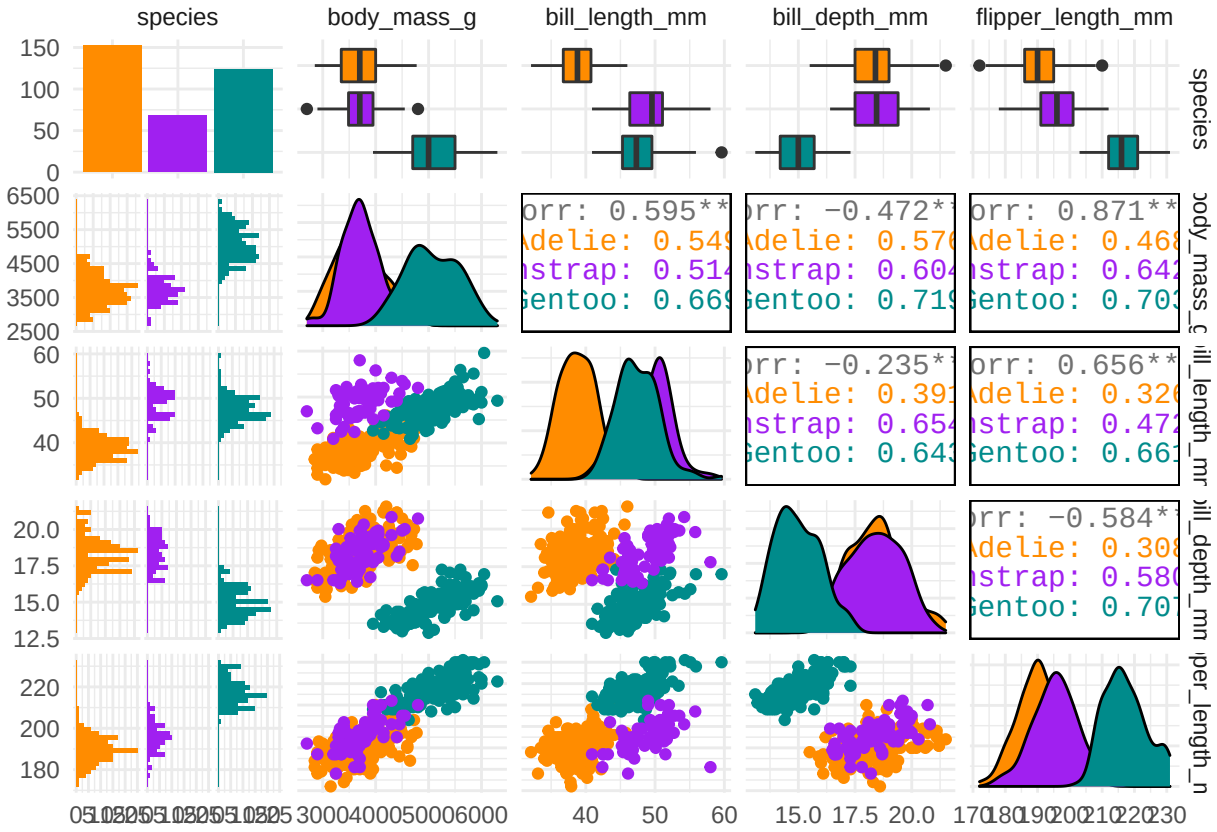
```
## Rows: 344
## Columns: 4
## $ body_mass_g       <int> 3750, 3800, 3250, NA, 3450, 3650, 3625, 4675, 347...
## $ bill_length_mm    <dbl> 39.1, 39.5, 40.3, NA, 36.7, 39.3, 38.9, 39.2, 34....
## $ bill_depth_mm     <dbl> 18.7, 17.4, 18.0, NA, 19.3, 20.6, 17.8, 19.6, 18....
## $ flipper_length_mm <int> 181, 186, 195, NA, 193, 190, 181, 195, 193, 190, ...
```

```
# Scatterplot example 1: penguin flipper length versus body mass
ggplot(data = penguins, aes(x = flipper_length_mm, y = body_mass_g)) +
  geom_point(aes(color = species,
                 shape = species),
             size = 2) +
  scale_color_manual(values = c("darkorange", "darkorchid", "cyan4"))
```



The above scatterplot of body mass shows a strong similarity between the species Adelie and Chinstrap with a clear difference from Gentoo. A valuable observation in regards to the binary logistic regression model.

```
ds %>%
  dplyr::select(species, body_mass_g, ends_with("_mm")) %>%
  GGally::ggpairs(aes(color = species)) +
  scale_color_manual(values = c("darkorange","purple","cyan4")) +
  scale_fill_manual(values = c("darkorange","purple","cyan4"))
```



The above plot shows the additional strong similarities between the Adelie and Chinstrap species as compared to the Gentoo species.

Model Definitions

Prep dataset for logistic regression. First step is to remove rows containing an NA.

```
# Create dataset for binary logistic regression: species Gentoo or Not
data_binary <- penguins

# Only use complete instances ... actually come back to this as I don't want to exclude because of sex
train_data_binary <- na.omit(data_binary)

dim(train_data_binary)
```

```
## [1] 333   8
```

Based on the result, 11 rows are removed, which would equal the number of NAs in variable *sex*.

Binary Logistic Regression

The following approach attempts to construct a logistic regression model based on a binary outcome. As the penguins dataset is based on a dependent variable (species) containing three values, a dummy variable

Gentoo is defined to identify penguins of the species Gentoo or of the other two values (Adelie and Chinstrap). Based on the exploratory data analysis indicating independent variable overlap for body mass, bill depth, and flipper length between the Adelie and Chinstrap species, the decision was made to group these two species based on the similarities.

```
# Create new column
train_data_binary$gentoo <- ifelse(train_data_binary$species=="Gentoo", 1, 0)

summary(train_data_binary)
```

```
##       species          island    bill_length_mm  bill_depth_mm
##  Adelie   :146   Biscoe   :163   Min.   :32.10   Min.   :13.10
##  Chinstrap: 68   Dream    :123   1st Qu.:39.50   1st Qu.:15.60
##  Gentoo   :119   Torgersen: 47   Median :44.50   Median :17.30
##                                  Mean   :43.99   Mean   :17.16
##                                  3rd Qu.:48.60   3rd Qu.:18.70
##                                  Max.   :59.60   Max.   :21.50
##  flipper_length_mm  body_mass_g       sex           year          gentoo
##  Min.   :172       Min.   :2700   female:165   Min.   :2007   Min.   :0.0000
##  1st Qu.:190       1st Qu.:3550   male  :168   1st Qu.:2007   1st Qu.:0.0000
##  Median :197       Median :4050                Median :2008   Median :0.0000
##  Mean   :201       Mean   :4207                Mean   :2008   Mean   :0.3574
##  3rd Qu.:213       3rd Qu.:4775                3rd Qu.:2009   3rd Qu.:1.0000
##  Max.   :231       Max.   :6300                Max.   :2009   Max.   :1.0000
```

With the derived dummy variable *Gentoo*, the variable *species* is removed from the initial dataset, so as not to impact the logistic regression models.

```
# Drop species column, as now just using gentoo column as Y variable

drops <- c("species")
train_data_binary <- train_data_binary[ , !(names(train_data_binary) %in% drops)]

summary(train_data_binary)
```

```
##        island    bill_length_mm  bill_depth_mm   flipper_length_mm
##  Biscoe   :163   Min.   :32.10   Min.   :13.10   Min.   :172
##  Dream    :123   1st Qu.:39.50   1st Qu.:15.60   1st Qu.:190
##  Torgersen: 47   Median :44.50   Median :17.30   Median :197
##                  Mean   :43.99   Mean   :17.16   Mean   :201
##                  3rd Qu.:48.60   3rd Qu.:18.70   3rd Qu.:213
##                  Max.   :59.60   Max.   :21.50   Max.   :231
##   body_mass_g       sex           year          gentoo
##  Min.   :2700   female:165   Min.   :2007   Min.   :0.0000
##  1st Qu.:3550   male  :168   1st Qu.:2007   1st Qu.:0.0000
##  Median :4050                Median :2008   Median :0.0000
##  Mean   :4207                Mean   :2008   Mean   :0.3574
##  3rd Qu.:4775                3rd Qu.:2009   3rd Qu.:1.0000
##  Max.   :6300                Max.   :2009   Max.   :1.0000
```

In order to validate the models properly, the initial penguins dataset is partitioned into training data at 70% of the given dataset with the remaining 30% used as test data completely unseen by the model.

```r
set.seed(123)
trainIndex <-createDataPartition(train_data_binary$gentoo, p = 0.7, list = FALSE, times = 1)
train <- train_data_binary[trainIndex,]
test <- train_data_binary[-trainIndex,]
```

Three versions of a binary logistic regression model are constructed in order to evaluate the accuracy of each and also provide to narrow the model to the least number of variables to identify the most parsimonious model.

Baseline Model

The first model uses all the available independent variables in order to define a baseline evaluation of the model.

```r
# All variables
model1 <- glm(gentoo ~ ., data = train, family = "binomial"(link="logit"))
#Accuracy 100%, AIC is 18
summary(model1)
```

```
##
## Call:
## glm(formula = gentoo ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -3.722e-05  -2.100e-08  -2.100e-08   2.100e-08   2.985e-05
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)        7.439e+03  1.581e+08       0        1
## islandDream       -1.043e+01  1.197e+05       0        1
## islandTorgersen   -1.180e+01  1.124e+05       0        1
## bill_length_mm     7.064e-01  1.136e+04       0        1
## bill_depth_mm     -9.278e+00  3.578e+04       0        1
## flipper_length_mm  9.491e-01  6.324e+03       0        1
## body_mass_g        1.516e-02  1.527e+02       0        1
## sexmale            1.990e+00  1.590e+05       0        1
## year              -3.773e+00  7.865e+04       0        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.0884e+02  on 233  degrees of freedom
## Residual deviance: 4.3248e-09  on 225  degrees of freedom
## AIC: 18
##
## Number of Fisher Scoring iterations: 25
```

Resulting AIC: 18.

Stepwise Model

Next, the *stepAIC* function is applied to the full model to determine the most predictive variables for the model.

```
# All variables then applied with stepAIC
model2 <- glm(gentoo ~ ., data = train, family = "binomial"(link="logit")) %>% stepAIC(trace=F, directi
# Accuracy 100% an AIC is 6
summary(model2)
```

```
##
## Call:
## glm(formula = gentoo ~ bill_depth_mm + flipper_length_mm, family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -6.826e-05  -2.100e-08  -2.100e-08   2.100e-08   6.510e-05
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -428.130 474946.766  -0.001    0.999
## bill_depth_mm        -14.834  12021.243  -0.001    0.999
## flipper_length_mm      3.274   1957.819   0.002    0.999
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.0884e+02  on 233  degrees of freedom
## Residual deviance: 9.5967e-09  on 231  degrees of freedom
## AIC: 6
##
## Number of Fisher Scoring iterations: 25
```

Resulting AIC: 6.

Hand Selected Model

Finally, a hand-selected list of independent variables are chosen based on the evaluation of the exploratory data analysis.

```
# Hand selected variables
model3 <- glm(gentoo ~ island + bill_depth_mm + flipper_length_mm + body_mass_g, data = train, family =
# Accuracy 100%, AIC is 12
summary(model3)
```

```
##
## Call:
## glm(formula = gentoo ~ island + bill_depth_mm + flipper_length_mm +
##     body_mass_g, family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
```

```
## -4.069e-05  -2.100e-08  -2.100e-08   2.100e-08   2.804e-05
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -1.402e+02  6.440e+05   0.000    1.000
## islandDream       -1.390e+00  4.055e+04   0.000    1.000
## islandTorgersen   -5.044e+00  7.407e+04   0.000    1.000
## bill_depth_mm     -1.049e+01  1.142e+04  -0.001    0.999
## flipper_length_mm  1.098e+00  3.818e+03   0.000    1.000
## body_mass_g        1.958e-02  5.657e+01   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.0884e+02  on 233  degrees of freedom
## Residual deviance: 4.7527e-09  on 228  degrees of freedom
## AIC: 12
##
## Number of Fisher Scoring iterations: 25
```

Resulting AIC: 12.

Make predictions

Predictions are performed on the test dataset based on the three above binary logistic regression models.

```
## use the test data set to make predictions for the 3 models
mod1.predict.probs <- predict.glm(model1, type="response", newdata=test)
mod1.predict.manual <- ifelse(mod1.predict.probs > 0.5, '1','0')
attach(test)

mod2.predict.probs <- predict.glm(model2, type="response", newdata=test)
mod2.predict.manual <- ifelse(mod2.predict.probs > 0.5, '1','0')
attach(test)

mod3.predict.probs <- predict.glm(model3, type="response", newdata=test)
mod3.predict.manual <- ifelse(mod3.predict.probs > 0.5, '1','0')
attach(test)
```

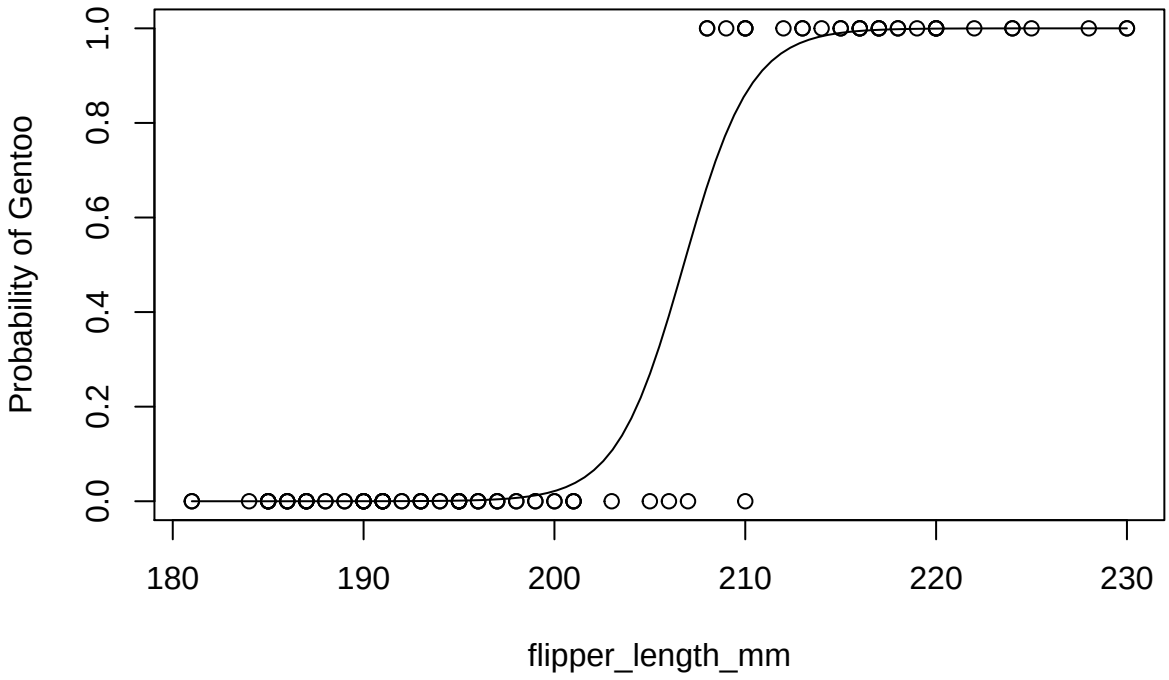
Model Visualizations

Plots of the data to visualize the independent variable's value compared to the logit function of the dependent variable.

First, plot the variable *flipper_length_mm* against the logit value of the *Gentoo* result.

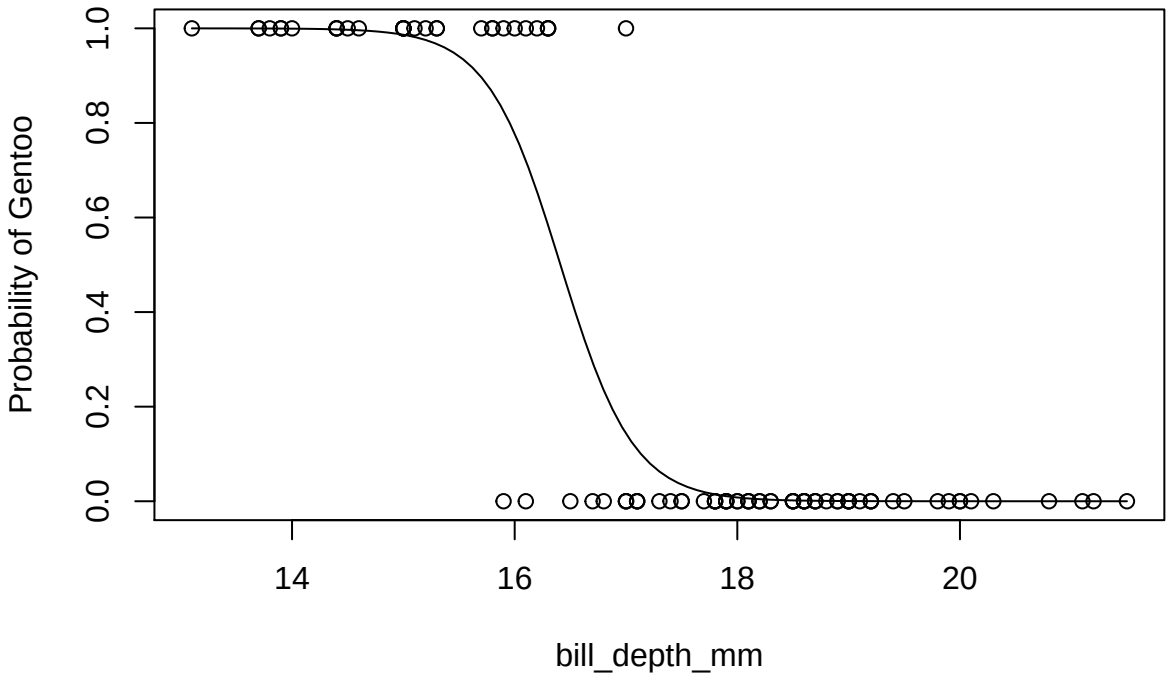
```
# Plot the dependent variable interpretation
# https://sites.google.com/site/daishizuka/toolkits/plotting-logistic-regression-in-r

# plot with flipper_length_mm on x-axis and Gentoo species (0 or 1) on y-axis
plot(flipper_length_mm,gentoo,xlab="flipper_length_mm",ylab="Probability of Gentoo")
g=glm(gentoo ~ flipper_length_mm, data = train, family = "binomial"(link="logit"))
curve(predict(g,data.frame(flipper_length_mm=x),type="resp"),add=TRUE)
```

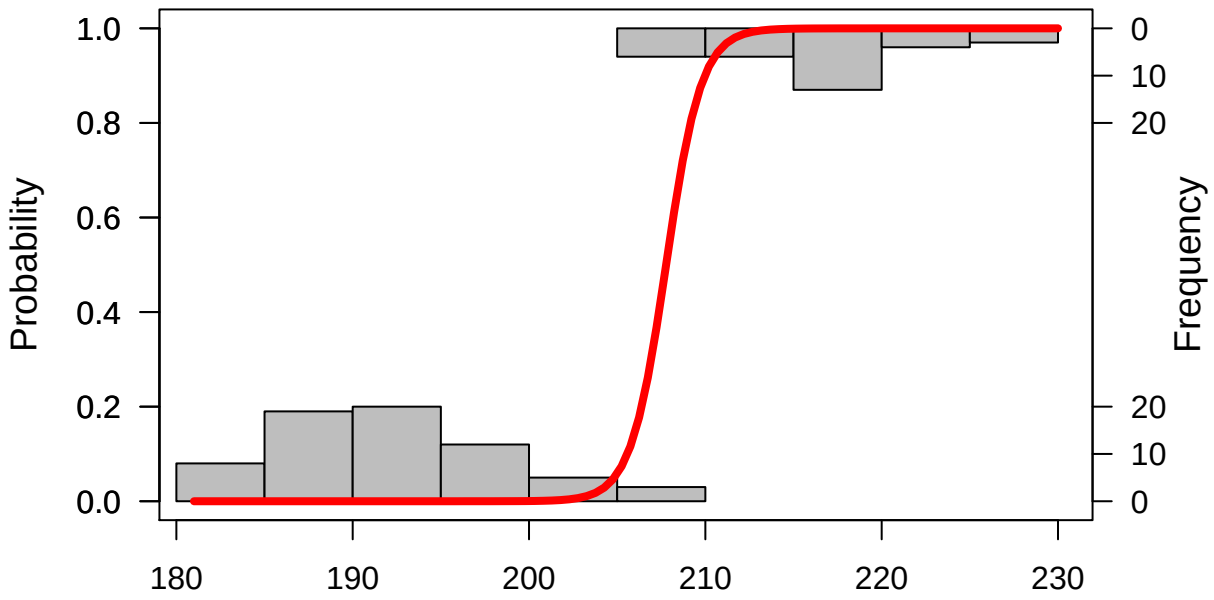


Second, plot the variable *bill_depth_mm* against the logit value of the *Gentoo* result.

```
# plot with bill_depth_mm on x-axis and Gentoo species (0 or 1) on y-axis
plot(bill_depth_mm,gentoo,xlab="bill_depth_mm",ylab="Probability of Gentoo")
g=glm(gentoo ~ bill_depth_mm, data = train, family = "binomial"(link="logit"))
curve(predict(g,data.frame(bill_depth_mm=x),type="resp"),add=TRUE)
```




```
# plot using another function
logi.hist.plot(flipper_length_mm,gentoo,boxp=FALSE,type="hist",col="gray")
```



Third plot above was just an attempt to use another library function for plotting the data against the logit function.

Model 1 Results

The baseline model shows:

- Accuracy: 1 or 100%
- Area Under the Curve: 1.0
- True Positive Rate (Sensitivity): 1.0
- True Negative Rate (Specificity: 1.0
- False Negative Rate (Miss Rate: 1-TPR): 0
- False Positive Rate (Fall-out: 1-TNR): 0

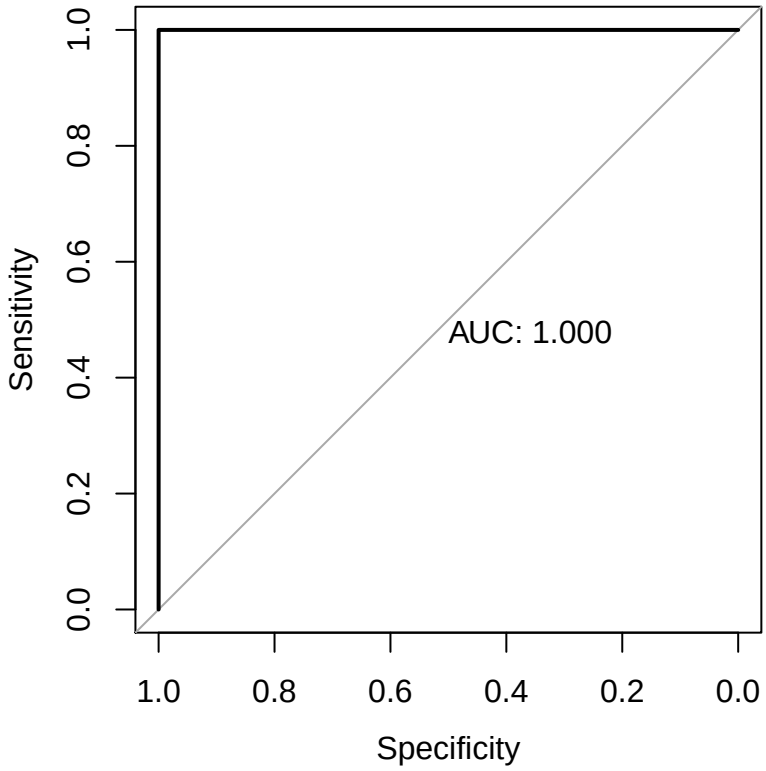
```
# Model1
# now can use the caret function
cm.var <- caret::confusionMatrix(factor(mod1.predict.manual), factor(test$gentoo), positive='1')
cm.var$table
```

```
##            Reference
## Prediction  0  1
##          0 67  0
##          1  0 32
```

```r
# print metrics
mod1.CMmetrics <- c(cm.var$overall[c(1)], cm.var$byClass[c(1,2,5,6,7)])
mod1.CMmetrics
```

```
##    Accuracy Sensitivity Specificity   Precision      Recall          F1
##           1           1           1           1           1           1
```

```r
# ROC and AUC
par(pty="s")
roc.stepwise <- roc(train$gentoo, model1$fitted.values, plot=TRUE, print.auc=TRUE)
```



Model 2 Results

The *stepAIC* model shows:

- Accuracy: 1 or 100%
- Area Under the Curve: 1.0
- True Positive Rate (Sensitivity): 1.0
- True Negative Rate (Specificity: 1.0
- False Negative Rate (Miss Rate: 1-TPR): 0
- False Positive Rate (Fall-out: 1-TNR): 0

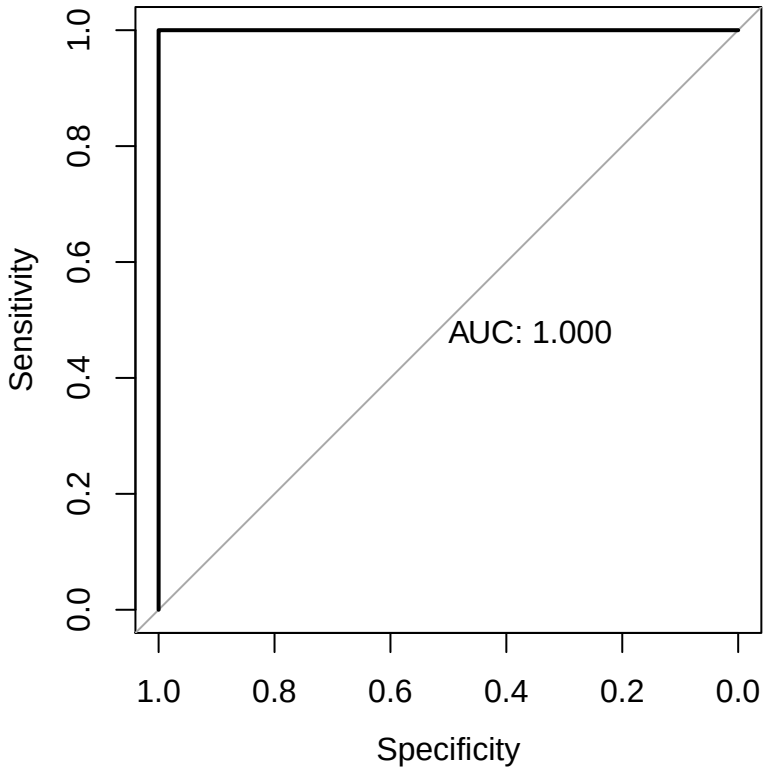
```r
# Model2
# now can use the caret function
cm.var <- caret::confusionMatrix(factor(mod2.predict.manual), factor(test$gentoo), positive='1')
cm.var$table
```

```
##            Reference
## Prediction  0  1
##          0 67  0
##          1  0 32
```

```r
# print metrics
mod2.CMmetrics <- c(cm.var$overall[c(1)], cm.var$byClass[c(1,2,5,6,7)])
mod2.CMmetrics
```

```
##    Accuracy Sensitivity Specificity   Precision      Recall          F1
##           1           1           1           1           1           1
```

```r
# ROC and AUC
par(pty="s")
roc.stepwise <- roc(train$gentoo, model2$fitted.values, plot=TRUE, print.auc=TRUE)
```



Model 3 Results

The hand-selected model shows:

- Accuracy: 1 or 100%
- Area Under the Curve: 1.0
- True Positive Rate (Sensitivity): 1.0
- True Negative Rate (Specificity: 1.0
- False Negative Rate (Miss Rate: 1-TPR): 0
- False Positive Rate (Fall-out: 1-TNR): 0

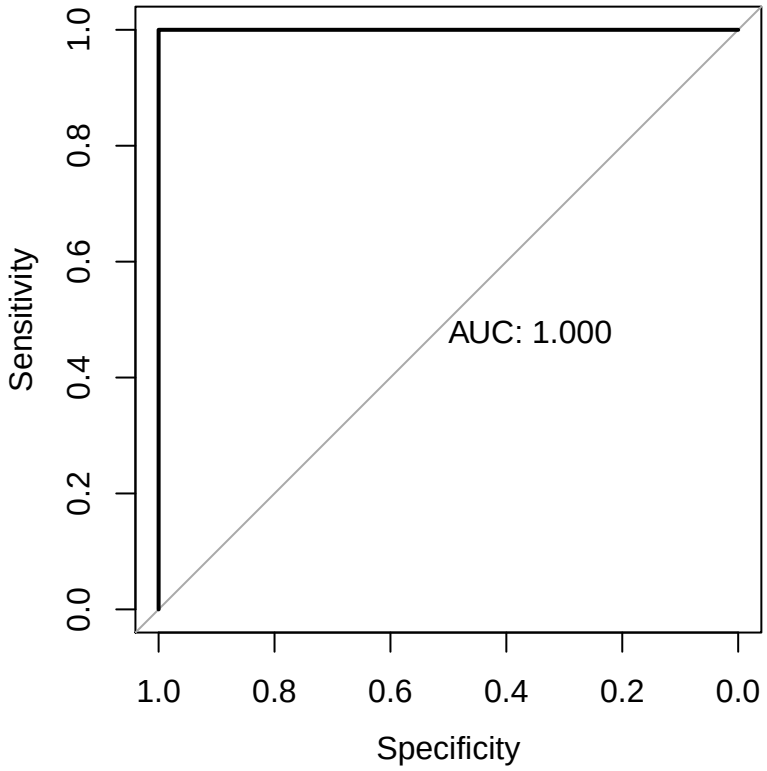
```
# Model3
# now can use the caret function
cm.var <- caret::confusionMatrix(factor(mod3.predict.manual), factor(test$gentoo), positive='1')
cm.var$table
```

```
##           Reference
## Prediction  0  1
##          0 67  0
##          1  0 32
```

```
# print metrics
mod3.CMmetrics <- c(cm.var$overall[c(1)], cm.var$byClass[c(1,2,5,6,7)])
mod3.CMmetrics
```

```
##    Accuracy Sensitivity Specificity   Precision      Recall          F1
##           1           1           1           1           1           1
```

```
# ROC and AUC
par(pty="s")
roc.stepwise <- roc(train$gentoo, model3$fitted.values, plot=TRUE, print.auc=TRUE)
```



Variable Interpretation

For Model 3 above, the variable interpretations are as follows.

A one-unit increase in the variable *island* for value *Dream* is associated with the decrease in the log odds of being in species Gentoo in the amount of 1.39.

A one-unit increase in the variable *island* for value *Torgersen* is associated with the decrease in the log odds of being in species Gentoo in the amount of 5.044.

A one-unit increase in the variable *bill_depth_mm* is associated with the decrease in the log odds of being in species Gentoo in the amount of 10.49.

A one-unit increase in the variable *flipper_length_mm* is associated with the increase in the log odds of being in species Gentoo in the amount of 1.098.

A one-unit increase in the variable *body_mass_g* is associated with the increase in the log odds of being in species Gentoo in the amount of .01958.

Alternate Binary Model

Looking at the summary results for the three models attempting to identify Gentoo species or not, I decided to make the goal a bit more difficult as Adelie species appears on all three islands, and statistically, the Adelie species does overlap with Chinstrap, I decided to create logistic regression models to identify Adelie instead of Gentoo.

The first model uses the same baseline model approach as above including all independent variables. The second model re-uses the *stepAIC* approach to algorithmically select the best independent variables.

```r
# Create dataset for binary logistic regression: species Adelie or Not
data_binary <- penguins

# Only use complete instances ... actually come back to this as I don't want to exclude because of sex
train_data_binary <- na.omit(data_binary)

train_data_binary$adelie <- ifelse(train_data_binary$species=="Adelie", 1, 0)

summary(train_data_binary)
```

```
##       species          island    bill_length_mm  bill_depth_mm
##  Adelie   :146   Biscoe   :163   Min.   :32.10   Min.   :13.10
##  Chinstrap: 68   Dream    :123   1st Qu.:39.50   1st Qu.:15.60
##  Gentoo   :119   Torgersen: 47   Median :44.50   Median :17.30
##                                  Mean   :43.99   Mean   :17.16
##                                  3rd Qu.:48.60   3rd Qu.:18.70
##                                  Max.   :59.60   Max.   :21.50
##  flipper_length_mm  body_mass_g       sex           year          adelie
##  Min.   :172       Min.   :2700   female:165   Min.   :2007   Min.   :0.0000
##  1st Qu.:190       1st Qu.:3550   male  :168   1st Qu.:2007   1st Qu.:0.0000
##  Median :197       Median :4050                Median :2008   Median :0.0000
##  Mean   :201       Mean   :4207                Mean   :2008   Mean   :0.4384
##  3rd Qu.:213       3rd Qu.:4775                3rd Qu.:2009   3rd Qu.:1.0000
##  Max.   :231       Max.   :6300                Max.   :2009   Max.   :1.0000
```

```r
drops <- c("species")
train_data_binary <- train_data_binary[ , !(names(train_data_binary) %in% drops)]

set.seed(123)
trainIndex <-createDataPartition(train_data_binary$adelie, p = 0.7, list = FALSE, times = 1)
train <- train_data_binary[trainIndex,]
test <- train_data_binary[-trainIndex,]

# All variables
model1_ad <- glm(adelie ~ ., data = train, family = "binomial"(link="logit"))
summary(model1_ad)
```

```
##
## Call:
## glm(formula = adelie ~ ., family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -8.484e-05  -2.100e-08  -2.100e-08   2.100e-08   8.052e-05
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -1.625e+04  5.517e+07   0.000    1.000
## islandDream       -8.202e+00  1.179e+05   0.000    1.000
## islandTorgersen    1.094e+01  1.264e+05   0.000    1.000
## bill_length_mm    -2.301e+01  7.614e+03  -0.003    0.998
```

```
## bill_depth_mm      2.986e+01  1.415e+04   0.002    0.998
## flipper_length_mm -1.174e+00  4.091e+03   0.000    1.000
## body_mass_g        3.811e-02  6.454e+01   0.001    1.000
## sexmale            9.225e+00  4.531e+04   0.000    1.000
## year               8.359e+00  2.785e+04   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.2103e+02  on 233  degrees of freedom
## Residual deviance: 2.8522e-08  on 225  degrees of freedom
## AIC: 18
##
## Number of Fisher Scoring iterations: 25
```

```
# All variables then applied with stepAIC
model2_ad <- glm(adelie ~ ., data = train, family = "binomial"(link="logit")) %>% stepAIC(trace=F, dire
summary(model2_ad)
```

```
##
## Call:
## glm(formula = adelie ~ bill_length_mm + bill_depth_mm + body_mass_g,
##     family = binomial(link = "logit"), data = train)
##
## Deviance Residuals:
##        Min          1Q      Median          3Q         Max
## -1.024e-04  -2.100e-08  -2.100e-08   2.100e-08   1.099e-04
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     3.328e+02  2.819e+05   0.001    0.999
## bill_length_mm -2.980e+01  7.288e+03  -0.004    0.997
## bill_depth_mm   4.525e+01  1.268e+04   0.004    0.997
## body_mass_g     3.582e-02  1.075e+01   0.003    0.997
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.2103e+02  on 233  degrees of freedom
## Residual deviance: 4.3658e-08  on 230  degrees of freedom
## AIC: 8
##
## Number of Fisher Scoring iterations: 25
```

```
## use the test data set to make predictions for the 3 models
mod1_ad.predict.probs <- predict.glm(model1_ad, type="response", newdata=test)
mod1_ad.predict.manual <- ifelse(mod1_ad.predict.probs > 0.5, '1','0')
attach(test)

mod2_ad.predict.probs <- predict.glm(model2_ad, type="response", newdata=test)
mod2_ad.predict.manual <- ifelse(mod2_ad.predict.probs > 0.5, '1','0')
attach(test)

# Model1
# now can use the caret function
```

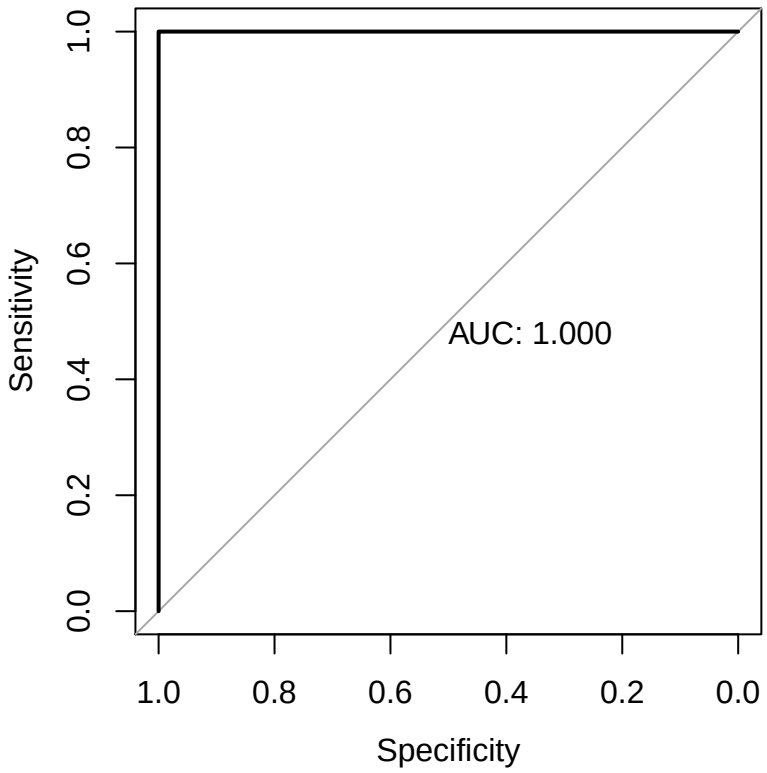
```
cm.var <- caret::confusionMatrix(factor(mod1_ad.predict.manual), factor(test$adelie), positive='1')
cm.var$table
```

```
##            Reference
## Prediction  0  1
##          0 56  2
##          1  0 41
```

```
# print metrics
mod1_ad.CMmetrics <- c(cm.var$overall[c(1)], cm.var$byClass[c(1,2,5,6,7)])
mod1_ad.CMmetrics
```

```
##    Accuracy Sensitivity Specificity   Precision      Recall          F1
##   0.9797980   0.9534884   1.0000000   1.0000000   0.9534884   0.9761905
```

```
# ROC and AUC
par(pty="s")
roc.stepwise <- roc(train$adelie, model1_ad$fitted.values, plot=TRUE, print.auc=TRUE)
```



The baseline model shows:

- Accuracy: 0.9797980 or ~98%
- Area Under the Curve: 1.0

- True Positive Rate (Sensitivity): 0.9534884
- True Negative Rate (Specificity: 1.0
- False Negative Rate (Miss Rate: 1-TPR): 0.0465116
- False Positive Rate (Fall-out: 1-TNR): 0

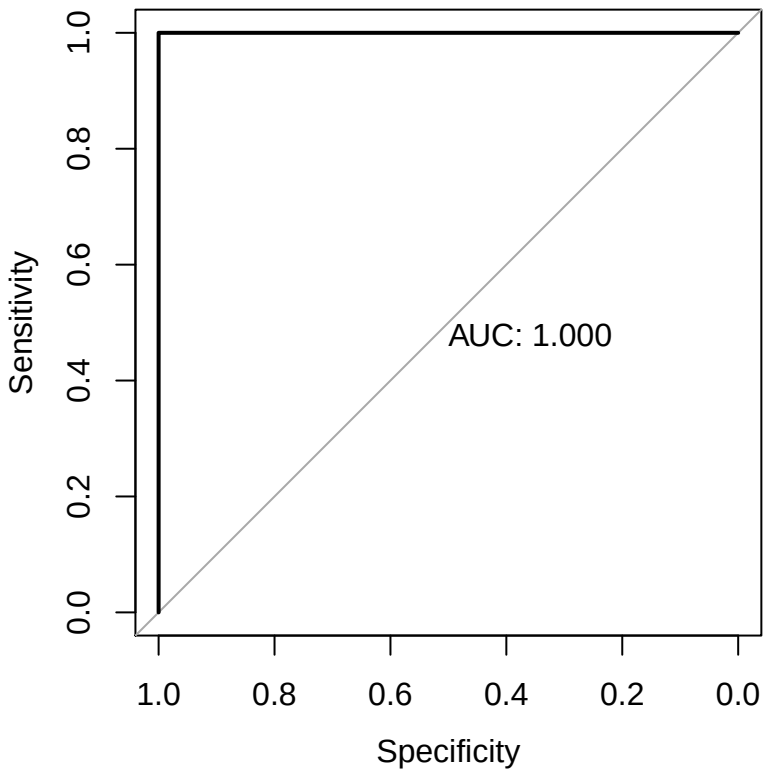
```
# Model2
# now can use the caret function
cm.var <- caret::confusionMatrix(factor(mod2_ad.predict.manual), factor(test$adelie), positive='1')
cm.var$table
```

```
##           Reference
## Prediction  0  1
##          0 56  2
##          1  0 41
```

```
# print metrics
mod2_ad.CMmetrics <- c(cm.var$overall[c(1)], cm.var$byClass[c(1,2,5,6,7)])
mod2_ad.CMmetrics
```

```
##    Accuracy Sensitivity Specificity   Precision      Recall          F1
##   0.9797980   0.9534884   1.0000000   1.0000000   0.9534884   0.9761905
```

```
# ROC and AUC
par(pty="s")
roc.stepwise <- roc(train$adelie, model2_ad$fitted.values, plot=TRUE, print.auc=TRUE)
```



The *stepAIC* model shows:

- Accuracy: 0.9797980 or ~98%
- Area Under the Curve: 1.0
- True Positive Rate (Sensitivity): 0.9534884
- True Negative Rate (Specificity: 1.0
- False Negative Rate (Miss Rate: 1-TPR): 0.0465116
- False Positive Rate (Fall-out: 1-TNR): 0

Interestingly, the *stepAIC* did not perform at 100% accuracy. The *stepAIC* model indicates *bill_length_mm*, *bill_depth_mm*, and *body_mass_g* are the three most predictive independent variables.

Multinomial Logistic Regression

The following approach attempts to construct a multinomial logistic regression model based on a multivariate outcome. As the penguins dataset is based on a dependent variable (species) containing three values, these models attempt to predict the species of each penguin subject.

Model 1

The baseline model is attempted for the multinomial logistic regression to predict the penguin species.

```
# using https://www.r-bloggers.com/2020/05/multinomial-logistic-regression-with-r/

mlr_data <- penguins

mlr_data <- na.omit(mlr_data)

index <- createDataPartition(mlr_data$species, p = .70, list = FALSE)
train <- mlr_data[index,]
test <- mlr_data[-index,]

# Set the reference
train$species <- relevel(train$species, ref = "Adelie")

# Training the multinomial model
multinom_model1 <- multinom(species ~ ., data = mlr_data)
```

```
## # weights:  30 (18 variable)
## initial  value 365.837892
## iter  10 value 38.413009
## iter  20 value 1.753187
## iter  30 value 0.032963
## final  value 0.000004
## converged
```

```r
# Checking the model
summary(multinom_model1)
```

```
## Call:
## multinom(formula = species ~ ., data = mlr_data)
##
## Coefficients:
##           (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap  -0.2496435   211.62912        97.64471       32.96624     -60.84017
## Gentoo      0.7633414   -68.84639      -140.79025       28.45770     -42.86214
##           flipper_length_mm body_mass_g   sexmale       year
## Chinstrap        0.01515612  -0.0193027 -70.73544 -0.2242371
## Gentoo          -2.79321661   0.1330873 -67.23109 -0.2391631
##
## Std. Errors:
##           (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap  0.02157370  0.02157370    4.311999e-15      0.2205894     0.1689163
## Gentoo     0.02114133  0.02114131    5.401394e-53      1.0993488     0.4376254
##           flipper_length_mm body_mass_g    sexmale     year
## Chinstrap          4.937946    25.79546 0.02113835 42.20323
## Gentoo             4.439678   101.47836 0.02114132 42.45178
##
## Residual Deviance: 7.087421e-06
## AIC: 36.00001
```

The final negative log-likelihood value of 0.000004 is produced by running the model. This value multiplied by two is then seen in the model summary as the Residual Deviance.

```r
stargazer(multinom_model1, type="text", out="multinom_model1.htm")
```

```
##
## ================================================
##                         Dependent variable:
##                     ---------------------------
##                       Chinstrap       Gentoo
##                          (1)            (2)
## ------------------------------------------------
## islandDream           211.629***    -68.846***
##                        (0.022)        (0.021)
##
## islandTorgersen       97.645***     -140.790***
##                        (0.000)        (0.000)
##
## bill_length_mm        32.966***      28.458***
##                        (0.221)        (1.099)
##
## bill_depth_mm         -60.840***    -42.862***
##                        (0.169)        (0.438)
##
## flipper_length_mm       0.015         -2.793
##                        (4.938)        (4.440)
##
```

```
## body_mass_g           -0.019         0.133
##                      (25.795)      (101.478)
##
## sexmale             -70.735***    -67.231***
##                      (0.021)        (0.021)
##
## year                  -0.224         -0.239
##                      (42.203)      (42.452)
##
## Constant            -0.250***      0.763***
##                      (0.022)        (0.021)
##
## ------------------------------------------------
## Akaike Inf. Crit.     36.000         36.000
## ================================================
## Note:              *p<0.1; **p<0.05; ***p<0.01
```

The *stargazer* produces a clear table to view the coefficients of the independent variables in the model.

```
z <- summary(multinom_model1)$coefficients/summary(multinom_model1)$standard.errors
(z)
```

```
##           (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap   -11.57166    9809.589    2.264488e+16      149.44619    -360.17936
## Gentoo       36.10660   -3256.487   -2.606554e+54       25.88596     -97.94253
##           flipper_length_mm   body_mass_g   sexmale         year
## Chinstrap       0.003069316 -0.0007482985 -3346.308 -0.005313268
## Gentoo         -0.629148457  0.0013114851 -3180.080 -0.005633758
```

```
p <- (1 - pnorm(abs(z), 0, 1)) * 2
(p)
```

```
##           (Intercept) islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap           0           0               0              0             0
## Gentoo              0           0               0              0             0
##           flipper_length_mm body_mass_g sexmale      year
## Chinstrap         0.9975510   0.9994029       0 0.9957606
## Gentoo            0.5292519   0.9989536       0 0.9955049
```

The above results of the 2-tailed z test show variables *island*, *bill_length_mm*, and *bill_depth_mm* play no role in the prediction of the species for baseline model.

```
# Convert the coefficients to odds by taking the exponential of the coefficients.
exp(coef(multinom_model1))
```

```
##           (Intercept)  islandDream islandTorgersen bill_length_mm bill_depth_mm
## Chinstrap   0.7790785 8.116346e+91    2.550108e+42   2.075179e+14  3.779617e-27
## Gentoo      2.1454330 1.260059e-30    7.170851e-62   2.285713e+12  2.427777e-19
##           flipper_length_mm body_mass_g      sexmale      year
## Chinstrap        1.01527155   0.9808824 1.905408e-31 0.7991256
## Gentoo           0.06122396   1.1423498 6.337337e-30 0.7872865
```

```r
head(pp <- fitted(multinom_model1))
```

```
##   Adelie     Chinstrap        Gentoo
## 1      1 3.772453e-149 7.048130e-167
## 2      1  9.694696e-79 1.031731e-111
## 3      1  1.805382e-78 1.058514e-155
## 4      1 4.743238e-166 2.084333e-210
## 5      1 1.363740e-195 1.795385e-216
## 6      1  1.825593e-96 1.267031e-130
```

The above results indicate the species odds for each penguin observed.

```r
# Predicting and validating the model

# Predicting the values for train dataset
train$speciesPredicted <- predict(multinom_model1, newdata = train, "class")

# Building classification table
tab <- table(train$species, train$speciesPredicted)

# Calculating accuracy - sum of diagonal elements divided by total obs
round((sum(diag(tab))/sum(tab))*100,2)
```

```
## [1] 100
```

The accuracy of the predictions against the initial training dataset is 100%.

```r
# Predicting the class for test dataset
test$speciesPredicted <- predict(multinom_model1, newdata = test, "class")

# Building classification table
tab <- table(test$species, test$speciesPredicted)
tab
```

```
##
##             Adelie Chinstrap Gentoo
##   Adelie        43         0      0
##   Chinstrap      0        20      0
##   Gentoo         0         0     35
```

The output table for the predictions of the test dataset show 100% accuracy, also.

Model 2

Given the results of the baseline model and desire to great a parsimonious model, the below model uses independent variables selected based on value to the model.

```r
index <- createDataPartition(mlr_data$species, p = .70, list = FALSE)
train <- mlr_data[index,]
test <- mlr_data[-index,]
```

```r
# Set the reference
train$species <- relevel(train$species, ref = "Adelie")

# Training the multinomial model
multinom_model2 <- multinom(species ~ island + bill_depth_mm + bill_length_mm, data = mlr_data)
```

```
## # weights:  18 (10 variable)
## initial  value 365.837892
## iter  10 value 7.150707
## iter  20 value 3.076699
## iter  30 value 1.110433
## iter  40 value 0.923906
## iter  50 value 0.797524
## iter  60 value 0.572400
## iter  70 value 0.124402
## iter  80 value 0.096157
## iter  90 value 0.094787
## iter 100 value 0.073060
## final  value 0.073060
## stopped after 100 iterations
```

```r
# Checking the model
summary(multinom_model2)
```

```
## Call:
## multinom(formula = species ~ island + bill_depth_mm + bill_length_mm,
##     data = mlr_data)
##
## Coefficients:
##           (Intercept) islandDream islandTorgersen bill_depth_mm bill_length_mm
## Chinstrap  -122.91871    1.288634       -33.22371     -16.84742       9.933821
## Gentoo       -1.42714  -20.848914       -18.53702     -24.91635      10.469682
##
## Std. Errors:
##           (Intercept) islandDream islandTorgersen bill_depth_mm bill_length_mm
## Chinstrap    122.5958    32.82534      0.01924641      18.15363       9.740966
## Gentoo       593.4594    86.87926     67.87526633      59.83588      13.983616
##
## Residual Deviance: 0.1461206
## AIC: 20.14612
```

The final negative log-likelihood value of 0.073060 is produced by running the model. This value multiplied by two (0.1461206) is then seen in the model summary as the Residual Deviance.

Variable Interpretation

- A one-unit increase in the variable bill_depth_mm is associated with the decrease in the log odds of being a Chinstrap species vs. Adelie species in the amount of 16.84742.
- A one-unit increase in the variable bill_depth_mm is associated with the decrease in the log odds of being a Gentoo species vs. Adelie species in the amount of 24.91635.

- A one-unit increase in the variable bill_length_mm is associated with the increase in the log odds of being a Chinstrap species vs. Adelie species in the amount of 9.933821.
- A one-unit increase in the variable bill_length_mm is associated with the increase in the log odds of being a Gentoo species vs. Adelie species in the amount of 10.469682.
- The log odds of being a Chinstrap species vs. Adelie species will increase by 1.288634 if moving from island Biscoe to island Dream.
- The log odds of being a Chinstrap species vs. Adelie species will decrease by -33.22371 if moving from island Biscoe to island Torgersen.
- The log odds of being a Gentoo species vs. Adelie species will decrease by 20.848914 if moving from island Biscoe to island Dream.
- The log odds of being a Gentoo species vs. Adelie species will decrease by 18.53702 if moving from island Biscoe to island Torgersen.

```
z <- summary(multinom_model2)$coefficients/summary(multinom_model2)$standard.errors
(z)
```

```
##            (Intercept) islandDream islandTorgersen bill_depth_mm bill_length_mm
## Chinstrap -1.002633881  0.03925728   -1726.2285547    -0.9280471      1.0197983
## Gentoo    -0.002404782 -0.23997572      -0.2731042    -0.4164116      0.7487106
```

```
# 2-tailed z test
p <- (1 - pnorm(abs(z), 0, 1)) * 2
(p)
```

```
##           (Intercept) islandDream islandTorgersen bill_depth_mm bill_length_mm
## Chinstrap   0.3160375   0.9686853       0.0000000     0.3533831      0.3078241
## Gentoo      0.9980813   0.8103491       0.7847731     0.6771088      0.4540316
```

The above results of the 2-tailed z test show each selected independent variable plays a factor in the prediction of the species for the second model.

```
# Convert the coefficients to odds by taking the exponential of the coefficients.
exp(coef(multinom_model2))
```

```
##            (Intercept)  islandDream islandTorgersen bill_depth_mm
## Chinstrap 4.140788e-54 3.627826e+00    3.725015e-15  4.822325e-08
## Gentoo    2.399942e-01 8.819254e-10    8.901715e-09  1.509961e-11
##           bill_length_mm
## Chinstrap       20615.96
## Gentoo          35231.01
```

```
head(pp <- fitted(multinom_model2))
```

```
##   Adelie    Chinstrap       Gentoo
## 1      1 1.123216e-36 5.769037e-34
## 2      1 1.940490e-25 4.438315e-18
## 3      1 2.234745e-26 6.196373e-21
## 4      1 2.024551e-51 2.269146e-51
## 5      1 1.026813e-49 1.289802e-53
## 6      1 5.925047e-31 3.896272e-25
```

Again, the first six rows are displayed to show the probabilities of the species for each penguin observed.

```
# Predicting and validating the model

# Predicting the values for train dataset
train$speciesPredicted <- predict(multinom_model2, newdata = train, "class")

# Building classification table
tab <- table(train$species, train$speciesPredicted)

# Calculating accuracy - sum of diagonal elements divided by total obs
round((sum(diag(tab))/sum(tab))*100,2)
```

```
## [1] 100
```

As in the baseline model, the accuracy of the predictions against the initial training dataset is 100%.

```
# Predicting the class for test dataset
test$speciesPredicted <- predict(multinom_model2, newdata = test, "class")

# Building classification table
tab <- table(test$species, test$speciesPredicted)
tab
```

```
##
##             Adelie Chinstrap Gentoo
##   Adelie        43         0      0
##   Chinstrap      0        20      0
##   Gentoo         0         0     35
```

And fortunately, the output table for the predictions of the test dataset show 100% accuracy, also. As this model relies on just three independent variables (*island*, *bill_depth_mm*, and *bill_length_mm*), the hand-selected model proves to be the best parsimonious model.

Attempt at Extra Credit

After some Internet searching, it appears there isn't much direction in how to measure model fit for multinomial logistic regression models. Pearson residual and overdispersion are ways to measure the model. Approaches to comparing two or more models would include likelihood ratio test, Wald test, cross validation, and parallel lines assumption.

Conclusion

Overall, the simplicity of the palmer penguins dataset allowed for the creation of highly accurate binary and multinomial logistic regression models. Even with the purposely more difficult binary prediction, the model still performed well. The variable interpretation provided gives a clear picture of the weights by each selected variable impacts the model. Given the lack of clear fit evaluation for the multinomial model, I do think further analysis is warranted to ensure the model fit is reasonable.