Lab 07 - Classification

Team gitData - Blossom Mojekwu, Avanti Shah, Nathan Kim

due Thursday, March 19, 11:59p

Data and packages

```
library(tidyverse)
library(class)
library(broom)
library(usethis)

wine <- read_csv("data/wine.csv")
```

Setting random seed

```
# DO NOT MODIFY!

set.seed(03052020)
indices <- sample(nrow(wine), 25) # 25 observations to test on
```

Exercises

Exercise 1

```
wine_test <- wine %>%
  slice(indices) %>%
  select(type, alcohol, malic, ash, phenols, color)

wine_train <- wine %>%
  slice(-indices) %>%
  select(type, alcohol, malic, ash, phenols, color)

train_type <- wine %>%
  slice(-indices) %>%
  pull(type)
```

Exercise 2

```
true_type <- wine_test %>%
  pull(type)
```

Exercise 3

```
wine_train_ex_3 <- wine_train %>%
  select(alcohol, malic, ash, phenols, color)

wine_test_ex_3 <- wine_test %>%
  select(alcohol, malic, ash, phenols, color)

ex_3_knn <- knn(wine_train_ex_3, wine_test_ex_3, train_type, k = 10, prob = F,
                use.all = T)

ex_3_knn == true_type
```

```
##  [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
## [13]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
## [25]  TRUE
```

```
mean(ex_3_knn == true_type)
```

```
## [1] 0.92
```

Answer: From this output, we can see that the fitted k-nearest neighbours model predicts all but 2 test wine types correctly (indicated by two "FALSE" values in the vector) using the training data. The mean accuracy is 92%, which means that 92% of the test data was correctly classified by the `knn` model.

Exercise 4

Repeat Exercise 1, but with k varying from 1 to 12. Which value of k results in the greatest prediction accuracy in our test dataset, and what is the associated prediction accuracy? In doing this exercise, do not "repeat" the same code twelve times.

```
wine_train_ex_4 <- wine_train %>%
  select(alcohol, malic, ash, phenols, color)

wine_test_ex_4 <- wine_test %>%
  select(alcohol, malic, ash, phenols, color)

result <- numeric(12)

for (i in 1:12) {
  ex_4_knn <- knn(wine_train_ex_4, wine_test_ex_4, train_type, k = i, prob = F,
                  use.all = FALSE)
  result[i] <- mean(ex_4_knn == true_type)
}
result
```

```
##  [1] 0.92 0.92 0.92 0.88 0.92 0.96 0.92 0.92 0.92 0.92 0.92 0.92
```

```r
which.max(result)
```

```
## [1] 6
```

Answer: The k = 6 results in the greatest prediction accuracy. k = 6 has a prediction accuracy of 96%.

Exercise 5

Create new variables in the original wine dataset that consist of standardized values for the predictors of interest. To create a standardized value, subtract each observation from the sample mean of the variable, and then divide this by the sample standard deviation of the variable. This gives us an estimate of how many standard deviations away from the mean each observation is. Overwrite the wine dataset with these additional variables so they can be used for later analyses.

```r
wine <- wine %>%
  mutate(alcohol_std = (mean(alcohol) - alcohol) / sd(alcohol)) %>%
  mutate(malic_std = (mean(malic) - malic) / sd(malic)) %>%
  mutate(ash_std = (mean(ash) - ash) / sd(ash)) %>%
  mutate(phenols_std = (mean(phenols) - phenols) / sd(phenols)) %>%
  mutate(color_std = (mean(color) - color) / sd(color))
```

Exercise 6- CHECK

Create wine_train_std and wine_test_std as new training and testing datasets that contain only these new standardized predictors. Fit the k-NN model on the training dataset, with k varying from 1 to 12. Which value of k results in the greatest prediction accuracy in our new testing dataset, and what was the prediction accuracy? Comment on your findings.

```r
wine_test_std <- wine %>%
  slice(indices) %>%
  select(alcohol_std, malic_std, ash_std, phenols_std, color_std)

wine_train_std <- wine %>%
  slice(-indices) %>%
  select(alcohol_std, malic_std, ash_std, phenols_std, color_std)

std_result <- numeric(12)

for (i in 1:12) {
  std_knn <- knn(wine_train_std, wine_test_std, train_type, k = i, prob = F,
                 use.all = F)
  std_result[i] <- mean(std_knn == true_type)
}
std_result
```

```
##  [1] 0.88 0.88 0.92 0.88 0.88 0.92 0.92 0.96 0.96 0.96 1.00 1.00
```

```r
which.max(std_result)
```

```
## [1] 11
```

Answer: k = 11 results in the greatest prediction accuracy. k = 11 has a prediction accuracy of 100%. When using the standardized values, we can see more values closer to 100% accuracy, which means that the k-nearest neighbors model can fit the training dataset with perfect accuracy more often than it can using

the original dataset with unstandardized values. Moreover, we can see that these prediction values are more spread apart (have more variation) than using the original dataset, which is often 0.92.

Exercise 7

The R function glm() requires for logistic regression that the response variable takes on values of 0 or 1. Create a new variable in the training dataset named bin_type that is 0 if the wine was type A, and 1 if the wine was type B.

```
wine_train <- wine_train %>%
  mutate(bin_type = ifelse(type == "A", 0, 1))
```

Exercise 8

Create a logistic regression model using the training for the estimated probability of being type B wine, based on the original (unstandardized) predictors as found in wine_train. Interpret each coefficient.

```
logm_1 <- glm(bin_type ~ alcohol + malic + ash + phenols + color,
              data = wine_train, family = "binomial")

logm_1 %>%
  tidy()
```

```
## # A tibble: 6 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)   72.5      20.5       3.54 0.000399
## 2 alcohol       -4.30      1.26     -3.41 0.000642
## 3 malic         -0.609     0.596    -1.02 0.307
## 4 ash           -1.99      1.51     -1.32 0.188
## 5 phenols       -1.53      1.23     -1.24 0.214
## 6 color         -1.58      0.707    -2.23 0.0259
```

Holding all other variables constant, for each unit increase in alcohol percentage, we would expect the log-odds of wine being Type B to decrease, on average, by approximately 4.30.

Holding all other variables constant, for each unit increase in malic, we would expect the log-odds of wine being Type B to decrease, on average, by approximately 0.61.

Holding all other variables constant, for each unit increase in ash, we would expect the log-odds of wine being Type B to decrease, on average, by approximately 1.99.

Holding all other variables constant, for each unit increase in phenols, we would expect the log-odds of wine being Type B to decrease, on average, by approximately 1.53.

Holding all other variables constant, for each unit increase in color, we would expect the log-odds of wine being Type B to decrease, on average, by approximately 1.58.

Exercise 9

Calculate the predicted probabilities of being type B wine for each one of the 25 individuals in the test dataset based on the logistic regression model, and then classify them, using 0.5 as the decision boundary. Display the predictions (type A vs. B). What is the prediction accuracy?

```
predictions <- augment(logm_1, newdata = wine_test) %>%
  mutate(trueprob = exp(.fitted) / (1 + exp(.fitted)),
         classification = case_when(
           trueprob > 0.5 ~ "B",
           trueprob <= 0.5 ~ "A"
         ))

wine_train %>%
  count(type)
```

```
## # A tibble: 2 x 2
##   type      n
##   <chr> <int>
## 1 A        51
## 2 B        54
```

```
wine_test %>%
  count(type)
```

```
## # A tibble: 2 x 2
##   type      n
##   <chr> <int>
## 1 A         8
## 2 B        17
```

```
pred_class <- predictions %>%
  pull(classification)

pred_class %>%
  tidy()
```

```
## Warning: 'tidy.character' is deprecated.
## See help("Deprecated")
```

```
## # A tibble: 25 x 1
##    x
##    <chr>
##  1 B
##  2 A
##  3 B
##  4 B
##  5 B
##  6 B
##  7 B
##  8 B
##  9 B
## 10 A
## # ... with 15 more rows
```

```
mean(pred_class == true_type)
```

```
## [1] 0.96
```

The prediction accuracy is 96%