Section 3 - More on Linear Regression: Exercises

Jacob Jameson

As Professor Saghafian noted on Slide 14 of lecture 6, there are certain skills you are expected to have about inference in general, particularly when it comes to linear regression models. The goal of today's section is to practice (some of) these skills. The session involves executing coding exercises and answering conceptual questions along the way. We will work with the ***Credit*** data set, which is a part of the ***ISLR*** package. I will give you time to work on each subsection and then I will share my proposed code and answers to the questions. You are encouraged to work in pairs.

Note that in some cases there are several ways to write the code to yield the same result.

1 Exploratory data analysis

1. Load the Credit data set from ISLR package. Check the codebook to understand the structure of the data set and the definition and unit of each variable.

```
# Store the data in a clean object and cast the data into a "data.table" object
# As noted earlier this package simplifies some of the data cleaning...
credit_data  <- as.data.table(Credit)
```

2. How many observations and variables does the data set include?

```
dim(credit_data)
```

```
## [1] 400  12
```

3. What are the categorical variables in the data set?

```
str(credit_data)
```

```
## Classes 'data.table' and 'data.frame':   400 obs. of  12 variables:
##  $ ID       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Income   : num  14.9 106 104.6 148.9 55.9 ...
##  $ Limit    : int  3606 6645 7075 9504 4897 8047 3388 7114 3300 6819 ...
##  $ Rating   : int  283 483 514 681 357 569 259 512 266 491 ...
##  $ Cards    : int  2 3 4 3 2 4 2 2 5 3 ...
##  $ Age      : int  34 82 71 36 68 77 37 87 66 41 ...
##  $ Education: int  11 15 11 11 16 10 12 9 13 19 ...
##  $ Gender   : Factor w/ 2 levels " Male","Female": 1 2 1 2 1 1 2 1 2 2 ...
##  $ Student  : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 1 1 2 ...
##  $ Married  : Factor w/ 2 levels "No","Yes": 2 2 1 1 2 1 1 1 1 2 ...
##  $ Ethnicity: Factor w/ 3 levels "African American",..: 3 2 2 2 3 3 1 2 3 1 ...
##  $ Balance  : int  333 903 580 964 331 1151 203 872 279 1350 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```r
# The categorical variables are: Gender, Student, Married, and Ethnicity.
```

4. Are they rows with missing values? If so, how many? Hint: checkout the `complete.cases` function.

```r
nrow(credit_data[!complete.cases(credit_data),])
```

```
## [1] 0
```

5. Generate summary statistics of all the variables. What is the mean and standard deviation of income?

```r
summary(credit_data)
```

```
##        ID             Income           Limit           Rating     
##  Min.   :  1.0   Min.   : 10.35   Min.   :  855   Min.   : 93.0  
##  1st Qu.:100.8   1st Qu.: 21.01   1st Qu.: 3088   1st Qu.:247.2  
##  Median :200.5   Median : 33.12   Median : 4622   Median :344.0  
##  Mean   :200.5   Mean   : 45.22   Mean   : 4736   Mean   :354.9  
##  3rd Qu.:300.2   3rd Qu.: 57.47   3rd Qu.: 5873   3rd Qu.:437.2  
##  Max.   :400.0   Max.   :186.63   Max.   :13913   Max.   :982.0  
##      Cards            Age          Education        Gender    Student  
##  Min.   :1.000   Min.   :23.00   Min.   : 5.00    Male :193   No :360  
##  1st Qu.:2.000   1st Qu.:41.75   1st Qu.:11.00   Female:207   Yes: 40  
##  Median :3.000   Median :56.00   Median :14.00                         
##  Mean   :2.958   Mean   :55.67   Mean   :13.45                         
##  3rd Qu.:4.000   3rd Qu.:70.00   3rd Qu.:16.00                         
##  Max.   :9.000   Max.   :98.00   Max.   :20.00                         
##  Married              Ethnicity      Balance       
##  No :155   African American: 99   Min.   :   0.00  
##  Yes:245   Asian           :102   1st Qu.:  68.75  
##            Caucasian       :199   Median : 459.50  
##                                   Mean   : 520.01  
##                                   3rd Qu.: 863.00  
##                                   Max.   :1999.00  
```

The mean income is:

```r
round(mean(credit_data$Income, na.rm = TRUE), 2)
```

```
## [1] 45.22
```

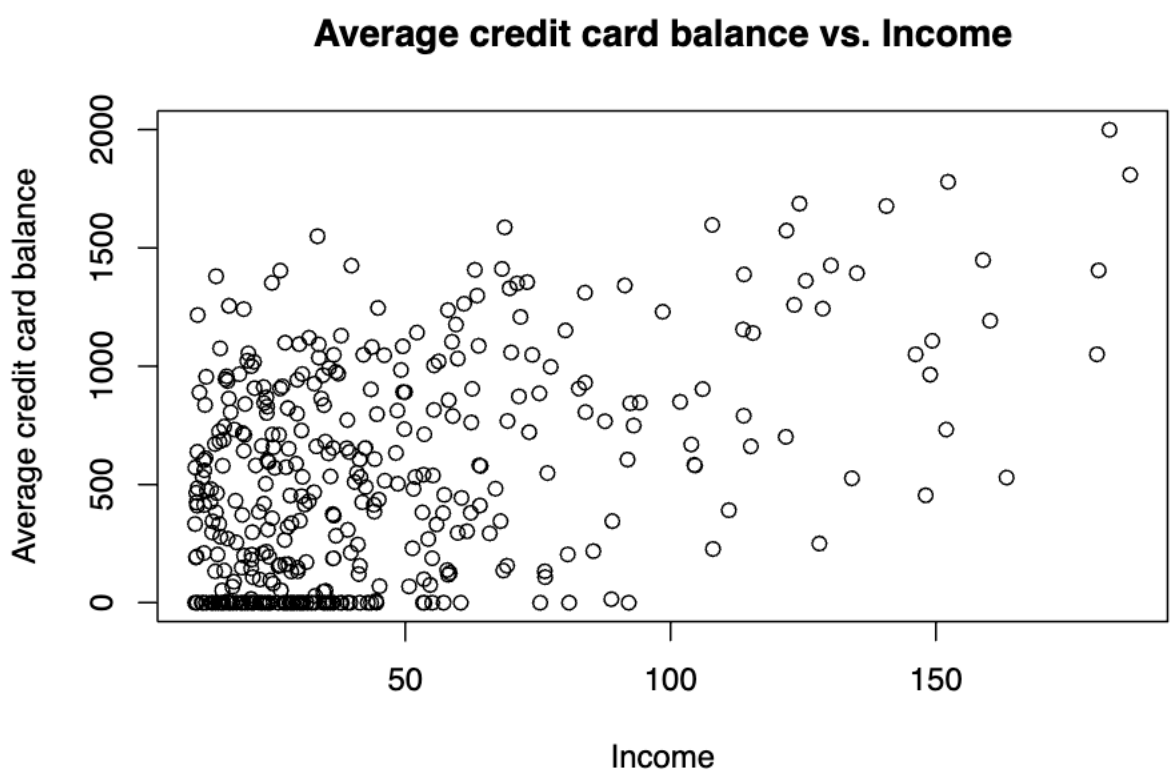
The standard deviation of income is:

```r
round(sd(credit_data$Income, na.rm = TRUE), 2)
```

```
## [1] 35.24
```

6. Plot the relationship between balance (y-axis) and income (x-axis). What do you notice about the relationship?

```r
plot(x = credit_data$Income, y = credit_data$Balance,
     main = "Average credit card balance vs. Income",
     xlab = "Income",
     ylab = "Average credit card balance")
```



2 Inference

1. Regress balance (y-variable) on income (x-variable). Interpret the income coefficient.

```r
mod1 <- lm(Balance ~ Income, credit_data)
summary(mod1)
```

```
##
## Call:
## lm(formula = Balance ~ Income, data = credit_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -803.64 -348.99  -54.42  331.75 1100.25
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 246.5148    33.1993   7.425  6.9e-13 ***
## Income        6.0484     0.5794  10.440  < 2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 407.9 on 398 degrees of freedom
## Multiple R-squared:  0.215,	Adjusted R-squared:  0.213
## F-statistic:   109 on 1 and 398 DF,  p-value: < 2.2e-16
```

2. Now add gender as an explanatory variable.

```r
mod2 <- lm(Balance ~ Income + Gender, credit_data)
summary(mod2)
```

```
##
## Call:
## lm(formula = Balance ~ Income + Gender, data = credit_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -791.23 -351.34  -51.57  328.18 1112.87
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  233.7663    39.5322   5.913 7.24e-09 ***
## Income         6.0521     0.5799  10.437  < 2e-16 ***
## GenderFemale  24.3108    40.8470   0.595    0.552
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 408.2 on 397 degrees of freedom
## Multiple R-squared:  0.2157,	Adjusted R-squared:  0.2117
## F-statistic: 54.58 on 2 and 397 DF,  p-value: < 2.2e-16
```

(a) Interpret all three coefficients (intercept, income coefficient, gender coefficient).

```r
# - The average balance for males is $233.77.

# - The average balance for females is $24.31 higher than the average
#   balance of males, when controlling for income. Note however, that this
#   difference is not statistically significant.

# - A $1,000 increase in income is associated with an average balance
#   increase of \$6.05. The coefficient is statistically significant.
```

(b) Test the null hypothesis that there is no relationship between balance and gender (i.e. $\beta_{gender} = 0$). What do you conclude about the test?

```r
# - H0: the difference in balance between females and males
#   (after controlling for income) is 0, that is $\beta_{gender} = 0$.

# - Ha: the difference in balance between females and males
#   (after controlling for income) is different from 0, that is $\beta_{gender} \neq 0$.
```

```r
# P-value suggests we cannot reject the NULL at any reasonable level of
# significance (1\%, 5\%, 10\%)
```

(c) What is the confidence interval of the gender coefficient? Interpret this coefficient. Hint: checkout the *confint* function.

```r
confint(mod2)
```

```
##                   2.5 %     97.5 %
## (Intercept)  156.04762 311.485030
## Income         4.91210   7.192038
## GenderFemale -55.99259 104.614265
```

```r
# We can be 95\% confident that the true difference in balance is between
# females and males is between -55.99 and 104.61. Notice this interval
# includes 0, which is consistent with our conclusion on the hypothesis test above.
```

(d) Find and interpret the R^2 of this regression.

```r
# The $R^2$ is 0.2157. This means that income and gender together
# explain ~22\% of the variation in average card balance.
```

3. Now add an interaction term between income and gender to the regression in part 2.

(a) Interpret the coefficient on the interaction term.

```r
mod3 <- lm(Balance ~ Income*Gender, credit_data)
summary(mod3)
```

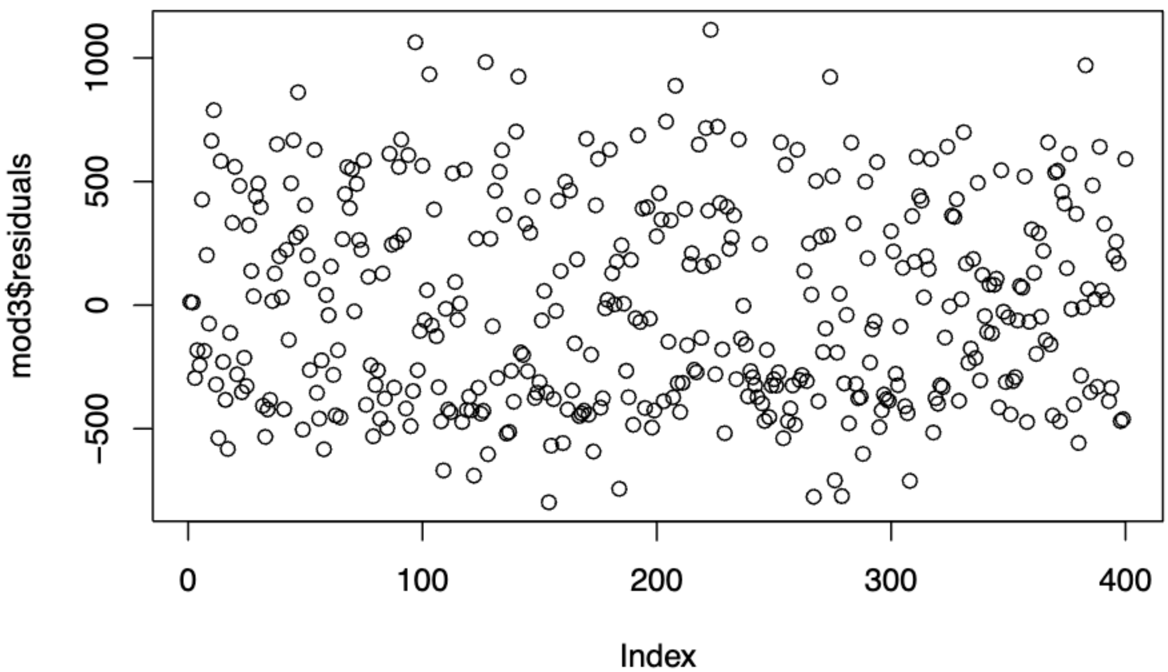
```
##
## Call:
## lm(formula = Balance ~ Income * Gender, data = credit_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -797.35 -352.35  -53.42  328.98 1114.47
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         227.7682    47.8567   4.759 2.73e-06 ***
## Income                6.1836     0.8276   7.472 5.12e-13 ***
## GenderFemale         36.0236    66.5744   0.541    0.589
## Income:GenderFemale  -0.2589     1.1612  -0.223    0.824
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 408.7 on 396 degrees of freedom
## Multiple R-squared:  0.2158, Adjusted R-squared:  0.2098
## F-statistic: 36.32 on 3 and 396 DF,  p-value: < 2.2e-16
```

(b) What is R^2 and the adjusted R^2 of this regression. What do these two values tell you about the usefulness of the interaction term?

```
# The $R^2$ is 0.2158 and the adjusted $R^2$ is 0.2098. In the model without the
# interaction term the $R^2$ was 0.2157, and the adjusted $R^2$ was 0.2117.
# The $R^2$ has increased as expected given we have a added a term.
# However, the adjusted adjusted $R^2$ has decreased suggesting the
# interaction term does not add value
# (when considering the complexity it adds to the model).
```

(c) Plot the residuals. What does the plot tell you about your model fit?

```
plot(mod3$residuals)
```




```
# There's no discernible pattern in the residuals, the model fit appears reasonable.
```

4. Rerun the model in part 3 using the log-transformed version of the balance and income variables. Interpret the coefficient on the income term.

```
mod4 <- lm(log(Balance + 0.0001) ~ log(Income)*Gender, credit_data)
summary(mod4)
```

```
##
```

```
## Call:
## lm(formula = log(Balance + 1e-04) ~ log(Income) * Gender, data = credit_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.5434  -0.1351   2.4202   4.1069   7.2967
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -6.2982     2.3019  -2.736 0.006498 **
## log(Income)                  2.4470     0.6340   3.859 0.000133 ***
## GenderFemale                -0.4506     3.2929  -0.137 0.891238
## log(Income):GenderFemale     0.3034     0.9073   0.334 0.738248
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.264 on 396 degrees of freedom
## Multiple R-squared:  0.0789, Adjusted R-squared:  0.07192
## F-statistic: 11.31 on 3 and 396 DF,  p-value: 3.939e-07
```

```
# A 1\% increase in income is associated with 2.45\% decreases
# average balance, when controlling for gender.
```

3 BONUS: Prediction Now let's revisit prediction models using linear regression and KNN.

1. Prepare the input datasets

(a) Drop the ID column, the categorical columns, and any rows with missing values.

```
# Remove rows with non-missing values
credit_data_complete <- credit_data[complete.cases(credit_data), ]

# Drop the district and municipality variables
credit_data_complete[, c("ID", "Gender", "Student",
                         "Married", "Ethnicity") := NULL] # "data.table" syntax
```

(b) Randomly split the data into a training set (75% of the observations) and a test set (the remaining 25% of the observations).

```
# Set a seed
set.seed(222)
# Extract the random test and training IDs
test_ids <- sample(seq(nrow(credit_data_complete)),
                   round(0.25 * nrow(credit_data_complete)))

training_ids <- which(!(seq(nrow(credit_data_complete)) %in% test_ids))

# Now use the IDs to get the two sets
test_data <- credit_data_complete[test_ids,]
training_data <- credit_data_complete[training_ids,]
```

2. When you use your training data to build a linear model that regresses account balance on all other features available in the data (plus an intercept), what is your test Mean Squared Error?

```
# The model
mod5 <- lm(Balance ~ ., training_data)

# Generate test predictions
predicted_bal <- predict(mod5, test_data[, -7])

## Let's see how well we did in terms of MSE
MSE_lm_bal <- mean((predicted_bal - test_data$Balance)^2)
print(MSE_lm_bal)
```

```
## [1] 33096.38
```

3. When you use your training data to build a KNN model that regresses account balance on all other features in the data, what is your test Mean Squared Error with $K = 1$?

```
# Library for KNN regression
library(FNN)

# The model
knn_reg1 <- knn.reg(training_data[, -c(7)],
                    test_data[, -c(7)],
                    training_data$Balance,
                    k = 1)

# The MSE
mse_knn1 <- mean((knn_reg1$pred - test_data$Balance)^2)
print(mse_knn1)
```

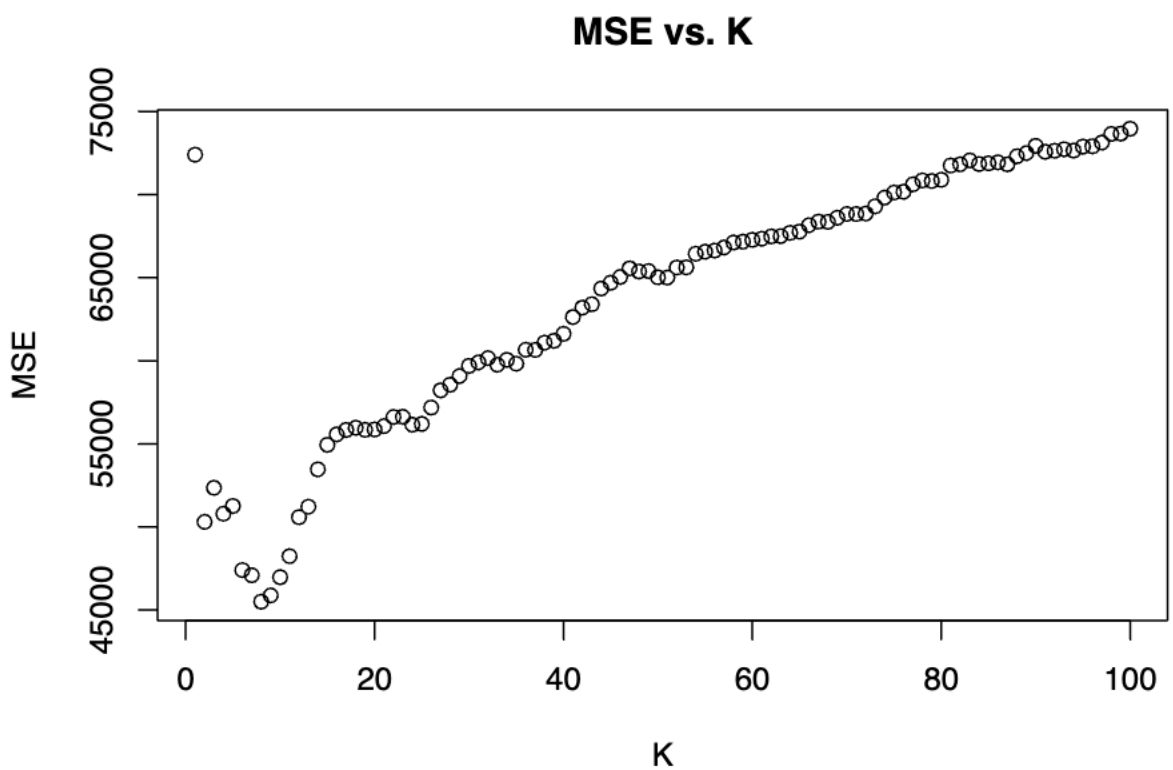
```
## [1] 72397.03
```

4. In last Friday's review session, one of your classmates asked: "Instead of testing a few individual values of K, could we use a more systematic approach that computes the Mean Squared Error for many values of K and then plot model performance as a function of K."

(a) In the first review session, we went through the basics of looping. Use a "for" loop to implement the approach your classmate suggested. Test K values going from 1 to 100.

```
# Define the range of K values to test
k_guesses <- 1:100

# Initialize a tracker for the MSE values for each K
mse_res <- NULL

# Now loop through all the values
for(i in 1:length(k_guesses)){
  # For each value, run the model using the current K guess
  knn_reg <- knn.reg(training_data[, -c(7)],
                     test_data[, -c(7)],
```

```r
                      training_data$Balance,
                      k = k_guesses[i]) # key line here

  # The MSE
  mse_knn <- mean((knn_reg$pred - test_data$Balance)^2)

  # Now update the tracker
  mse_res[i] <- mse_knn
}

# Now plot the results
plot(x = k_guesses, y = mse_res, main = "MSE vs. K", xlab = "K", ylab = "MSE")
```



(b) What can you conclude about the optimal K value for this model.

```r
# Find the K that gives the minimum MSE
which.min(mse_res)
```

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

```r
# It looks like $K = 8$ would give you the lowest MSE in this case.
# Note: this result may be different from yours depending on how your sampling played out.
```