CS 699 Assignment 6

Katherine Rein

```
# Import libraries
```

Problem 1

```
My work is in the written document. The IDs in each cluster are...
Cluster 1: 1,2,3,4,7
Cluster 2: 5,6
```

Problem 2

```
My work is in the written document. The maximum distance between the red and
the green clusters is 8. The centroid distance is 4.09.
```

Problem 3

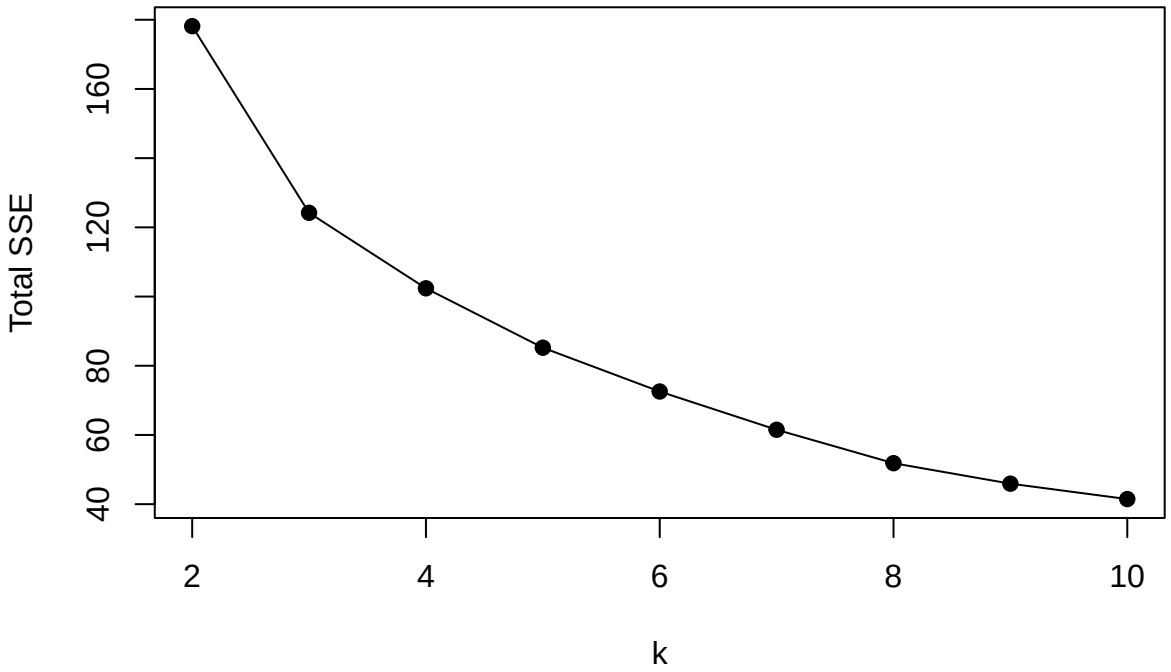
(1). Apply k-means clustering to the data. Which value of k do you recommend? (2). Which cluster or clusters seem to be associated with high Examination and high Education scores? Which Swiss provinces are associated with these clusters?

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(dplyr)

# Read in data
data(swiss)

# Standardize data
scaled = scale(swiss)

# Best K
set.seed(31)
tot_sse = c()
```

```
for(k in 2:10) {
  km = kmeans(scaled, centers=k, nstart=25)
  tot_sse[k] = km$tot.withinss}

# Visualize best k
plot(2:10, tot_sse[2:10], type='o', pch=19, xlab='k', ylab='Total SSE')
```




```
tot_tbl = tibble(k=2:10, SSE=tot_sse[2:10])
tot_tbl = tot_tbl %>%
  mutate(gain=lag(SSE)-SSE) %>%
  mutate(base_gain=gain[which(!is.na(gain))[1]]) %>%
  mutate(gain_pct=round(100*gain/base_gain,2)) %>%
  select(-base_gain)

knitr::kable(tot_tbl, caption='SSE, marginal gain, gain % of first')
```

Table 1: SSE, marginal gain, gain % of first

k	SSE	gain	gain_pct
2	178.14377	NA	NA
3	124.20515	53.938620	100.00
4	102.38664	21.818512	40.45
5	85.22272	17.163912	31.82
6	72.55739	12.665330	23.48
7	61.50080	11.056595	20.50
8	51.83916	9.661642	17.91
9	45.91634	5.922812	10.98
10	41.46795	4.448392	8.25

```r
# Run with best k
set.seed(31)
best_k = 3
k3 = kmeans(scaled, centers=best_k, nstart=25)
swiss$cluster = k3$cluster
cent = data.frame(k3$centers)
cent$cluster <- 1:nrow(cent)
print('Centroids:')
```

```
## [1] "Centroids:"
```

```r
print(cent)
```

```
##     Fertility Agriculture Examination  Education   Catholic Infant.Mortality
## 1 -1.40333639 -1.33786636   1.3958136  1.7312087 -0.3435471      -0.37080867
## 2  0.83314915  0.65426591  -0.8839264 -0.4527862  1.3189394       0.28579935
## 3 -0.09146501  0.01020332   0.1294049 -0.2871779 -0.7980284      -0.06984001
##   cluster
## 1       1
## 2       2
## 3       3
```

```r
swiss <- swiss %>%
  rownames_to_column(var = "canton")

target <- cent %>%
  arrange(desc(Examination), desc(Education)) %>%
  slice(1) %>%
  pull(cluster)

states_high <- swiss %>%
  filter(cluster == target) %>%
  pull(canton)

print('Highest Examination and Education:')
```

```
## [1] "Highest Examination and Education:"
```

```r
print(states_high)
```

```
## [1] "Lausanne"     "La Vallee"    "Vevey"        "La Chauxdfnd" "Neuchatel"
## [6] "V. De Geneve" "Rive Droite"  "Rive Gauche"
```

1) The biggest drop is from 2 to 3 to 4. From the table, we see that k = 4 only benefited 40% of what k = 3 benefited. Because of this we will likely choose k = 3.
2) The cluster containing Lausanne, La Vallee, Vevey, La Chauxdfnd, Neuchatel, V. De Geneve, Rive Droite, and Rive Gauche has the highest Examination and Education scores.

Problem 4

My work is in the written document. The clusters were created in the following order: bc, ef, efh, dg, abc, abcdg, abcdgefh.

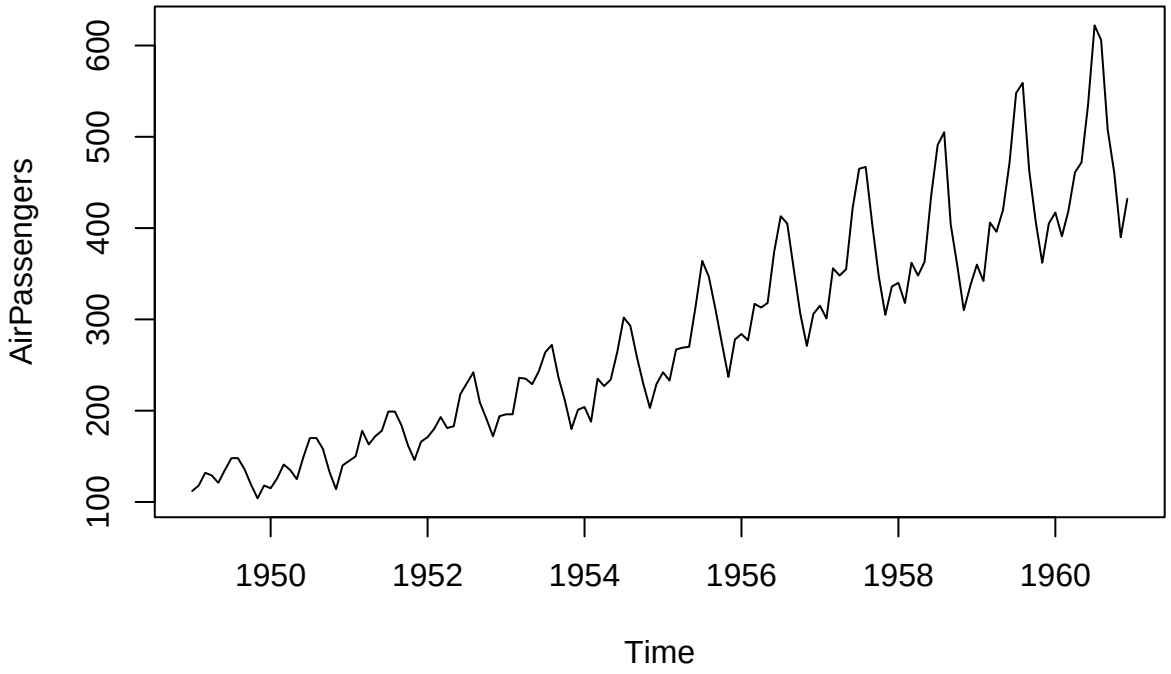
Problem 5

(1). Plot the time series. (2). Create either a seasonal subseries plot or a seasonal boxplot. Comment on the patterns you see. (3). Hold out the last year of data. Attempt at least two regression-based forecasts. Measure each model's performance using one or more appropriate metrics of your choice. (4). Report the best model. Explain the trend.

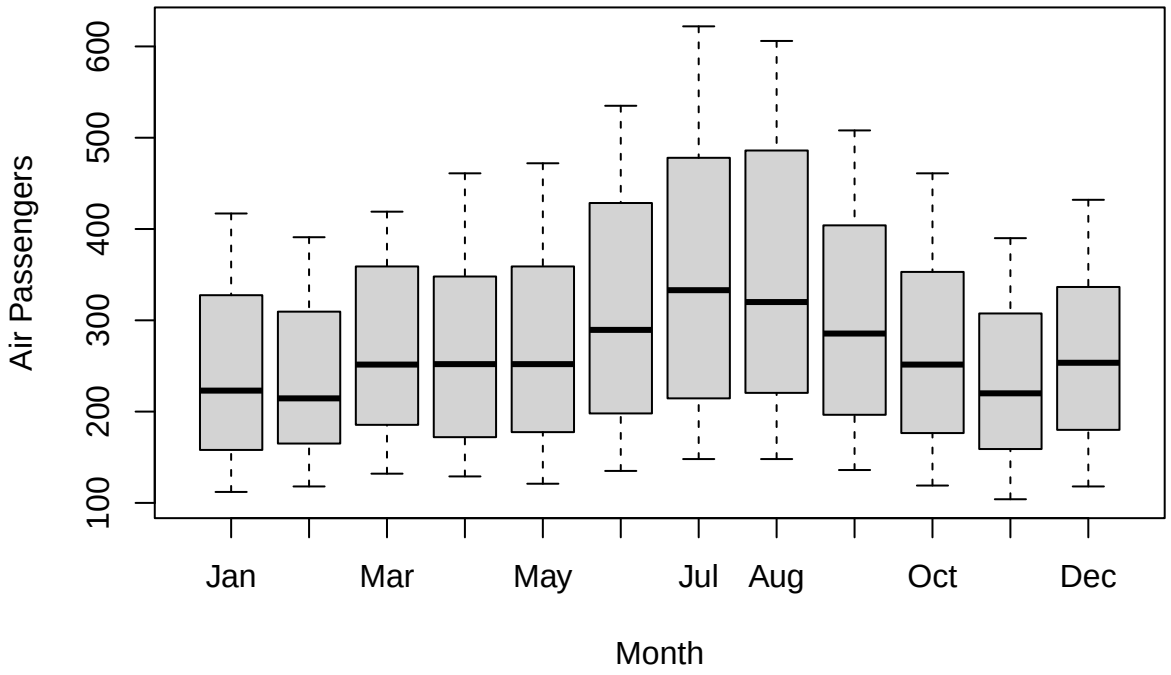
```
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(tidyverse)

# Plot time series
data(AirPassengers)
plot(AirPassengers, type = "l")
```




```
# Create dataframe
ap.df <- data.frame(
  passengers = as.numeric(AirPassengers),
  month = factor(month.abb[cycle(AirPassengers)], levels = month.abb)
)

# Seasonal Boxplot
boxplot(passengers ~ month, data = ap.df,
        xlab = "Month", ylab = "Air Passengers",
        main = "Monthly Air Passengers (1949-1960)")
```




```r
# Train test split
train <- window(AirPassengers, end = c(1959, 12))
test <- window(AirPassengers, start = c(1960, 1))

# Linear model
lm <- tslm(train ~ trend)

# Linear with seasonality
linear.season <- tslm(train ~ trend + season)

# Exponential model
expmod <- tslm(train ~ trend, lambda = 0)

# Quadratic model
quadratic <- tslm(train ~ trend + I(trend^2))

# Quadratic with seasonality
quadratic.season <- tslm(train ~ trend + I(trend^2) + season)

# Actual vs predicted
nValid = 12
actual = as.numeric(test)
pred_lm = as.numeric(forecast(lm, h = nValid)$mean)
pred_linear.season = as.numeric(forecast(linear.season, h = nValid)$mean)
pred_expmod = as.numeric(forecast(expmod, h = nValid)$mean)
pred_quadratic = as.numeric(forecast(quadratic, h = nValid)$mean)
pred_quadratic.season = as.numeric(forecast(quadratic.season, h = nValid)$mean)

# RMSE
rmse <- function(actual, predicted) {
```

```
  sqrt(mean((actual - predicted)^2))
}

rmse_lm = rmse(actual, pred_lm)
rmse_linear.season = rmse(actual, pred_linear.season)
rmse_expmod = rmse(actual, pred_expmod)
rmse_quadratic = rmse(actual, pred_quadratic)
rmse_quadratic.season = rmse(actual, pred_quadratic.season)

print('RMSE')
```

```
## [1] "RMSE"
```

```
cat('Linear Model:', rmse_lm, "\n")
```

```
## Linear Model: 78.82273
```

```
cat('Linear Season Model:', rmse_linear.season, "\n")
```

```
## Linear Season Model: 49.47908
```

```
cat('Exponential Model:', rmse_expmod, "\n")
```

```
## Exponential Model: 79.36673
```

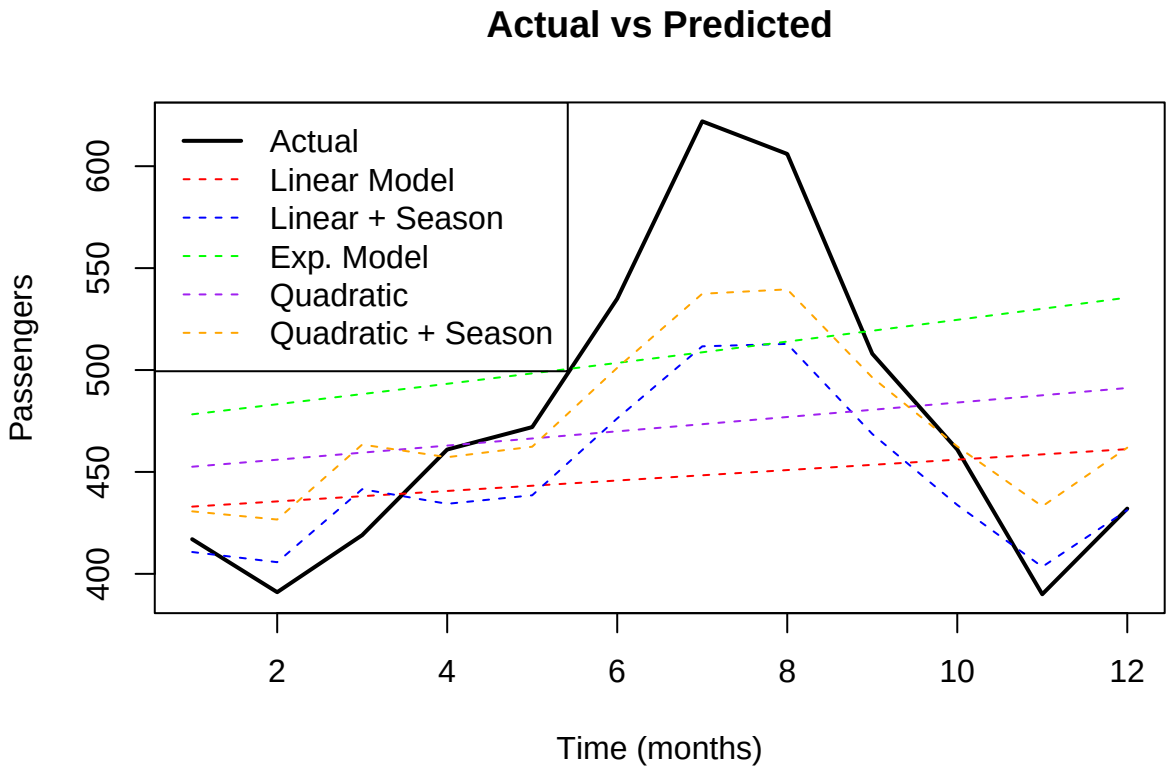
```
cat('Quadratic Model:', rmse_quadratic, "\n")
```

```
## Quadratic Model: 73.28095
```

```
cat('Quadratic Season Model:', rmse_quadratic.season, "\n")
```

```
## Quadratic Season Model: 39.94693
```

```
# Plot all 4 models (actual vs predicted)
predictions <- cbind(pred_lm, pred_linear.season, pred_expmod, pred_quadratic, pred_quadratic.season)
plot(actual, type = "l", col = "black", lwd = 2, ylim = range(c(actual, predictions)),
     ylab = "Passengers", xlab = "Time (months)", main = "Actual vs Predicted")
matlines(predictions, col = c("red", "blue", "green", "purple", "orange"), lty = 2)
legend("topleft", legend = c("Actual", "Linear Model", "Linear + Season", "Exp. Model",
                             "Quadratic", "Quadratic + Season"),
       col = c("black", "red", "blue", "green", "purple", "orange"), lty = c(1, 2, 2, 2, 2, 2),
       lwd = c(2, 1, 1, 1, 1, 1))
```



2) From the seasonal boxplot, I can tell that lots of people fly in the summer and there is also a slight up tick in December. Both of these make sense as both kids are out of school in the summer and weather is nicer. Additionally December makes sense as this is when Christmas is and often times companies give time off for that.

3) The best model was the quadratic plus seasonality model. It had the lowest RMSE of 39.94. The overall trend is relatively even until the summer months. During the summer it pops up and then goes back down.