SDS322E Project Report 2

2022-11-16

Names: Yunseo Hur, Jack Misukanis, Harini Shanmugam

---

1. Title and Introduction

Completed by Yunseo and Jack

For this project, we wanted to look at overseas travel and tourism of the UK. More specifically, we looked at the expenditure and quantity of overseas visitors in the UK and UK visitors abroad.

The dataset was pulled from Kaggle, https://www.kaggle.com/datasets/prajittr/uk-tourism-information-19862020?resource=download, but was originally sourced from the International Passenger Survey data of the Office for National Statistics. The original data included seasonally and non-seasonally adjusted estimates. However, for this project, we only looked at the non-seasonally adjusted numbers.

The variables observed are **Date (Year and Month)**, **Pounds (in millions) spent by overseas visitors in the UK**, **Number (in thousands) of overseas visitors in the UK**, **Pounds (in millions) spent by UK visitors abroad**. **Number (in thousands) of UK visitors abroad**, **Above Average Visitor Traffic to the UK**, **Above Average Tourism Abroad of UK Citizens**.

We wanted to see the relationship between visitor traffic and economic stimulation supplied to the UK economy. A unique row represents the year and month between 1986-2020 and the amount of money spent as well as number of visitors. We tidy-ed the data by removing the non-seasonally adjusted columns. We also created a categorical variable that determines if the number of tourists to the UK that month were high or not (denoted by 0 for not high and 1 for high). One potential relationship we expect to see is the correlation between number of overseas visitors to the UK and pounds spent by overseas visitors in the UK. Another correlation we expect to see is the number of UK visitors abroad and pounds spent by UK visitors abroad.

---

2. Exploratory Data Analysis

Completed by Harini

Tidying

```r
# Load packages
library(readxl)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.8     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(plotROC)
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```r
library(factoextra)
```

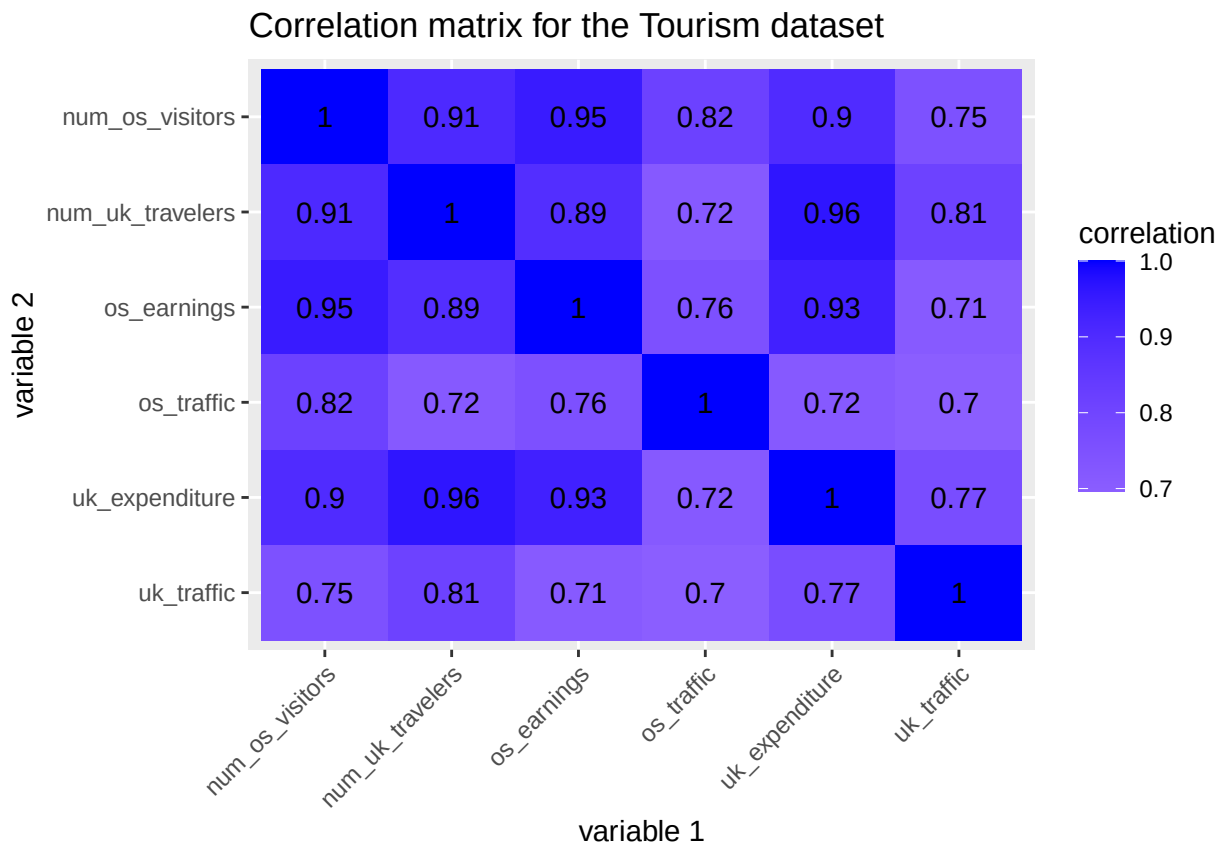
```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

```r
library(cluster)

# Load dataset in as an object
tourism_raw <- read_excel("UK_tourism_data.xls")

# Tidying the original dataset
# Save changes to a new object
tourism <- tourism_raw %>%
  # These are the specific columns we want to look at
  select("Title",
         "OS visitors Earnings: Mn £ (NSA)",
         "OS visitors: # in Thousands (NSA)",
         "UK visitors Expenditure: Mn £ (NSA)",
         "UK visitors: # in Thousands (NSA)") %>%
  # Renaming the existing column names with new names
  rename(date = "Title",
         os_earnings = "OS visitors Earnings: Mn £ (NSA)",
         num_os_visitors = "OS visitors: # in Thousands (NSA)",
```

```
        uk_expenditure = "UK visitors Expenditure: Mn £ (NSA)",
        num_uk_travelers = "UK visitors: # in Thousands (NSA)") %>%
  # Creating categorical variables
  # If average number of OS visitors is high then, 1. Else 0
  mutate(os_traffic = ifelse(num_os_visitors > mean(num_os_visitors), 1, 0)) %>%
  # If average number of UK travelers is high then, 1. Else 0
  mutate(uk_traffic = ifelse(num_uk_travelers > mean(num_uk_travelers), 1, 0)) %>%
  # Create a duplicate of the date column
  mutate(date2 = date) %>%
  # Split date2 column into two new columns year and month
  separate(date2, c("year", "month"), convert=TRUE) %>%
  filter(year!=2020) %>%
  # Order columns by date, month, year, and then everything else as they were
  select(date, month, year, everything())
```

Data Analysis

```
# Correlation matrix
tourism %>%
  # Do not include year and month in the matrix
  select(-date, -year, -month) %>%
  # Find the correlations among the other variables
  cor(use = "pairwise.complete.obs") %>%
  # Save as a data frame
  as.data.frame %>%
  # Convert row names to an explicit variable
  rownames_to_column %>%
  # Pivot so that all correlations appear in the same column
  pivot_longer(-1,
               names_to = "other_var",
               values_to = "correlation") %>%
  # Define ggplot (reorder values on y-axis)
  ggplot(aes(x = rowname,
             y = ordered(other_var, levels = rev(sort(unique(other_var)))),
             fill = correlation)) +
  # Heat map with geom_tile
  geom_tile() +
  # Change the scale to make the middle appear neutral
  scale_fill_gradient2(low = "red", mid = "white", high = "blue") +
  # Overlay values
  geom_text(aes(label = round(correlation,2)), color = "black", size = 4) +
  # Angle the x-axis label to 45 degrees
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  # Give title and labels
  labs(title = "Correlation matrix for the Tourism dataset",
       x = "variable 1", y = "variable 2")
```



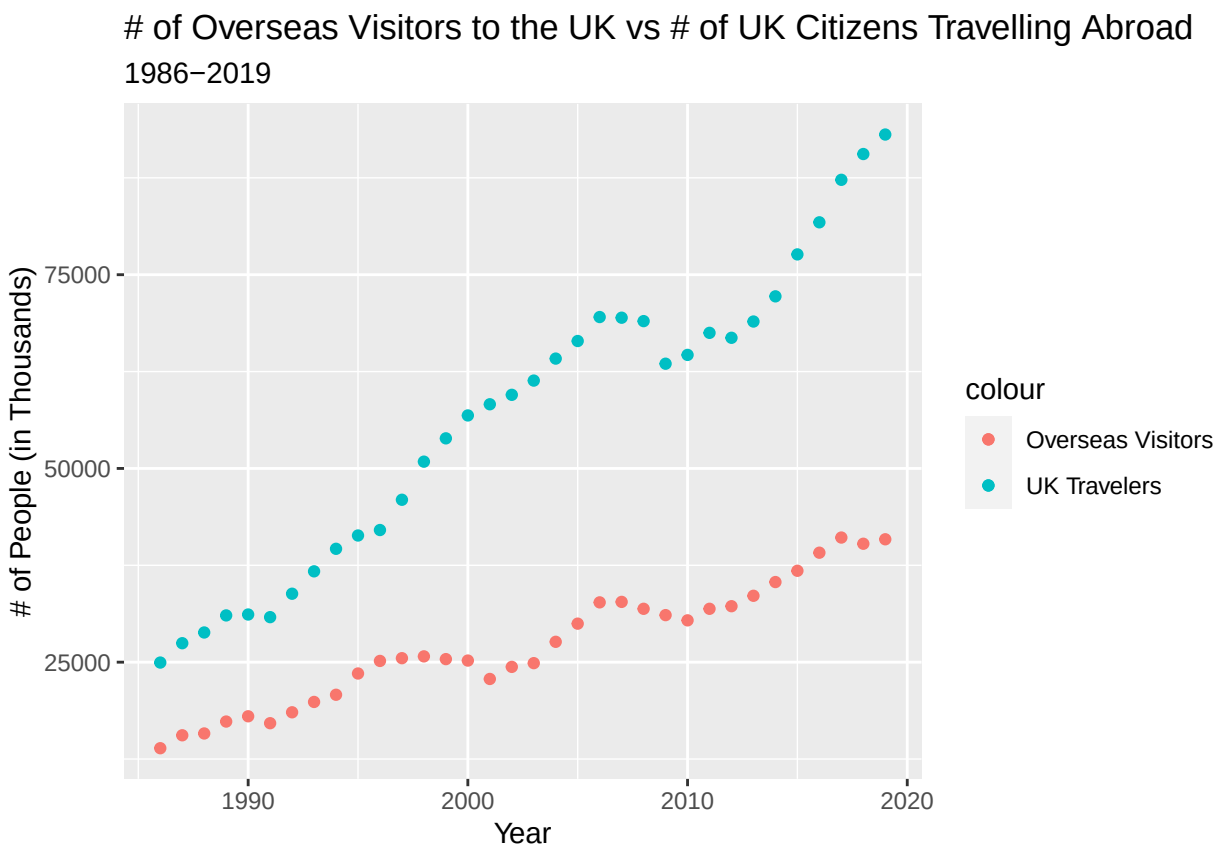
The variables within our data that appeared to be most correlated were the variables related to the number of travelers between individuals from overseas and those from the UK. Likewise, there is a strong correlation between the number of travelers within a group and the expenditure amount of that group. That is to say, when the number of overseas travelers was high, so too was overseas expenditures. The variables that appeared to be the least correlated were variables regarding high traffic travel years between overseas and UK travelers. That is, when it was a high traffic travel year for UK citizens, this did not necessarily correlate to a high traffic travel year for overseas travelers.

Visualizations

Completed by Yunseo

```
tourism %>%
  # Order rows by year
  group_by(year) %>%
  # Calculate number of os visitors and uk travelers by each year
  summarise(os_visitors_by_year = sum(num_os_visitors),
            uk_travelers_by_year = sum(num_uk_travelers)) %>%
  # Plot number of people by year
  ggplot(aes(x=year)) +
  # Two y variables
  geom_point(aes(y=os_visitors_by_year, color="Overseas Visitors")) +
```

```
  geom_point(aes(y=uk_travelers_by_year, color="UK Travelers")) +
  labs(title = "# of Overseas Visitors to the UK vs # of UK Citizens Travelling Abroad",
       subtitle = "1986-2019",
       x = "Year",
       y = "# of People (in Thousands)")
```

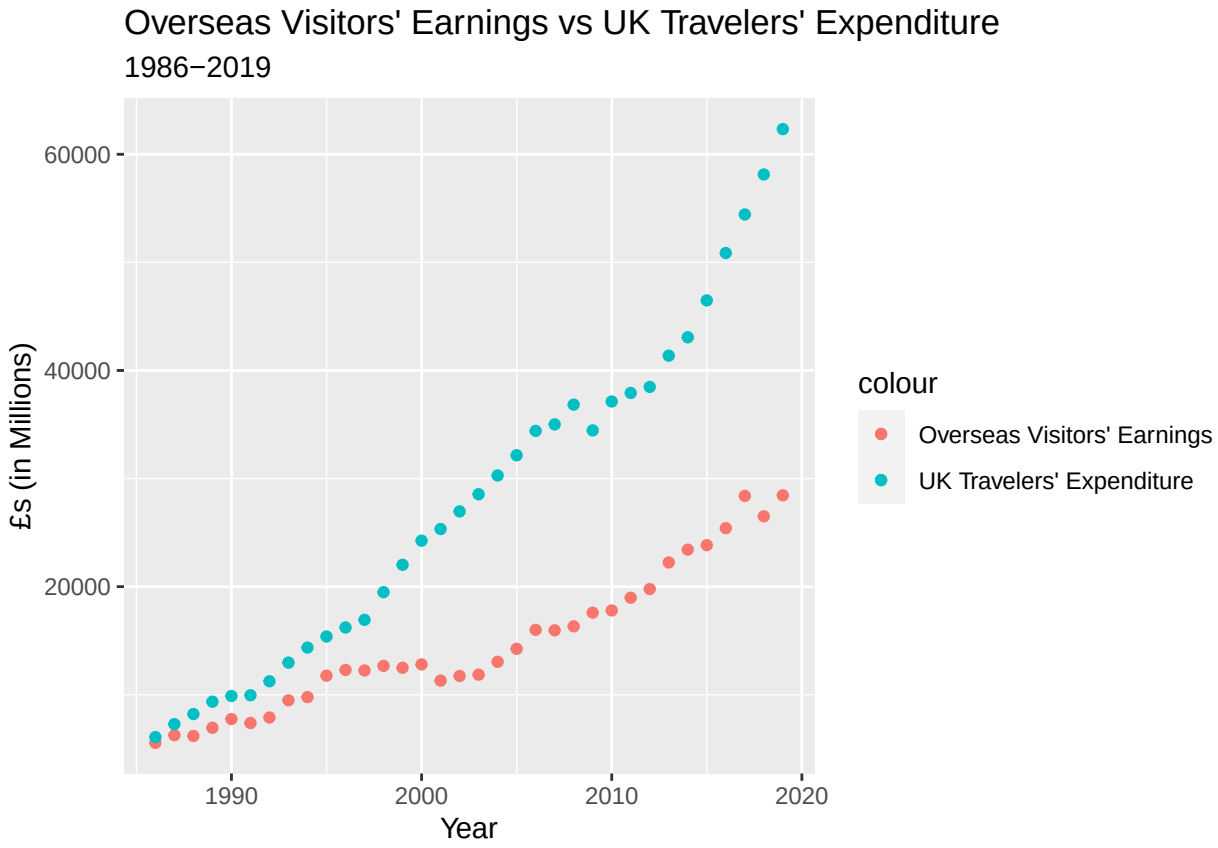


1)

As evidenced in the visualization, # of Overseas Visitors to the UK and # of UK Citizens Traveling abroad are both highly related. These variables share a positive relationship in that both are steadily increasing over time, with UK Travelers beginning to outpace Overseas visitors in 1997 and beyond.

```
tourism %>%
  # Order rows by year
  group_by(year) %>%
  # Calculate os earnings and uk expenditures by each year
  summarise(os_earnings_by_year = sum(os_earnings),
            uk_expenditure_by_year = sum(uk_expenditure)) %>%
  # Plot amount of money by year
  ggplot(aes(x=year)) +
  # Two y variables
  geom_point(aes(y=os_earnings_by_year, color="Overseas Visitors' Earnings")) +
  geom_point(aes(y=uk_expenditure_by_year, color="UK Travelers' Expenditure")) +
  labs(title = "Overseas Visitors' Earnings vs UK Travelers' Expenditure",
       subtitle = "1986-2019",
```

```
       x = "Year",
       y = "£s (in Millions)")
```



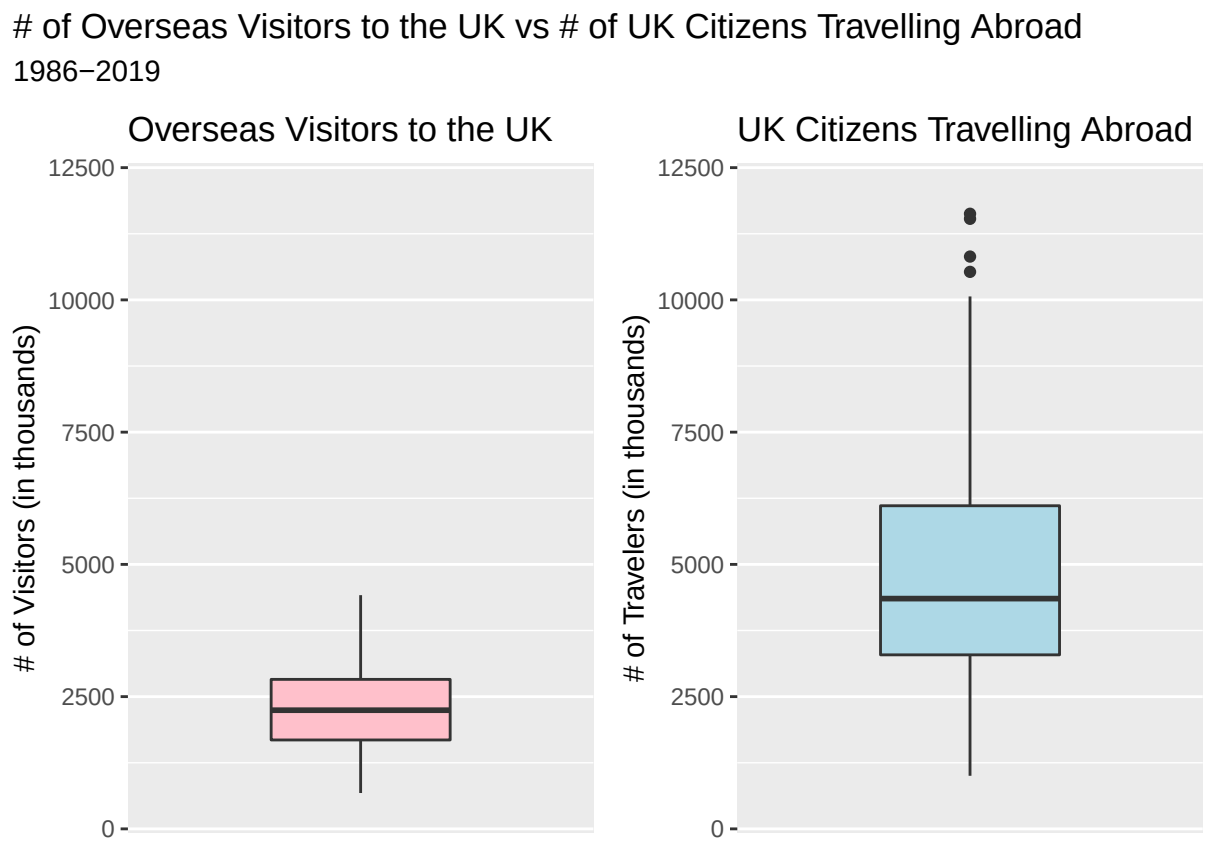
2)

There is a positive relationship between UK Travelers' Expenditures and Overseas Visitors' Earnings. This means that, over time, UK Travelers' Expenditures and Overseas Visitors' Earnings both increase with each passing year, where UK Travelers' Expenditures begins to outpace Overseas Visitors' Earnings in roughly 1997. This conclusion matches that from the previous visualization, and this makes sense given the high correlation between number of travelers and expenditures in each subgroup.

3) *Completed by Jack*

```
library(patchwork)

# Boxplot of number of os visitors
box1 <- tourism %>%
  ggplot(aes(y = num_os_visitors)) +
  # Color plot
  geom_boxplot(fill="pink") +
  labs(title="Overseas Visitors to the UK",
       y="# of Visitors (in thousands)") +
  # mute x-axis markers
  scale_x_discrete() +
  # limit y-axis
  ylim(500,12000)
```

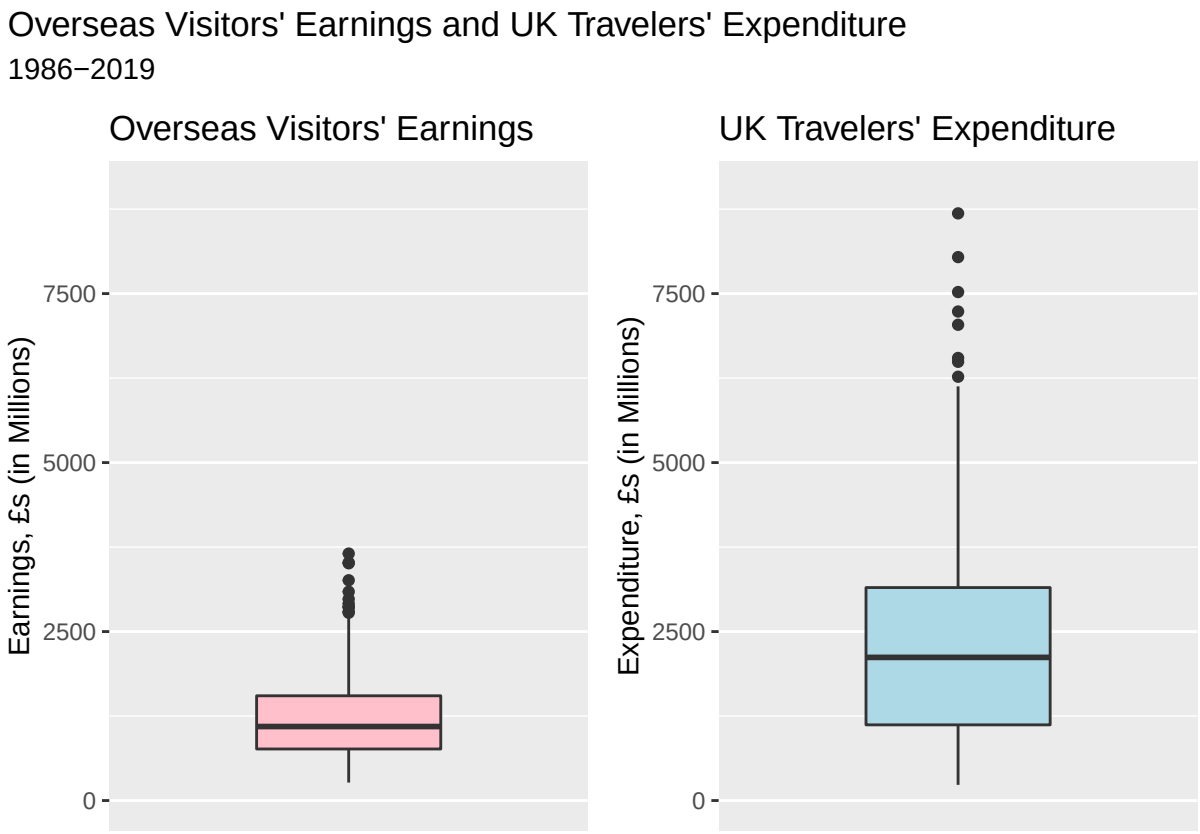
```
# Boxplot of number of uk travelers
box2 <- tourism %>%
  ggplot(aes(y = num_uk_travelers)) +
  geom_boxplot(fill="light blue") +
  labs(title="UK Citizens Travelling Abroad",
       y="# of Travelers (in thousands)") +
  scale_x_discrete() +
  ylim(500,12000)

# Put boxplots side-by-side and give a joint title
box1 + box2 + plot_annotation(
  title="# of Overseas Visitors to the UK vs # of UK Citizens Travelling Abroad",
  subtitle = "1986-2019")
```



Source for how we learned to put the boxplots side-by-side: https://www.statology.org/side-by-side-plots-ggplot2/

This visualization indicates that the spread between the two subgroups, UK Citizens Traveling Abroad and Overseas Visitors to the UK, is quite different from one another. UK Citizens Traveling Abroad has a similar minimum as the Overseas Visitors to the UK, but a much higher maximum as well as a higher 1st quartile, 3rd quartile and median. There are also more outliers in the "UK Citizens Traveling Abroad" subgroup. The spread for the data of the "Overseas Visitors to the UK" subgroup is much smaller and this indicates that traveling for this group did not grow nearly as much as the "UK Citizens Traveling Abroad" group over time.

```
# Boxplot of amount of os earnings
box3 <- tourism %>%
  ggplot(aes(y = os_earnings)) +
  geom_boxplot(fill="pink") +
  labs(title = "Overseas Visitors' Earnings",
       y = "Earnings, £s (in Millions)") +
  scale_x_discrete() +
  ylim(0,9000)

# Boxplot of amount of uk expenditure
box4 <- tourism %>%
  ggplot(aes(y = uk_expenditure)) +
  geom_boxplot(fill="light blue") +
  labs(title = "UK Travelers' Expenditure",
       y = "Expenditure, £s (in Millions)") +
    scale_x_discrete() +
  ylim(0,9000)

# Put boxplots side-by-side and give a joint title
box3 + box4 + plot_annotation(
  title="Overseas Visitors' Earnings and UK Travelers' Expenditure",
  subtitle = "1986-2019")
```



4)

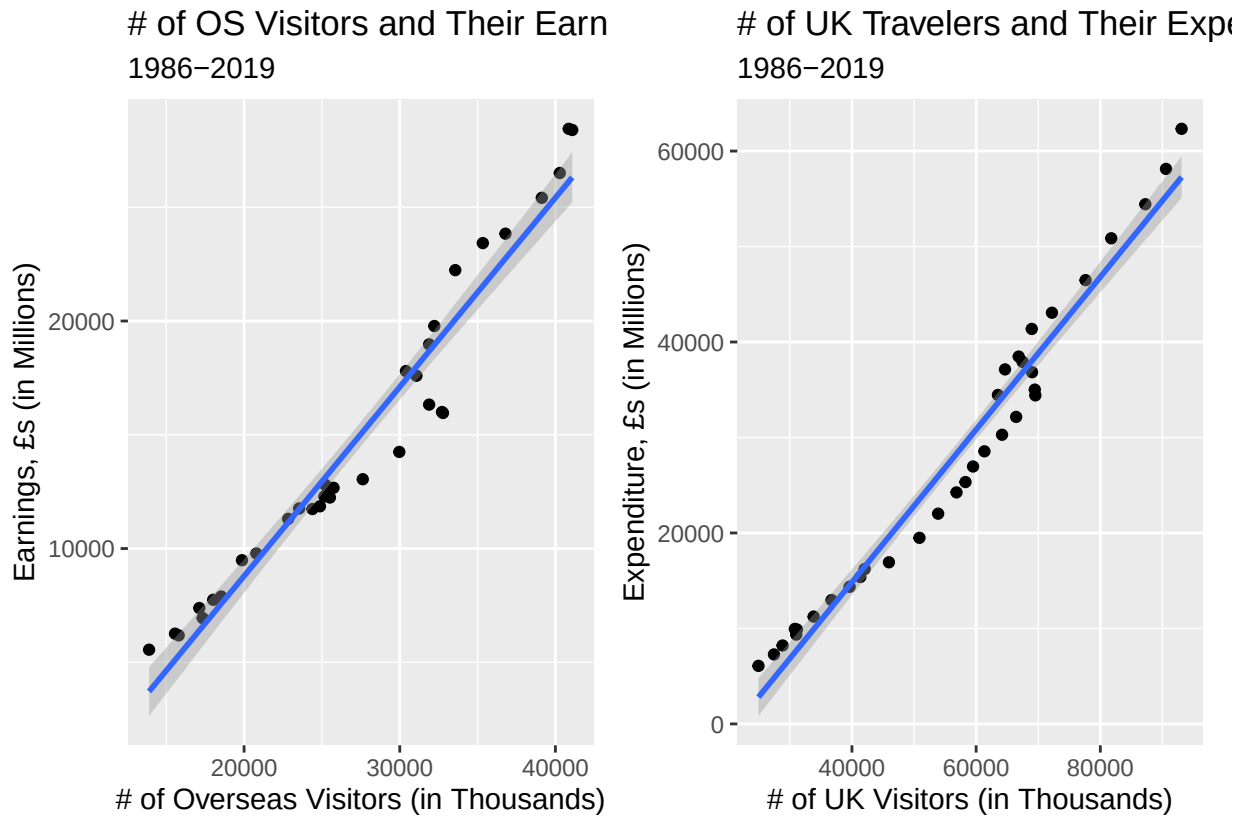
The above boxplot visualization indicates that the spread for expenditures was much higher for the UK

Traveler's group than the earnings for the Overseas Visitors group. This makes inherent sense as this was the conclusion drawn from the previous visualization looking at number of travelers within these subgroups. For the above visualization, the spread is much smaller in the "Overseas Visitors' Earnings" Group, indicating a smaller minimum, maximum, 1st quartile, 3rd quartile and median value. This indicates that the spread of the data for "Overseas Visitors' Earnings" is smaller than the "UK Travelers' Expenditures" group.

5) *Completed by Harini*

```
plot1 <- tourism %>%
  # Order rows by year
  group_by(year) %>%
  # Calculate os vistors and os earnings by each year
  summarise(os_visitors_by_year = sum(num_os_visitors),
            os_earnings_by_year = sum(os_earnings)) %>%
  ggplot(aes(x=os_visitors_by_year, y=os_earnings_by_year)) +
  geom_point() +
  # Add regression line
  geom_smooth(method="lm") +
  labs(title = "# of OS Visitors and Their Earnings",
       subtitle = "1986-2019",
       x = "# of Overseas Visitors (in Thousands)",
       y = "Earnings, £s (in Millions)")

plot2 <- tourism %>%
  group_by(year) %>%
  summarise(uk_travelers_by_year = sum(num_uk_travelers),
            uk_expenditure_by_year = sum(uk_expenditure)) %>%
  ggplot(aes(x=uk_travelers_by_year, y=uk_expenditure_by_year)) +
  geom_point() +
  geom_smooth(method="lm") +
  labs(title = "# of UK Travelers and Their Expenditure",
       subtitle = "1986-2019",
       x = "# of UK Visitors (in Thousands)",
       y = "Expenditure, £s (in Millions)")

plot1 + plot2 + plot_annotation()
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



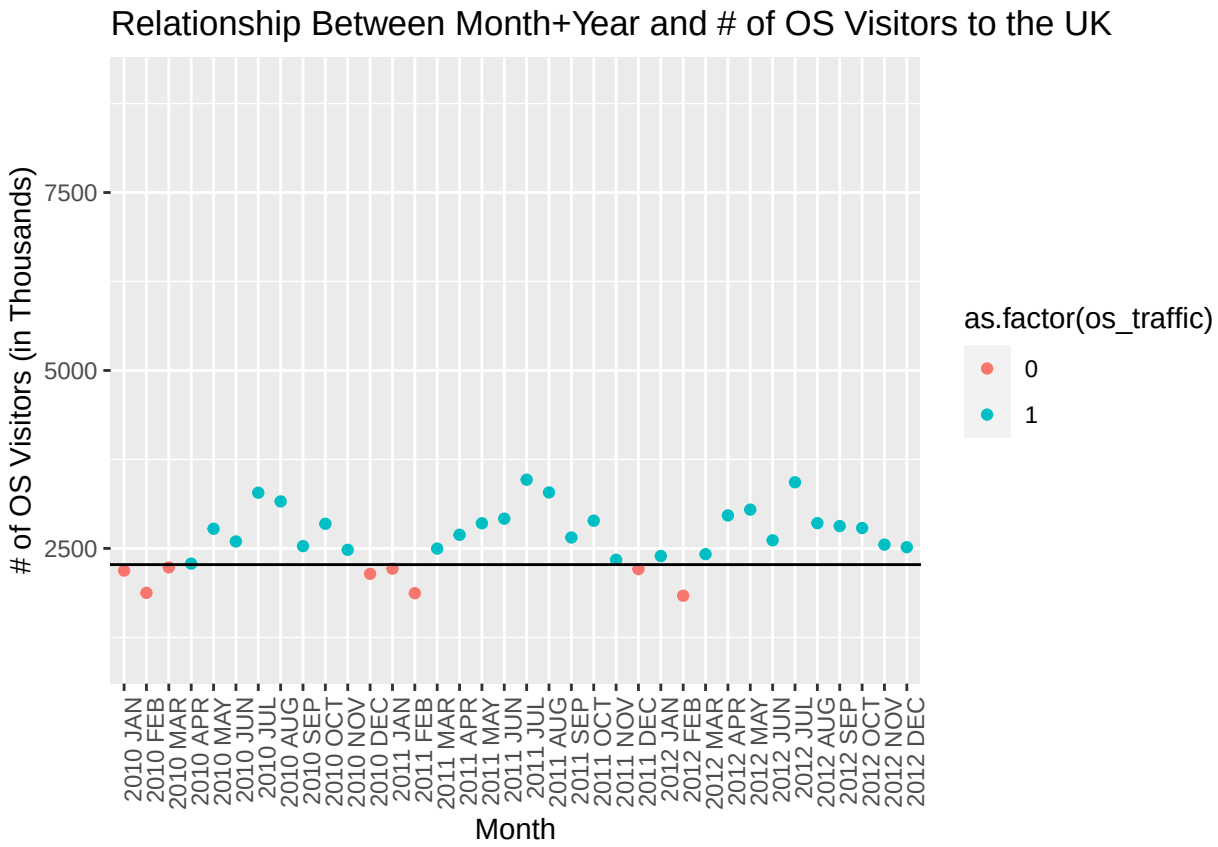

```
# Can find slope by saving previous calculations as objects and running the following does:
# model_os <- lm(os_visitors_by_year ~ os_earnings_by_year, data=tourism)
# model_os
# model_uk <- lm(uk_travelers_by_year ~ uk_expenditure_by_year, data=tourism)
# model_uk
```

The first visualization is highlighting the relationship between overseas visitors and their respected earnings. As expected, the relationship between these two variables is a positive one indicating that, typically as the number of overseas visitors is increasing, so too is their respected earnings in a given year. The gray band around the regression line indicating the 95% confidence interval for the regression line is larger in the upper and lower sections of the line and this is due to outliers which pull away from the regression line and expand the size of the confidence interval in these given areas.

As we noted with overseas visitors and earnings, the relationship between UK travelers and their expenditure also appears to be quite strong. That is, as the number of UK visitors is increasing, so too is their level of expenditure in a given year. As seen in the overseas visitor/earning visualization, the gray band indicating the 95% confidence interval of the regression line is expanded in the highest and lowest parts of the graph. This is due to outliers which expand the size of the CI in these areas.

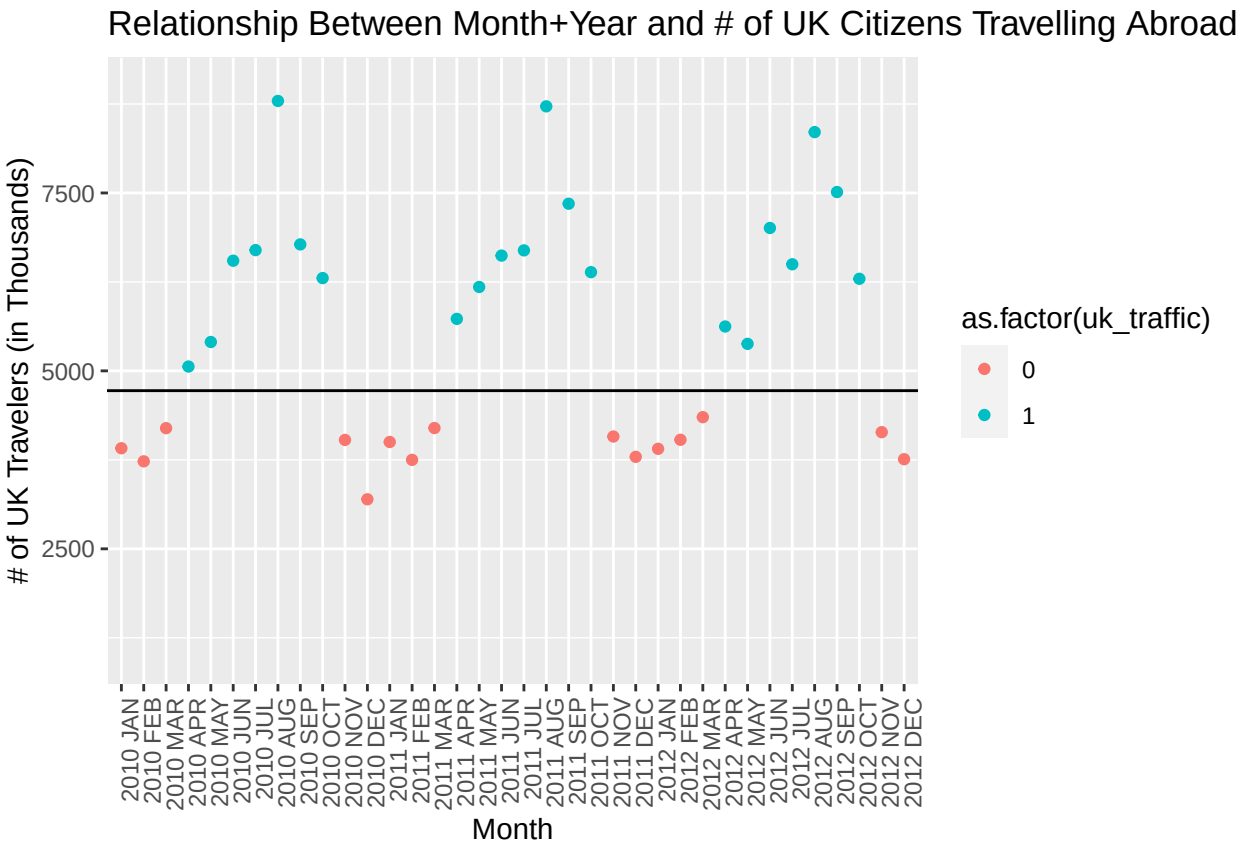
```
tourism %>%
  # Keep only rows with years between 2020 nd 2012
  filter(between(year, "2010", "2012")) %>%
  # Keep dates in the original order in the data set
```

```
  ggplot(aes(x=fct_inorder(date), y=num_os_visitors, color=as.factor(os_traffic))) +
  geom_point() +
  labs(title = "Relationship Between Month+Year and # of OS Visitors to the UK",
       x = "Month",
       y = "# of OS Visitors (in Thousands)") +
  # Plot horizontal line marking average
  geom_abline(intercept=2272, slope=0) +
  # Angle text of x-axis marker
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  # Set y-axis range
  ylim(1000,9000)
```



6)

```
tourism %>%
  filter(between(year, "2010", "2012")) %>%
  ggplot(aes(x=fct_inorder(date), y=num_uk_travelers, color=as.factor(uk_traffic))) +
  geom_point() +
  labs(title = "Relationship Between Month+Year and # of UK Citizens Travelling Abroad",
       x = "Month",
       y = "# of UK Travelers (in Thousands)") +
  geom_abline(intercept=4722, slope=0) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))+
  ylim(1000,9000)
```



Source for how we learned to keep order of categorical values in the original order that's in the dataset: https://stackoverflow.com/questions/20041136/avoid-ggplot-sorting-the-x-axis-while-plotting-geom-bar#

We looked over a three year period at the amount of travelers going into the United Kingdom as a result of the marriage between Prince William and Kate Middleton to see if this historic event would impact the number of travelers going into the UK or impact the number of UK citizens traveling to other countries. The wedding itself took place in April 2011, but we wanted to look at travel from the previous year and the following year as points of comparison. Our hypothesis is that such a historic event would bring travelers into the country to view the spectacle. From the above visualization we determined that the wedding of Prince William and Catherine had nearly no effect on the amount of overseas visitors traveling to the United Kingdom in 2011. While April was a "high travel" month at that time, that specific month appeared to have no higher level of traffic than the year before or the year after.

Our hypothesis was that the wedding would have little impact on the number of UK Citizens traveling abroad as the event itself was happening in the United Kingdom. Looking at the same three year period, there is no evidence that the wedding of Prince William and Catherine Middleton had any impact on the number of UK Citizens Traveling Abroad.

---

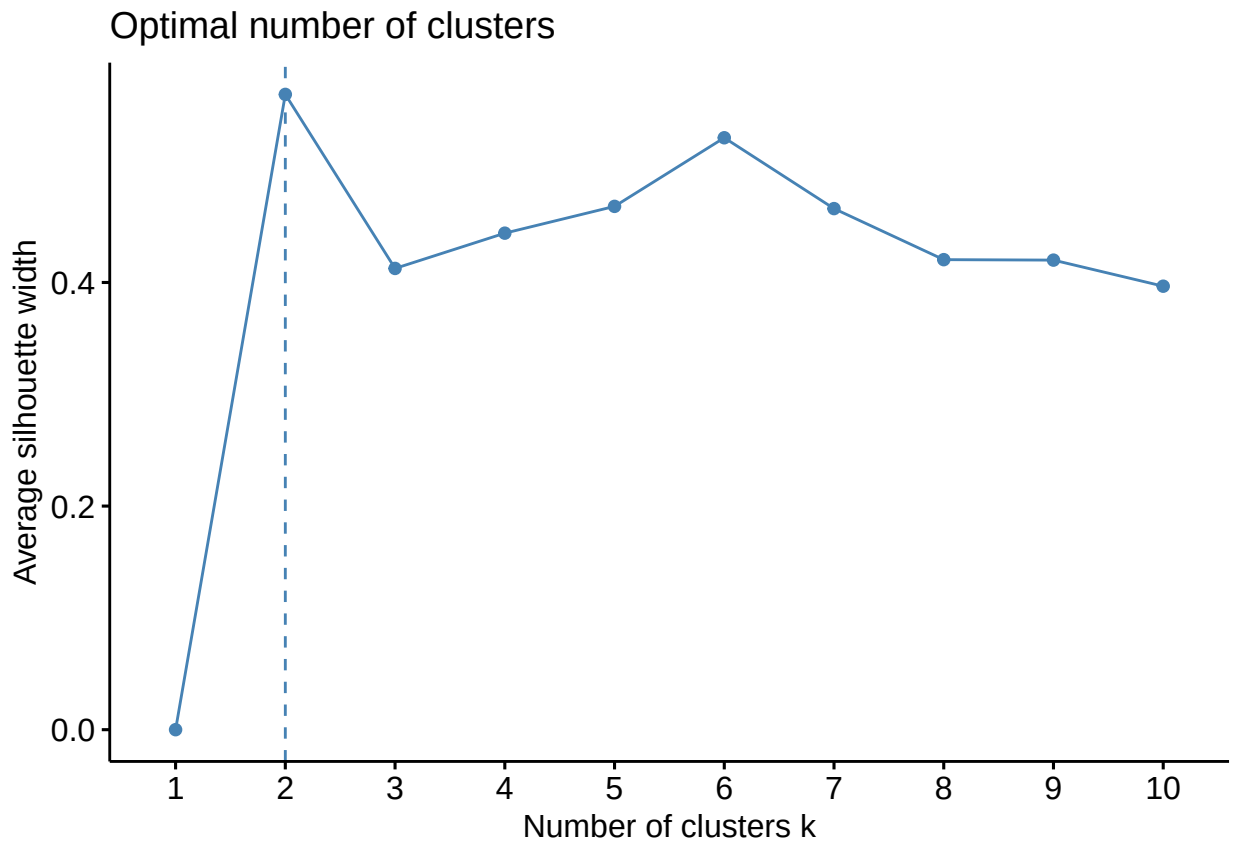
3. Clustering

Completed by Jack

```r
# Prepare data
tourism_scaled <- tourism %>%
  select(-date, -year, -month) %>%
  scale

# Use the function pam() to find clusters
pam_results <- tourism_scaled %>%
  pam(k = 2) # k is the number of clusters

# Save cluster assignment as a column in the original dataset
tourism_pam <- tourism %>%
  mutate(cluster = as.factor(pam_results$clustering))

# Compare the cluster and os_traffic
table(tourism_pam$cluster, tourism_pam$os_traffic)
```

```
##
##       0   1
##   1 204  34
##   2   8 162
```

```r
# Maximize the silhouette while keeping a small number of clusters
fviz_nbclust(tourism_scaled, pam, method = "silhouette")
```



```
# Let's visualize our data with cluster assignment
fviz_cluster(pam_results, data = tourism_scaled)
```


```
# Statistic about the centers of the clusters
tourism[pam_results$id.med, ]
```

```
## # A tibble: 2 x 9
##   date     month  year os_earnings num_os_visi~1 uk_ex~2 num_u~3 os_tr~4 uk_tr~5
##   <chr>    <chr> <int>       <dbl>         <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 1998 MAR MAR    1998         808          1767    1364    3327       0       0
## 2 2011 JUN JUN    2011        1797          2918    3640    6620       1       1
## # ... with abbreviated variable names 1: num_os_visitors, 2: uk_expenditure,
## #   3: num_uk_travelers, 4: os_traffic, 5: uk_traffic
```

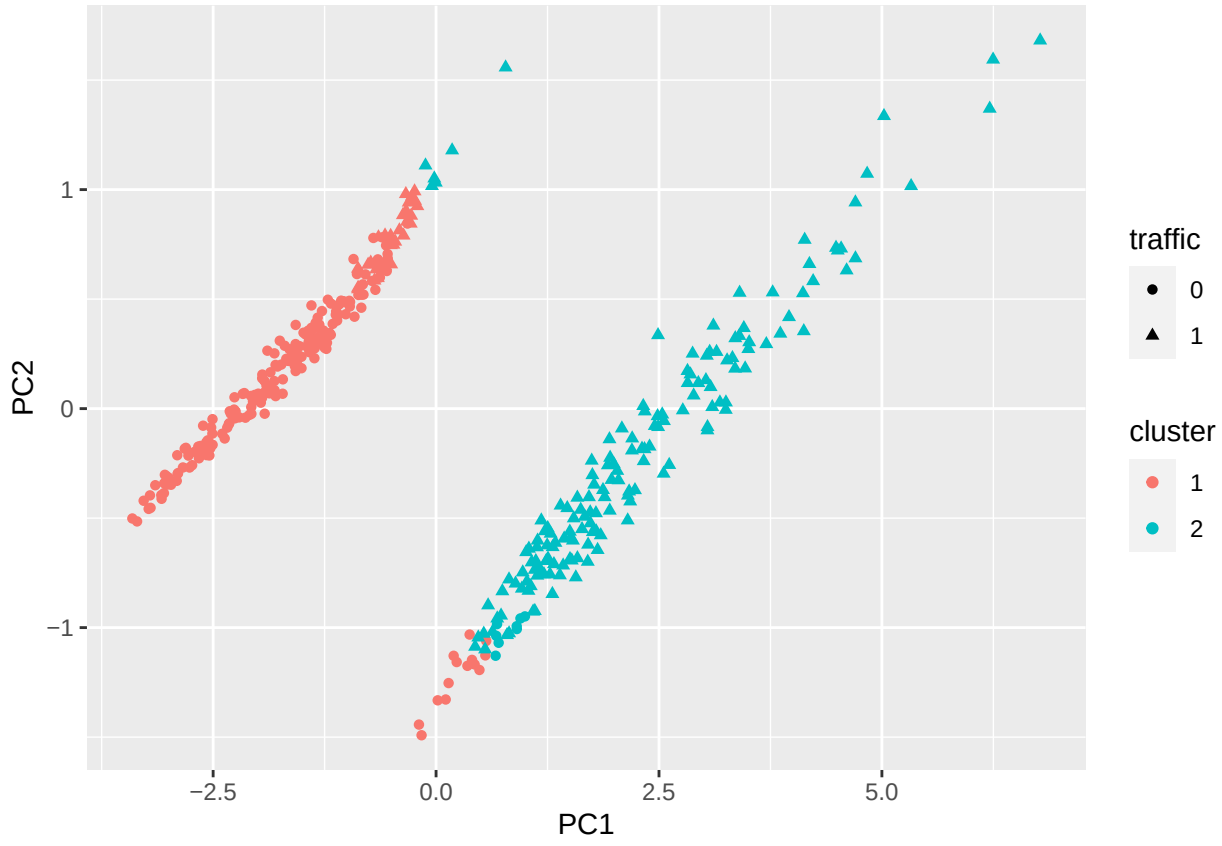
From the silhouette width, it is recommended that we have two clusters. After applying k=2 clusters we see the two clusters 1) low overseas traffic and low UK traffic and 2) high overseas traffic and high UK traffic. The center of the first cluster is the data point March 1998 and the center of the second cluster is June 2011.

---

4. Dimensionality Reduction

Completed by Yunseo

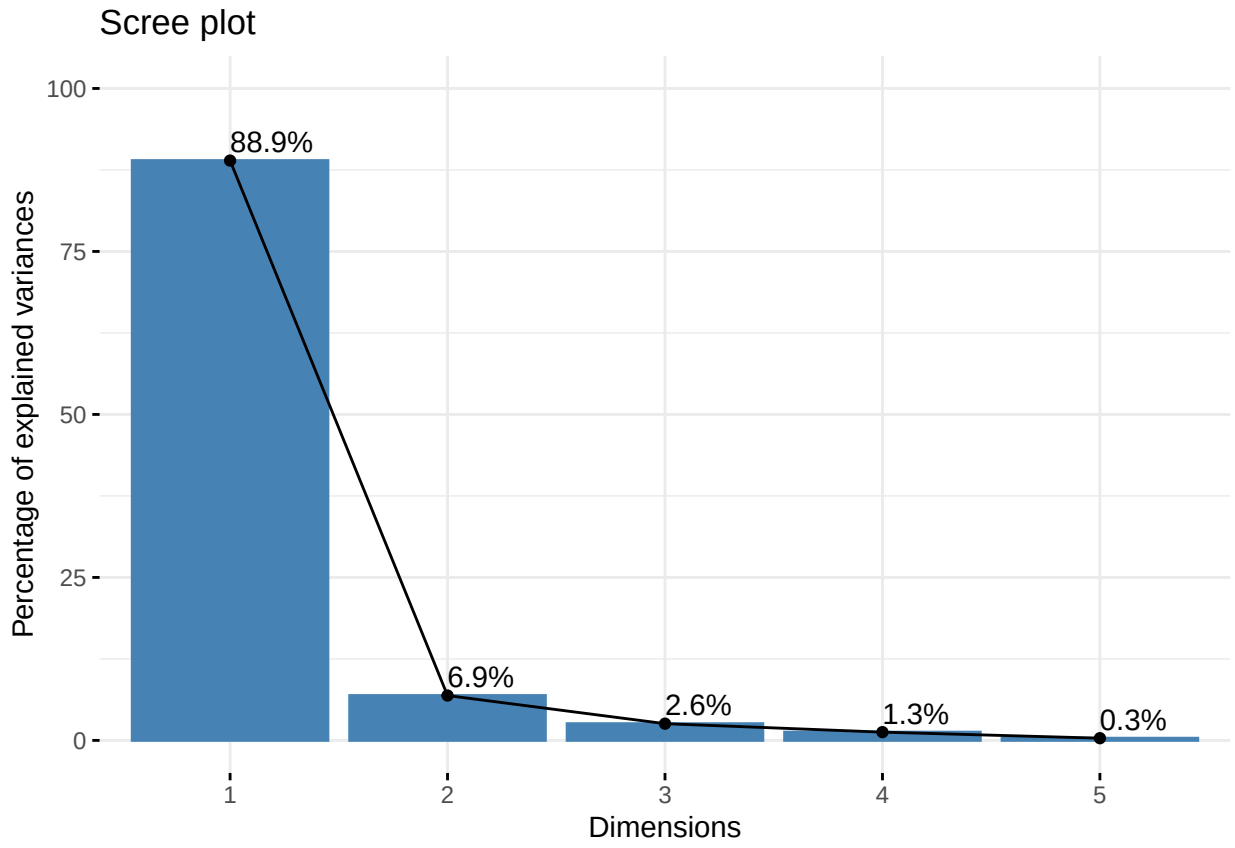
```
# Prepare the data
tourism_reduced <- tourism %>%
  select(-date, -year, -month) %>%
  mutate(os_traffic = as.factor(os_traffic))

# Apply PCA to numeric variables only
pca <- tourism_reduced %>%
  select_if(is.numeric) %>%
  scale %>%
  prcomp

# Represent the clusters on PC1 and PC2
pca$x %>%
  as.data.frame %>%
  mutate(cluster = as.factor(pam_results$clustering),
         traffic = tourism_reduced$os_traffic) %>% # also add traffic
  ggplot(aes(x = PC1, PC2,
             color = cluster, shape = traffic)) +
  geom_point()
```



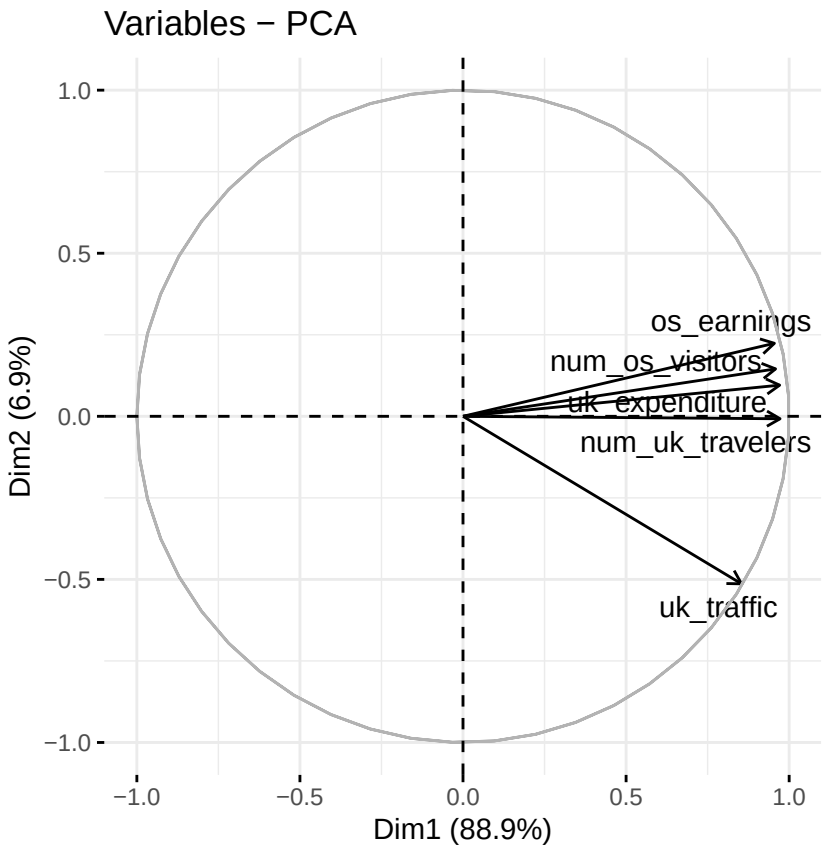

```
# Average silhouette width
pam_results$silinfo$avg.width
```

```
## [1] 0.5682335
```

```
# Percentage of variance explained for each PC in a scree plot
fviz_eig(pca, addlabels = TRUE, ylim = c(0, 100))
```




```
# PCA correlation circle
fviz_pca_var(pca, col.var = "black",
             repel = TRUE)
```



The Principal Component Analysis (PCA) correlation circle indicates that os_earnings, num_os_visitors, uk_expenditure, and num_uk_visitors are grouped together and thus are positively correlated variables. Of the components analyzed, the variable representing high traffic travel days for UK citizens was not included in this grouping.

The small discrepancy can be ween by the handful of pink triangles towards the top of the left hand cluster and the blue circles towards the bottom of the right hand cluster.

From the scree plot, we see that Dimension 1 explains the majority of the variation for each principal component.

---

5. Classification and Cross-Validation

Completed by Yunseo, Jack, and Harini

Classify with KNN model

```
# Consider the kNN classifier with k = 5
tourism_kNN <- knn3(os_traffic ~ os_earnings + num_os_visitors +
                      uk_expenditure + num_uk_travelers,
                data = tourism,
```

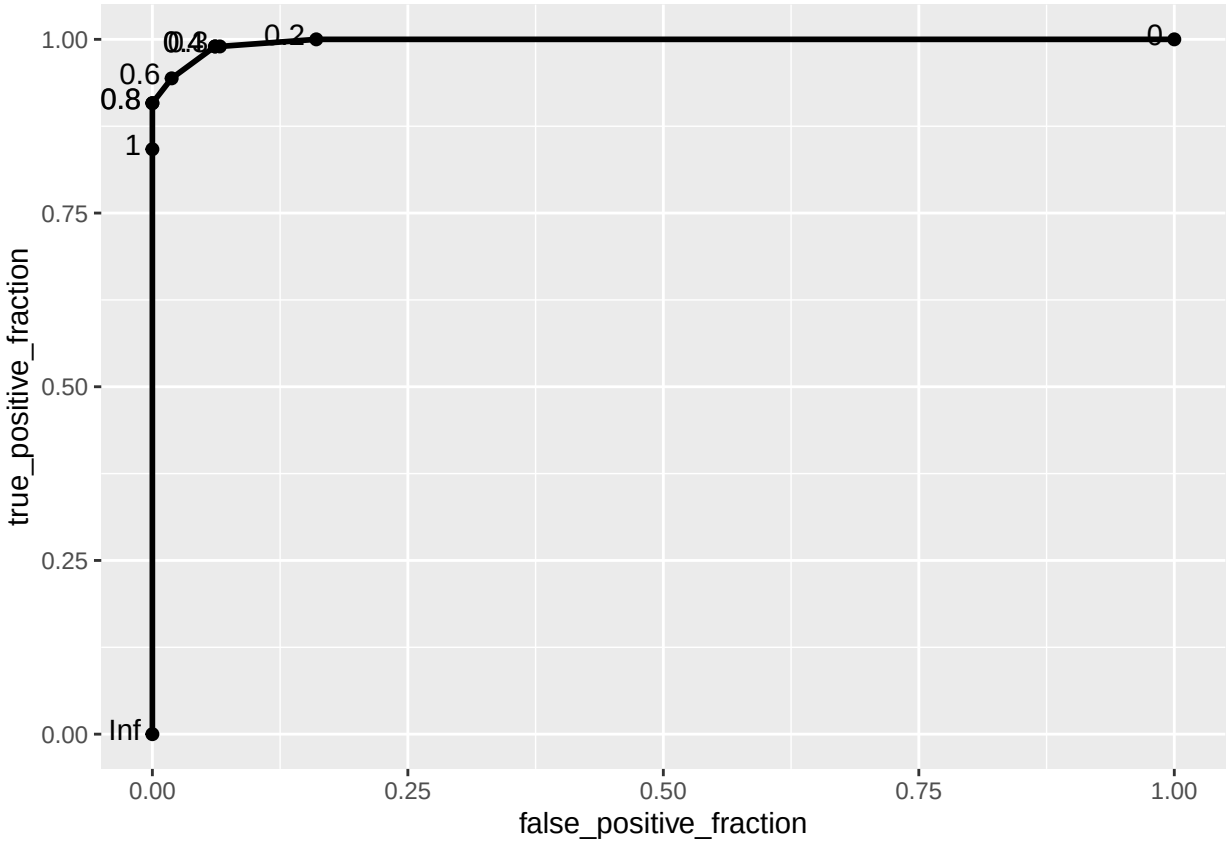
```
                k = 5) # number of neighbors

# Find the proportion of nearest neighbors with os_traffic
predict(tourism_kNN, tourism) %>% as.data.frame %>% head
```

```
##   0 1
## 1 1 0
## 2 1 0
## 3 1 0
## 4 1 0
## 5 1 0
## 6 1 0
```

```
tourism_pred <- tourism %>%
  mutate(predictions = predict(tourism_kNN, tourism)[,2], # index second column
         predicted = ifelse(predictions > 0.5, "high traffic", "low traffic"))

# Accuracy (by confusion matrix)
table(tourism_pred$predicted, tourism_pred$os_traffic) %>% addmargins
```

```
##
##                   0   1 Sum
##   high traffic    4 185 189
##   low traffic   208  11 219
##   Sum           212 196 408
```

```
#ROC Curve
ROC <- ggplot(tourism_pred) +
  geom_roc(aes(d = os_traffic, m = predictions), n.cuts = 10)
ROC
```




```
#AUC
calc_auc(ROC)$AUC
```

```
## [1] 0.9966668
```

Our KNN model predicts new observations very well as seen by the ROC curve with an AUC of 0.99

K-Fold Cross Validation

```
# Choose number of folds
# k = 10

# Randomly order rows in the dataset
# data <- tourism[sample(nrow(tourism)), ]

# Create k folds from the dataset
# folds <- cut(seq(1:nrow(data)), breaks = k, labels = FALSE)

# Cross-Validation
# Initialize a vector to keep track of the performance
# perf_k <- NULL

# Use a for loop to get diagnostics for each test set
```

```
# for(i in 1:k){
   # Create train and test sets
#  train <- data[folds != i, ] # all observations except in fold i
#  test <- data[folds == i, ]  # observations in fold i

   # Train model on train set (all but fold i)
#  tourism_kNN <- knn3(os_traffic ~ .,
#                 data = train,
#                 k = 5)

   # Test model on test set (fold i)
#  df <- data.frame(
#    predictions = predict(tourism_kNN, test)[,2],
#    os_traffic = test$os_traffic)

   # Consider the ROC curve for the test dataset
#  ROC <- ggplot(df) +
#    geom_roc(aes(d = os_traffic, m = predictions))

   # Get diagnostics for fold i (AUC)
#  perf_k[i] <- calc_auc(ROC)$AUC
#}

# Average performance
# mean(perf_k)
```

Unfortunately, we kept running into an error while running our code for the k-fold cross-validation. Above is what should have sucessfully been run. We would have seen a similarly well performing model.

---

6. Formatting