

# HW5

Arjun Ganesh

3/24/2020

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.0 --

## v tibble  2.1.3      v purrr   0.3.3
## v tidyr   1.0.2      v stringr 1.4.0
## v readr   1.3.1      v forcats 0.5.0

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

1. [8] Import the RestaurantRating1 dataset in R and save the resulting data frame. Use some of the data wrangling techniques to transform the dataset into a tidy data. [Note: use the glimpse() function to show the resulting dataframe.]

```
RR1<- read.csv("/cloud/project/RestaurantRating1.csv")%>%
  pivot_longer(c("Donalds", "Fila", "King", "Payes", "Wendi"),
               names_to = "Restaurant", values_to = "Ratings") %>%
  group_by(Restaurant) %>%
  arrange(Ratings, .by_group = TRUE)

glimpse(RR1)
```

```
## Observations: 50
## Variables: 2
## Groups: Restaurant [5]
## $ Restaurant <chr> "Donalds", "Donalds", "Donalds", "Donalds", "Donalds", "...
## $ Ratings      <int> 1, 2, 2, 3, 3, 3, 3, 4, 4, 5, 3, 3, 3, 3, 3, 3, 3, 3,...
```

2. Import the RestaurantRating2 dataset in R and save the resulting data frame as ResRat. Use the ResRat to answer the following questions:

```
ResRat<- read.csv("/cloud/project/RestaurantRating2.csv")
```

- (A) [6] Tabulate the Restaurant variable to see the various misspellings of the restaurant names. Write a code to add a column (with column name Rest) to ResRat dataframe with the correct spellings: Donalds, Fila, King, Payes and Wendi. [Note: use the glimpse() function to show the resulting dataframe.]

```
table(ResRat$Restaurant)
```

```
##
##      D's      don donald Donald Donalds  Fil a   fila   Fila      K      king
##      1        1        1        1        6        1        1        8        1        1
##      King     Pap papayes   paye   Payes   Wen   wend   wendi   Wendi
##      8        1        1        1        7        1        1        2        6
```

```
Rest_Tab<-ResRat%>%
  mutate(Rest = ifelse(grepl("on", Restaurant), "Donalds",
    ifelse(grepl("D", Restaurant), "Donalds",
    ifelse(grepl("il", Restaurant), "Fila ",
    ifelse(grepl("in",Restaurant), "King",
    ifelse(grepl("K", Restaurant), "Payes",
    ifelse(grepl("Pa", Restaurant), "Payes",
    ifelse(grepl("pa", Restaurant), "Payes","Wendi"))))))))
```

```
glimpse(Rest_Tab)
```

```
## Observations: 50
## Variables: 3
## $ Restaurant <fct> Donalds, Donald, D's, Donalds, Donalds, don, Donalds, Do...
## $ Ratings      <int> 1, 2, 2, 3, 3, 3, 3, 4, 4, 5, 3, 3, 3, 3, 3, 3, 3, 3, 3,...
## $ Rest         <chr> "Donalds", "Donalds", "Donalds", "Donalds", "Donalds", "...
```

- (B) [3] Write a code to find the mean rating by Rest. [Note: be sure to include the output of your code.]

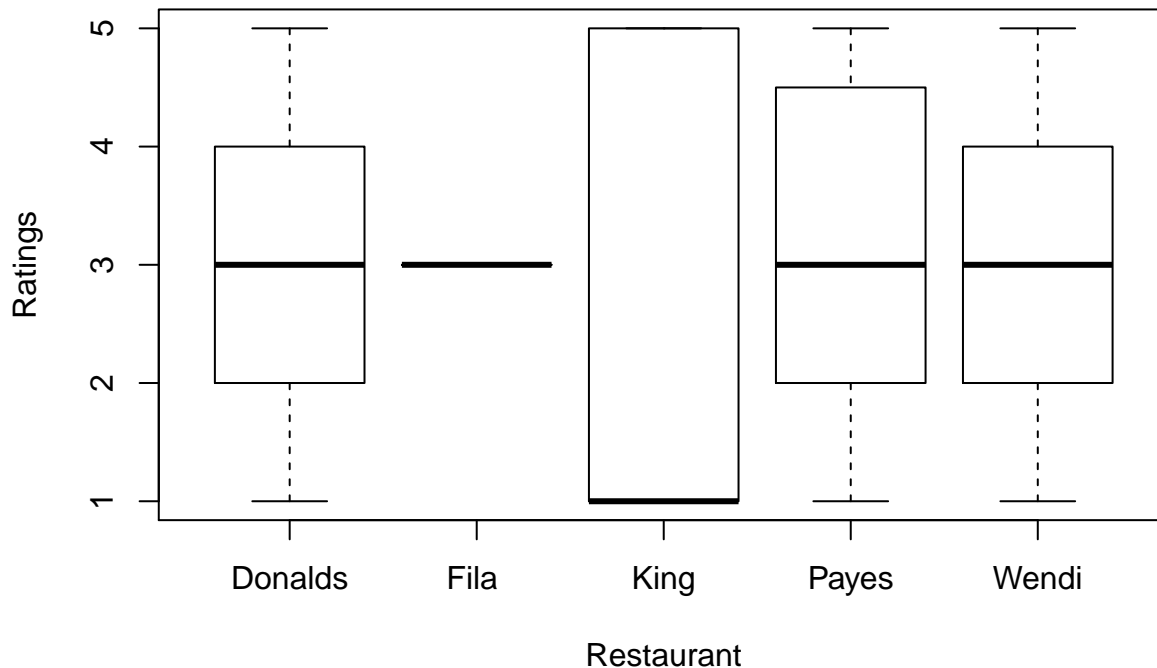
```
Mean_Rest<- Rest_Tab%>%
  group_by(Rest)%>%
  summarise(Mean_Rating=mean(Ratings))
Mean_Rest
```

```
## # A tibble: 5 x 2
##   Rest      Mean_Rating
##   <chr>      <dbl>
## 1 "Donalds"      3
## 2 "Fila "        3
## 3 "King"        2.78
## 4 "Payes"       3.18
## 5 "Wendi"       3
```

- (C) [4] Create a side-by-side boxplot of the ratings by Rest. [Note: be sure to include the output of your code.]

```
boxplot(Ratings~Rest,Rest_Tab, main="Ratings of Restaurants",
  xlab= "Restaurant", ylab="Ratings")
```

## Ratings of Restaurants



- (D) [5] Write a code to add a column (with column name Score) to add a column to ResRat that categorises the Ratings variable as follows: • "Poor" for customer rating 1 • "Meh" for customer rating 2 • "Okay" for customer rating 3 • "Good" for customer rating 4 • "Super" for customer rating 5 [Note: use the glimpse() function to show the resulting dataframe.]

```
Rest_With_Scores<- Rest_Tab %>%
  mutate(Scores = ifelse(grepl(1, Ratings), "Poor",
    ifelse(grepl(2, Ratings), "Meh",
      ifelse(grepl(3, Ratings), "Okay",
        ifelse(grepl(4, Ratings), "Good",
          ifelse(grepl(5, Ratings), "Super", "Other"))))))))

glimpse(Rest_With_Scores)
```

```
## Observations: 50
## Variables: 4
## $ Restaurant <fct> Donalds, Donald, D's, Donalds, Donalds, don, Donalds, Do...
## $ Ratings <int> 1, 2, 2, 3, 3, 3, 3, 4, 4, 5, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,...
## $ Rest <chr> "Donalds", "Donalds", "Donalds", "Donalds", "Donalds", "...
## $ Scores <chr> "Poor", "Meh", "Meh", "Okay", "Okay", "Okay", "Okay", "G...
```

- (E) [4] Write a code to extract and tabulate (count) the number customers with either a Poor or Super rating for each restaurant. [Note: be sure to include the output of your code.]

```
Rating_Tab<- Rest_With_Scores%>%
  filter(Scores == "Super" | Scores == "Poor")

Rating_Tab%>%
  group_by(Rest)%>%
  summarize(count=n())
```

```
## # A tibble: 4 x 2
```

```
## Rest count
## <chr> <int>
## 1 Donalds 2
## 2 King 9
## 3 Payes 5
## 4 Wendi 2
```