Project book

Group O Final Project

- Las Vegas: The Strip or Downtown? – Comparative Analysis of Restaurants in Old and New Las Vegas
- Members: Veronica Lee, Eileen (Yei Rim Suh)
- Published Github Page:(https://vl2354.github.io/LasVegas_Restaurants)

Overview and Motivation

There are two main districts that travelers to Las Vegas visit: the Downtown or Strip Las Vegas. The Downtown Las Vegas (DTLV) is also often called as Old Las Vegas. As the later name implies, it is the historic center of Las Vegas, Nevada. All the casinos and hotels are relatively close together, mostly in walking distance. They are also less expensive that those in the Strip Las Vegas as it requires less minimum bet for different table games. For this reason, DTLV is "for the budget minded and the serious gambler who do not like all the glitz [and] glamour of the Strip [Las Vegas]".

The Las Vegas Strip, on the other hand, features mega resorts that one often sees in television shows or movies. It has "Disney-like attractions and nothing less than 5 star hotels" All the famous casino and resorts like the Bellagio, Wynn, Venetian, and Mandalay Bay could be found in the Strip Las Vegas. Thus sightseers or less serious gamblers would prefer to stay on the Strip.

We are mainly interested in comparing Yelp reviews, particularly those related to restuarants, in DTLV and the Strip. We will use a vareity of different data visualization techniques in order to conduct conpartive analysis.

Research Questions

1. How restaurants are visually distributed in the map in Las Vegas?

2. How high/low rated restaurants in Old and New Las Vegas differ in terms of their business attributes?

3. Would the sentiment toward key business attribute differ in two regions, due to different preferences of the traveler groups in each region?

Data

We used the Yelp's challenge Round 9 academic dataset. This dataset has 4.1 million reviews by 1 million users for 144,000 businesses. It provides dataset for different cities like Edinburgh, U.K; Karlsruhe, Germany; Montreal and Waterloo, Canada; Pittsburgh, Charlotte, Urbana-Champaign, Phoenix, Las Vegas, Madison, Cleveland, USA. The dataset also includes 1.1 million business attributes like hours, parking availability, and ambience.

It consists of two datasets. The first is business dataset where it includes Name of business (with business ID), Categories, Attributes, Star ratings, and Review Count. The other is review Dataset which includes review text, star rating, and user ID.

For our project, we are only interested in the restaurant data in Las Vegas in the United States. Thus we plan to use subset of this large dataset.

Processing data:

- Download Yelp dataset from: https://www.yelp.com/dataset_challenge
- Download Zillow Nevada Shapefile dataset from: https://www.zillow.com/howto/api/neighborhood-boundaries.htm
- Download Census Tract level Cartographic Boundary Shapefiles from the U.S. Census Bureau from :https://www.census.gov/geo/maps-data/data/tiger-cart-boundary.html
- Create a data folder containing two folders named processed and raw_subset under the local repo
- Unzip Yelp dataset to get raw dataset in .json files in local direcotry
- Run src/data/process_raw_data.py

For businesses data set: - Run src/data/process_business_data.R - Run src/data/b_categories.R

For reviews data set: - Run src/data/subset_reviews.py - Run src/data/clean_review_data.R - Run src/data/summarise_review.R

Once we processed our data into raw_subset in .csv or .xls files, we used dplyr and plyr to wrangle data into the best structure for our further analysis directly using R scripts in each RMD files.

Data Analysis:

- Run src/attribute_unnest.R for unnest categories and attributes in nested .ujson format in Business data subset
- Run src/eda_progress.R for data wrangling/cleaning/creating a subset used for exploratory data analysis
- Run src/text_progress.R for text and sentiment analysis
- Run src/map_progress.R for spatial data wrangling and cleaning for geo-spatial analysis

All the source codes are accessible in our github repository: https://github.com/Columbia-University-QMSS/final-project-team-vegas To view full codebook, please refer to /src folder in above repo.

Prior to check our github repo for review, please read our README file accounting for project repo organization: https://github.com/Columbia-University-QMSS/final-project-team-vegas/blob/master/README.md

Process Steps:

Data Table

We created two data tables in D3 interactive format to show how our main data subsets look to the audience.

The Business dataset includes the geogrphic information of business such as state,neighborhood and coordinates,name of Business (with business ID), categories, star ratings, Review Count, more than 1.1 million business attributes such as hours, parking availability, and ambience, etc. We created a subset which sampled restaurant business of the city of Las Vegas in Nevada, U.S. from Business Dataset for our comparative analysis. Following Data Table gives a flavor of the business data used for our anlaysis:

Review Data contains Review text, Star ratings (out of 5), business ID, user ID, etc. We create a subset by sampling 10% of review data for restaurants in the city of Las Vegas of Nevada, U.S. from Review Dataset for our comparative analysis. Data Visualizaiton result is at: https://vl2354.github.io/LasVegas_Restaurants/Data.html

```
# Importing the data
restaurants <- read.csv("./data/lv_business_categories_matrix_v2.csv")
library(dplyr)
s_restaurants <- restaurants %>%
  group_by(neighborhood) %>%
  filter(neighborhood %in% c("The Strip", "Downtown")) %>% arrange(neighborhood)
```

```
s_restaurants_subset <- s_restaurants[,c(4,5,11,12,13)]

library(DT)
datatable(s_restaurants_subset)


review <- read.csv("./data/lv_restaurant_reviews_10.csv") # data 10 samples

review_summary_bid <- review %>% group_by(neighborhood, business_id, name) %>%
  filter(neighborhood %in% c("The Strip", "Downtown")) %>%
  summarise(
    name_ct = n()
    , avg_stars = mean(stars) ) %>% arrange(name)

review_summary <- review %>% group_by(neighborhood, name) %>%
  filter(neighborhood %in% c("The Strip", "Downtown")) %>%
  summarise(
    restaurant_ct = n()
    , avg_stars = mean(stars) ) %>% arrange(neighborhood)

# Renaming the columns
review_summary2 <- review_summary %>% rename("Restaurant Counts"=restaurant_ct,
                                             "Average Rating" = avg_stars,
                                             "Restaurant Name" = name,
                                             "Neighborhood" = neighborhood)
is.num <- sapply(review_summary2, is.numeric)
review_summary2[is.num] <- lapply(review_summary2[is.num], round, 2)


# Create a data table of count of reviews and avg stars per restaurant in DTLV & The Strip
#write.csv(review_summary, file = "review_summary.csv")
library(DT)
datatable(review_summary2) %>% DT::formatRound(columns="Average Rating", digits=1)
```

Exploratory Analysis

For exploratory analysis, we checked proportion of restaurant ratings by all neighborhood in Las Vegas, then particularly focused on two selected neighborhoods for our anlaysis, Downtown and the Strip.

We checked price range, rate range, review counts and attribute proportion by neighborhood level in Las Vegas in this part.

First, we wrangled business data se using below code:

```
library(tidyverse)
library(stringr)
library(ggplot2)
library(ggthemes)
library(readr)
library(plotly)
library(dplyr)


restaurants_count <- restaurants %>%
```

```
  group_by(neighborhood) %>%
  summarise(
    n = n()) %>%
  arrange(neighborhood)

s_restaurants_count <- restaurants %>%
  group_by(neighborhood) %>%
  summarise(
    n = n()
  ) %>%
  filter(neighborhood %in% c("The Strip", "Downtown"))

n1 <- unlist(unique(restaurants_count[, "neighborhood"]))
```

Then, we draw ggplot graphs for review counts and higlight our selected neighborhoodS: Downtown and the Strip to get this result: https://vl2354.github.io/LasVegas_Restaurants/EDA.html#restaurants_count_in_las_vegas

As labeling was confusing at beginning with original data field names, I replace them by wrangling the data.

```
# original graph
original <- ggplot(restaurants_count, aes(x = reorder(neighborhood, n), y = n)) +
  geom_col(width = 0.7, fill = "#c41200") +
  labs(x = "Neighborhood", y = "Number of Restaurants") +
  theme_tufte() +
  ggtitle("Restaurants per Neighborhood in Las Vegas") +
  coord_flip() +
  geom_text(aes(label = n), vjust = 0)

# Create a new "selected" Column, marks the Strip and DTLV in dummy variables (selected =1, not selecte
restaurants_count$selected <- ifelse(grepl("The Strip|Downtown",restaurants_count$neighborhood),1, 0)

# updated graph, highlighting selected cities in different colors, and show counts of restaurants per a
ggplot(restaurants_count, aes(x = reorder(neighborhood, n), y = n, fill = as.factor(selected))) +
  geom_col(width = 0.7) +
  labs(x = "Neighborhood", y = "Number of Restaurants") +
  theme_tufte() +
  ggtitle("Restaurants per Neighborhood in Las Vegas") +
  coord_flip() +
  geom_text(aes(label = n), vjust = 0) + theme(legend.position = "None")
```

For Restaurant Rating proportion visualized here: https://vl2354.github.io/LasVegas_Restaurants/EDA.html#proportion_of_ratings_by_neighborhood, I wranged the processed business data set and created D3 interactive plot.

```
restaurants_rating <- restaurants %>%
  group_by(neighborhood, stars) %>%
#  filter(neighborhood %in% c("The Strip", "Downtown")) %>%
  summarise(
    n = n()
  ) %>%
  spread(key = stars, value = n) %>%
  mutate(
    star1 = `1` + `1.5`,
    star2 = `2` + `2.5`,
```

```
    star3 = `3` + `3.5`,
    star4 = `4` + `4.5`,
    star5 = `5`
  ) %>%
  select(neighborhood, star1, star2, star3, star4, star5) %>%
  gather(star1, star2, star3, star4, star5, key = "star", value = "n")

# Omitting neighborhoods with misisng variables
restaurants_rating <- restaurants_rating %>% na.omit()

s_restaurants_rating <- restaurants_rating %>%
  group_by(neighborhood) %>%
  filter(neighborhood %in% c("The Strip", "Downtown"))

s_restaurants_rating <- s_restaurants_rating %>% na.omit()


# Renaming the columns to better account for values asked by an assignment Part 2 question
restaurants_rating <- restaurants_rating %>% rename(Rating=star)

library(plotly)

# distribution of restaurant ratings per neighborhood
g1 <- ggplot(restaurants_rating) +
  geom_bar(aes(x = neighborhood, y = n, fill = Rating), stat = "identity", position = "fill") +
  theme_tufte() +
  theme(axis.text.x = element_text(colour = "grey20", size = 11, angle = 45, hjust = 1,
                                   vjust = 1, face = "italic")) +
  ggtitle("Distribution of Restaurants Ratings per Neighborhood in Las Vegas") +
  labs(x = "Neighborhood", y = "Proportion")

ggplotly(g1)

g2 <- ggplot(s_restaurants_rating) +
  geom_bar(aes(x = neighborhood, y = n, fill = star), stat = "identity", position = "fill") +
  annotate("text", x = 'Downtown', y =  0.50, label = "Avg. Star=3.65", family="serif", fontface="italic
  annotate("text", x = 'The Strip', y =  0.50, label = "Avg. Star=3.32", family="serif", fontface="itali
  theme_tufte() +
  theme(axis.text.x = element_text(colour = "grey20", size = 11, angle = 45, hjust = 1,
                                   vjust = 1, face = "italic")) +
  ggtitle("Distribution of Restaurants Ratings in DTLV & the Strip") +
  labs(x = "Neighborhood", y = "Proportion")

ggplotly(g2)
```

I also conducted the analysis on review counts per neighborhood to see whether review base used for analyzing the consumer preferences in Downtown and the Strip is suitable. As a result, I found the Strip has a number of reviews, comparing to Downtown. As the Strip had more restaurant counts and total reviews, after this step, I decided to mark proportion of restaurants (%Restaurant) for comparison analysis on restaurant categories and attributes.

```
restaurants_review_bp <- restaurants %>%
  select(neighborhood, review_count) %>%
  group_by(neighborhood) %>%
```

```
  filter(neighborhood %in% n1) %>% arrange(neighborhood)

s_restaurants_review_bp <- restaurants %>%
  select(neighborhood, review_count) %>%
  group_by(neighborhood) %>%
  filter(neighborhood %in% c('The Strip','Downtown')) %>% arrange(neighborhood)
```

For attribute analysis, I did data wrangling to get price scale out of 4, rating scale out of 5, and proportion of restaurant in percentage in Downtown and the Strip, by utilizing restaurant business data subset and draw ggplots. Following is sampled codes giving a flavor on this step:

```
restaurants$price <- ifelse(grepl("-",restaurants$price, ignore.case = TRUE),0,as.numeric(restaurants$pr

restaurants_cuisine <- restaurants %>%
  group_by(neighborhood, categories) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

restaurants_alcohol <- restaurants %>%
  group_by(neighborhood, Alcohol) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

restaurants_ambience <- restaurants %>%
  group_by(neighborhood, Ambience) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

restaurants_attire <- restaurants %>%
  group_by(neighborhood, attire) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

restaurants_kids <- restaurants %>%
  group_by(neighborhood, kids) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

restaurants_meal <- restaurants %>%
  group_by(neighborhood, Meal) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

restaurants_noise <- restaurants %>%
  group_by(neighborhood, noise) %>%
 summarise(
 n = n(), avg_price=mean(price), avg_rate=mean(stars))

att.cuisine <- ggplot(restaurants_cuisine, aes(x = categories,
                                    y = as.numeric(n))) +   geom_bar(alpha = 0.3) +   geom_line(alpha =
  coord_flip() +   facet_wrap(~neighborhood, scales="free") +   theme_excel() +   ggtitle("Restaurant D

att.cuisine <- ggplot(restaurants_cuisine, aes(x = categories, y = as.numeric(n))) + geom_bar(alpha = 0
  geom_line(alpha = 0.3) +   coord_flip() +   facet_wrap(~neighborhood, scales="free") +   theme_excel()
```

```
  ggtitle("Restaurant Distribution by Cuisine in Downtown and the Strip")

restaurants_alcohol_dt <- restaurants_alcohol %>% filter(neighborhood == 'Downtown')
restaurants_alcohol_strp <- restaurants_alcohol %>% filter(neighborhood == 'The Strip')

restaurants_Ambience_dt <- restaurants_ambience %>% filter(neighborhood == 'Downtown')
restaurants_Ambience_strp <- restaurants_ambience %>% filter(neighborhood == 'The Strip')

restaurants_Meal_dt <- restaurants_meal %>% filter(neighborhood == 'Downtown')
restaurants_Meal_strp <- restaurants_meal %>% filter(neighborhood == 'The Strip')

restaurants_attire_dt <- restaurants_attire %>% filter(neighborhood == 'Downtown')
restaurants_attire_strp <- restaurants_attire %>% filter(neighborhood == 'The Strip')

restaurants_noise_dt <- restaurants_noise %>% filter(neighborhood == 'Downtown')
restaurants_noise_strp <- restaurants_noise %>% filter(neighborhood == 'The Strip')

restaurants_kids_dt <- restaurants_kids %>% filter(neighborhood == 'Downtown')
restaurants_kids_strp <- restaurants_kids %>% filter(neighborhood == 'The Strip')

att.cuisine <- ggplot(restaurants_cuisine, aes(x = price, y = as.numeric(n))) +   geom_bar(alpha = 0.3)
  geom_line(alpha = 0.3) +   coord_flip() +   facet_wrap(~neighborhood, scales="free") +   theme_excel()
  ggtitle("Restaurant Distribution by Cuisine in Downtown and the Strip")
```

Text/Sentiment Analysis

The following is the last data wrangling process we went through for text and sentient analysis.

```
# bring in the final dataset that was cleaned using python
## this dataset contains business attribute files
data <- read.csv("final_data.csv")
# bring in the final review dataset that was cleaned using python
## this dataset contains text review files, business ids, and star reviews
review <- read.csv('data_r.csv', fileEncoding = "latin1")

# exclude non-ASCII texts
review$text <- gsub("[^\x20-\x7E]", "", review$text)

# merge attribute data frame and review text file
total <- merge(data, review, by = "business_id", all = FALSE)

# only keep unique texts (delete duplicated ones)
total_u <- distinct(total, text, .keep_all = TRUE)

# choose only downtown and The Strip
total_s <- total_u %>% filter(neighborhood == "Downtown" | neighborhood == "The Strip")
```

The Corpus cleaning process that we then conducted is pretty much same as what we did in class.

```
##### clean text #####
total_s$text <- as.character(total_s$text)
str(total_s$text)
text <- as.data.frame(total_s$text)
```

```
df_source <- DataframeSource(text[1])
df_corpus <- VCorpus(df_source)
df_corpus

# check the contents
df_corpus[[1]][1]
df_corpus[[13312]][1]

# Text cleaning
df_corpus <- tm_map(df_corpus, content_transformer(tolower))
df_corpus <- tm_map(df_corpus, content_transformer(removeWords), c(stopwords("english")))
# df_corpus <- tm_map(df_corpus, content_transformer(removeWords), c("list", "w/"))
df_corpus <- tm_map(df_corpus, content_transformer(removeNumbers))
df_corpus <- tm_map(df_corpus, content_transformer(removePunctuation))
# df_corpus <- tm_map(df_corpus, content_transformer(replace_abbreviation))
# df_corpus <- tm_map(df_corpus, replace_symbol)
# delete non-english characters

## stem document
df_corpus_stem <- tm_map(df_corpus, stemDocument)
df_corpus_stem <- tm_map(df_corpus_stem, stripWhitespace)

# check stemmed document
df_corpus_stem
df_corpus_stem[[1]][1]
df_corpus_stem[[2]][1]

# givr row names to data frame to match neighbhorhood in tdm and dtm
total_s$row.names <- 1:nrow(total_s)

# dtm and tdm
tdm <-TermDocumentMatrix(df_corpus_stem)
tdm_td <- tidy(tdm)
tdm_td$neighborhood <- total_s[match(tdm_td[['document']], total_s[['row.names']]), 'neighborhood']
head(tdm_td)

dtm <- DocumentTermMatrix(df_corpus_stem)

dtm_td <- tidy(dtm)
head(dtm_td)
dtm_td$neighborhood <- total_s[match(dtm_td[['document']], total_s[['row.names']]), 'neighborhood']
head(dtm_td)
```

The following is the code for the data wrangling process we went through for the sentiment analysis. The main challenge here was to merge two datasets.

```
pos <- read.table("positive-words.txt", as.is=T)
neg <- read.table("negative-words.txt", as.is=T)

library(quanteda)

sentiment <- function(words=c("really great good stuff bad")){
  require(quanteda)
  tok <- quanteda::tokenize(words)
```

```
  pos.count <- sum(tok[[1]]%in%pos[,1])
  neg.count <- sum(tok[[1]]%in%neg[,1])
  out <- (pos.count - neg.count)/(pos.count+neg.count)
  return(out)
}

Strip <- total_s %>% filter(neighborhood == "The Strip")
Downtown <- total_s %>% filter(neighborhood == "Downtown")

Sent_s <- data.frame(matrix(0, ncol = 1, nrow = nrow(Strip)))
colnames(Sent_s)[1] <- "sent"
Sent_s$text <- Strip$text
Sent_s$star <- Strip$stars

Sent_d <- data.frame(matrix(0, ncol = 1, nrow = nrow(Downtown)))
Sent_d$text <- Downtown$text
colnames(Sent_d)[1] <- "sent"
Sent_d$star <- Downtown$stars

for(i in 1:nrow(Strip)) {
  Sent_s[[i, 1]] <- sentiment(Strip[[i, 495]])
}

for(i in 1:nrow(Downtown)) {
  Sent_d[[i,1]] <- sentiment(Downtown[[i, 495]])
}

sentiment(Strip[[7, 495]])
sentiment(Downtown$text)

Sent_d$neigh <- "Downtown"
Sent_d$neigh <- factor(Sent_d$neigh)
Sent_s$neigh <- "The Strip"
Sent_s$neigh <- factor(Sent_s$neigh)
```

Also, Many exploratory text analysis was done on the text files.

```
# The bar graph of frquent words used in review of each neighbor
# frequent terms in general
tdm_td %>% group_by(term) %>%
  summarise(n = sum(count)) %>%
  arrange(desc(n)) %>%
  top_n(n = 20, wt = n)  %>%
  mutate(term = reorder(term, n)) %>%
  ggplot(aes(term, n)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) +  coord_flip() + theme_tufte() +
  ggtitle("The Words Most Frequently Used in Both Neighborhood") +
  theme(plot.title = element_text(lineheight=.8, face="bold")) +
  ylab("Term Frequency")

# Bind the TF,DF, and IDF frequency
# of a tidy text dataset to the dataset
tf_idf <- tdm_td %>%
```

```
  bind_tf_idf(term, document, count)  %>%
  arrange(desc(tf_idf))

tf_idf

tf_idf2 <- tdm_td %>%
  bind_tf_idf(term, document, count)  %>%
  arrange(desc(tf))

# frequent words by used in reviews of each neighborhood
### The Strip
#### by tf
tf_idf2 %>% filter(neighborhood=="The Strip") %>%
  arrange(desc(tf)) %>%
  top_n(n = 10, wt = tf)  %>%
  ggplot(aes(x = term, y = tf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("Term Frequency") + coord_flip() + theme_tufte() +
  ggtitle("Top 10 Frequent Words Used in Reviews of The Strip") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))

#### by tf_idf
tf_idf %>% filter(neighborhood=="The Strip") %>%
  top_n(n = 10, wt = tf_idf)  %>%
  ggplot(aes(x = reorder(term, tf_idf), y = tf_idf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("TF-IDF") + coord_flip() + theme_tufte() +
  ggtitle("The Frequent Words Used in The Strip") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))


### Downtown
#### by tf
tf_idf2 %>% filter(neighborhood=="Downtown") %>%
  #arrange(desc(tf)) %>%
  top_n(n = 10, wt = tf)  %>%
  ggplot(aes(x = term, y = tf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("Term Frequency") + coord_flip() + theme_tufte() +
  ggtitle("Top 10 Frequent Words Used in Reviews of Downtown") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))

#### by tf_idf
tf_idf %>% filter(neighborhood=="Downtown") %>%
  arrange(desc(tf_idf)) %>%
  top_n(n = 10, wt = tf_idf)  %>%
  ggplot(aes(x = reorder(term, tf_idf), y = tf_idf)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  xlab(NULL) + ylab("TF-IDF") + coord_flip() + theme_tufte() +
  ggtitle("The Frequent Words Used in Downtown") +
  theme(plot.title = element_text(lineheight=.8, face="bold"))

# comparison
```

```
plot <- tf_idf %>%
  group_by(neighborhood) %>%
  top_n(n = 10, wt = tf_idf) %>%
  mutate(key = 1:10)
plot %>% ggplot() +
  geom_bar(aes(x = term))

######################## wordcloud ######################

# wordcloud by neighborhood
library(wordcloud)
# Create purple_orange
purple_orange <- brewer.pal(10, "PuOr")
# Drop 2 faintest colors
purple_orange <- purple_orange[-(1:2)]

## The Strip
term_frequency_s <- tf_idf %>% filter(neighborhood=="The Strip")
set.seed(100)

# Create a wordcloud for the review in The Strip
wordcloud(term_frequency_s$term, term_frequency_s$tf,
          max.words = 200,
          colors= purple_orange)

### includes words "overpriced"

## Downtown
term_frequency_DT <- tf_idf %>% filter(neighborhood=="Downtown")
set.seed(213)

# Create a wordcloud for the review in Downtown
wordcloud(term_frequency_DT$term, term_frequency_DT$tf,
          max.words = 100, colors = purple_orange)

#### includes the world "cheap"
```

Geo-Spatial Map Analysis

Based on exploratory data analysis and textual anallysis done above, we anted to built a map showing all of our visual insights of the Las Vegas restaurants on a map. To do so, we found the census tract level shape file from U.S. Census Bureau and neighborhood level shape file of Nevada from Zillow. As both shape files provided the entire state of Nevada's spatial information within their data sets, we did a data wrangling by creating shapefile subsets only account for the city of Las Vegas. We selected neighborhood level shapefile against census tract level shapefile, as this version of visualization not only described the restaurant distribution better, but also clearly stressed Downtown and the Strip, two selected neighborhoods of our research interests.

Originally, we provided static ggmap plots of restaurant distribution in our [presentation slide] (https://github.com/Columbia-University-QMSS/final-project-team-vegas/blob/master/Presentation_Slide_Old%20%26%20New%20LV%20restaurants%20comparative%20analysis.pdf), but we later converted this to ineractive leaflet map to give more freedome to observe to the audience. This leaflet map can be viewed at our published website's "Map" menu: https://vl2354.github.io/LasVegas_Restaurants/Map.html

```
library(maps)
library(sp)
library(leaflet)

#load Nevada County Level Shape File
NV_z <- readOGR("./shape/ZillowNeighborhoods-NV","ZillowNeighborhoods-NV", verbose = FALSE) #Ziller Nei

#Subset a Las Vegas map by neighborhood-level from above
LV_z <- subset(NV_z, NV_z$City %in% c("Las Vegas"))
#Subset a Vegas map by neighborhood-level from above
two_z <- subset(LV_z, LV_z$Name %in% c("The Strip", "Downtown"))
two_z@data$r_count <- ifelse(two_z@data$Name == "The Strip", 818, 340)
two_z@data$avg_rating <- ifelse(two_z@data$Name == "The Strip", '3.32/5.00', '3.65/5.00')

# Load Las Vegas Business Data Frame
lv_b <- readr::read_csv("C:/Users/Veronica/Desktop/Spring 2017/Data Visualization 4063/final/LV_bis.csv

# Add restaurant scatter as circles on a map
# Set Palette
library(RColorBrewer)
pal = colorFactor("RdYlGn", domain = lv_b$stars)
color_rating <- pal(lv_b$stars)

# base map
m2 <- leaflet(LV_z) %>% setView(-115.14,36.16,10) %>% addTiles() %>%
  addPolygons(data = LV_z, color = "#444444", weight = 1, smoothFactor = 0.5,
              opacity = 1.0, fillOpacity = 0.5,
              popup = paste("<b>Neighborhood:</b>",LV_z$Name, "<br/>"),
              highlightOptions = highlightOptions(color = "white", weight = 2,
                                                  bringToFront = F)) %>%
  addPolygons(data = two_z, color = "#c41200", weight = 1, smoothFactor = 1,
              opacity = 1, fillOpacity = 0.3,
              popup = paste("<b>Neighborhood:</b>",two_z$Name, "<br/>",
                            "<b>Restaurant Count:</b>",two_z$r_count,"<br/>",
                            "<b>Avg. Rating:</b>",two_z$avg_rating,"<br/>"),
              highlightOptions = highlightOptions(color = "white", weight = 2,
                                                  bringToFront = FALSE))
# add locate me button

m2 <- m2 %>%
  addEasyButton(easyButton(
    icon="fa-globe", title="Zoom to Entire World Map",
    onClick=JS("function(btn, map){ map.setZoom(1.5); }"))) %>%
  addEasyButton(easyButton(
    icon="fa-crosshairs", title="Locate Me",
    onClick=JS("function(btn, map){ map.locate({setView: true}); }")))

#Add Business Attribute Layers

m2 <- m2 %>%  # First Data Layer: Rating
  addCircles(group="Rating",data = lv_b, lng = ~longitude, lat = ~latitude, col = color_rating,
                popup = paste("<b>Neighborhood:</b>",lv_b$neighborhood, "<br/>",
                              "<b>Name:</b>",lv_b$name, "<br/>",
```

```
                              "<b>Rating:</b>",lv_b$stars,"<br/>",
                              "<b>Price:</b>",lv_b$price_range,"<br/>")) %>%
addCircles(group="Price",data = lv_b, lng = ~longitude, lat = ~latitude, col = color_price,
               popup = paste("<b>Neighborhood:</b>",lv_b$neighborhood, "<br/>",
                              "<b>Name:</b>",lv_b$name, "<br/>",
                              "<b>Rating:</b>",lv_b$stars,"<br/>",
                              "<b>Price:</b>",lv_b$price_range,"<br/>")) %>%
# Layers control
addLayersControl(
  baseGroups = c("OpenStreetMap"),
  overlayGroups = c("Rating","Price"),
  options = layersControlOptions(collapsed = TRUE) ) %>%
  addLegend("bottomright",
          pal = colorNumeric("RdYlGn", domain = lv_b$stars), values = ~lv_b$stars,
          title = "Rate Range<br>in Las Vegas", opacity = 0.5) %>%
  addLegend("bottomright",
          pal = colorFactor("RdYlGn", domain = lv_b$price), values = ~lv_b$price,
          title = "Price Range<br>in Las Vegas", opacity = 0.5)
```

Evaluation

1 More spatial clusters of restaurants in New Vegas
2. High-End, expensive restaurants are preferred and exist more in New Vegas
3. Overall sentiment score for two regions are similarly distributed and somewhat popular
4. However, consumers in the Strip are more likely to get negative sentiment in low rated restaurants.
5. Casual restaurants are welcomed in both areas
6. Up-scaled/romantic restaurants are welcomed in both areas

All in all, consumers in Old Vegas are more budget-minded than those of New Vegas Consumers. Consumer in New Vegas are more likely to be the traveler for specific events, funs, and traveling.