An Examination of Bangalore Restaurants

**Gil Graybill – Final Project, August 7, 2021**

**BAN 530, Professor Scott**

# Overview

Bangalore is a fast-growing, culturally diverse city in Karnataka, India with a population of over 12 million people. It is ethnically diverse, with 4 different languages spoken by more than 10% of the population (Kannada – 38%, Tamil – 28%, Telugu -17%, Urdu – 13%, <https://worldpopulationreview.com/world-cities/bangalore-population> ). Much of its growth is due to migration for tech jobs, as it is known as the Silicon Valley of India (<https://www.aalpha.net/articles/why-bangalore-is-called-silicon-valley-of-india/>) . This influx of people and cultures has led to an interesting phenomena: a vibrant and growing restaurant scene that seemingly can’t keep up with demand.

The Zomato dataset can help us make sense of the current restaurant state in Bangalore. The Zomato dataset consists of over 50,000 rows of data of Bangalore area restaurants, their location, cuisine type, ratings, the number of votes to get the ratings, the type of restaurant, favorite dishes, approximate cost, the neighborhood it’s located in, and online reviews. By analyzing this dataset, we hope to understand what restaurants are popular, why they are popular, and provide valuable insight for entrepreneurial individuals who wish to get a foothold in this thriving market.

# Dataset Discoveries

While there are over 50,000 entries in the dataset, that does not mean that there are 50,000 restaurants. If the restaurant is classified as more than one type (Dine-in, Delivery, Desserts, Cafes, etc.), it is listed multiple times in the dataset.

There are 30 different neighborhoods specified, with the number of restaurants ranging from 696 (New BEL Road) to 3062 (BTM). This will provide excellent insight as to what cuisine types are more or less popular in what neighborhoods. This brings up a good question to ponder: If an area has a lower demand for a certain cuisine, is there an under-served market there? Or is the demand lower for reasons not reflected in the data, hence not a good candidate for bringing that cuisine there?

There are 108 different categories of Cuisine listed, with many restaurants serving more than one type of cuisine. The Cuisine type can be regional (North Indian, Chinese, Biryani), food-specific (Pizza, Burger, Ice Cream), or a food style (Fast Food, Desserts, Bakery, Street Food). The leaders in this category are North Indian (21083), Chinese (15546), South Indian (8642), and Fast Food (8095). 30 cuisines are represented more than 1000 times, and 50 of the cuisines are listed less than 100 times.

Analyzing the dataset by “name” allows us to get an idea of the number of restaurant entities that exist in the dataset, and gives us an idea of the existence of chains. There are 8792 unique ones, and most of the restaurants listed are larger chains. Café Coffee Day leads the way with 96 entries. 36 restaurant chains have 50 or more entries. 7558 entities have 10 or less entries. It may be notable that Starbucks has “only” 39 locations for a population of 12 million people, which is the same number of locations within a mile of my house in Raleigh.

The field “rate” shows the average rating from scraping data from the restaurant website, and “votes” is the number of ratings received. We will keep these numbers in mind as we try to determine who serves the best food, remembering that a larger number of votes will yield a rate that is more likely to be accurate.  
The field “approx\_cost” shows the cost for two people to dine there. By itself this field may not be significant, but if combined with rating, it could indicate a correlation between cost and satisfaction. If a high-priced meal does not meet high expectations, that is certainly a red flag. Likewise, if expectations are low because of price and those expectations are exceeded, this could be an indication of a restaurant that is at the top of its game.

The “reviews\_list” field could be examined with Text Analytics to determine the vibe or attitude of the customers who chose to leave reviews.

There are Boolean fields for online ordering and “book\_table”, which sounds like the ability to make a reservation.

# Issues and Problems with the dataset

When initially loading the data, Excel believes the CSV file has over 1000 columns, instead of 15. Some rows are empty, and some rows are from reviews that have run amuck. It appears that the scraping of the reviews has created these issues, which is a standard problem for text that can be unstructured. This will have to be cleaned up, and determinations made about how much effort will be put into preserving this data. There is also the standard “missing data” when a field shows up as “NA”. There are various strategies for dealing with these rows of data, and hopefully we can still find useful information from rows.

The fields “dish\_liked” and “cuisines” can have comma-separated lists in them, which does not fit the definition of “tidy” data. At some point during the analysis these fields will have to be broken up and separate rows of data created for each one.

Querying the dataset to find the number of unique locations is problematic. Getting a count by using the street address gives over 11,000 unique locations. However, further examination of those results show that the address listed may be for the “headquarters”, as the same listing/data shows up in multiple neighborhoods. If further analysis is needed in this area, a series of operations may need to be performed to get the data in the proper format.

The “menu\_item” field appears to be sparsely populated and may not be very useful for analytical purposes.

As mentioned earlier “reviews\_list” may contain a lot of information, but it’s unknown how useful it will be, even if we can clean it up. If the list of reviews are in an expected “tidy” format, it can be analyzed and trends can be looked for.

The rating is a text field and will need to be converted to numerical if any analysis will need to be done with it.

# Proposed Project Scope

My project proposal is based on the following premises:

1. There is an unfulfilled demand for quality restaurant food in the Bangalore area.
2. The ratings of the restaurants in the Zomato dataset are indicative of how all of its customers feel about it. Higher rated restaurants do better business.
3. Opening a new restaurant is a high-risk proposal.
4. Not all knowledge needed to determine if a restaurant should be opened can be obtained from analyzing the Zomato dataset.

## The Story

My client, Betty Bigbucks, has recently sold her hi-tech company in Bangalore. From running her company, she has seen the “mad-dash” at lunch time by her employees. She tried arranging to have food trucks come by on certain days, but with the demand so high, many times the food trucks just wouldn’t show up. Even when they did show up the novelty of the trucks diminished after a few months, even after including employees in the selection process. She has also seen the issues at night when her employees who stay late try to order delivery. While some of the food arrived in a timely manner, some would take a long time, and some wouldn’t make it there at all, which leads to low morale for those who are sacrificing their evenings for her company. She is now flush with cash and believes there is an opportunity not only improve the food offerings in Bangalore, but also to make some money in the process.

Betty would like to be an “angel investor” for one or more up-and-coming restaurants and take them to the next level, and beyond. First, she’d like to identify local restaurants that are small (3 or fewer physical locations) whose ratings are higher than average for the cuisines that they serve. This list would be a starting point for her to visit the restaurants and talk with the owners about their willingness to expand into other neighborhoods. This list will be referred to as the “Ganesh List”, named for the Hindu god of beginnings.

Which neighborhoods would they expand into? We could find neighborhoods where the number of restaurants for a given cuisine seem low, but that may be because of cultural factors and local preferences. Rather than find the “untapped market”, she prefers to go into a neighborhood that has an established demand for a cuisine but where the restaurants that currently serve the population have low customer satisfaction ratings. She feels that a fresh new face with a food that’s in demand is the key to success. With the current demand for food being very high, she knows she could just flood the market with quick, cheap food, but she is also looking to the future. At some point the market will have to be saturated, and then quality will win out. She has placed no limit on the number of restaurants to open but wants to start with one or two from the Ganesh list and go to those neighborhoods that are perceived to have the highest demand for a new choice. In her wildest dreams, she sees one or more of her chosen restaurants being a respected chain first across Bangalore, then nationwide.

The additional variable to look at is the cost of the meals. Restaurants on the Ganesh list should have a competitive price point with the restaurants in neighborhoods they are trying to compete with.

To keep start-up costs low, she feels we should concentrate our efforts on Delivery and Take-out, which means a smaller footprint for the restaurant. Future options could include food trucks and dine-in, but that is only if a location proves to be wildly successful and there is a perceived demand for it.

# Analytical Methods

## Regression Analysis

We’re putting a lot of emphasis on the rating for each restaurant, which was obtained by scraping data from websites. Is there a correlation between rating and any of the other variables? Are more expensive restaurants rated higher? Do the number of reviews have anything to do with it? While we know that all cuisines probably do not have the same overall ratings, can we detect if the cuisine type affects the ratings? Are people in certain neighborhoods “tough graders” or easy to please?

Betty hopes that the ratings are guided by factors that she can control, like price, cuisine type, neighborhood, and restaurant type. If it’s more guided by votes then the ratings are utterly useless and we’ll have to find some other yardstick to determine where to open restaurants and which members of the Ganesh list to use.

First we’ll answer the question of whether or not chain restaurants get higher or lower ratings. This will be done with some simple linear regression.

Next we’ll do a simple multiple linear regression with our predictor being “rating” and the quantitative variables of votes and meal cost. Those variables will be put in a new dataset and run against R’s “ggpairs” to give us both visual and numerical representation of correlation between the variables.

Next we’ll add in the categorical variables of cuisine, neighborhood, and restaurant type, and plot that against ratings. Forward and backward stepwise regression should prove helpful here to eliminate any unnecessary variables from the equation. We’ll see if r-squared is a respectable value and look for multicollinearity and overfitting.

We’ll split the data into training and testing datasets, and run the same tests again to make sure we’re not overfitting the data. Then we’ll run k-fold cross-validation to make sure nothing gets left out.

## Predictive Model

Most restaurant owners, including Betty, hope that their published ratings are determined by customer reviews, and those opinions are guided by the quality of the food. A good restauranteur adds their own special something to try and make their food stand out.

However, there are a lot of restaurants who are not as concerned about quality. They just keep slinging their hash and get their money and provide a necessary service for the customers of Bangalore.

Since we are focused on ratings, it would be nice to develop a model to predict what the rating of a new restaurant would be based on the inputs of price, location, cuisine, and restaurant type. It would not be perfect, since hopefully the quality of the food will have something to do with that. But, if cheap food of cuisine X in neighborhood Y has a high chance of a certain rating, that could be useful information. When a new restaurant comes into a neighborhood, Betty can just look at a menu for the price and cuisine and get a prediction about the potential rating. Maybe they’ll surprise, and maybe they’ll disappoint, but it’s a data-driven starting point.

Betty could also use this for her own restaurants. She could find out what the predicted rating is for any restaurant she opens, and they see if she is able to surpass it after a certain time period (one year?). If she goes over, her methodology is working. If not, then she’s just another face in the crowd.

We can use the “predict” method with our regression models. Models will be trained using training data with a 70/30 split. We’ll expand our models to include various types of Logistical Regression (K-fold and Threshold), Neural Networks, Random Forests (with Ensemble methods), and Hierarchical clustering. With all methods we’ll look at their respective values for accuracy (R-squared, AIC, ROC, glm, etc.).

## Prescriptive Model

Betty becomes a household name, all-knowing about the restaurant scene in Bangalore! People ask her for guides for the budget conscious consumer. We’ll use the “solver” add-in in Excel to answer questions like "For x money, find the five highest rated meals in y neighborhood for z cuisine. Linear programming at it’s finest.

## The Process

Most of this will be done in R, using the methodologies used in BAN 502, MIS 504, and MIS 506, the remainder will be done in Excel.

# Data Exploration using R Markdown

## Library Setup, Data Cleaning, and Loading Data

The dataset was contained in a file zomato\_1.csv. The following was done to clean the data:

1. Using Excel, the first 18 columns were copied, which eliminated many unneeded dirty columns from the data.
2. All blank rows were removed.
3. All rows with a URL value that did not start with “https” were removed.
4. Columns that were awkwardly named were renamed.
5. Unicode characters in the “name” column were removed.

library(readr)  
library(dplyr)  
library(tidyr)  
library(tidyverse)

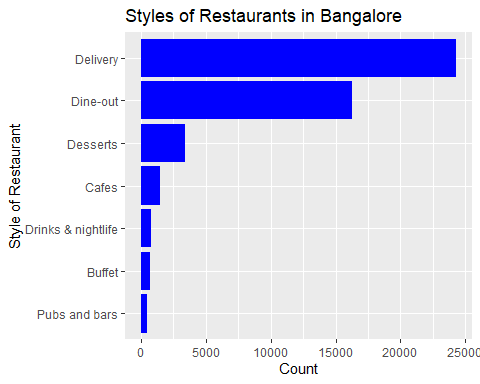
## Warning: package 'ggplot2' was built under R version 4.0.5

library(ggplot2)  
options(scipen = 999)  
zomato\_1\_cleaned <- read\_csv("zomato-1\_cleaned.csv")

# clean the data, rename columns, select the desired columns, and remove Unicode characters from the name.   
  
zomato\_1\_cleaned <- rename(zomato\_1\_cleaned,   
 cost\_for\_two = `approx\_cost(for two people)`,  
 restaurant\_style = `listed\_in(type)`,  
 neighborhood = `listed\_in(city)`)  
zomato\_1\_selected <- dplyr::select(zomato\_1\_cleaned, url, name, rate, votes, location,   
 rest\_type, cuisines, cost\_for\_two, restaurant\_style, neighborhood)  
zomato\_1\_selected$name <- iconv(zomato\_1\_selected$name, "latin1", "ASCII", sub="")

## Restaurant Styles

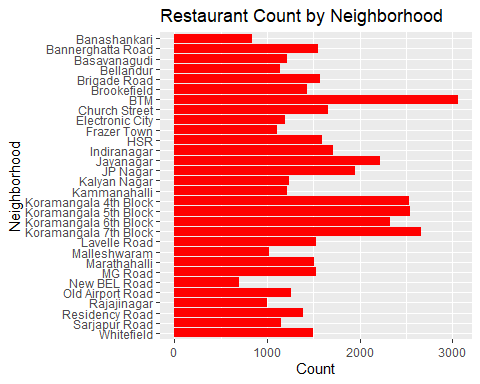
# Break down the columns  
dining\_type <- zomato\_1\_selected %>% count(restaurant\_style) %>% drop\_na(restaurant\_style)  
p1 <- ggplot(dining\_type, aes(x = n, y = reorder(restaurant\_style, n))) +   
 geom\_col(fill = "blue")  
p1 + labs(title = "Styles of Restaurants in Bangalore", x = "Count", y = "Style of Restaurant")



We can see that a majority of restaurants are Carry-out or Delivery, which is exactly what Betty was hoping for. Startup costs can be kept low by keeping the footprint of the restaurant small. It’s easy to do both out of one location.

## Count of Restaurants by Neighborhood

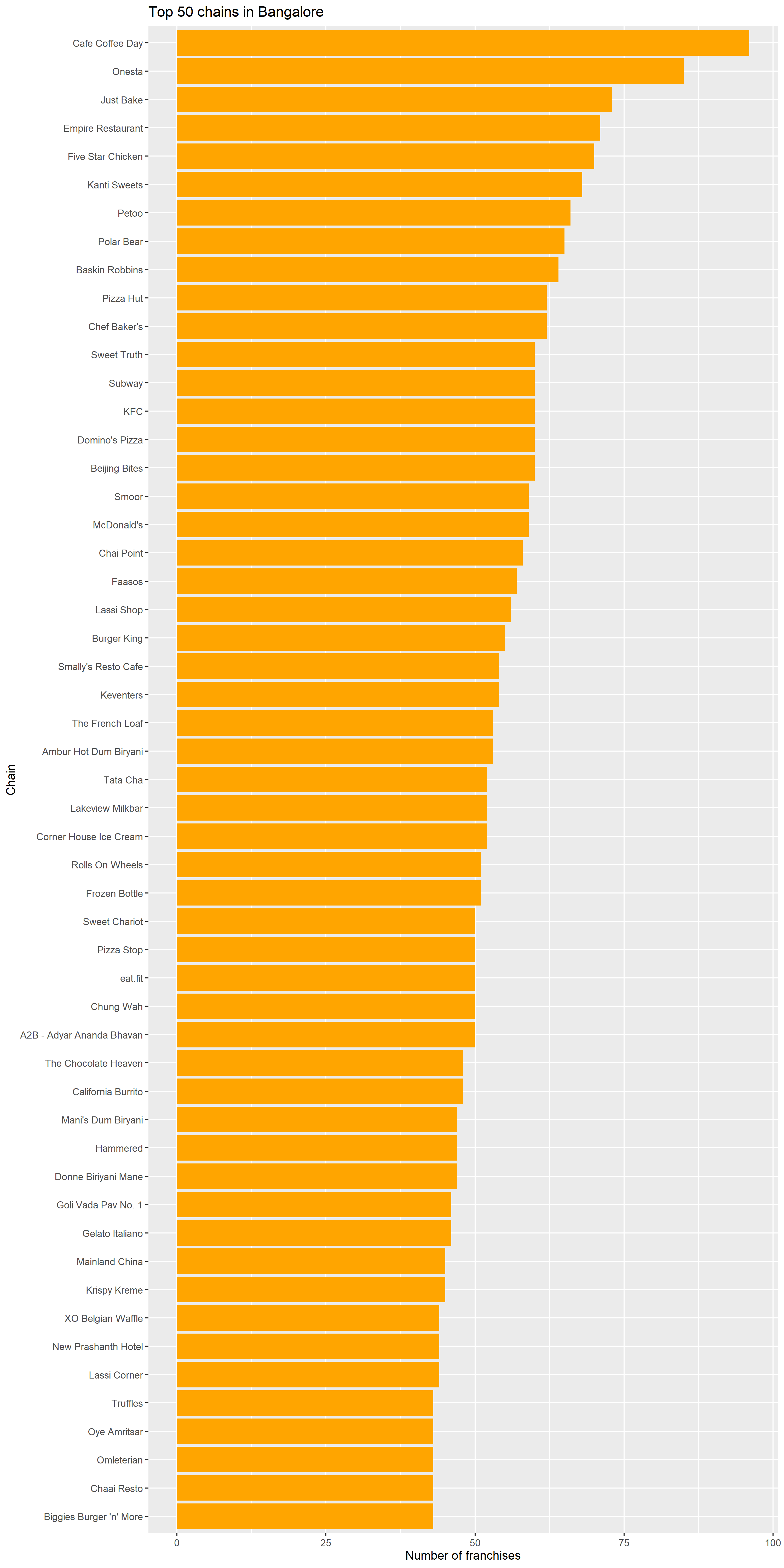
city\_list <- zomato\_1\_selected %>% count(neighborhood) %>% drop\_na(neighborhood)  
city\_list2 <- transform(city\_list, neighborhood = reorder(neighborhood, order(neighborhood, decreasing = TRUE)))  
p2 <- ggplot(city\_list2, aes(x = n, y = neighborhood)) +   
 geom\_col(fill = "red")  
p2 + labs(title = "Restaurant Count by Neighborhood", x = "Count", y = "Neighborhood")



There are 30 different neighborhoods, and each of them have enough restaurants such that we can determine what the dining trends are and how we can capitalize on them. The opportunities may exist where there are lots of restaurants, or it may be where there are fewer restaurants.

## Chain Restaurant Analysis

# restaurant\_type <- zomato\_1\_selected %>% count(rest\_type) # I don't like this one  
chains <- zomato\_1\_selected %>% count(name, sort = TRUE) # list the restaurants and # of franchises  
chains\_50 <- chains[order(chains$n), ] %>%  
 top\_n(50)  
chains\_50$name <- factor(chains\_50$name, levels = chains\_50$name)  
p3 <- ggplot(chains\_50, aes(x = n, y = name)) +   
 geom\_col(fill = "orange", width = 0.9) +   
 labs(title = "Top 50 chains in Bangalore", x = "Number of franchises", y = "Chain")  
  
ggsave(p3, filename = "50\_chains.png", height = 20, width = 10)  
knitr::include\_graphics("50\_chains.png")

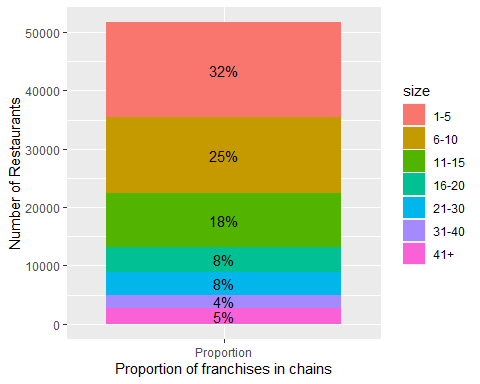


chains\_count <- chains %>% count(n, sort = TRUE)

It is useful to see what the largest chains are in Bangalore. It gives us an idea of what the saturation is for franchises. It looks like franchising has a higher prevalence of Western chains. Is India ready for some more Indian franchises? It doesn’t look like franchises dominate the landscape. Let’s go a little deeper into that.

chains\_count <- chains %>% count(n, sort = TRUE)   
chains\_count <- rename(chains\_count, franchises\_in\_chain = n, instances = nn)  
chains\_count$total <- chains\_count$franchises\_in\_chain \* chains\_count$instances

chain\_total <- sum(chains\_count$total)  
chain\_to\_graph <- data.frame(size = c("1-5", "6-10", "11-15",   
 "16-20", "21-30", "31-40", "41+"),  
 total = c(sum(chains\_count$total[1:5]),   
 sum(chains\_count$total[6:10]),  
 sum(chains\_count$total[11:15]),  
 sum(chains\_count$total[16:20]),  
 sum(chains\_count$total[21:30]),  
 sum(chains\_count$total[31:40]),  
 sum(chains\_count$total[41:69])),  
 chain\_prop = c(sum(chains\_count$total[1:5])/chain\_total\*100,   
 sum(chains\_count$total[6:10])/chain\_total\*100,  
 sum(chains\_count$total[11:15])/chain\_total\*100,  
 sum(chains\_count$total[16:20])/chain\_total\*100,  
 sum(chains\_count$total[21:30])/chain\_total\*100,  
 sum(chains\_count$total[31:40])/chain\_total\*100,  
 sum(chains\_count$total[41:69])/chain\_total\*100))  
  
chain\_to\_graph <- chain\_to\_graph %>%  
 mutate(Style = "Proportion")  
  
chain\_to\_graph$size <- factor(chain\_to\_graph$size, levels = chain\_to\_graph$size)  
  
ggplot(chain\_to\_graph, aes(x = Style, y = total, fill = size)) +  
 geom\_col() +  
 geom\_text(aes(label = paste0(round(chain\_prop), "%")),  
 position = position\_stack(vjust = 0.5)) +  
# theme\_minimal(base\_size = 16) +  
 ylab("Number of Restaurants") +  
 xlab("Proportion of franchises in chains")

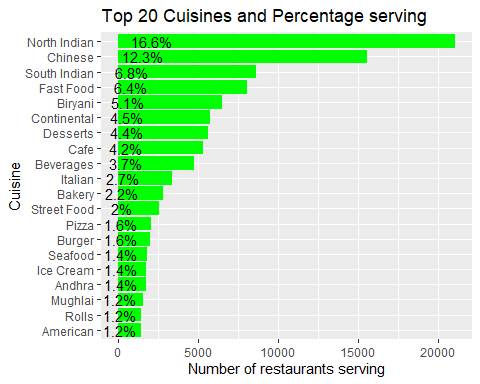


Numerically, more than half of the restaurants in Bangalore are in chains of less than 10 franchises, and almost a third are in chains of 5 or less franchises. Large chains make up a very small proportion of the restaurants. Note: the restuarant listings in the dataset may include multiple listing for the same location if they serve more than one style (i.e. Dine-out, Delivery). Hence the reason for using the cutoff number of 5.

For this example we had to create a dataframe using the chains data to create a stacked bar graph with customized categories.

## Cuisine Distribution

cuisines\_list <- zomato\_1\_selected %>%   
 separate\_rows(cuisines, sep = ",") %>%   
 dplyr::select(cuisines)  
  
# How many of each cuisine  
cuisines\_list$cuisine <- trimws(cuisines\_list$cuisines, which = c("both"))  
cuisine\_count <- cuisines\_list %>%   
 count(cuisine, sort = TRUE)  
  
cuisine\_total <- sum(cuisine\_count$n)  
cuisine\_count$percent <- round(cuisine\_count$n/cuisine\_total\*100,1)  
  
cuisine\_20 <- cuisine\_count[order(cuisine\_count$n), ] %>%  
 top\_n(20)  
cuisine\_20$cuisine <- factor(cuisine\_20$cuisine, levels = cuisine\_20$cuisine)  
p4 <- ggplot(cuisine\_20, aes(x = n, y = cuisine)) +   
 geom\_col(fill = "green", width = 0.9) +   
 geom\_text(aes(label = paste0(round(percent, 2), "%")),  
 position = position\_stack(vjust = 0.1)) +  
 labs(title = "Top 20 Cuisines and Percentage serving", x = "Number of restaurants serving", y = "Cuisine")  
p4



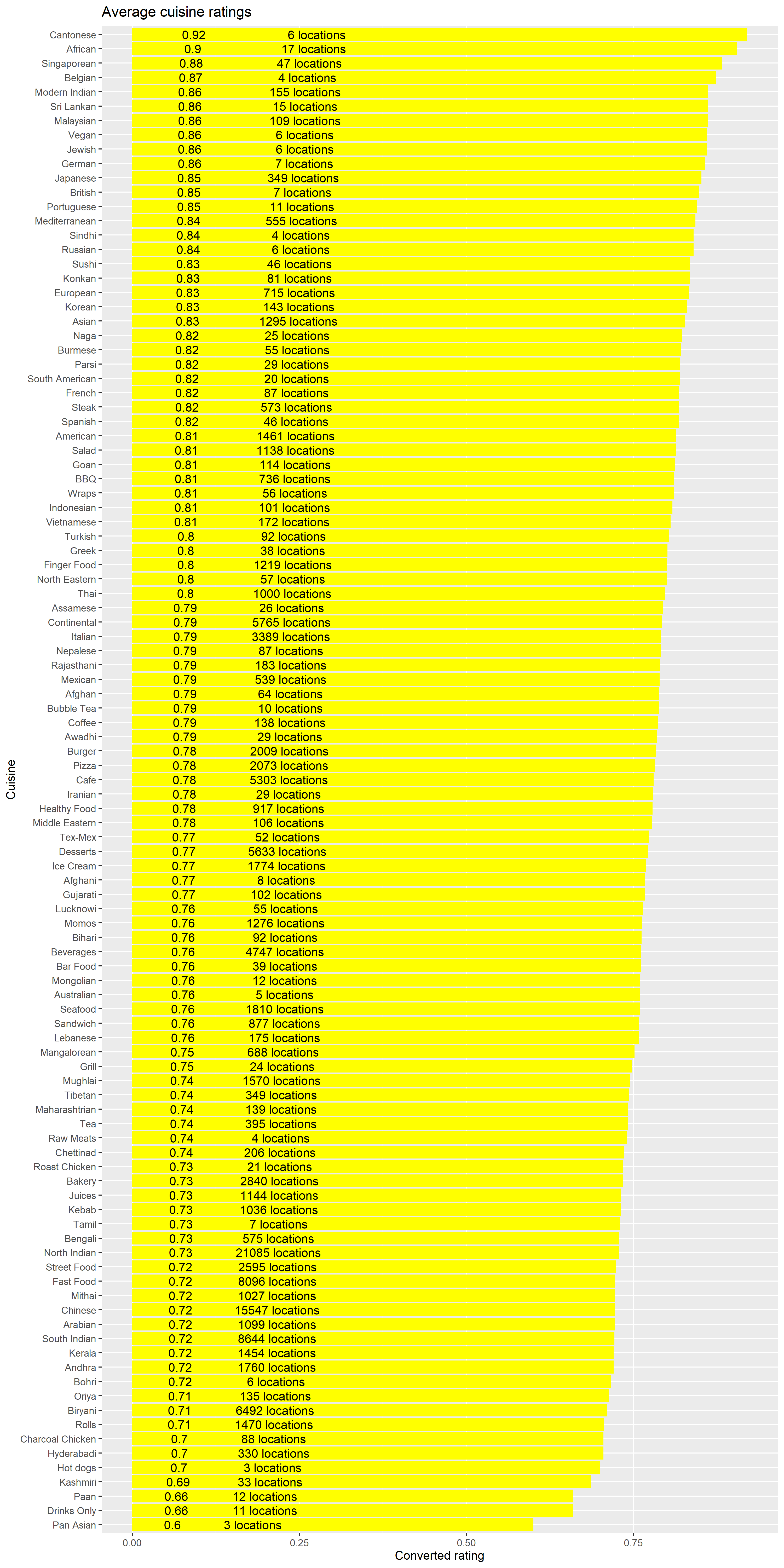
There are 108 different cuisines. Over 20,000 restaurants serve North Indian cuisine, with Chinese coming in at 15,000. South Indian and Fast Food are both around 8000, with a good handful of others coming in above 5000. We’ll look at all cuisines to see which ones are vulnerable in a neighborhood. Note that many restaurants serve more than one type of cuisine.

To get this information we had to parse out the “cuisines” column and create a separate row for each to have normalized data. We will now be able to do more in-depth analysis using the individual cuisines.

# Per cuisine, what's the average rating  
# First, separate the cuisines and create a new row for each  
zomato\_1\_tidy\_cuisines <- zomato\_1\_selected %>%   
 separate\_rows(cuisines, sep = ",")   
  
# Gotta do it this way or it will crash. Trim whitespace from cuisines, then drop NAs  
NEW=TRUE  
zomato\_1\_tidy\_cuisines$cuisine <- trimws(zomato\_1\_tidy\_cuisines$cuisines, which = c("both"))  
zomato\_1\_tidy\_cuisines <- zomato\_1\_tidy\_cuisines %>% drop\_na(rate) %>% drop\_na(cuisine)  
  
# The rating is a character string. Painstakingly convert it to a number  
zomato\_1\_tidy\_cuisines$new\_rate <- data.frame(do.call('rbind', strsplit(as.character(zomato\_1\_tidy\_cuisines$rate),'/',fixed=TRUE)))  
zomato\_1\_tidy\_cuisines = zomato\_1\_tidy\_cuisines %>%   
 mutate(num\_rate = as.numeric(new\_rate$X1)/as.numeric(new\_rate$X2)) %>%  
 dplyr::select(-new\_rate)

## Warning in mask$eval\_all\_mutate(quo): NAs introduced by coercion  
  
## Warning in mask$eval\_all\_mutate(quo): NAs introduced by coercion

cuisine\_rating <- group\_by(zomato\_1\_tidy\_cuisines, cuisine) %>%  
 summarize(cuisine\_mean = mean(num\_rate, na.rm=TRUE)) %>%  
 drop\_na(cuisine\_mean)  
  
cuisine\_rating$cuisine <- factor(cuisine\_rating$cuisine, levels = cuisine\_rating$cuisine)  
cuisine\_rating <- cuisine\_rating %>%   
 mutate(restaurant\_count = cuisine\_count$n[match(cuisine, cuisine\_count$cuisine)])  
  
  
p5 <- ggplot(cuisine\_rating, aes(x = cuisine\_mean, y = reorder(cuisine, cuisine\_mean))) +   
 geom\_col(fill = "yellow", width = 0.9) +   
 geom\_text(aes(label = paste0(round(cuisine\_mean, 2), "")),  
 position = position\_stack(vjust = 0.1)) +  
 geom\_text(aes(label = paste0(round(restaurant\_count, 2), " locations")),  
 position = position\_stack(vjust = 0.3)) +  
 labs(title = "Average cuisine ratings", x = "Converted rating", y = "Cuisine")  
  
ggsave(p5, filename = "cuisine\_rating.png", height = 20, width = 10)  
knitr::include\_graphics("cuisine\_rating.png")



Some cuisines are ranked better than others. Do we try to take over Bangalore with a good version of a low ranked cuisine, like hot dogs or Pan Asian? If we do we run the risk of opening up a restaurant with a cuisine that is not in demand. Instead, we will look for a neighborhoods where the ratings for a cuisine is below what it should be. We’ve added the number of locations so we can see if that cuisine is popular or not.

To get this data, the “rating” character string needed to be converted to number so mathematical operations could be performed on it.

## Find candidates for Ganesh List

zomato\_1\_carry\_delivery <- zomato\_1\_tidy\_cuisines %>%  
 filter((restaurant\_style == "Dine-out") | (restaurant\_style == "Delivery")) %>%  
 drop\_na(num\_rate)  
  
avg\_rating\_cuisine\_by\_neighborhood <- zomato\_1\_carry\_delivery %>% group\_by(neighborhood, cuisine) %>%   
 filter(n() > 4 ) %>%  
 summarise(avg\_rating = mean(num\_rate, na.rm=TRUE))  
  
# If more than 5 instances of a cuisine are in a neighborhood, get the average rating. This tells us that the cuisine is at least liked in the neighborhood.   
  
small\_chains <- chains %>% filter(n <=5 )  
small\_chains\_carry\_deliver <- subset(zomato\_1\_carry\_delivery, name %in% small\_chains$name)  
# Don't be fooled by multiple cuisines  
  
small\_chains\_carry\_deliver <- small\_chains\_carry\_deliver %>%  
 left\_join(avg\_rating\_cuisine\_by\_neighborhood, c("cuisine" = "cuisine",   
 "neighborhood" = "neighborhood"))  
  
small\_chains\_carry\_deliver$rating\_diff <-   
 small\_chains\_carry\_deliver$num\_rate -   
 small\_chains\_carry\_deliver$avg\_rating  
  
small\_chains\_carry\_deliver\_display <- small\_chains\_carry\_deliver %>%   
 dplyr::select(name, votes, neighborhood, cuisine, num\_rate, rating\_diff, cost\_for\_two, restaurant\_style)  
  
small\_chains\_carry\_deliver\_display[order(-small\_chains\_carry\_deliver\_display$rating\_diff), ] %>% top\_n(100, rating\_diff)

## # A tibble: 100 x 8  
## name votes neighborhood cuisine num\_rate rating\_diff cost\_for\_two  
## <chr> <dbl> <chr> <chr> <dbl> <dbl> <dbl>  
## 1 The Big Ba~ 544 Electronic City Chinese 0.94 0.250 1500  
## 2 The Big Ba~ 544 Electronic City North I~ 0.94 0.246 1500  
## 3 TBC Sky Lo~ 6745 HSR North I~ 0.94 0.230 1000  
## 4 Sant Spa C~ 246 Old Airport Road Healthy~ 0.98 0.228 1000  
## 5 Brahmin's ~ 2679 Basavanagudi South I~ 0.96 0.226 100  
## 6 Hunger Camp 311 Bannerghatta Ro~ Chinese 0.92 0.226 1300  
## 7 Oota Banga~ 478 Whitefield South I~ 0.92 0.220 1700  
## 8 Hunger Camp 311 Bannerghatta Ro~ North I~ 0.92 0.219 1300  
## 9 Oota Banga~ 463 Brookefield South I~ 0.92 0.217 1700  
## 10 ECHOES Kor~ 3214 Koramangala 4th~ North I~ 0.94 0.215 750  
## # ... with 90 more rows, and 1 more variable: restaurant\_style <chr>

Who are the restaurants who are going to make it on the list? The process we used to find them is the following:

1. Create a table for the average rating for the restaurants for a given cuisine in a given neighborhood, if there are more than 5 restaurants. I feel if there are at least 5 restaurants serving a given cuisine in a neighborhood that at least that cuisine is deemed acceptable for people there, and that there’s a demand for that type of cuisine there.
2. Take the “chains” dataframe, and find the chains that have 5 or less franchises. This gives us a list of all the “small” chains. Filter out by “Dine-out” and “Deliver”
3. Join the two tables by cuisine and neighborhood, which adds the average rating for each neighborhood/cuisine for each restaurant. That makes it easy to subtract to get the difference.
4. Sort by the difference. This shows us which small chains are rated higher (or lower) than the average for that cuisine/area. This gives us our tentative list. Next we’ll need to join that list with the average rating for cuisine/neighborhood and see which combination of small restaurant/neighborhood bubble to the top.

## Final Results: Which restaurants to open where

neighborhoods\_possible <- small\_chains\_carry\_deliver\_display %>%   
 left\_join(avg\_rating\_cuisine\_by\_neighborhood, by = c("cuisine" = "cuisine")) %>%  
 mutate(diff\_neighborhood\_avg = num\_rate - avg\_rating) %>%  
 filter(diff\_neighborhood\_avg > 0.15)  
  
neighborhoods\_possible\_filtered <- neighborhoods\_possible %>%  
 dplyr::select(name, cuisine, num\_rate, neighborhood.y, diff\_neighborhood\_avg, cost\_for\_two, restaurant\_style ) %>%  
 unique()  
  
neighborhoods\_possible\_filtered[order(-neighborhoods\_possible\_filtered$diff\_neighborhood\_avg), ] %>% top\_n(100, diff\_neighborhood\_avg)

## # A tibble: 101 x 7  
## name cuisine num\_rate neighborhood.y diff\_neighborhoo~ cost\_for\_two  
## <chr> <chr> <dbl> <chr> <dbl> <dbl>  
## 1 Sant Spa C~ Healthy ~ 0.98 Rajajinagar 0.37 1000  
## 2 Sant Spa C~ Healthy ~ 0.98 Electronic City 0.343 1000  
## 3 Sant Spa C~ Salad 0.98 Marathahalli 0.307 1000  
## 4 Sant Spa C~ Healthy ~ 0.98 Malleshwaram 0.283 1000  
## 5 Brahmin's ~ South In~ 0.96 Electronic City 0.276 100  
## 6 Sant Spa C~ Salad 0.98 Bannerghatta R~ 0.271 1000  
## 7 Dhyaana Healthy ~ 0.88 Rajajinagar 0.27 600  
## 8 Dhyaana Healthy ~ 0.88 Rajajinagar 0.27 600  
## 9 Sant Spa C~ Salad 0.98 Electronic City 0.267 1000  
## 10 Brahmin's ~ South In~ 0.96 Marathahalli 0.261 100  
## # ... with 91 more rows, and 1 more variable: restaurant\_style <chr>

# gugray  
# write.csv(neighborhoods\_possible\_filtered, "restaurants\_to\_create.csv")

Now we put it all together, joining the data from the Ganesh List and the neighborhood/cuisine average ratings by cuisine.

1. Join the data
2. We calculate the difference between the ratings of a restaurant on the Ganesh List (high) and the average rating for that cuisine in a specific neighborhood.
3. Only use those entries where the difference is more than 0.15 (filter)
4. Some chains have multiple entries on the Ganesh List, which causes the same name to be listed more than once. Filter out the neighborhood where the restaurant is located.
5. Sort results by the difference.

The results can be interpreted by reading a row as follows (using row 1 as an example):

“Sant Spa Cuisine, which serves Healthy Food cuisine and has a rating of 0.98, should expand into Rajajinagar, since they have a rating that is 0.37 higher than the average Healthy Food cuisine there”.

We now have a list of small restaurants with high ratings, and what neighborhoods would be most welcoming of their expansion. Betty now has some restaurants to visit, some owners to charm, and some real estate to acquire.

# Linear Regression, Cross Validation and Sampling using R Markdown

## Library Setup, Data Cleaning, and Loading Data

library(readr)  
library(dplyr)  
library(tidyr)  
library(tidyverse)  
library(ggplot2)  
options(scipen = 999)  
zomato\_1\_cleaned <- read\_csv("zomato-1\_cleaned.csv")  
zomato\_1\_cleaned <- rename(zomato\_1\_cleaned,   
 cost\_for\_two = `approx\_cost(for two people)`,  
 restaurant\_style = `listed\_in(type)`,  
 neighborhood = `listed\_in(city)`)

# clean the data, rename columns, select the desired columns, and remove Unicode characters from the name.   
  
zomato\_1\_selected <- dplyr::select(zomato\_1\_cleaned, url, name, rate, votes, location,   
 rest\_type, cuisines, cost\_for\_two, restaurant\_style, neighborhood)  
zomato\_1\_selected$name <- iconv(zomato\_1\_selected$name, "latin1", "ASCII", sub="")

## Simple Linear Regression of Ratings vs. Chains

We’ve been talking about finding a great small restaurant and turning it into a chain. A question that should be asked (and easily answered) is if there is a correlation (positive or negative) between ratings and restaurant chains. Do consumers like the small restaurants more, or do they seek the comfort of food they know and love?

We determine the average rating for each chain, and the number of restaurants in the chain. Then we run a simple linear regression to see if there is a relationship.

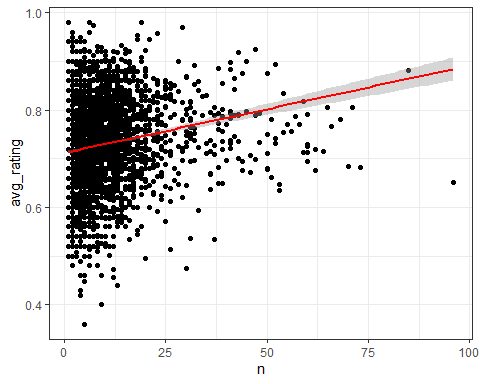
chains <- zomato\_1\_selected %>%   
 dplyr::select(name, rate)  
chains$new\_rate <- data.frame(do.call('rbind', strsplit(as.character(chains$rate),'/',fixed=TRUE)))  
chains <- chains %>%   
 mutate(num\_rate = as.numeric(new\_rate$X1)/as.numeric(new\_rate$X2)) %>%  
 dplyr::select(-new\_rate, -rate)

## Warning in mask$eval\_all\_mutate(quo): NAs introduced by coercion  
  
## Warning in mask$eval\_all\_mutate(quo): NAs introduced by coercion

chains\_count <- chains %>% count(name)  
  
chains2 <- chains %>%  
 drop\_na(num\_rate) %>%  
 group\_by(name) %>%  
 summarise\_at(vars(num\_rate), list(mean)) %>%  
 rename(avg\_rating = num\_rate)  
  
chains3 <- chains2 %>%  
 left\_join(chains\_count, c("name" = "name"))  
  
mod1 = lm(formula = avg\_rating ~ n, data = chains3)  
summary(mod1)

##   
## Call:  
## lm(formula = avg\_rating ~ n, data = chains3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.36078 -0.05540 0.00281 0.05602 0.26640   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.7118118 0.0013305 535.01 <0.0000000000000002 \*\*\*  
## n 0.0017929 0.0001343 13.35 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07997 on 6619 degrees of freedom  
## Multiple R-squared: 0.02622, Adjusted R-squared: 0.02607   
## F-statistic: 178.2 on 1 and 6619 DF, p-value: < 0.00000000000000022

ggplot(chains3,aes(x=n,y=avg\_rating)) + geom\_point() + geom\_smooth(method = "lm", color = "red") + theme\_bw()



The line on the graph shows a slight upward trajectory, but this is somewhat illusionary. Looking at the grouping of the data points suggests that as a chain gets bigger, the rating for it tends to settle toward the middle. Note the distribution of the points is in a triagular shape, converging towards the mean as the number of franchises gets larger. This is backed up by the calculation of the value for R-Squared, which is 0.02622, which is pretty low. So, it doesn’t look like creating a chain is inherently a bad idea. Good to know. Moving on.

## More cleaning of data

library(GGally)

## Warning: package 'GGally' was built under R version 4.0.5

# Convert the rate to a number  
zomato\_1\_selected$new\_rate <- data.frame(do.call('rbind', strsplit(as.character(zomato\_1\_selected$rate),'/',fixed=TRUE)))  
zomato\_1\_selected = zomato\_1\_selected %>%   
 mutate(num\_rate = as.numeric(new\_rate$X1)/as.numeric(new\_rate$X2)) %>%  
 dplyr::select(-new\_rate)

## Warning in mask$eval\_all\_mutate(quo): NAs introduced by coercion

## Warning in mask$eval\_all\_mutate(quo): NAs introduced by coercion

# break up the cuisines  
zomato\_1\_selected <- zomato\_1\_selected %>%   
 separate\_rows(cuisines, sep = ",")   
  
# Gotta do it this way or it will crash. Trim whitespace from cuisines, then drop NAs  
NEW=TRUE  
zomato\_1\_selected$cuisine <- trimws(zomato\_1\_selected$cuisines, which = c("both"))  
zomato\_1\_selected <- zomato\_1\_selected %>% drop\_na(rate) %>% drop\_na(cuisine)

## Simple Linear regression of Ratings vs votes and cost

We only have two other quantitative values in our dataset: the cost for two people to eat there, and the number of votes the restaurant has received. Intiutively, one would hope that the votes don’t determine the ratings and that they are a natural offshoot of the quality of the food. Let’s check it out.

ggpairs(zomato\_1\_selected, columns = c("votes", "cost\_for\_two", "num\_rate"))

## Warning in ggally\_statistic(data = data, mapping = mapping, na.rm = na.rm, :  
## Removed 615 rows containing missing values

## Warning in ggally\_statistic(data = data, mapping = mapping, na.rm = na.rm, :  
## Removed 5419 rows containing missing values

## Warning: Removed 615 rows containing missing values (geom\_point).

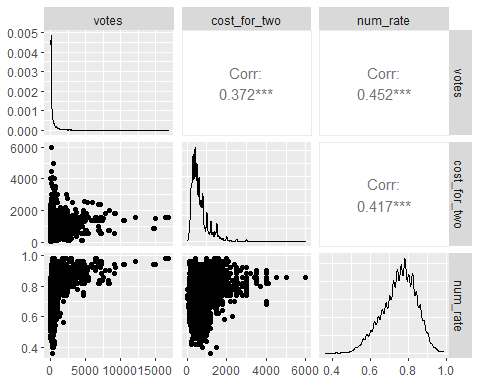
## Warning: Removed 615 rows containing non-finite values (stat\_density).

## Warning in ggally\_statistic(data = data, mapping = mapping, na.rm = na.rm, :  
## Removed 6024 rows containing missing values

## Warning: Removed 5419 rows containing missing values (geom\_point).

## Warning: Removed 6024 rows containing missing values (geom\_point).

## Warning: Removed 5419 rows containing non-finite values (stat\_density).



mod2 = lm(num\_rate ~ votes + cost\_for\_two, zomato\_1\_selected)  
summary(mod2)

##   
## Call:  
## lm(formula = num\_rate ~ votes + cost\_for\_two, data = zomato\_1\_selected)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.41228 -0.04158 0.01151 0.05178 0.25263   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.6992485732 0.0004070127 1718.0 <0.0000000000000002 \*\*\*  
## votes 0.0000300055 0.0000002445 122.7 <0.0000000000000002 \*\*\*  
## cost\_for\_two 0.0000552376 0.0000005402 102.3 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07665 on 105859 degrees of freedom  
## (6024 observations deleted due to missingness)  
## Multiple R-squared: 0.2766, Adjusted R-squared: 0.2766   
## F-statistic: 2.024e+04 on 2 and 105859 DF, p-value: < 0.00000000000000022

# Interaction effect  
mod3 = lm(num\_rate ~ votes \* cost\_for\_two, zomato\_1\_selected)  
summary(mod3)

##   
## Call:  
## lm(formula = num\_rate ~ votes \* cost\_for\_two, data = zomato\_1\_selected)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.41689 -0.03878 0.01187 0.05020 0.25299   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.6903666617746 0.0004352380510 1586.18  
## votes 0.0000600254489 0.0000006154124 97.54  
## cost\_for\_two 0.0000665481208 0.0000005742680 115.88  
## votes:cost\_for\_two -0.0000000253250 0.0000000004776 -53.02  
## Pr(>|t|)   
## (Intercept) <0.0000000000000002 \*\*\*  
## votes <0.0000000000000002 \*\*\*  
## cost\_for\_two <0.0000000000000002 \*\*\*  
## votes:cost\_for\_two <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07566 on 105858 degrees of freedom  
## (6024 observations deleted due to missingness)  
## Multiple R-squared: 0.2953, Adjusted R-squared: 0.2953   
## F-statistic: 1.479e+04 on 3 and 105858 DF, p-value: < 0.00000000000000022

Using R’s ggpairs function, we can see there is some positive correlation between these variables, but not a significant amount. R-squared is 0.295, which is not great.

* votes and num\_rate have correlation of 0.45 . The better your food, the more people will review it. But this correlation is not that high, which is instinctively correct. Giving a restaurant more reviews doesn’t make the food better.
* num\_rate and cost\_for\_two have a correlation of 0.417. One would hope that the more expensive the food is the better it is. But raising the price doesn’t make the food better.
* cost\_for\_two and votes have a correlation of 0.372. The more expensive it is, the more likely people are to vote for it. But again, it’s not a high correlation.

## Linear Regression with categorical variables - Simple

Now we can add some categorical variables and see if the addition of any of those will contribute to the rating. We’ll be adding cuisine, restaurant\_style, and neighborhood. We’ve already seen that there are differences in ratings between the different cuisines and the neighborhood. But can we tell how much predictive value they actually have? We’ll create factors for the categorical variables and then run a simple linear regression on them, along with our quantitative variables.

**Please note that for brevity and readability, the results of our regression analysis will have the values for the categorical values removed. It will be noted where the editing is done with “…SNIP…” . The source code will remain, allowing for easy reproducibility if needed.**

library(MASS)

## Warning: package 'MASS' was built under R version 4.0.5

# run num\_rate vs cost\_for\_two, num\_rate, cuisine, restaurant\_style, and neighborhood  
zomato\_1\_regress = zomato\_1\_selected %>%   
 mutate(cuisine, as\_factor(cuisine)) %>%  
 mutate(restaurant\_style, as\_factor(restaurant\_style)) %>%  
 mutate(neighborhood, as\_factor(neighborhood))  
  
zomato\_1\_regress <- subset(zomato\_1\_regress, select = c(cost\_for\_two, num\_rate, cuisine, restaurant\_style, neighborhood))  
zomato\_1\_regress\_omit <- na.omit(zomato\_1\_regress)  
  
allmod = lm(num\_rate ~ cost\_for\_two + restaurant\_style + cuisine + neighborhood, zomato\_1\_regress\_omit)  
summary(allmod)

##   
## Call:  
## lm(formula = num\_rate ~ cost\_for\_two + restaurant\_style + cuisine +   
## neighborhood, data = zomato\_1\_regress\_omit)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.37968 -0.04395 0.00929 0.05266 0.25426   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.7358691086 0.0112774084 65.252  
## cost\_for\_two 0.0000622514 0.0000007113 87.519  
## restaurant\_styleCafes 0.0026163111 0.0021882841 1.196

**“…SNIP…”**

## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07508 on 91898 degrees of freedom  
## Multiple R-squared: 0.2017, Adjusted R-squared: 0.2004   
## F-statistic: 165.8 on 140 and 91898 DF, p-value: < 0.00000000000000022

The big takeaway from this is that R-Squared is 0.20, so the addition of these categorical variables actually made the predictive value go down. So as a group, the quantitative variables don’t add predictive value according to simple linear regression model.

## Forward and Backward Stepwise Regression

We’ll use the “allmod” regression model from above, create the empty model, and run forward and backward stepwise regression to add/remove variables to see if any of them are unneeded to try and predict num\_rate.

# create empty model  
emptymod = lm(num\_rate ~1, zomato\_1\_regress\_omit)  
summary(emptymod)

##   
## Call:  
## lm(formula = num\_rate ~ 1, data = zomato\_1\_regress\_omit)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.33844 -0.05844 0.00156 0.06156 0.24156   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.7384391 0.0002768 2668 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.08396 on 92038 degrees of freedom

### Forward first

# forward Stepwise  
forwardmod = stepAIC(emptymod, direction = "forward", scope=list(upper=allmod,lower=emptymod),  
 trace = TRUE)

## Start: AIC=-456034.9  
## num\_rate ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + cost\_for\_two 1 76.843 571.97 -467635  
## + cuisine 104 65.865 582.95 -465679  
## + restaurant\_style 6 20.954 627.86 -459044  
## + neighborhood 29 20.392 628.42 -458916  
## <none> 648.81 -456035  
##   
## Step: AIC=-467635.1  
## num\_rate ~ cost\_for\_two  
##   
## Df Sum of Sq RSS AIC  
## + cuisine 104 41.287 530.68 -474323  
## + neighborhood 29 14.536 557.43 -469946  
## + restaurant\_style 6 8.657 563.31 -469027  
## <none> 571.97 -467635  
##   
## Step: AIC=-474322.9  
## num\_rate ~ cost\_for\_two + cuisine  
##   
## Df Sum of Sq RSS AIC  
## + neighborhood 29 11.5964 519.09 -476298  
## + restaurant\_style 6 1.1819 529.50 -474516  
## <none> 530.68 -474323  
##   
## Step: AIC=-476298.4  
## num\_rate ~ cost\_for\_two + cuisine + neighborhood  
##   
## Df Sum of Sq RSS AIC  
## + restaurant\_style 6 1.11 517.98 -476483  
## <none> 519.09 -476298  
##   
## Step: AIC=-476483.5  
## num\_rate ~ cost\_for\_two + cuisine + neighborhood + restaurant\_style

summary(forwardmod)

##   
## Call:  
## lm(formula = num\_rate ~ cost\_for\_two + cuisine + neighborhood +   
## restaurant\_style, data = zomato\_1\_regress\_omit)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.37968 -0.04395 0.00929 0.05266 0.25426   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.7358691086 0.0112774084 65.252  
## cost\_for\_two 0.0000622514 0.0000007113 87.519  
## cuisineAfghani -0.0428889291 0.0287472087 -1.492

**“…SNIP…”**

## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07508 on 91898 degrees of freedom  
## Multiple R-squared: 0.2017, Adjusted R-squared: 0.2004   
## F-statistic: 165.8 on 140 and 91898 DF, p-value: < 0.00000000000000022

### Now Backward

# backward stepwise  
backmod = stepAIC(allmod, direction = "backward", trace = TRUE)

## Start: AIC=-476483.5  
## num\_rate ~ cost\_for\_two + restaurant\_style + cuisine + neighborhood  
##   
## Df Sum of Sq RSS AIC  
## <none> 517.98 -476483  
## - restaurant\_style 6 1.110 519.09 -476298  
## - neighborhood 29 11.525 529.50 -474516  
## - cuisine 104 31.386 549.36 -471277  
## - cost\_for\_two 1 43.172 561.15 -469117

summary(backmod)

##   
## Call:  
## lm(formula = num\_rate ~ cost\_for\_two + restaurant\_style + cuisine +   
## neighborhood, data = zomato\_1\_regress\_omit)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.37968 -0.04395 0.00929 0.05266 0.25426   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.7358691086 0.0112774084 65.252  
## cost\_for\_two 0.0000622514 0.0000007113 87.519  
## restaurant\_styleCafes 0.0026163111 0.0021882841 1.196

**“…SNIP…”**

## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07508 on 91898 degrees of freedom  
## Multiple R-squared: 0.2017, Adjusted R-squared: 0.2004   
## F-statistic: 165.8 on 140 and 91898 DF, p-value: < 0.00000000000000022

Both Forward and Backward Stepwise Regression keep all variables and contribute to AIC=-469117 and R-Squared of 0.20 . We’re not having a lot of luck creating a model that explains the high ratings. That’s a good thing. We hope that the ratings are all about how good that food is in that restaurant, regardless of the neighborhood, the cuisine, or the restaurant type.

## Utilizing a Test/Train split

We split the data into testing and training data. This is used to test the performance of the models we’ve created and to make sure that our model is not over-fitting the data. Fortunately our dataset is large enough so that we can split it into 2 reasonably sized chunks.

library(caret)

## Warning: package 'caret' was built under R version 4.0.5

## Loading required package: lattice

##   
## Attaching package: 'caret'

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

train.rows = createDataPartition(y = zomato\_1\_regress\_omit$num\_rate, p=0.7, list = FALSE) #70% in training  
train = zomato\_1\_regress\_omit[train.rows,]   
test = zomato\_1\_regress\_omit[-train.rows,]

Now we’ll run the same lm() creation, except we’ll compare the R-squared values and look for overfitting.

train\_allmod = lm(num\_rate ~ cost\_for\_two + restaurant\_style + cuisine + neighborhood, train)  
summary(train\_allmod)

##   
## Call:  
## lm(formula = num\_rate ~ cost\_for\_two + restaurant\_style + cuisine +   
## neighborhood, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.38241 -0.04369 0.00943 0.05282 0.25434   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.7359262753 0.0143042536 51.448  
## cost\_for\_two 0.0000627500 0.0000008563 73.278  
## restaurant\_styleCafes 0.0009054426 0.0026170842 0.346

**“…SNIP…”**

## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07508 on 64290 degrees of freedom  
## Multiple R-squared: 0.2031, Adjusted R-squared: 0.2014   
## F-statistic: 118.8 on 138 and 64290 DF, p-value: < 0.00000000000000022

test\_allmod = lm(num\_rate ~ cost\_for\_two + restaurant\_style + cuisine + neighborhood, test)  
summary(test\_allmod)

##   
## Call:  
## lm(formula = num\_rate ~ cost\_for\_two + restaurant\_style + cuisine +   
## neighborhood, data = test)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.37944 -0.04448 0.00891 0.05243 0.22078   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.733829362 0.018407636 39.865  
## cost\_for\_two 0.000061413 0.000001282 47.901  
## restaurant\_styleCafes 0.006765611 0.004007202 1.688

**“…SNIP…”**

## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07511 on 27474 degrees of freedom  
## Multiple R-squared: 0.2012, Adjusted R-squared: 0.1973   
## F-statistic: 51.27 on 135 and 27474 DF, p-value: < 0.00000000000000022

R-squared for the training data was 0.20 R-squared for the test data was 0.195 . There is some degradation, but only an insignificant amount. These values are in line with what we were seeing without splitting the data.

## K-fold cross validation

Next we’ll run a k-fold cross validation. Instead of separating the data into testing and training components, for k-fold we are essentially separating out a slice consisting of 1/K of the data, and doing that K times so each observation gets to be part of the testing data. We wouldn’t want any data to get jealous now, would we? We’ll run it against the standard linear regression “lm()”. For this test we’ll use K=10.

# K-fold Cross Validation  
library(caret)  
ctrl = trainControl(method = "cv",number = 10) #set up caret 10 fold cross validation  
  
set.seed(123) #set random number seed for cross validation  
modkFold = train(num\_rate ~., zomato\_1\_regress\_omit, method = "lm", trControl = ctrl)

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit  
## may be misleading

summary(modkFold)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.37968 -0.04395 0.00929 0.05266 0.25426   
##   
## Coefficients:  
## Estimate Std. Error t value  
## (Intercept) 0.7358691086 0.0112774084 65.252  
## cost\_for\_two 0.0000622514 0.0000007113 87.519  
## cuisineAfghani -0.0428889291 0.0287472087 -1.492

**“…SNIP…”**

## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07508 on 91898 degrees of freedom  
## Multiple R-squared: 0.2017, Adjusted R-squared: 0.2004   
## F-statistic: 165.8 on 140 and 91898 DF, p-value: < 0.00000000000000022

The value for R-squared is 0.2004, showing once again that the data we have just isn’t a good predictor for the rating of the food. Which is still a good thing.

## Conclusion for cross-validating and sampling for food rating

With R-squared valued hovering in the 0.20 area for linear regression, Forward and Backward stepwise regression, and splitting the data using k-fold cross validation and train/test split, it’s safe to conclude that the variables in the Zomato dataset have a small positive correlation with the food rating, but nothing seems to drive that value too much.

# Predictive Analytics

How good are these models at predicting the rating for a given set of variables? If a new restaurant opens up, can we give a guess as to what the rating will be?

## Predictive for Simple Linear model

We can take our models we created from the train and test data and calculate what the values should be and compare them, and look at R-squared to see what the predictive value is.

predict\_train\_allmod = predict(train\_allmod)  
SSE = sum((train$num\_rate - predict\_train\_allmod)^2)  
SST = sum((train$num\_rate - mean(train$num\_rate))^2)  
ptr\_rsq = 1 - SSE/SST  
# 0.202  
  
predict\_test\_allmod = predict(test\_allmod)  
SSE = sum((test$num\_rate - predict\_test\_allmod)^2)  
SST = sum((test$num\_rate - mean(test$num\_rate))^2)  
ptest\_rsq = 1 - SSE/SST  
# 0.204

With the calculated R-squared hovering around 0.20 for both, we come to the conclusion (again) that the given dataset does not have a lot of correlation for the ratings and the other variables.

## Use our restaurant list

We have our list of restaurants that we’d like to open, and all of the information about them. We’ve come full circle. Instead of test data, we have potential restaurant data. What better data to try and predict ratings against? Which are the best small restaurants in the neighborhoods that will welcome them the most. How accurate would this be?

restaurants\_to\_create\_filtered <-  
 neighborhoods\_possible\_filtered[order(-neighborhoods\_possible\_filtered$diff\_neighborhood\_avg),]  
  
restaurants\_to\_create\_filtered <-  
 dplyr::select(restaurants\_to\_create\_filtered, cost\_for\_two, num\_rate, cuisine, restaurant\_style, neighborhood.y)  
  
rest\_to\_create\_allmod = lm(num\_rate ~ cost\_for\_two + restaurant\_style + cuisine + neighborhood.y, restaurants\_to\_create\_filtered)  
  
predict\_rest\_to\_create\_allmod = predict(rest\_to\_create\_allmod)  
SSE = sum((restaurants\_to\_create\_filtered$num\_rate - predict\_rest\_to\_create\_allmod)^2)

## Warning in restaurants\_to\_create\_filtered$num\_rate -  
## predict\_rest\_to\_create\_allmod: longer object length is not a multiple of shorter  
## object length

SST = sum((restaurants\_to\_create\_filtered$num\_rate - mean(restaurants\_to\_create\_filtered$num\_rate))^2)  
prod\_rsq = 1 - SSE/SST  
prod\_rsq

## [1] 0.3725513

Surprise! We get an R-Squared of 0.37 . For our list of potential restaurants we get a much higher r-squared value. What’s the reason? Perhaps ratings for higher rated restaurants are more predictable than average and lower rated restaurants. Are smaller restaurants more predictable? This is an interesting phenomena.

## Predictions when a new restaurant opens up

We’ve seen that there is a small amount of correlation between the ratings and the data in the dataset. Now what if a new restaurant opens up? Can we get an idea of what the rating for the restaurant might be?

We would know the cuisine, neighborhood, restaurant type, and the cost for two by walking into it. That’s all we need to get a potential rating (which will be very inaccurate, but it’s fun to look into it). First we’ll do it for one. Then I’ve created a .csv file with 20 rows of random sample data.

new\_restaurants = data.frame(cuisine = "North Indian", neighborhood = "Banashankari", restaurant\_style = "Delivery", cost\_for\_two = 1000)  
predict(allmod, new\_restaurants, type="response")

## 1   
## 0.754052

# Now do a list  
new\_restaurants\_to\_predict <- read\_csv("~/BAN 530/Week 5/Final5/new\_restaurants\_to\_predict.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## cuisine = col\_character(),  
## neighborhood = col\_character(),  
## restaurant\_style = col\_character(),  
## cost\_for\_two = col\_double()  
## )

new\_restaurants\_to\_predict$poss\_rating <- predict(allmod, new\_restaurants\_to\_predict, type="response")  
top\_n(new\_restaurants\_to\_predict,20)

## Selecting by poss\_rating

## # A tibble: 20 x 5  
## cuisine neighborhood restaurant\_style cost\_for\_two poss\_rating  
## <chr> <chr> <chr> <dbl> <dbl>  
## 1 North Indian Rajajinagar Dine-out 1000 0.738  
## 2 Chinese Electronic City Delivery 500 0.680  
## 3 South Indian Marathahalli Dine-out 300 0.685  
## 4 Biryani Malleshwaram Delivery 800 0.729  
## 5 Healthy Food Rajajinagar Dine-out 750 0.771  
## 6 North Indian Electronic City Delivery 1500 0.744  
## 7 Chinese Marathahalli Dine-out 2100 0.787  
## 8 South Indian Malleshwaram Delivery 600 0.728  
## 9 Biryani Rajajinagar Dine-out 600 0.710  
## 10 Healthy Food Electronic City Delivery 800 0.748  
## 11 North Indian Marathahalli Dine-out 500 0.690  
## 12 Chinese Malleshwaram Delivery 250 0.697  
## 13 South Indian Rajajinagar Delivery 350 0.708  
## 14 Biryani Electronic City Dine-out 800 0.695  
## 15 Healthy Food Marathahalli Delivery 1000 0.771  
## 16 North Indian Malleshwaram Dine-out 550 0.715  
## 17 Chinese Rajajinagar Delivery 600 0.714  
## 18 South Indian Electronic City Dine-out 700 0.700  
## 19 Biryani Marathahalli Delivery 500 0.689  
## 20 Healthy Food Malleshwaram Dine-out 300 0.747

# Linear Programming

## The Story

Betty Bigbucks’ plan has succeeded beyond her wildest dreams! She has turned three different small restaurants into 3 chains of 30 restaurants each. She has stuck with Dine-Out and Delivery because she seems to have found her sweet spot.

But haters are gonna hate. Her critics say that since she is just concentrating on dine-out and delivery that she doesn’t understand what REAL food is. Also, since she is developing chains, which is a Western concept, that she has forgotten her Indian roots. Betty likes the money, but she wants her reputation back. What can she do?

What if she decides to publish dining guides for North Indian and South Indian food on a per neighborhood basis for formal sitdown restaurants? That would make it look like she hasn’t lost touch with her Indian roots, and enjoys a good sitdown dinner. But that would mean going to all of the cafes and buffets in a given neighborhood and trying them all out. Who’s got time for that?

Fortunately our parsed-out dataset can help. We can get all of the buffets and cafes in a neighborhood that have North or South Indian food. She could just list the top 5 in each category. Instead she’ll put her own twist on them, with “gotta-read” headline like “5 Can’t Miss Meals in Malleshwaram for Two for 4000 Rupees or Less”.

## Methodology

After filtering out to get our list of restaurants to fit the criteria, we plug the data into Excel using solver. The solving criteria will be as follows:

* Objective function: Maximize ratings for 5 restaurants
* Constraints
  + A total of 5 restaurants
  + Each of the 5 restaurants can only be visited once.
  + Total cost for two for all 5 meals is less than or equal to 4000

For proof of concept, we’ll find two neighborhoods that have between 10 and 20 restaurants that fit the criteria.

buffets\_and\_cafes <- zomato\_1\_selected %>%  
 filter(restaurant\_style == "Cafes" | restaurant\_style == "Buffet") %>%  
 filter(cuisine == "North Indian" | restaurant\_style == "South Indian") %>%  
 count(neighborhood) %>%   
 drop\_na(neighborhood)  
buffets\_and\_cafes

## # A tibble: 30 x 2  
## neighborhood n  
## <chr> <int>  
## 1 Banashankari 8  
## 2 Bannerghatta Road 19  
## 3 Basavanagudi 15  
## 4 Bellandur 23  
## 5 Brigade Road 40  
## 6 Brookefield 20  
## 7 BTM 34  
## 8 Church Street 44  
## 9 Electronic City 16  
## 10 Frazer Town 12  
## # ... with 20 more rows

## Results

We decided to choose Brookefield with 19 restaurants. Get the name, rating, and cost to plug into the Excel file, with a price constraint of 6000 rupees.

brookefield\_buffets\_and\_cafes <- zomato\_1\_selected %>%  
 filter(restaurant\_style == "Cafes" | restaurant\_style == "Buffet") %>%  
 filter(cuisine == "North Indian" | restaurant\_style == "South Indian") %>%  
 filter(neighborhood == "Brookefield") %>%  
 dplyr::select(name, cost\_for\_two, num\_rate) %>%  
 drop\_na()  
  
brookefield\_buffets\_and\_cafes

## # A tibble: 19 x 3  
## name cost\_for\_two num\_rate  
## <chr> <dbl> <dbl>  
## 1 Afoozoo - Ginger Hotel 800 0.7   
## 2 Barbeque Nation 1600 0.76   
## 3 Cafe Palmyra 2000 0.84   
## 4 Citrus Cafe - Lemon Tree Hotel 1900 0.76   
## 5 Elements - Mapple Express 600 0.580  
## 6 Flavors of India 800 0.72   
## 7 Flavours Radha Hometel 800 0.7   
## 8 HIGH SKY 1000 0.72   
## 9 Imperio Restaurant 700 0.84   
## 10 Kava Kitchen & Bar - Fairfield by Marriott 1200 0.78   
## 11 Keys Cafe - Keys Hotel 1200 0.560  
## 12 M Cafe - Bengaluru Marriott Hotel Whitefield 2500 0.82   
## 13 Mega Bite 1100 0.72   
## 14 The Restaurant 1200 0.74   
## 15 The Terrace at Windmills Craftworks 1800 0.82   
## 16 Tiffin Room - Miraya Hotel & Residences 2000 0.76   
## 17 UBCR 1000 0.72   
## 18 Zaica Dine & Wine 750 0.76   
## 19 Zodiac 1500 0.72

* The 5 restaurants chosen are (with price and rating):
  + Cafe Palmyra (2000, 0.84)
  + Imperio Restaurant (700, 0.84)
  + Kava Kitchen & Bar - Fairfield by Marriott (1200, 0.78)
  + The Restaurant (1200, 0.74)
  + Zaica Dine & Wine (750, 0.76)

The total cost of the meals is 5850.

Now let’s do it for Malleshwaram, using 4000 rupees for the price contraint:

malleshwaram\_buffets\_and\_cafes <- zomato\_1\_selected %>%  
 filter(restaurant\_style == "Cafes" | restaurant\_style == "Buffet") %>%  
 filter(cuisine == "North Indian" | restaurant\_style == "South Indian") %>%  
 filter(neighborhood == "Malleshwaram") %>%  
 dplyr::select(name, cost\_for\_two, num\_rate) %>%  
 drop\_na()  
  
malleshwaram\_buffets\_and\_cafes

## # A tibble: 18 x 3  
## name cost\_for\_two num\_rate  
## <chr> <dbl> <dbl>  
## 1 24/7 - The Lalit Ashok Bangalore 3000 0.8   
## 2 1947 950 0.8   
## 3 Barbecued - By The Orchard 1200 0.86  
## 4 Chef's Bank 700 0.68  
## 5 Coastal Spice 900 0.74  
## 6 Kwatlay Rest-O-Cafe 800 0.82  
## 7 La Boulangerie - Le Meridien 2000 0.74  
## 8 La Brasserie - Le Meridien 4100 0.82  
## 9 Mint Masala 800 0.76  
## 10 Mumbai Bistro 500 0.84  
## 11 Sangam Restaurant 600 0.74  
## 12 Sattvam 1200 0.88  
## 13 Tata Cha 500 0.64  
## 14 The Green Path - Forgotten Food 1100 0.86  
## 15 The Higher Taste 800 0.86  
## 16 The Kabab Studio - Goldfinch Hotel 1500 0.84  
## 17 The Raj Pavilion - ITC Windsor 2400 0.84  
## 18 Village - The Soul of India 1000 0.8

* The restaurants chosen are:
  + Kwatlay Rest-O-Cafe (800, 0.82)
  + Mint Masala (800, 0.76)
  + Mumbai Bistro (500, 0.84)
  + The Green Path - Forgotten Food (1100, 0.86)
  + The Higher Taste (800, 0.86)

The total cost of the meals is 6000 rupees.

# Conclusion

## The Saga of Betty Bigbucks

Betty knows that starting up a restaurant can be risky, and that any advantage that can be had should be pursued. Sometimes a successful restaurant merely copies what the work that their parents did. Or someone is in the business and sees the trends change and then pivots or expands to seize that opportunity. Betty had a hunch there was a demand for more quality Dineout and Delivery options to serve the tech workers of Bangalore. She combined this hunch with the analysis of very real data about the restaurants in the area.

The Zomato dataset of Bangalore restaurants was packed with information about the area dining establishments. Using this dataset, we were able to determine what cuisines were in demand in a given neighborhood. We also found restaurants that we felt were a good match to expand into those neighborhoods. While someone might have “a feel” for this kind of thing, we all know these “feelings” can be biased. We used a data-driven approach to analyze the current restaurant environment, and make recommendations based on that. We did not bring our biases, and the conclusions about the cuisines and the neighborhoods that should be expanded into were derived through analytics.

With the dataset, we were able to determine the average rating per cuisine in a neighborhood. With this amount of data and possibilities, this is not something that could be determined with any degree of confidence using one’s gut. While we knew that some will be higher or lower than others, we were able to find where the data discovered the outliers. In this case, they are opportunities. The lack of a high quality cuisine in a neighborhood is not something that any one person or group did…it happened organically. Like Capitalism is helped by “the invisible hand”, we are able to use the conclusions to bring desired food where we believe it will be wanted, but in a more efficient way.

As far as Betty and her ambitions go, the restaurant analysis is just the beginning. She has to find restaurant owners to partner with her. She needs to find real estate to expand into, and remodel the places she finds. Inspections will need to be scheduled. Advertising bought and a social media presence created. The success of those small restaurants will have to be duplicated in another place with different personnel. She has a head start, but there’s a lot of work to go.

## The Role of Business Analytics

The restaurant business is a fickle one, with many establishments failing within the first 5 years. If analysis like this can make that less likely, everyone is better off. There are many factors that can determine the success and failure of a restaurant that can’t be quantified. Relying only on a dataset like this one would be a fool’s errand. Analytics is a tool, not the answer, to many of humanity’s problems. At the end of the day, the answers that are obtained and the decisions determined must be implemented by human beings, who are faced with dealing with factors that can’t be put in a spreadsheet, and more importantly, other people, with their biases, self-interests, and internal decisions making rubrics.

While Betty Bigbucks is a fictitious person (surprise!) and the scenario concocted for her is a contrived one, the methods used to examine, clean, process, and analyze the data are real. We can determine which data is important, and which is just noise. We can use a variety of methods to describe, predict, and make decisions. The ability of find the unexpected, the “needles in the haystack” is something that is needed. Whether it’s finding an unfilled hunger for food in a neighborhood on the other side of the world, or using data to determine which medicine is best used against a devastating disease, Analytics provides the tools and methodology to provide meaningful, actionable insight. What we do with it after that is up to us.

It is said that luck is when preparation meets opportunity. As we have seen, Analytics helps find the opportunity. If we can crunch the numbers to develop models to assist in determining outcomes, society will be better off because of it. Will be perfect? No. All models and predictions come with a stated percentage of success and failure. As we have learned the techniques in this curriculum, we have added them to our “bag of tricks”. In the bigger picture, Analytics is a powerful tool that society has for decision making. It’s only as good as the data that goes into it. But if that data is a true representation of the subject at hand, Analytics will help determine the proper course of action.