Final Project

Kitu Komya

November 28, 2016

Final Report

by Kitu Komya

Introduction

We've been given three sets of datasets: edmunds, cell, and irs. These data files will be used to answer questions regarding the relationships between variables within the set. Since all three of the data files are related, they will eventually be joined so that more analysis can be run on them. We will understand if there is a significant difference between # Of single tax returns vs # of joint tax returns which may be of interest to people better trying to understand how the irs system works and how each location is affected by these two variables. We will also learn the relationship between the dependencies and the number of single returns which may be of interest to those who file their taxes so they may better understand the dynamics of the system.

Data

Describe the individual data files

The edmunds original data dimensions is 2445924 by 24. After cleanup, it was 2445924 by 16. The irs original data dimensions is 288 by 111. After cleanup, it was 288 by 7. The cell original data dimensions is 9248 by 22. After cleanup, it was 9248 by 22.

The edmunds data set comes from edmunds.com in which each car is assigned multiple aspects such as a unique car ID, information about the car, pricing information, information on the dealer, and other location tidbits. Most of the variables were descriptive in describing the car physically or numerical in describing tidbits like its price.

The irs data set is aggregated tax information from the IRS directly from the year 2014. They are based on returns from residents from different zipcodes. The variables here are all numeric since they relate to the number of tax returns.

The cell data set contains information about cell towers in Los Angeles. The data set includes information on the cell tower's location and type, and thus the variables range from descriptive to numerical.

Clean the data

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

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

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

```
## Warning: package 'stringr' was built under R version 3.4.2
```

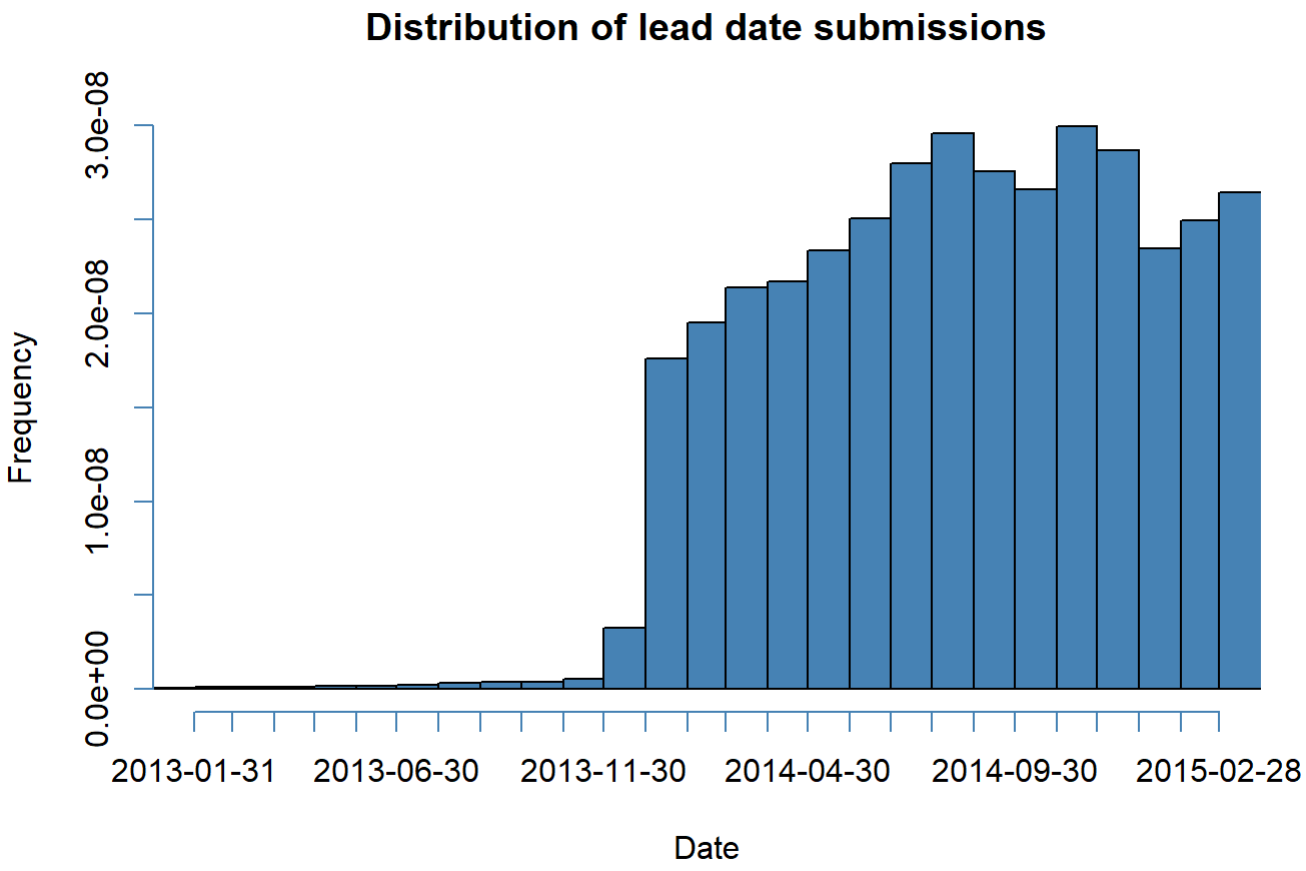
```
## Warning: package 'knitr' was built under R version 3.4.2
```

```
## Parsed with column specification:
## cols(
##   .default = col_character(),
##   OBJECTID = col_integer(),
##   post_id = col_integer(),
##   info2 = col_double(),
##   ext_id = col_integer(),
##   ZIP = col_integer(),
##   longitude = col_double(),
##   latitude = col_double()
## )
```

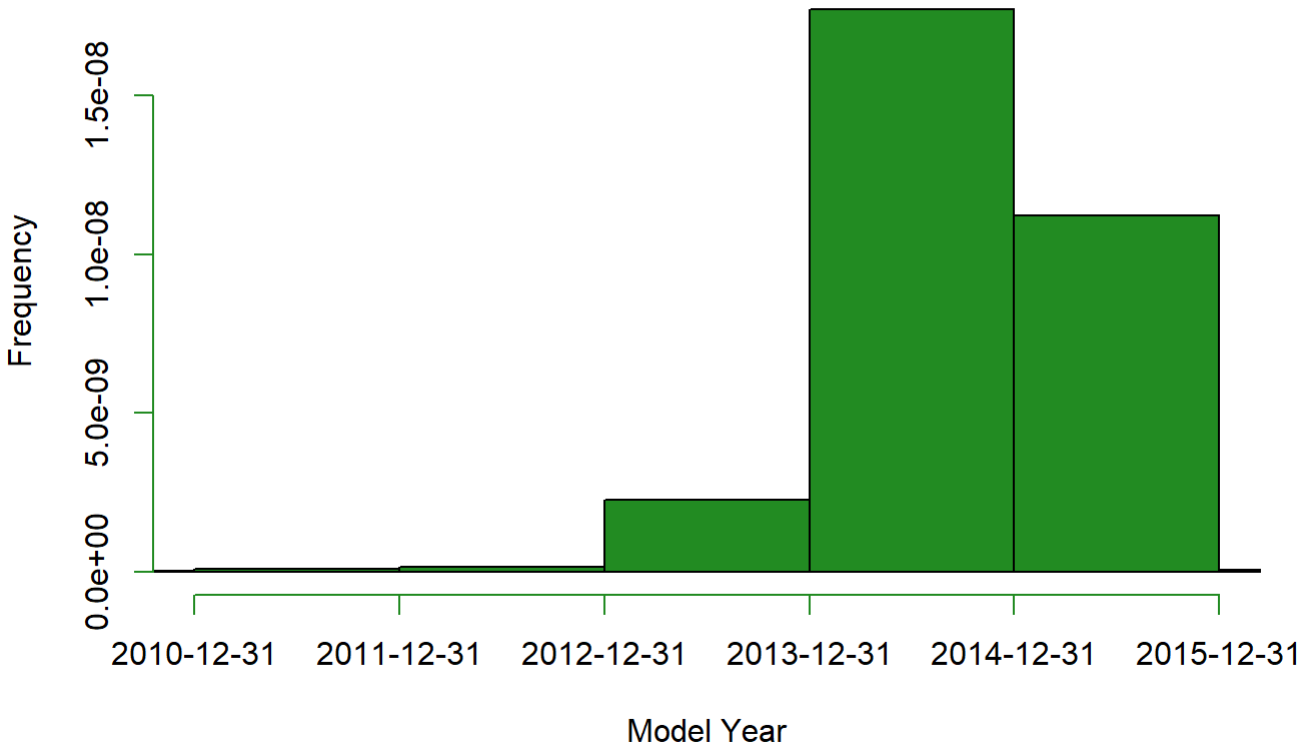
```
## See spec(...) for full column specifications.
```

In cleaning the data, all the unneeded variables were dropped, the missing values wered changed into NAs, date/datetime variables were changed to date/POSIX variables, numerics were ensured to be numerical variables and not characters, and factors were also ensured to be factor variables and not characters. Many dplyr and pipes techniques were used to change the variable class types, the most consuming part of data cleaning.

Summarize



After zooming in on this histogram data, we see that the shape of the distribution of lead date submissions is skewed to the left, with the majority of submissions occurring in 2014 and 2015. The spread is huge since the range encompasses nearly 2 years. The center is around the middle of 2014.



We see that the model year distribution is skewed to the left. The spread of the model years is from 2010 to 2016. We see that we achieve the peak frequency occurs in the year 2014, which is probably also the center of our distribution.

```
## Selecting by most_popular_makes
```

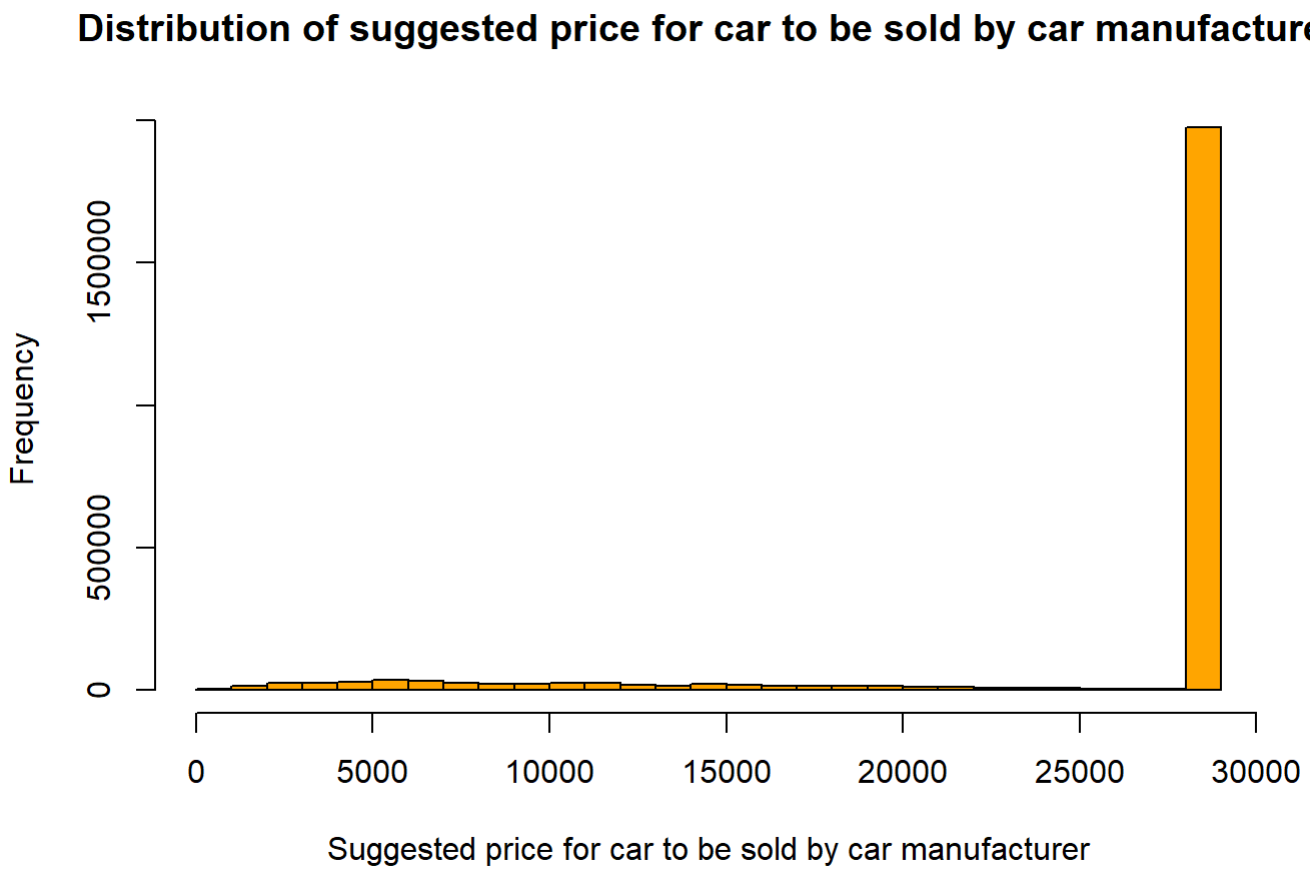
make	most_popular_makes
honda	440720
toyota	353630
ford	144697
nissan	118091
subaru	106417

```
## Selecting by least_popular_makes
```

make	least_popular_makes
isuzu	1
Oldsmobile	1
geo	3

make	least_popular_makes
Isuzu	3
Lotus	3
oldsmobile	3

As we can see from the outputted table, the most popular make is honda, and the least popular make is isuzu and oldsmobile.



What an ugly graph. However, I will not change the x value range because that is omitting data. As we can see, the distribution of suggested price for car to be sold by car manufacturer is highly concentrated at the end range at about 30000. The mean is also at 30000. Although the spread ranges from 0 to 30000, the values before 30000 are basically irrelevant in comparison to the 30000 values. This makes sense because the price suggested by the car manufacturer for a vehicle to be sold is pretty high and around those values.

In this IRS data, we discover a few relationships between the variables. Firstly, the N1 refers to the number of returns, and thus, the variables MARS1 (# of single returns), MARS2 (# of joint returns), and MARS4 (# of head of household returns) will all add up together to approximately equal N1. It's not exact because in our dataset we have omitted a few other types of returns. A00100 is the adjusted group income when taken into account all of the returns in each zipcode area.

```
## Selecting by most_cell_towers
```

ZIP	most_cell_towers
91042	398
90275	250
90012	248
91311	198
90045	160

```
## Selecting by least_cell_towers
```

ZIP	least_cell_towers
90222	1
90506	1
90630	1
90814	1
91046	1
91607	1
91710	1
91764	1
92280	1
92305	1
92311	1
92382	1
92404	1
92518	1
92545	1
92571	1
92592	1
92612	1
92618	1
92648	1
92651	1
92676	1

ZIP	least_cell_towers
92703	1
92707	1
92801	1
92806	1
92821	1
93004	1
93010	1
93030	1
93033	1
93105	1
93108	1
93205	1
93225	1
93274	1
93420	1
93513	1
93518	1
93527	1
93545	1
93549	1
94513	1
95607	1
95670	1

The zipcode with the most amount of cell towers is 91042. There are many zipcodes with only 1 cell tower, so I will not list them all since they are included in the table, but a few include 90222, 90506, and 90630. The cities in the zipcode 91042 are Tujunga, Montrose, La Crescenta, Los Angeles, Sunland, Pasadena, Mount Lukens, Valencia, Highway Highlands, Burbank, La Canada, and Sylmar.

Join data files

```
## Warning: Column `dealer_zip`/`ZIPCODE` joining factors with different
## levels, coercing to character vector
```

```
## Selecting by most_celltowers
```

dealer_zip	most_celltowers
90703	18031

Zipcode 90703 has the most cell towers. This new join vector has 18031 observations.

Bonus Task

We have created a map in which we can see the locations of all of the cell towers. They are mainly concentrated in Southern California.

Analysis

t-test

```
ARS1, irs$MARS2, alternative = "greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  irs$MARS1 and irs$MARS2
## t = 9.9343, df = 497.73, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  2404.404      Inf
## sample estimates:
## mean of x mean of y
##  7884.340  5001.771
```

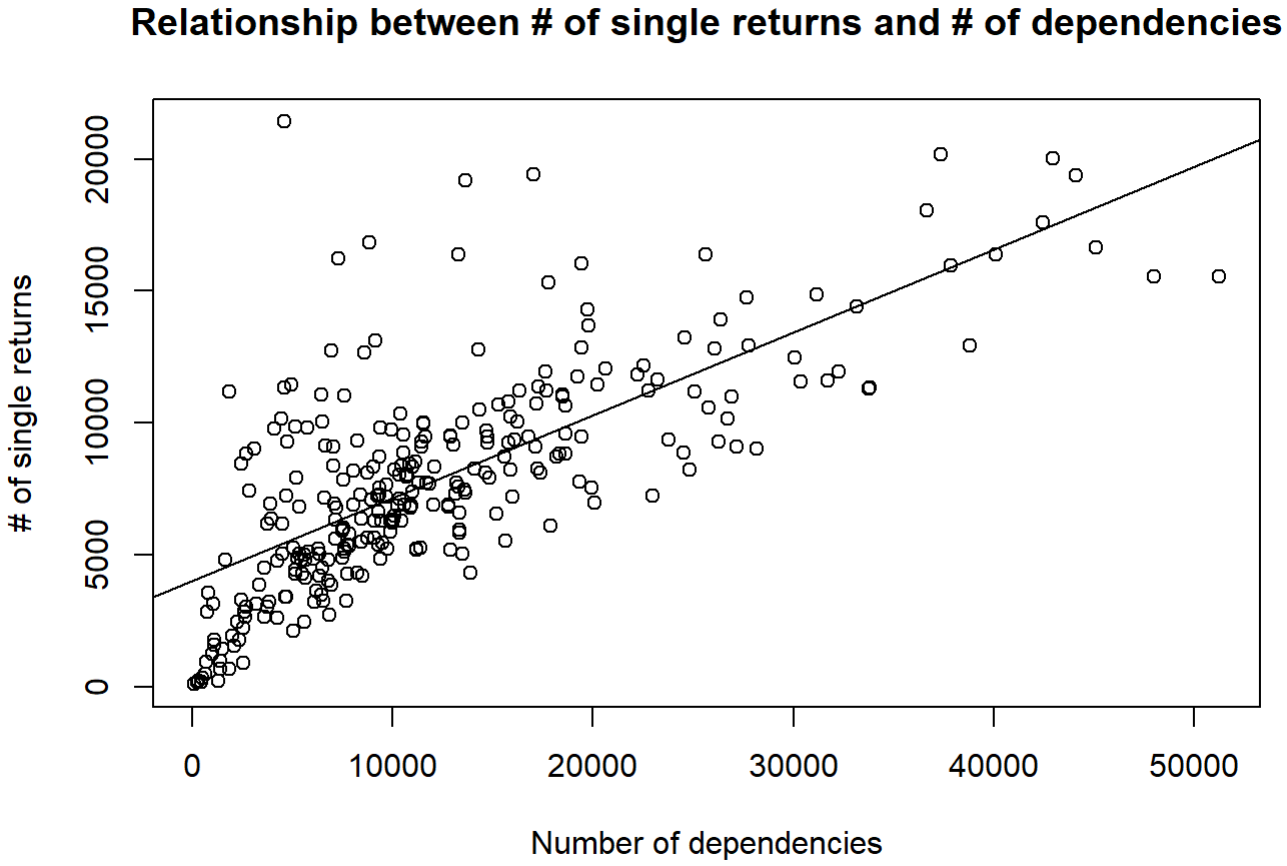
I am choosing to compare the means between the MARS1 and MARS2 data from the irs dataset. Basically, I am testing to see whether there is a difference in means in the number of single returns and the number of joint returns. By looking at the difference in means, I will be able to discern which kind of tax return is more significantly filed.

Linear Regression

```
fit <- lm(MARS1~NUMDEP, data = irs)
plot(MARS1~NUMDEP, data = irs, main = "Relationship between # of single returns and # of dependencie
s", xlab = "Number of dependencies", ylab = "# of single returns")
summary(fit)
```

```
##
## Call:
## lm(formula = MARS1 ~ NUMDEP, data = irs)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4554.1 -1981.9  -585.5  1334.9 15945.4
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.036e+03  2.730e+02   14.78   <2e-16 ***
## NUMDEP      3.133e-01  1.759e-02   17.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2833 on 286 degrees of freedom
## Multiple R-squared:  0.526,  Adjusted R-squared:  0.5243
## F-statistic: 317.3 on 1 and 286 DF,  p-value: < 2.2e-16
```

```
abline(fit)
```



I have included the plot to better understand the relationship between the number of single returns and number of dependencies. In the conclusions section, I will elaborate on this plot as well as the summary output.

Custom Functions

```
zipcode <- function(zip) {

# I have created blank vectors for the outputs
cell <- c()
tax <- c()
car <- c()

n <- length(zip) # this is the length of the zipcode vector

# makes function robust by only allowing for correct input types
if(!is.factor(zip)) {
  stop(paste("Your input needs to be a vector of zipcode values in the factor format. Your class is
 currently", class(x)))
}

# for loop adds the table values into output for each zipcode
for (i in 1:n) {
  cell[i] <- nrow(subset(cell, zipcode == zip[i])) # counts all cell towers
  tax[i] <- sum(subset(irs, zipcode == zip[i][2])) # counts all tax returns since the total tax retu
rns is stored in N1
  car[i] <- nrow(subset(edmunds, zipcode == zip[i])) # counts all car leads
}

# creates data frame output for all zipcodes
output <- data.frame(zip, cell, tax, car) %>%
  kable()

# returns the output
return(output)
}
```

Results & Conclusions

The p-value is 2.2 * 10^-16 which is clearly less than 0.05. This means that at the 95% significance level, there is a significant difference in means between the number of single returns and the number of joint returns. More specifically, since we ran at an alternative value of “greater,” the number of single returns is significantly greater than the number of joint returns. Thus, they are not statistically equal. Our output actually tells us the mean of the two variables. The mean of the number of single returns is 7884.340, and the mean of joint returns is 5001.771, and since they really are not values close to each other, we can be confident that our t-test worked.

The adjusted r square value is 0.5243 which means that 52.43% of the variability in the # of single returns is explained by our linear model. And honestly, that’s not too bad. This means that there is a positive, linear, moderately strong correlation between the number of dependencies and the # of single returns. This is surprising, because we should expect that if there are more dependencies in a family, for instance, then the # of single returns should decrease and instead the # of joint returns increase. That is why I believe that there is a lurking variable. The lurking variable is population. In a city, for instance, if there are many dependencies, then in that city there will be a higher number of single and joint returns. Thus, our linear model is useless since it fails to recognize this confounding variable, which has a profound effect on our conclusions.

```
#zipcode(x = c(97477, 92545))
```

I genuinely do not know why my function is not working, unfortunately. What a grievance this is, honestly. I have uncommented it out because the function does not let the entire html to knit otherwise. Aaah!