

CS424 Big Yellow Taxi

Group 2 - Jack Martin and Shoaib Jakvani

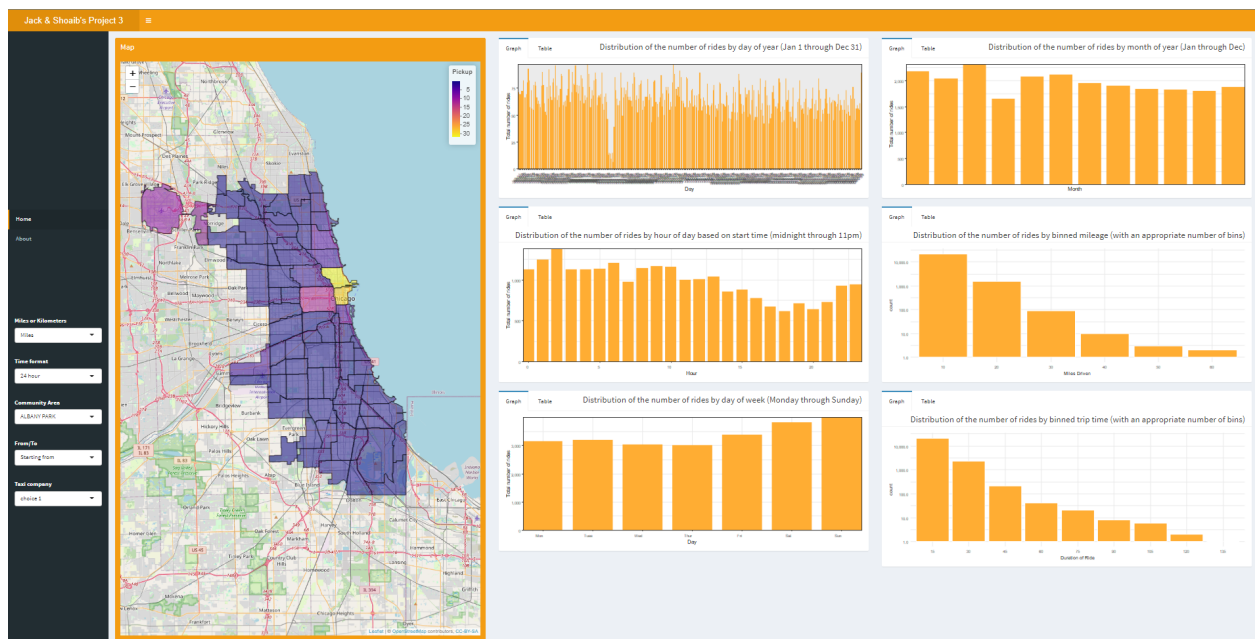
Links

[Shiny App](#)

[Github Repo](#)

[Youtube Video](#)

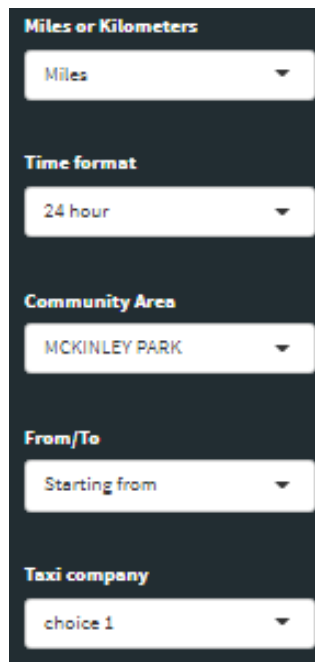
Introduction



Hello, welcome to our application documentation for Project 3 of CS 424 - Visual Analytics. One of our project's main purposes is to teach how to clean a large collection of data into a much smaller and focused selection of data in order to visualize it in a more intuitive way. The libraries that I will be using are Shiny, leaflet and Shiny Dashboard in order to visualize this. These libraries enable an R program to create a webpage that can display all the plots created by ggplot and maps created by leaflet

and it lets users be able to dynamically change the input data and have the plots change accordingly based on what the user does.

For example, if someone wanted to see only plots related to the Loop community, they would have to select the Loop through either a dropdown menu or selecting it on the map and it would update the plots dynamically and without crashing the application. The figure below highlights the user input controls: “**Miles or Kilometers**” allows the user to view the binned mileage in either unit, “**Time format**” was unimplemented but should allow the user to toggle between viewing charts scaled by hours in either military or standard time, although our app used military as default. “**Community Area**” allows the user to select any given community from the derived communities found [here](#). “**From/To**” determines whether a user is viewing the data as the taxi driver picking the client up or dropping them off. “**Taxi company**” is unimplemented but similarly to *Community Area* should show a list of taxi companies that provided services in the area.



Miles or Kilometers

Miles

Time format

24 hour

Community Area

MCKINLEY PARK

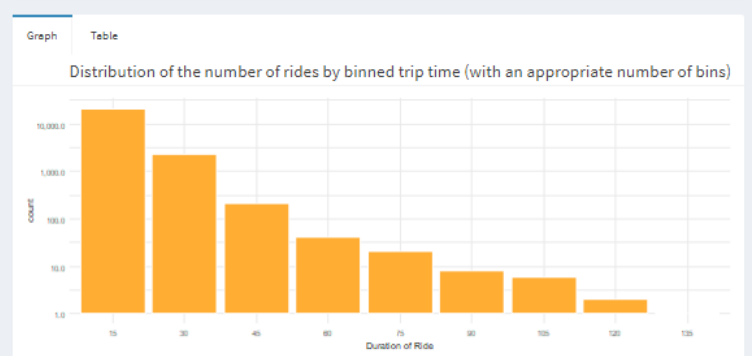
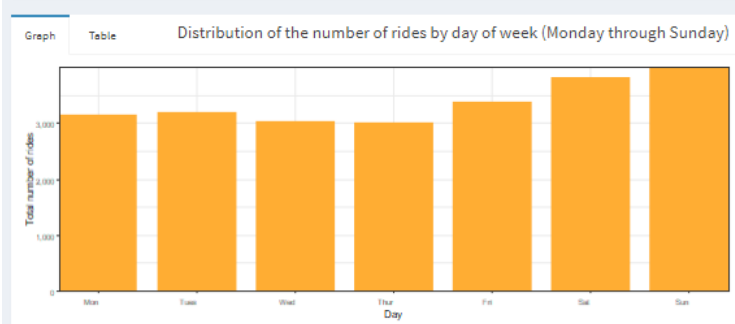
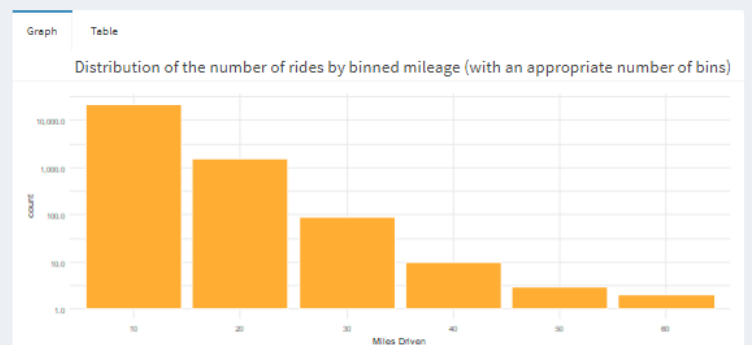
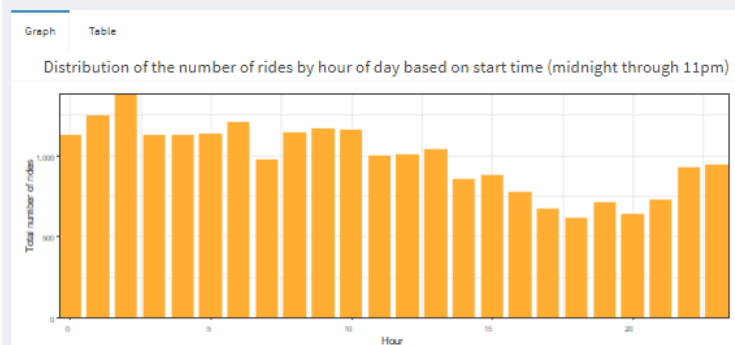
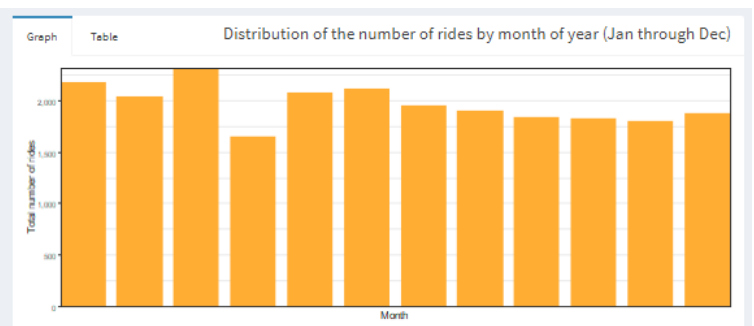
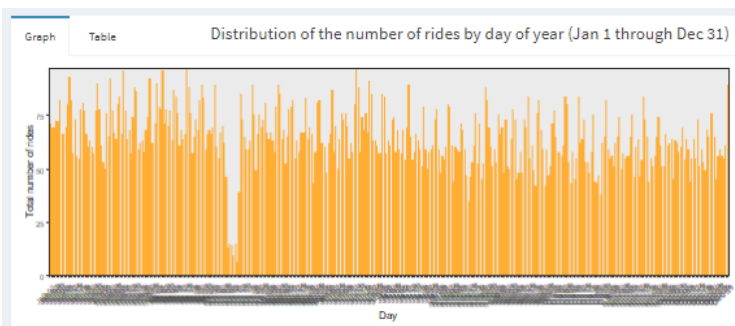
From/To

Starting from

Taxi company

choice 1

The data that we are looking into for visualization are the collection of all Taxi rides and the attributes associated with each of them during the year of 2019. The data that we'll be looking into for each of these Taxi rides are when the trip started, how long in seconds each trip lasted, how much distance each ride covered, the community that the rider got picked up from, and the community that the rider got dropped off from. The plots that we visualized were the number of rides by the day of the year, hour of day, day of the week, month of year, rides by binned mileage, and rides by trip time. For a lot of these plots, we chose to select the y axis on a logarithmic scale in order to see selections that had a very small amount of rides compared to ones that had many magnitudes higher than them. The figure below shows a snapshot of the mentioned graphs



Data and Reproduction

To begin the data cleaning process, we took our initial dataset from the [City of Chicago Data Portal](#) that was approximately 7 GB and contained over 16 million rows, and aimed to cut it down into small enough file sizes so that we could upload them onto the Shiny server and efficiently parse through the datasets. The first thing we did was load the large datafile into R using `fread()`. Then, we selected only the columns of data that we were interested in plotting.

```
# # list of needed columns
# col_names <- c(
#   "Trip Start Timestamp",
#   "Trip Seconds",
#   "Trip Miles",
#   "Pickup Community Area",
#   "Dropoff Community Area",
#   "Company"
# )
```

Then, we filtered the data to delete any outliers in the above data still existing. The outliers in our cases were defined as any trips less than 0.5 miles or those more than 100 miles. Any trips less than 60 seconds, and any trips greater than 18,000 seconds (5 hours). And we also dropped trips that started or ended outside Chicago Community areas denoted by an NA.

```
# # 1) all trips less than 0.5 miles
# TaxiSelect <- TaxiSelect[!TaxiSelect$'Trip Miles' < 0.5]
# # 2) more than 100 miles
# TaxiSelect <- TaxiSelect[!TaxiSelect$'Trip Miles' > 100]
# # 3) less than 60 seconds
# TaxiSelect <- TaxiSelect[!TaxiSelect$'Trip Seconds' < 60]
# # 4) greater than 5 hours == 18,000 seconds
# TaxiSelect <- TaxiSelect[!TaxiSelect$'Trip Seconds' > 18000]
# # 5) drop NA values (trips outside Chicago community)
# TaxiSelect <- TaxiSelect[!is.na(TaxiSelect$`Pickup Community Area`)]
# TaxiSelect <- TaxiSelect[!is.na(TaxiSelect$`Dropoff Community Area`)]
# #
```

In order to reduce file size even more, we used a map to convert cab company names to numbers and did this by assigning the first mentioned company to a 1 and the second to a 2 and so on.

```
# unique <- unique(TaxiSelect$)
# view(unique)
# valMap = c(1:55)
# #use integers to denote different companies in order to reduce file size
# TaxiSelect$Company = mapvalues(TaxiSelect$Company, unique, valMap)
#
```

Then we turned the timestamp of the starting trip time to a datetime object that lubridate could work on easier.

```
# # turns timestamp of starting trip to easier data to work with: hour/date
# TaxiSelect$NewDate <- parse_date_time(TaxiSelect$'Trip Start Timestamp', "%m/%d/%Y %I:%M:%S Op")
# TaxiSelect$Hour <- hour(TaxiSelect$NewDate)
# TaxiSelect$Date <- date(TaxiSelect$NewDate)
# #removes unneeded columns to save file size
# TaxiSelect <- subset (TaxiSelect, select = -NewDate)
# TaxiSelect <- subset (TaxiSelect, select = -`Trip Start Timestamp`)
# #
```

Next, we wrote the result of this to a csv file and exported it out.

```
# # #writes result to csv file
# # fwrite(TaxiSelect,"TaxiData.csv")
#
```

The resulting file is too large for GitHub so we split it into multiple csv files that were smaller using a program called CSVSplitter.

[Home](#) / [Windows](#) / [Productivity Software](#) / [Contact Management Software](#) / [CSV Split](#)




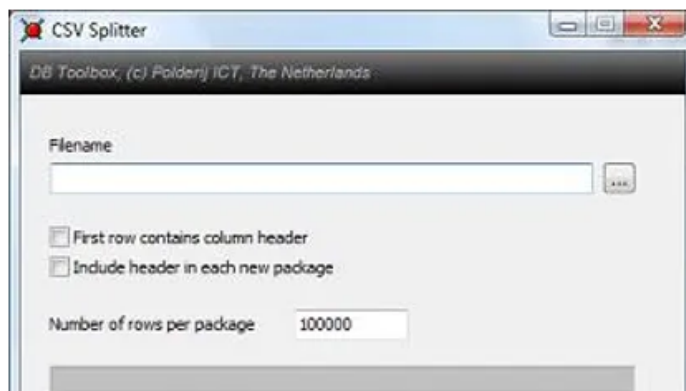
CSV Splitter

By [Polderij ICT](#) **FREE**

✓ **DOWNLOAD NOW**

Key Details of CSV Splitter

- Split large sized comma separated files into smaller ones
- Last updated on 04/25/13
- There have been 0 updates within the past 6 months
- The current version has [0 flags on VirusTotal](#) 



Finally, to load them all in, we merged all the files together and used that object to store our data.

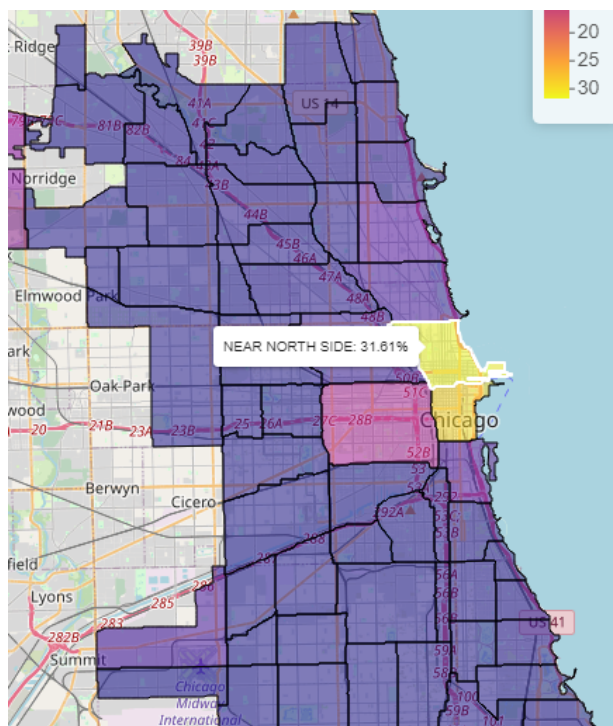
```

# merging files together start -----
Taxi <-
  list.files(pattern = "*.csv") %>%
  map_df(~fread(.))
#end-merging files together -----

```

Interesting things found for data

Areas with the most rides according to the heatmap ended up being the Loop and Near North Side which made sense to us as that's where people commonly take the taxi to work, while other areas tend to make use of other public transportation like the CTA L stops.



We also found that Taxi rides least commonly happen during weekends which was surprising to us. We also found that most rides had a duration of 15 minutes which

made sense if you only wanted a Taxi ride to somewhere closeby. The peak for ridership tends to be in the early-late afternoon.

