DA5020.A4.KatieMcCreedy

2023-02-05

libraries

library(dplyr) library(tidyr) library(stringr) library(ggplot2) library(corrplot) library(githubinstall) library(arrow)

Question 1

devtools::install_github("apache/arrow/r") tripdata_df<-read_parquet("green_tripdata_2020-02 (2).parquet")

knitr::opts_chunk$set(echo = TRUE)$dim(tripdata_df)$$as.factor(tripdata_df$PULocationID) as.factor(tripdata_df$DOLocation
PULocationID, DOLocationID, Store_and_fwd_flag are all variables which should be loaded as factors because they are non-numeric/have no natural order. As.factor will load these variables as factors. This can also be done when the data is loaded into R.

#Question 2 #Most Common Way to Hail a Cab as.numeric(tripdata_df$trip_type)$$is.na(tripdata_df$trip_type)

#Question 2 tripdata_df %>% count(tripdata_df$trip_type)

#Most Common Payment Type tripdata_df %>% count(tripdata_df$payment_type)

This returns that 1-trip_type or street hail is done 310,466 times. 2-trip_type or dispatch is done 7,272 times. People street hail to get cabs vastly more often. For trip_payment, the most common form was 1-payment_type or credit card at 176530 uses.

```
#Question 3 - Plotting trip pick-ups in February
#Plot most common payment type - for practice
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v tibble  3.1.8     v purrr   1.0.1
## v readr   2.1.3     v forcats 1.0.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

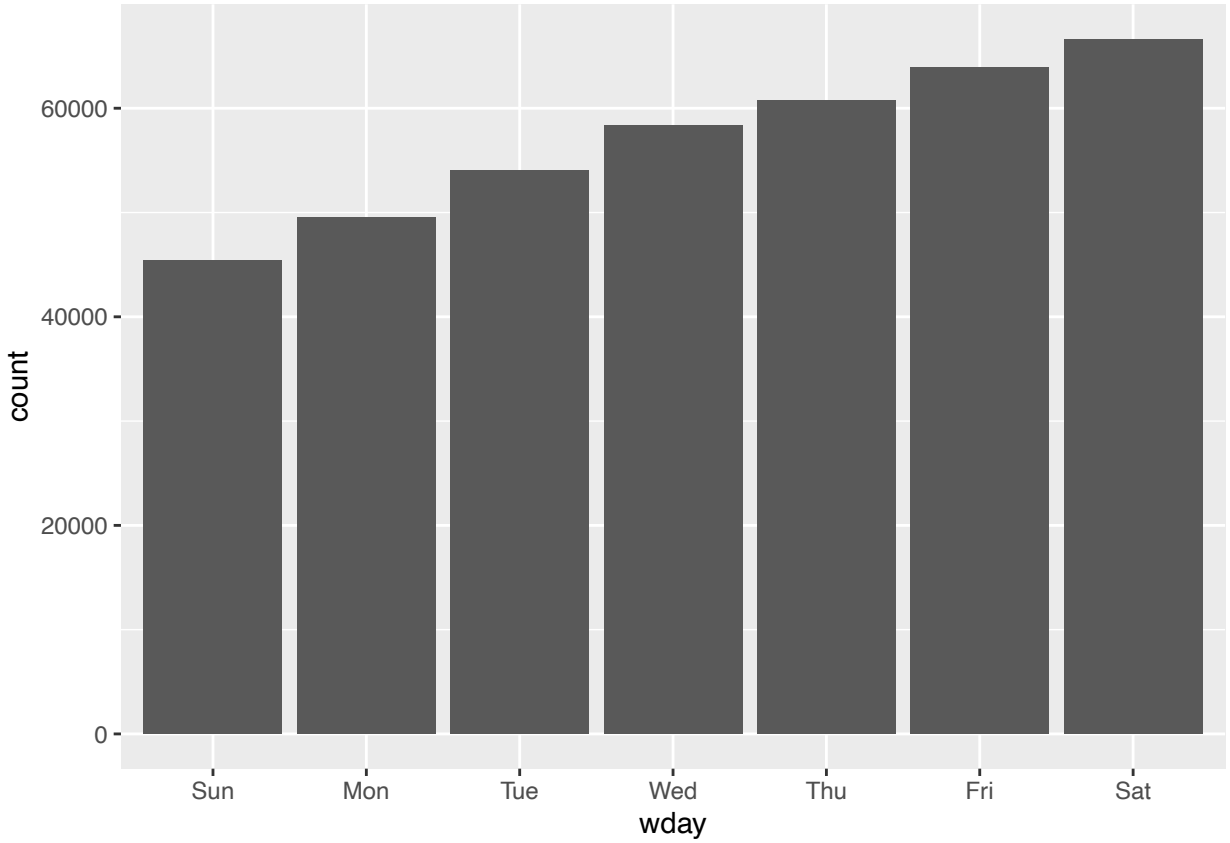
```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
##
## The following object is masked from 'package:arrow':
##
##     duration
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
tripdata_df %>%  filter(!row_number() %in% c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)) # remove Jan pick-ups
```

```
## # A tibble: 398,622 x 20
##    VendorID lpep_pickup_datetime lpep_dropoff_datetime store_a~1 Ratec~2 PULoc~3
##       <int> <dttm>               <dttm>                <chr>       <dbl>   <int>
##  1        2 2020-02-01 00:38:46  2020-02-01 00:47:36   N               1       7
##  2        1 2020-02-01 00:11:49  2020-02-01 00:42:38   N               4     130
##  3        2 2020-02-01 00:03:24  2020-02-01 00:17:45   N               1      74
##  4        2 2020-02-01 00:24:08  2020-02-01 00:26:54   N               1     116
##  5        1 2020-02-01 00:13:24  2020-02-01 00:18:06   N               1     166
##  6        2 2020-02-01 00:07:46  2020-02-01 00:17:20   N               1      42
##  7        2 2020-02-01 00:38:32  2020-02-01 00:56:40   N               1      83
##  8        2 2020-02-01 00:20:44  2020-02-01 00:34:06   N               1     223
##  9        2 2020-02-01 00:09:09  2020-02-01 00:31:18   N               1      52
## 10        2 2020-02-01 00:01:24  2020-02-01 00:10:09   N               1      74
## # ... with 398,612 more rows, 14 more variables: DOLocationID <int>,
## #   passenger_count <dbl>, trip_distance <dbl>, fare_amount <dbl>, extra <dbl>,
## #   mta_tax <dbl>, tip_amount <dbl>, tolls_amount <dbl>, ehail_fee <???>,
## #   improvement_surcharge <dbl>, total_amount <dbl>, payment_type <dbl>,
## #   trip_type <dbl>, congestion_surcharge <dbl>, and abbreviated variable names
## #   1: store_and_fwd_flag, 2: RatecodeID, 3: PULocationID
```

```
tripdata_df %>%
  mutate(wday = wday(lpep_pickup_datetime, label = TRUE)) %>%
  ggplot(aes(x = wday)) +
    geom_bar()
```



This chart shows that the pick-ups increase in frequency over the Sunday-Sat period. The least common day

for pick-ups was Sunday and the most frequent day was Saturday.

```
#Question 4
data<-"2020-02-01 00:10:25"
hour(data)
```

```
## [1] 0
```

```
attach(tripdata_df)
lpep_hour <- tripdata_df %>%
    mutate(lpep_hour = hour(lpep_pickup_datetime))

HourOfDay <- function(lpep_pickup_datetime) {
  lpep_hour <- hour(lpep_pickup_datetime)
  return(lpep_hour)
}

HourOfDay(lpep_pickup_datetime)
```

```
##     [1]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##    [25]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##    [49]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##    [73]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##    [97]  0  0  0  0  0  0  0  0  0  0  0  0 23  0  0  0  0 23  0  0  0  0  0  0  0
##   [121]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [145]  0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0
##   [169]  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [193]  0  0  0  1  0  0  0  0  0  0  0  0  0  0 23  0  0  0  0  0  0  0  0  0
##   [217]  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0
##   [241]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [265]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [289]  0  0  0  0  0  0  1  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [313]  0  0  0  0  0  0  0  0 23  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [337]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [361]  0 23  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [385]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0
##   [409]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  0  0  0  0  0
##   [433]  0  0 23  0  0  0  0  0  0  0  0  0  0 23  0  0 20  0  0  0  0  0  0  0
##   [457]  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  1  2  1  1  1  1  1  1
##   [481]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [505]  1  1  1  1  1  1  1  1  2  1  2  1  1  0  1  1  1  1  1  1  1  1  1  1
##   [529]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [553]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [577]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [601]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  2  1  1  1  1  1  1
##   [625]  1  1  1  1  1  1  0  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [649]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [673]  1  1  1  1  1  2  1  2  1  1  1  1  1  1  1  1  1  2  1  1  1  1  1  1
##   [697]  1  1  1  1  1  1  2  1  1  0  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [721]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  0  1
##   [745]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [769]  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
##   [793]  1  1  1  1  1  2  1  1  1  1  1  1 20  1  1  1  1  1  1  1  1  1  1  1
##   [817]  1  1  1  1  3  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2
##   [841]  2  3  2  2  2  2  2  2  2  2  1  2  2  2  2  2  2  2  2  2  2  2  2  2
##   [865]  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2  2
```

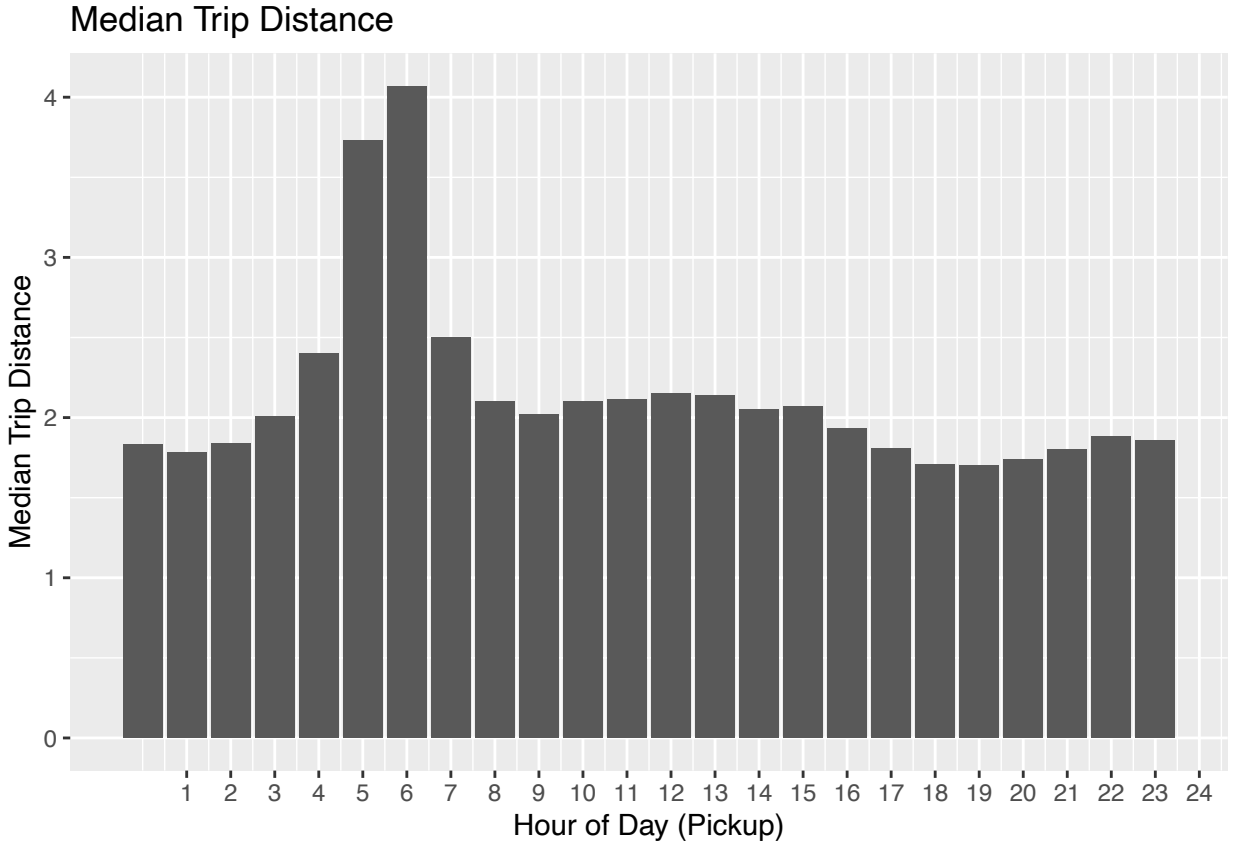
```
## [99385]  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  8  7
## [99409]  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
## [99433]  7  7  7  7  7  7  7  7  8  7  7  7  7  7  7  7  8  7  7  7  7  7  8
## [99457]  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
## [99481]  7  7  7  7  7  7  7  7  7  8  7  7  7  7  7  7  7  7  7  7  7  6  7  7
## [99505]  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7  7
## [99529]  7  7  7  7  7  7  7  7  7  7  8  7  7  7  7  7  7  7  7  7  7  7  7  8
## [99553]  8  9  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  9  8  8  8  8
## [99577]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  9  8  9  8  8  7  8  8
## [99601]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  9  8  8  8  8  8  8
## [99625]  8  8  8  9  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  9  8  8  8
## [99649]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99673]  8  8  8  8  8  8  8  8  8  8  8  9  8  8  8  8  8  8  8  8  8  7  7  8  8
## [99697]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99721]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  6  8  8
## [99745]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99769]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99793]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99817]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  7  8
## [99841]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99865]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99889]  9  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99913]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99937]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8  9  8  8  8  8  8  8  8
## [99961]  8  8  8  8  8  8  8  8  7  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
## [99985]  8  8  8  8  8  8  8  8  8  8  8  8  8  8  8
##  [ reached getOption("max.print") -- omitted 298633 entries ]
```

This code create the HourOfDay function which serves to extract the hour from the timestamp.

#Question 5 tripdata_df1<-tripdata_df %>% mutate(lpep_pickup_hour = HourOfDay(lpep_pickup_datetime))

This code applies the HourOfDay function to create a new column that is the hour of day from each listed time stamp.

```
#Question 6
# plot df
tripdata_df1 %>% group_by(lpep_pickup_hour) %>% summarise(med_trip_distance=median(trip_distance)) %>%
  ggplot(aes(lpep_pickup_hour, med_trip_distance)) + geom_col() +
  scale_x_continuous(breaks=seq(1,24,1)) +
  labs(title='Median Trip Distance',
       y="Median Trip Distance", x="Hour of Day (Pickup)")
```



This graph displays the Median Trip Distance grouped by the hour of day of the taxi cab pickup for February 2020. The most common hour for the longest medium trip distance rides is 06:00 AM EST followed closely by 05:00 AM EST. Outside of the 5-6AM peak, the remianing 22 hours have similar median trip distances around 2 miles. This is likely because 5-6AM coincides with overnight transit construction and people need to take cabs further.

‘‘‘