DA5020.A4.Hsiao-Yu.Peng

Hsiao-Yu Peng

2023-10-01

Bonus 1. See the attached DataCamp certificates.

Bonus 2.

```
# load dataset
tripdata_df <- read_csv("~/Desktop/2023Fall_Syllabus/DA5020/week3/2018_Green_Taxi_Trip_Data.csv")
```

```
## Warning: One or more parsing issues, call `problems()` on your data frame for details,
## e.g.:
##   dat <- vroom(...)
##   problems(dat)
```

```
## Rows: 1048575 Columns: 19
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): lpep_pickup_datetime, lpep_dropoff_datetime, store_and_fwd_flag
## dbl (15): VendorID, RatecodeID, PULocationID, DOLocationID, passenger_count,...
## lgl  (1): ehail_fee
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Add a "pickup" variable which is from "lpep_pickup_datetime" varialbe, convert "picup" data type to D
tripdata_df <- tripdata_df %>%
  mutate(pickup = as.Date(lpep_pickup_datetime, format = "%m/%d/%Y"))


tripdata_df %>%
  group_by(pickup) %>%
  summarize(count = n()) %>%
  arrange(desc(count))
```

```
## # A tibble: 69 x 2
##    pickup     count
##    <date>     <int>
##  1 2018-02-02 33953
##  2 2018-02-03 31272
##  3 2018-01-13 31203
##  4 2018-01-26 30818
```

```
##  5 2018-01-27 30599
##  6 2018-01-12 30462
##  7 2018-01-20 30276
##  8 2018-01-19 30262
##  9 2018-02-09 29853
## 10 2018-02-08 28548
## # i 59 more rows
```

2018-02-02, recorded the highest number of trips, with a total of 33,953 trips on that day. Upon further investigation of date in history, it shows that Fridays and Saturdays have significantly more trips than other weekdays.

Q1.

```
glimpse(tripdata_df)
```

```
## Rows: 1,048,575
## Columns: 20
## $ VendorID              <dbl> 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, ~
## $ lpep_pickup_datetime  <chr> "1/1/2018 0:18", "1/1/2018 0:30", "1/1/2018 0:07~
## $ lpep_dropoff_datetime <chr> "1/1/2018 0:24", "1/1/2018 0:46", "1/1/2018 0:19~
## $ store_and_fwd_flag    <chr> "N", "N", "N", "N", "N", "N", "N", "N", "N", "N"~
## $ RatecodeID            <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ PULocationID          <dbl> 236, 43, 74, 255, 255, 255, 189, 189, 129, 226, ~
## $ DOLocationID          <dbl> 236, 42, 152, 255, 255, 161, 65, 225, 82, 7, 129~
## $ passenger_count       <dbl> 5, 5, 1, 1, 1, 1, 5, 5, 1, 1, 2, 2, 1, 1, 2, 1, ~
## $ trip_distance         <dbl> 0.70, 3.50, 2.14, 0.03, 0.03, 5.63, 1.71, 3.45, ~
## $ fare_amount           <dbl> 6.0, 14.5, 10.0, -3.0, 3.0, 21.0, 8.5, 14.5, 10.~
## $ extra                 <dbl> 0.5, 0.5, 0.5, -0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.~
## $ mta_tax               <dbl> 0.5, 0.5, 0.5, -0.5, 0.5, 0.5, 0.5, 0.5, 0.5, 0.~
## $ tip_amount            <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 3.16, ~
## $ tolls_amount          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ ehail_fee             <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
## $ improvement_surcharge <dbl> 0.3, 0.3, 0.3, -0.3, 0.3, 0.3, 0.3, 0.3, 0.3, 0.~
## $ total_amount          <dbl> 7.30, 15.80, 11.30, -4.30, 4.30, 22.30, 9.80, 18~
## $ payment_type          <dbl> 2, 2, 2, 3, 2, 2, 2, 1, 2, 2, 2, 2, 2, 1, 1, 1, ~
## $ trip_type             <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ pickup                <date> 2018-01-01, 2018-01-01, 2018-01-01, 2018-01-01,~
```

```
# Check documenatation. VendorID, ratecodeID, payment_type, trip_type are factor variables, we can cove
variables_cat <- c("VendorID", "RatecodeID", "payment_type", "trip_type")

# Use lapply to convert the categorical variables
tripdata_df[variables_cat] <- lapply(tripdata_df[variables_cat], as.factor)

# Check the levels of each variable's factor
for (var in variables_cat) {
  cat(sprintf("%s factor is: %s\n", var, paste(levels(tripdata_df[[var]]), collapse = ", ")))
}
```

```
## VendorID factor is: 1, 2
```

```
## RatecodeID factor is: 1, 2, 3, 4, 5, 6, 99
## payment_type factor is: 1, 2, 3, 4, 5
## trip_type factor is: 1, 2
```

The dataset has dimension 1,048,575 x 20. The data type of "lpep_pickup_datetime" and "lpep_dropoff_datetime" are character, not datetime. Other categorical variables like VendorID, RatecodeID, trip_type, and payment_type have "numerical" datatype. So we convert them in factor type here.

It's worth noting that the RatecodeID factor includes the value '99,' which is considered invalid in the dataset.

Q2

```
# Analyze Trip Types (Hailing Method)
trip_type_analysis <- tripdata_df %>%
  group_by(trip_type) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

# Analyze Payment Types
payment_type_analysis <- tripdata_df %>%
  group_by(payment_type) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

trip_type_analysis
```

```
## # A tibble: 3 x 2
##   trip_type   count
##   <fct>       <int>
## 1 1         1030020
## 2 2           18552
## 3 <NA>            3
```

```
payment_type_analysis
```

```
## # A tibble: 5 x 2
##   payment_type  count
##   <fct>         <int>
## 1 1            564347
## 2 2            476995
## 3 3              5032
## 4 4              2133
## 5 5                68
```

```
# Analyze what payment type is the most when people hail a cab
payment_type_hailcab <- tripdata_df %>%
  filter(trip_type == 1) %>%
  group_by(payment_type) %>%
  summarize(counts = n())

payment_type_hailcab
```

```
## # A tibble: 5 x 2
##   payment_type counts
##   <fct>         <int>
## 1 1            557317
## 2 2            465880
## 3 3              4796
## 4 4              1967
## 5 5                60
```

Trip type 1, which is street-hail, is the most common way people hail a cab, with a total count of 1,030,020 trips.

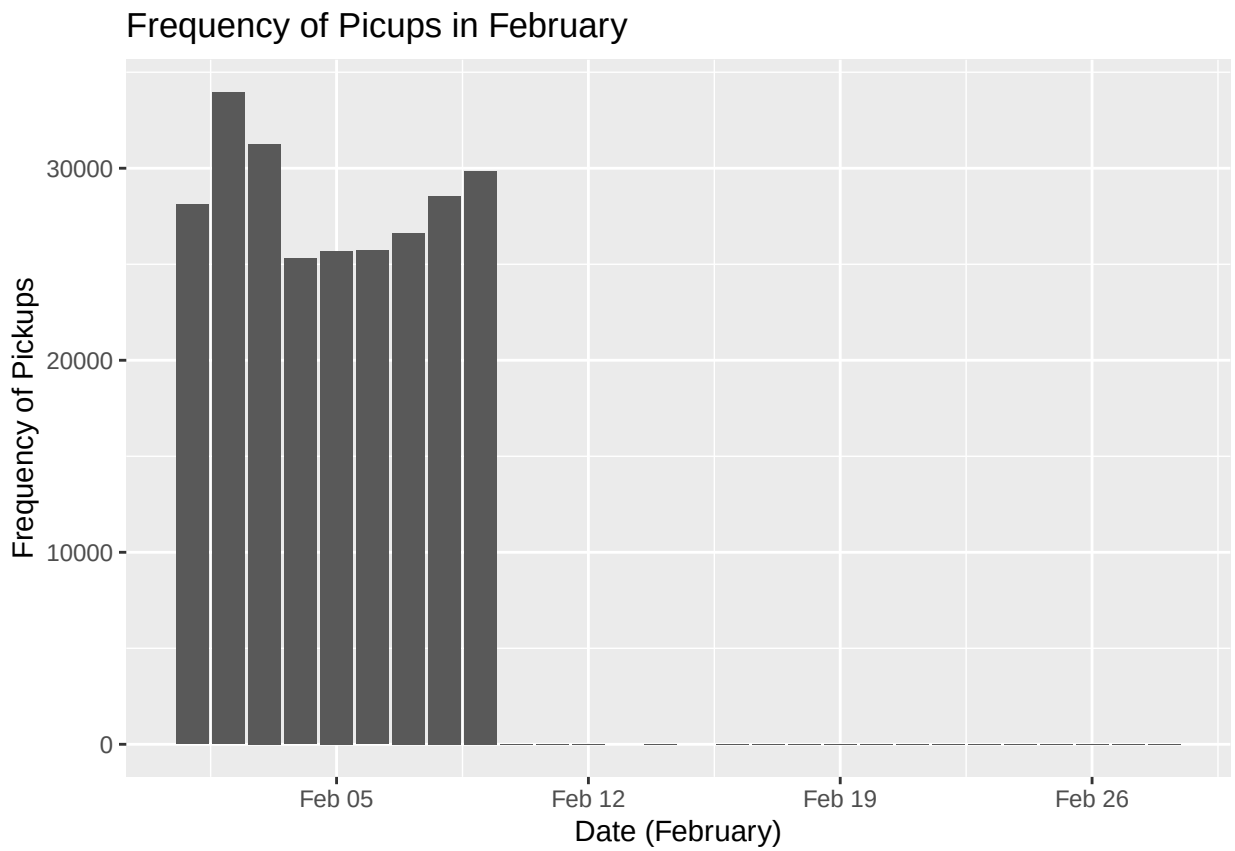
Payment type 1, which is credit card, is the most common way people pay for their trips, with a total count of 564,347.

In further analysis, we found Payment_type 1 (credit_card) is the most common method when people use "street-hail", with a total count of 557,317.

Q3

```
# February pickups
feb_pickups <- tripdata_df %>%
  filter(pickup >= "2018-02-01" & pickup < "2018-03-01") %>%
  group_by(pickup) %>%
  summarize(count = n())

# barplot
ggplot(feb_pickups, aes(x = pickup, y = count)) +
  geom_bar(stat = "identity") +
  labs(x = "Date (February)", y = "Frequency of Pickups") +
  ggtitle("Frequency of Picups in February")
```



From the bar plot, we observed that the majority of pickups occurred during the early part of February, specifically from February 1st to February 9th. After February 10th, the pickup frequency dramatically decreased to single digits, such as 3 or 6, and remained low.

Q4

```
HourOfDay <- function(timestamp){

  # Use regular expression to extract the hour (HH) part
  hour <- sub(".*\\s(\\d{1,2}):.*", "\\1", timestamp)
  return(hour)
}

# Example
timestamp <- "9/23/2010 11:17"
hour <- HourOfDay(timestamp)
cat("Hour:", hour, "\n")
```

```
## Hour: 11
```

Q5

```r
# Extract Hour element in all Pickup date
tripdata_df <- tripdata_df %>%
  mutate(lpep_pickup_hour = HourOfDay(lpep_pickup_datetime))

tripdata_df %>%
  select(lpep_pickup_hour) %>%
  head()
```

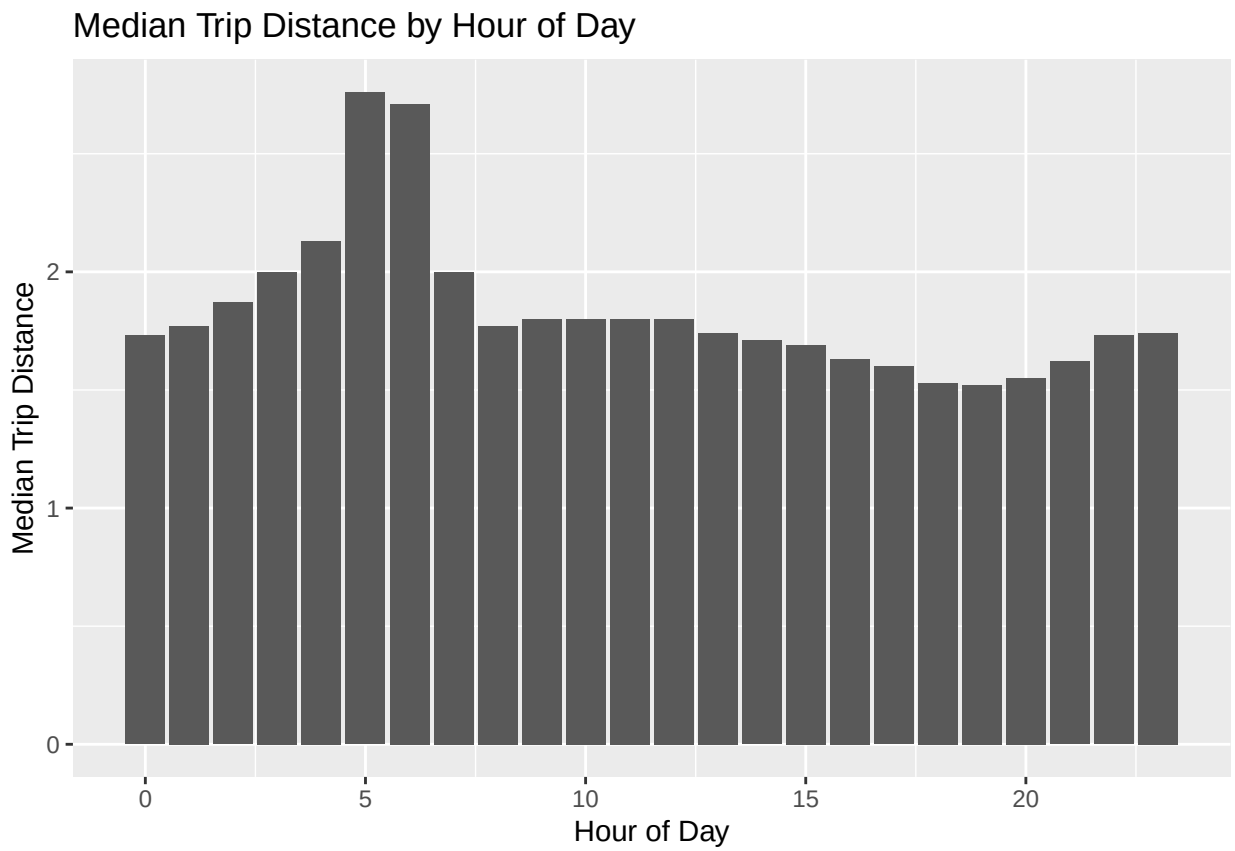
```
## # A tibble: 6 x 1
##   lpep_pickup_hour
##   <chr>
## 1 0
## 2 0
## 3 0
## 4 0
## 5 0
## 6 0
```

Q6

```r
median_distance <- tripdata_df %>%
  group_by(lpep_pickup_hour) %>%
  summarize(median_dist = median(trip_distance))


# Convert "lpep_pickup_hour" to numeric type
median_distance$lpep_pickup_hour <- as.numeric(median_distance$lpep_pickup_hour)

# Create the bar plot
ggplot(median_distance, aes(x = lpep_pickup_hour, y = median_dist)) +
  geom_bar(stat = "identity") +
  labs(x = "Hour of Day", y = "Median Trip Distance") +
  ggtitle("Median Trip Distance by Hour of Day")
```



The longest median trip distance occurs during pickups at 5 AM and 6 AM in the morning. It then gradually decreases as the day progresses, reaching its shortest median trip distance at 19:00. After 20:00, the median distance begins to gradullay increase once again.