Cleaning Cylistic Data 2023-05

2023-07-31

Import data

```
data_01 <- read.csv(file="dataset/202305-divvy-tripdata.csv")
```

Check data 01

Check the data type for each meta

```
str(data_01)
```

```
## 'data.frame':    604827 obs. of  13 variables:
##  $ ride_id           : chr  "0D9FA920C3062031" "92485E5FB5888ACD" "FB144B3FC8300187" "DDEB93BC2CE9AA
##  $ rideable_type     : chr  "electric_bike" "electric_bike" "electric_bike" "classic_bike" ...
##  $ started_at        : chr  "2023-05-07 19:53:48" "2023-05-06 18:54:08" "2023-05-21 00:40:21" "2023-
##  $ ended_at          : chr  "2023-05-07 19:58:32" "2023-05-06 19:03:35" "2023-05-21 00:44:36" "2023-
##  $ start_station_name: chr  "Southport Ave & Belmont Ave" "Southport Ave & Belmont Ave" "Halsted St &
##  $ start_station_id  : chr  "13229" "13229" "13162" "13196" ...
##  $ end_station_name  : chr  "" "" "" "Damen Ave & Cortland St" ...
##  $ end_station_id    : chr  "" "" "" "13133" ...
##  $ start_lat         : num  41.9 41.9 41.9 41.9 42 ...
##  $ start_lng         : num  -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ end_lat           : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num  -87.7 -87.7 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr  "member" "member" "member" "member" ...
```

```
summary(data_01)
```

```
##    ride_id          rideable_type       started_at          ended_at
##  Length:604827      Length:604827      Length:604827      Length:604827
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  start_station_name start_station_id   end_station_name   end_station_id
##  Length:604827      Length:604827      Length:604827      Length:604827
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
```

```
##
##
##
##    start_lat       start_lng         end_lat         end_lng
##  Min.   :41.65   Min.   :-87.87   Min.   :41.62   Min.   :-87.91
##  1st Qu.:41.88   1st Qu.:-87.66   1st Qu.:41.88   1st Qu.:-87.66
##  Median :41.90   Median :-87.64   Median :41.90   Median :-87.64
##  Mean   :41.90   Mean   :-87.65   Mean   :41.90   Mean   :-87.65
##  3rd Qu.:41.93   3rd Qu.:-87.63   3rd Qu.:41.93   3rd Qu.:-87.63
##  Max.   :42.07   Max.   :-87.53   Max.   :42.11   Max.   :-87.53
##                                   NA's   :710     NA's   :710
##  member_casual
##  Length:604827
##  Class :character
##  Mode  :character
##
##
##
##
```

From meta check we know that data type of column "started_at" and "end_at" should be datetime

Check duplicate data 01

```
print(data_01[duplicated(data_01), ])
```

```
##  [1] ride_id            rideable_type      started_at         ended_at
##  [5] start_station_name start_station_id   end_station_name   end_station_id
##  [9] start_lat          start_lng          end_lat            end_lng
## [13] member_casual
## <0 rows> (or 0-length row.names)
```

Duplicate data checking result : no data duplicate in data_01

Remove duplicate data

Remove Duplicate data result : No data to remove

Check missing value data in character data type

```
count(data_01[is.na(data_01$ride_id) | data_01$ride_id=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[is.na(data_01$rideable_type) | data_01$rideable_type=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[is.na(data_01$started_at) | data_01$started_at=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[is.na(data_01$ended_at) | data_01$ended_at=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[data_01$start_station_name=="", ])
```

```
##       n
## 1 89240
```

```
count(data_01[data_01$start_station_id=="", ])
```

```
##       n
## 1 89240
```

```
count(data_01[data_01$end_station_name=="", ])
```

```
##       n
## 1 95267
```

```
count(data_01[data_01$end_station_id=="", ])
```

```
##       n
## 1 95267
```

```
count(data_01[is.na(data_01$member_casual) | data_01$member_casual=="", ])
```

```
##   n
## 1 0
```

Missing value checking result :

ride_id: [0]

rideable_type: [0]

started_at: [0]

ended_at: [0]

start_station_name: [89,240]

start_station_id: [89,240]

end_station_name: [95,267]

end_station_id: [95,267]

member_casual: [0]

Fill Missing value with NA

Missing value (empty data) in start_station_name, start_station_id, end_station_name, end_station_id will be filling with NA

```
data_01 <- replace(data_01, data_01 == "", NA)
```

Fill missing value result : empty data was replace with NA

Check missing value data

```
count(data_01[is.na(data_01$start_lat) | data_01$start_lat=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[is.na(data_01$start_lng) | data_01$start_lng=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[is.na(data_01$end_lat) | data_01$end_lat=="", ])
```

```
##     n
## 1 710
```

```
count(data_01[is.na(data_01$end_lng) | data_01$end_lng=="", ])
```

```
##     n
## 1 710
```

Missing value checking result :

start latitude and langitude : [0]

end latitude and langitude : [710]

Remove Missing value with NA

Missing value in end_lat, end_lng will be delete by remove the row

```
# remove missing value data in this other data if there are also missing values
# data_01 <- data_01[!is.na(data_01$rideable_type), ]
# data_01 <- data_01[!is.na(data_01$started_at), ]
# data_01 <- data_01[!is.na(data_01$ended_at), ]
# data_01 <- data_01[!is.na(data_01$member_casual), ]

data_01 <- data_01[!is.na(data_01$end_lat), ]
data_01 <- data_01[!is.na(data_01$end_lng), ]

count(data_01[is.na(data_01$end_lat) | data_01$end_lat=="", ])
```

```
##   n
## 1 0
```

```
count(data_01[is.na(data_01$end_lng) | data_01$end_lng=="", ])
```

```
##   n
## 1 0
```

Remove missing value result : Row with missing value data was removed

Check outliers in coordinate data

```
print(cat("start_lat : mean max min : ",
    mean(data_01$start_lat),
    max(data_01$start_lat),
    min(data_01$start_lat)))
```

```
## start_lat : mean max min :  41.90341 42.07 41.6485NULL
```

```
print(cat("start_lng : mean max min : ",
    mean(data_01$start_lng), max(data_01$start_lng), min(data_01$start_lng)))
```

```
## start_lng : mean max min :  -87.64716 -87.52823 -87.87NULL
```

```
print(cat("end_lat : mean max min : ",
    mean(data_01$end_lat), max(data_01$end_lat), min(data_01$end_lat)))
```

```
## end_lat : mean max min :  41.90385 42.11 41.62NULL
```

```
print(cat("end_lng : mean max min : ",
    mean(data_01$end_lng), max(data_01$end_lng), min(data_01$end_lng)))
```

```
## end_lng : mean max min :  -87.64732 -87.52823 -87.91NULL
```

Outliers checking result : no outliers in coordinate data, max and min value for each data doesnt far from average value

Remove useless column data

Acording to the bussines task, start_station_name and end_station_name will be remove

```
data_01 <- data_01[, -which(names(data_01) == "start_station_name")]
data_01 <- data_01[, -which(names(data_01) == "end_station_name")]

head(data_01)
```

```
##            ride_id rideable_type          started_at            ended_at
## 1 0D9FA920C3062031 electric_bike 2023-05-07 19:53:48 2023-05-07 19:58:32
## 2 92485E5FB5888ACD electric_bike 2023-05-06 18:54:08 2023-05-06 19:03:35
## 3 FB144B3FC8300187 electric_bike 2023-05-21 00:40:21 2023-05-21 00:44:36
## 4 DDEB93BC2CE9AA77  classic_bike 2023-05-10 16:47:01 2023-05-10 16:59:52
## 5 C07B70172FC92F59  classic_bike 2023-05-09 18:30:34 2023-05-09 18:39:28
## 6 2BA66385DF8F815A  classic_bike 2023-05-30 15:01:21 2023-05-30 15:17:00
##   start_station_id end_station_id start_lat start_lng  end_lat   end_lng
## 1            13229           <NA>  41.93941 -87.66383 41.93000 -87.65000
## 2            13229           <NA>  41.93948 -87.66385 41.94000 -87.69000
## 3            13162           <NA>  41.85379 -87.64672 41.86000 -87.65000
## 4            13196          13133  41.89456 -87.65345 41.91598 -87.67733
## 5    TA1308000047          13229  41.95708 -87.66420 41.93948 -87.66375
## 6    TA1305000032   TA1306000029  41.88275 -87.64119 41.89259 -87.61729
##   member_casual
## 1        member
## 2        member
## 3        member
## 4        member
## 5        member
## 6        member
```

```
str(data_01)
```

```
## 'data.frame':    604117 obs. of  11 variables:
##  $ ride_id         : chr  "0D9FA920C3062031" "92485E5FB5888ACD" "FB144B3FC8300187" "DDEB93BC2CE9AA77" ...
##  $ rideable_type   : chr  "electric_bike" "electric_bike" "electric_bike" "classic_bike" ...
##  $ started_at      : chr  "2023-05-07 19:53:48" "2023-05-06 18:54:08" "2023-05-21 00:40:21" "2023-05-
##  $ ended_at        : chr  "2023-05-07 19:58:32" "2023-05-06 19:03:35" "2023-05-21 00:44:36" "2023-05-
##  $ start_station_id: chr  "13229" "13229" "13162" "13196" ...
##  $ end_station_id  : chr  NA NA NA "13133" ...
##  $ start_lat       : num  41.9 41.9 41.9 41.9 42 ...
##  $ start_lng       : num  -87.7 -87.7 -87.6 -87.7 -87.7 ...
##  $ end_lat         : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng         : num  -87.7 -87.7 -87.7 -87.7 -87.7 ...
##  $ member_casual   : chr  "member" "member" "member" "member" ...
```

Export clean data into csv

```
# write.csv(data_01, "dataclean/202305-clean.csv", row.names = FALSE)
```