Cleaning Cylistic Data 2022-09

2023-07-31

Import data

```
data_01 <- read.csv(file="dataset/202209-divvy-tripdata.csv")
```

Check data 01

Check the data type for each meta

```
str(data_01)
```

```
## 'data.frame':    701339 obs. of  13 variables:
##  $ ride_id           : chr  "5156990AC19CA285" "E12D4A16BF51C274" "A02B53CD7DB72DD7" "C82E05FEE872DF
##  $ rideable_type     : chr  "electric_bike" "electric_bike" "electric_bike" "electric_bike" ...
##  $ started_at        : chr  "2022-09-01 08:36:22" "2022-09-01 17:11:29" "2022-09-01 17:15:50" "2022-0
##  $ ended_at          : chr  "2022-09-01 08:39:05" "2022-09-01 17:14:45" "2022-09-01 17:16:12" "2022-0
##  $ start_station_name: chr  "" "" "" "" ...
##  $ start_station_id  : chr  "" "" "" "" ...
##  $ end_station_name  : chr  "California Ave & Milwaukee Ave" "" "" "" ...
##  $ end_station_id    : chr  "13084" "" "" "" ...
##  $ start_lat         : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng         : num  -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ end_lat           : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng           : num  -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ member_casual     : chr  "casual" "casual" "casual" "casual" ...
```

```
summary(data_01)
```

```
##    ride_id          rideable_type       started_at          ended_at
##  Length:701339      Length:701339      Length:701339      Length:701339
##  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:701339      Length:701339      Length:701339      Length:701339
##  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.   :-87.84   Min.   :41.55   Min.   :-87.92
##  1st Qu.:41.88   1st Qu.:-87.66   1st Qu.:41.88   1st Qu.:-87.66
##  Median :41.90   Median :-87.65   Median :41.90   Median :-87.65
##  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.   :42.15   Max.   :-87.30
##                                   NA's   :712     NA's   :712
##  member_casual
##  Length:701339
##  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[data_01$ride_id=="", ])
```

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

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

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

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

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

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

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

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

```
##        n
## 1 103780
```

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

```
##        n
## 1 103780
```

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

```
##        n
## 1 111185
```

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

```
##        n
## 1 111185
```

```
count(data_01[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: [103,789] start_station_id: [103,780] end_station_name: [111,185] end_station_id: [111,185] 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
## 1 712
```

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

```
##     n
## 1 712
```

Missing value checking result :

start latitude and langitude : [0] end latitude and langitude : [712]

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
```

```r
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

```r
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 :  41.90406 42.07 41.6485NULL
```

```r
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 :  -87.64853 -87.52823 -87.84NULL
```

```r
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 :  41.90426 42.15 41.55NULL
```

```r
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 :  -87.6488 -87.3 -87.92NULL
```

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

```r
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 5156990AC19CA285 electric_bike 2022-09-01 08:36:22 2022-09-01 08:39:05
## 2 E12D4A16BF51C274 electric_bike 2022-09-01 17:11:29 2022-09-01 17:14:45
## 3 A02B53CD7DB72DD7 electric_bike 2022-09-01 17:15:50 2022-09-01 17:16:12
## 4 C82E05FEE872DF11 electric_bike 2022-09-01 09:00:28 2022-09-01 09:10:32
## 5 4DEEB4550A266AE1 electric_bike 2022-09-01 07:30:11 2022-09-01 07:32:36
## 6 B1721F8C7C3AC6BF electric_bike 2022-09-01 12:04:25 2022-09-01 12:21:08
##   start_station_id end_station_id start_lat start_lng  end_lat   end_lng
## 1             <NA>          13084     41.93    -87.69 41.92269 -87.69715
## 2             <NA>           <NA>     41.87    -87.62 41.87000 -87.62000
## 3             <NA>           <NA>     41.87    -87.62 41.87000 -87.62000
## 4             <NA>           <NA>     41.93    -87.69 41.94000 -87.67000
## 5             <NA>           <NA>     41.92    -87.73 41.92000 -87.73000
## 6             <NA>           <NA>     41.89    -87.65 41.92000 -87.67000
##   member_casual
## 1        casual
## 2        casual
## 3        casual
## 4        casual
## 5        casual
## 6        casual
```

```
str(data_01)
```

```
## 'data.frame':    700627 obs. of  11 variables:
##  $ ride_id         : chr  "5156990AC19CA285" "E12D4A16BF51C274" "A02B53CD7DB72DD7" "C82E05FEE872DF11" ...
##  $ rideable_type   : chr  "electric_bike" "electric_bike" "electric_bike" "electric_bike" ...
##  $ started_at      : chr  "2022-09-01 08:36:22" "2022-09-01 17:11:29" "2022-09-01 17:15:50" "2022-09-
##  $ ended_at        : chr  "2022-09-01 08:39:05" "2022-09-01 17:14:45" "2022-09-01 17:16:12" "2022-09-
##  $ start_station_id: chr  NA NA NA NA ...
##  $ end_station_id  : chr  "13084" NA NA NA ...
##  $ start_lat       : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng       : num  -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ end_lat         : num  41.9 41.9 41.9 41.9 41.9 ...
##  $ end_lng         : num  -87.7 -87.6 -87.6 -87.7 -87.7 ...
##  $ member_casual   : chr  "casual" "casual" "casual" "casual" ...
```

Export clean data into csv

```
write.csv(data_01, "dataclean/202209-clean.csv", row.names = FALSE)
```