Cleaning Cylistic Data 2022-08

2023-07-31

Import data

```
data_01 <- read.csv(file="dataset/202208-divvy-tripdata.csv")
```

Check data 01

Check the data type for each meta

```
str(data_01)
```

```
## 'data.frame':    785932 obs. of  13 variables:
##  $ ride_id           : chr  "550CF7EFEAE0C618" "DAD198F405F9C5F5" "E6F2BC47B65CB7FD" "F597830181C2E1
##  $ rideable_type     : chr  "electric_bike" "electric_bike" "electric_bike" "electric_bike" ...
##  $ started_at        : chr  "2022-08-07 21:34:15" "2022-08-08 14:39:21" "2022-08-08 15:29:50" "2022-
##  $ ended_at          : chr  "2022-08-07 21:41:46" "2022-08-08 14:53:23" "2022-08-08 15:40:34" "2022-
##  $ start_station_name: chr  "" "" "" "" ...
##  $ start_station_id  : chr  "" "" "" "" ...
##  $ end_station_name  : chr  "" "" "" "" ...
##  $ end_station_id    : chr  "" "" "" "" ...
##  $ start_lat         : num  41.9 41.9 42 41.9 41.9 ...
##  $ start_lng         : num  -87.7 -87.6 -87.7 -87.7 -87.7 ...
##  $ end_lat           : num  41.9 41.9 42 42 41.8 ...
##  $ end_lng           : num  -87.7 -87.6 -87.7 -87.7 -87.7 ...
##  $ member_casual     : chr  "casual" "casual" "casual" "casual" ...
```

```
summary(data_01)
```

```
##    ride_id          rideable_type       started_at          ended_at
##  Length:785932      Length:785932      Length:785932      Length:785932
##  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:785932      Length:785932      Length:785932      Length:785932
##  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.   :41.60   Min.   :-88.05
##  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   :41.91   Mean   :-87.65   Mean   :41.91   Mean   :-87.65
##  3rd Qu.:41.93   3rd Qu.:-87.63   3rd Qu.:41.93   3rd Qu.:-87.63
##  Max.   :42.07   Max.   :-87.52   Max.   :42.12   Max.   :-87.50
##                                   NA's   :843     NA's   :843
##  member_casual
##  Length:785932
##  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

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 112037
```

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

```
##        n
## 1 112037
```

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

```
##        n
## 1 120522
```

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

```
##        n
## 1 120522
```

```
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: [112,037] start_station_id: [112,037] end_station_name: [120,522] end_station_id: [120,522] 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
## 1 843
```

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

```
##     n
## 1 843
```

Missing value checking result :

start latitude and langitude : [0] end latitude and langitude : [843]

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 :  41.90526 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 :  -87.64812 -87.52 -87.84NULL
```

```
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 :  41.90549 42.12 41.6NULL
```

```
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 :  -87.64836 -87.5 -88.05NULL
```

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 550CF7EFEAE0C618 electric_bike 2022-08-07 21:34:15 2022-08-07 21:41:46
## 2 DAD198F405F9C5F5 electric_bike 2022-08-08 14:39:21 2022-08-08 14:53:23
## 3 E6F2BC47B65CB7FD electric_bike 2022-08-08 15:29:50 2022-08-08 15:40:34
## 4 F597830181C2E13C electric_bike 2022-08-08 02:43:50 2022-08-08 02:58:53
## 5 0CE689BB4E313E8D electric_bike 2022-08-07 20:24:06 2022-08-07 20:29:58
## 6 BFA7E7CC69860C20 electric_bike 2022-08-08 13:06:08 2022-08-08 13:19:09
##   start_station_id end_station_id start_lat start_lng end_lat end_lng
## 1             <NA>           <NA>     41.93    -87.69   41.94  -87.72
## 2             <NA>           <NA>     41.89    -87.64   41.92  -87.64
## 3             <NA>           <NA>     41.97    -87.69   41.97  -87.66
## 4             <NA>           <NA>     41.94    -87.65   41.97  -87.69
## 5             <NA>           <NA>     41.85    -87.65   41.84  -87.66
## 6             <NA>           <NA>     41.79    -87.72   41.82  -87.69
##   member_casual
## 1        casual
## 2        casual
## 3        casual
## 4        casual
## 5        casual
## 6        casual
```

```
str(data_01)
```

```
## ’data.frame’:    785089 obs. of  11 variables:
##  $ ride_id         : chr  "550CF7EFEAE0C618" "DAD198F405F9C5F5" "E6F2BC47B65CB7FD" "F597830181C2E13C
##  $ rideable_type   : chr  "electric_bike" "electric_bike" "electric_bike" "electric_bike" ...
##  $ started_at      : chr  "2022-08-07 21:34:15" "2022-08-08 14:39:21" "2022-08-08 15:29:50" "2022-08
##  $ ended_at        : chr  "2022-08-07 21:41:46" "2022-08-08 14:53:23" "2022-08-08 15:40:34" "2022-08
##  $ start_station_id: chr  NA NA NA NA ...
##  $ end_station_id  : chr  NA NA NA NA ...
##  $ start_lat       : num  41.9 41.9 42 41.9 41.9 ...
##  $ start_lng       : num  -87.7 -87.6 -87.7 -87.7 -87.7 ...
##  $ end_lat         : num  41.9 41.9 42 42 41.8 ...
##  $ end_lng         : num  -87.7 -87.6 -87.7 -87.7 -87.7 ...
##  $ member_casual   : chr  "casual" "casual" "casual" "casual" ...
```

Export clean data into csv

```
write.csv(data_01, "dataclean/202208-clean.csv", row.names = FALSE)
```