EDA_615_final

Hui Xiong

2022-12-14

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(plotly)
```

```
## Loading required package: ggplot2

##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

```
library(fmsb)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --

## v tibble  3.1.8     v purrr   0.3.5
## v tidyr   1.2.1     v stringr 1.4.1
## v readr   2.1.3     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x plotly::filter() masks dplyr::filter(), stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
##Subway
LR_2021_Q4 <- read.csv("~/Desktop/MA615-Final/Travel_Times_2021 (2)/LRTravelTimesQ4_21.csv",header = T)

HR_2021_Q4 <- read.csv("~/Desktop/MA615-Final/Travel_Times_2021 (2)/HRTravelTimesQ4_21.csv",header = T)

LR_2022_Q1 <- read.csv("~/Desktop/MA615-Final/TravelTimes_2022/2022-Q1_LRTravelTimes.csv",header = T)

HR_2022_Q1 <- read.csv("~/Desktop/MA615-Final/TravelTimes_2022/2022-Q1_HRTravelTimes.csv",header = T)

LR_2022_Q2 <- read.csv("~/Desktop/MA615-Final/TravelTimes_2022/2022-Q2_LRTravelTimes.csv",header = T)

HR_2022_Q2 <- read.csv("~/Desktop/MA615-Final/TravelTimes_2022/2022-Q2_HRTravelTimes.csv",header = T)

LR_2022_Q3 <- read.csv("~/Desktop/MA615-Final/TravelTimes_2022/2022-Q3_LRTravelTimes.csv",header = T)

HR_2022_Q3 <- read.csv("~/Desktop/MA615-Final/TravelTimes_2022/2022-Q3_HRTravelTimes.csv",header = T)



##Bus
bus_10 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2021/MBTA-Bus-Arrival-Departu

bus_11 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2021/MBTA-Bus-Arrival-Departu

bus_12 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2021/MBTA-Bus-Arrival-Departu

bus_01 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_02 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_03 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_04 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_05 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu
```

```r
bus_06 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_07 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_08 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu

bus_09 <- read.csv("~/Desktop/MA615-Final/MBTA_Bus_Arrival_Departure_Times_2022/MBTA-Bus-Arrival-Departu
```

```r
## subway choose date
# The data I chose is 20-26 per month

date2021_10TO12 <- c('2021-10-20', '2021-10-21', '2021-10-22', '2021-10-23', '2021-10-24', '2021-10-25'
HR2021_10TO12 <- HR_2021_Q4 %>% filter(service_date %in% date2021_10TO12)
LR2021_10TO12 <- LR_2021_Q4 %>% filter(service_date %in% date2021_10TO12)


date2022_01TO03 <- c('2022-01-20', '2022-01-21', '2022-01-22', '2022-01-23', '2022-01-24', '2022-01-25'
HR2022_01TO03 <- HR_2022_Q1 %>% filter(service_date %in% date2022_01TO03)
LR2022_01TO03 <- LR_2022_Q1 %>% filter(service_date %in% date2022_01TO03)


date2022_04TO06 <- c('2022-04-20', '2022-04-21', '2022-04-22', '2022-04-23', '2022-04-24', '2022-04-25'
HR2022_04TO06 <- HR_2022_Q2 %>% filter(service_date %in% date2022_04TO06)
LR2022_04TO06 <- LR_2022_Q2 %>% filter(service_date %in% date2022_04TO06)


date2022_07TO09 <- c('2022-07-20', '2022-07-21', '2022-07-22', '2022-07-23', '2022-07-24', '2022-07-25'
HR2022_07TO09 <- HR_2022_Q3 %>% filter(service_date %in% date2022_07TO09)
LR2022_07TO09 <- LR_2022_Q3 %>% filter(service_date %in% date2022_07TO09)


dataHR <- rbind(HR2021_10TO12, HR2022_01TO03, HR2022_04TO06, HR2022_07TO09)
dataLR <- rbind(LR2021_10TO12, LR2022_01TO03, LR2022_04TO06, LR2022_07TO09)

Subway_Alldata <- rbind(dataHR, dataLR)
```

```r
## Bus choose data

choose_bus_data <- c('2021-10-20', '2021-10-21', '2021-10-22', '2021-10-23', '2021-10-24', '2021-10-25'
                     '2021-11-20', '2021-11-21', '2021-11-22', '2021-11-23', '2021-11-24', '2021-11-25'
                     '2021-12-20', '2021-12-21', '2021-12-22', '2021-12-23', '2021-12-24', '2021-12-25'
                     '2022-01-20', '2022-01-21', '2022-01-22', '2022-01-23', '2022-01-24', '2022-01-25'
                     '2022-02-20', '2022-02-21', '2022-02-22', '2022-02-23', '2022-02-24', '2022-02-25'
                     '2022-03-20', '2022-03-21', '2022-03-22', '2022-03-23', '2022-03-24', '2022-03-25'
                     '2022-04-20', '2022-04-21', '2022-04-22', '2022-04-23', '2022-04-24', '2022-04-25'
                     '2022-05-20', '2022-05-21', '2022-05-22', '2022-05-23', '2022-05-24', '2022-05-25'
                     '2022-06-20', '2022-06-21', '2022-06-22', '2022-06-23', '2022-06-24', '2022-06-25'
                     '2022-07-20', '2022-07-21', '2022-07-22', '2022-07-23', '2022-07-24', '2022-07-25'
                     '2022-08-20', '2022-08-21', '2022-08-22', '2022-08-23', '2022-08-24', '2022-08-25'
                     '2022-09-20', '2022-09-21', '2022-09-22', '2022-09-23', '2022-09-24', '2022-09-25'
```

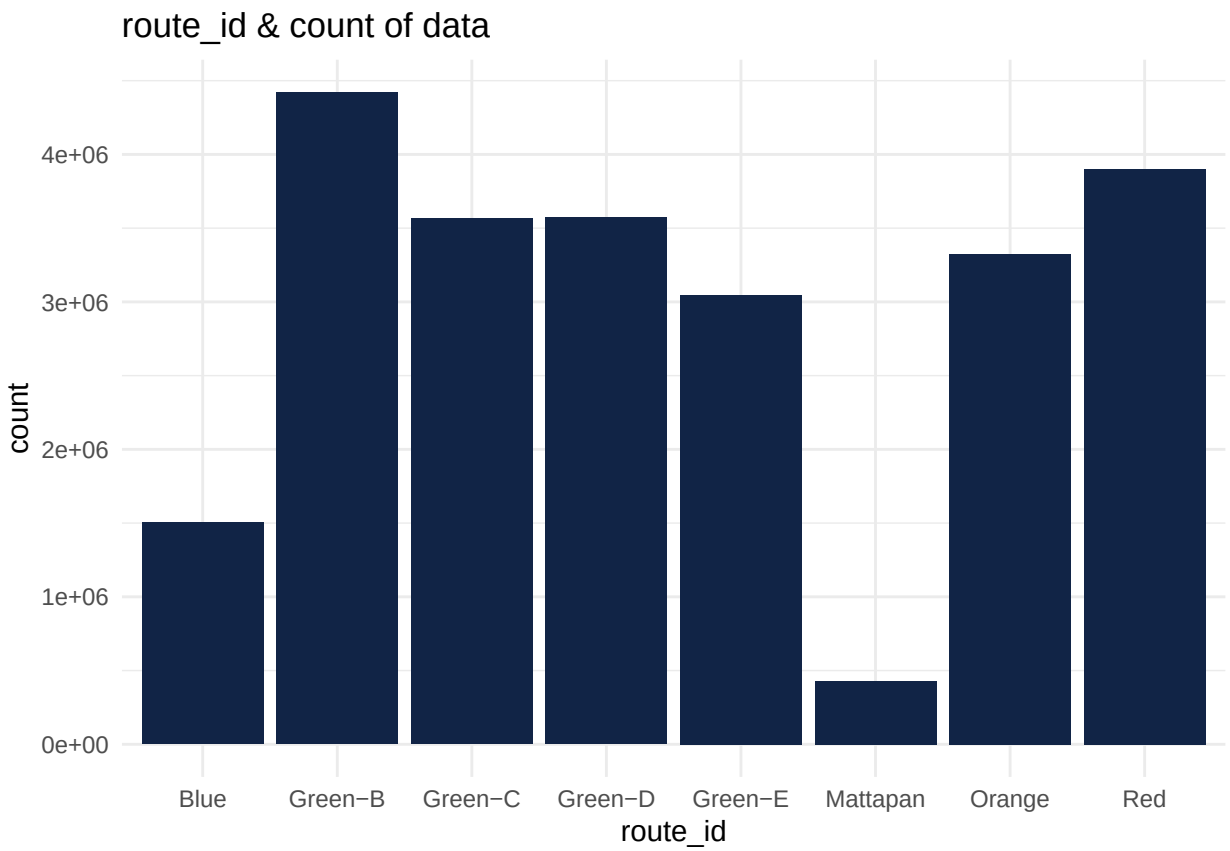
```
bus_10_new <- bus_10 %>% filter(service_date %in% choose_bus_data)
bus_11_new <- bus_11 %>% filter(service_date %in% choose_bus_data)
bus_12_new <- bus_12 %>% filter(service_date %in% choose_bus_data)
bus_01_new <- bus_01 %>% filter(service_date %in% choose_bus_data)
bus_02_new <- bus_02 %>% filter(service_date %in% choose_bus_data)
bus_03_new <- bus_03 %>% filter(service_date %in% choose_bus_data)
bus_04_new <- bus_04 %>% filter(service_date %in% choose_bus_data)
bus_05_new <- bus_05 %>% filter(service_date %in% choose_bus_data)
bus_06_new <- bus_06 %>% filter(service_date %in% choose_bus_data)
bus_07_new <- bus_07 %>% filter(service_date %in% choose_bus_data)
bus_08_new <- bus_08 %>% filter(service_date %in% choose_bus_data)
bus_09_new <- bus_09 %>% filter(service_date %in% choose_bus_data)

bus_all <- rbind(bus_10_new,bus_11_new,bus_12_new,bus_01_new,bus_02_new,bus_03_new,bus_04_new,bus_05_new

Bus_Alldata<- bus_all[order(bus_all$service_date),]
```

EDA

```
##Subway

unique(Subway_Alldata$route_id)
```

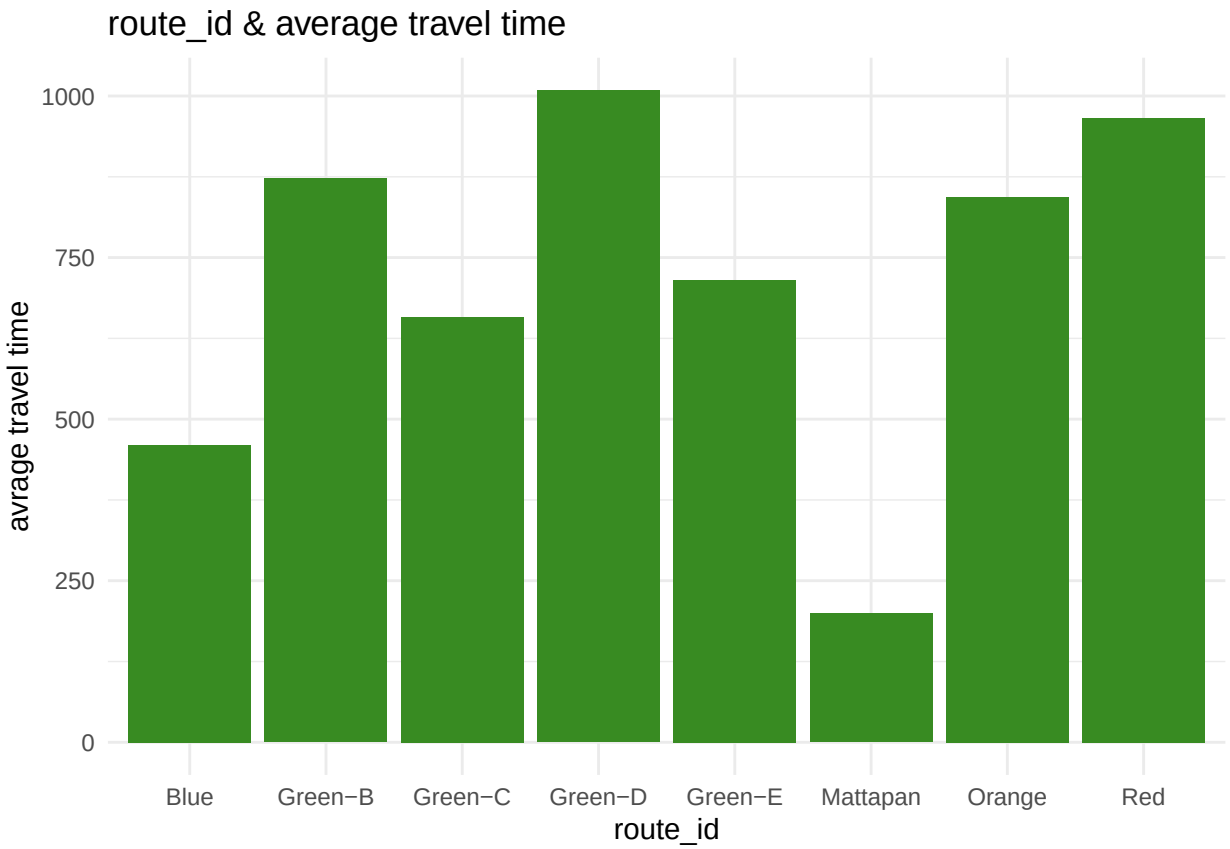
```
## [1] "Orange"   "Blue"     "Red"      "Green-B"  "Green-C"  "Green-D"  "Green-E"
## [8] "Mattapan"
```

```
data1_subway <- Subway_Alldata %>% group_by(route_id) %>% summarise(count = n())

ggplot(data1_subway) +
  aes(x = route_id, y = count) +
  geom_col(fill = "#112446") +
  labs(x = "route_id", y = "count",title = "route_id & count of data") +
  theme_minimal()
```



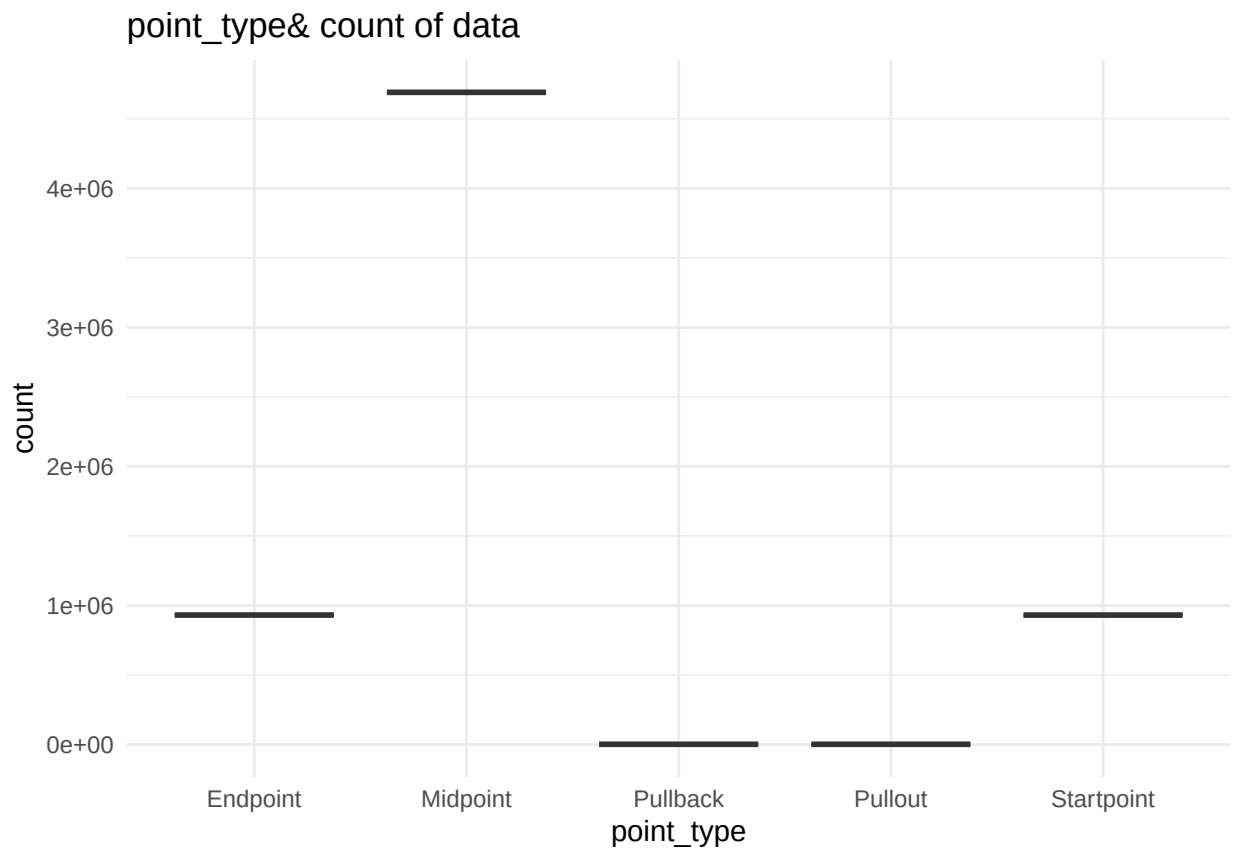

```
#As shown in the figure, Green-B has the highest, followed by red, and Green-C is similar to Green-D. T
```

```
##Subway

data2_subway <- Subway_Alldata %>% group_by(route_id) %>% summarise(average = mean(travel_time_sec))

ggplot(data2_subway) +
  aes(x = route_id, y = average) +
  geom_col(fill = "#388B22") +
  labs(
    x = "route_id",
    y = "avrage travel time",
    title = "route_id & average travel time"
  ) +
  theme_minimal()
```



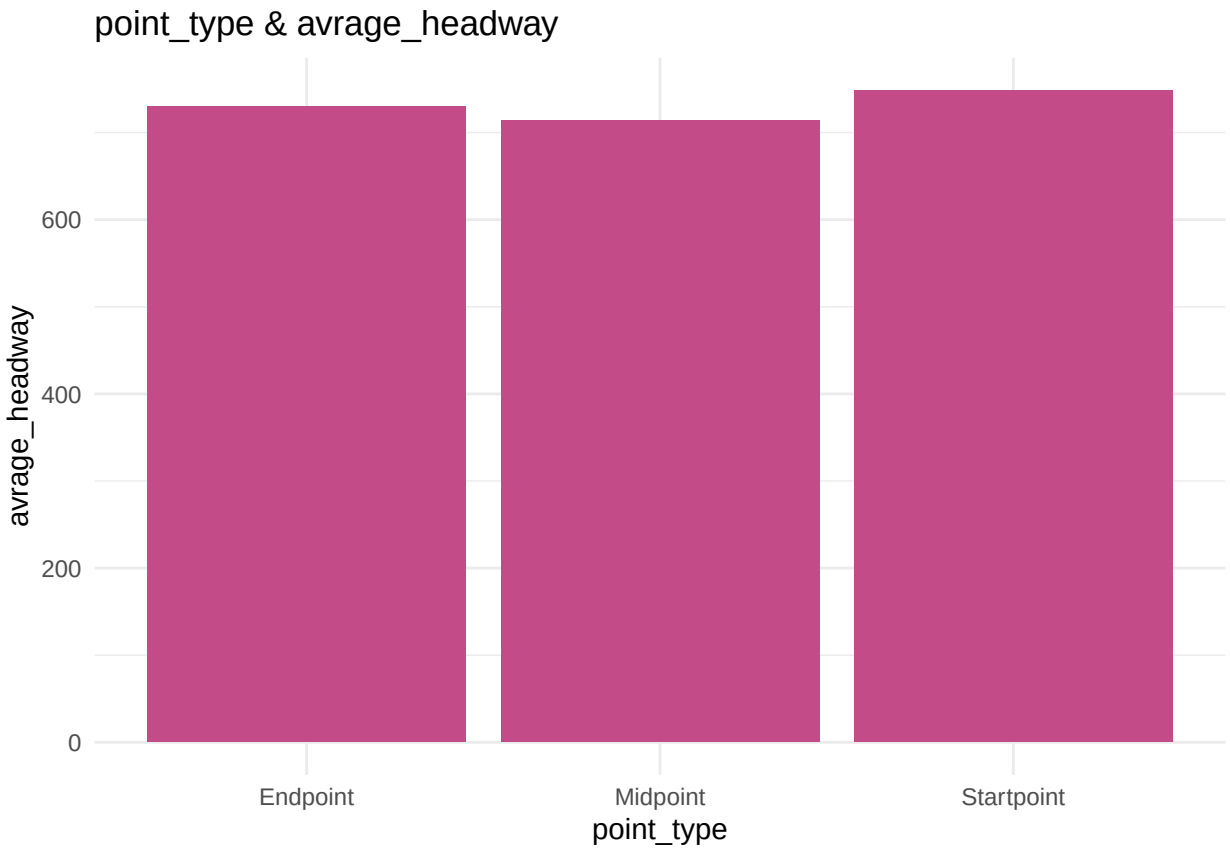

```
#The figure shows that Green-D has the highest average travel time, followed by red, and Green-B is abo
```

```
##Bus

Bus_Alldata<- bus_all[order(bus_all$service_date),]

data3_bus <- Bus_Alldata %>% group_by(point_type) %>% summarise(count = n())

ggplot(data3_bus) +
  aes(x = point_type, y = count) +
  geom_boxplot(fill = "#DC8A25") +
  labs(x = "point_type", y = "count", title = "point_type& count of data") +
  theme_minimal()
```



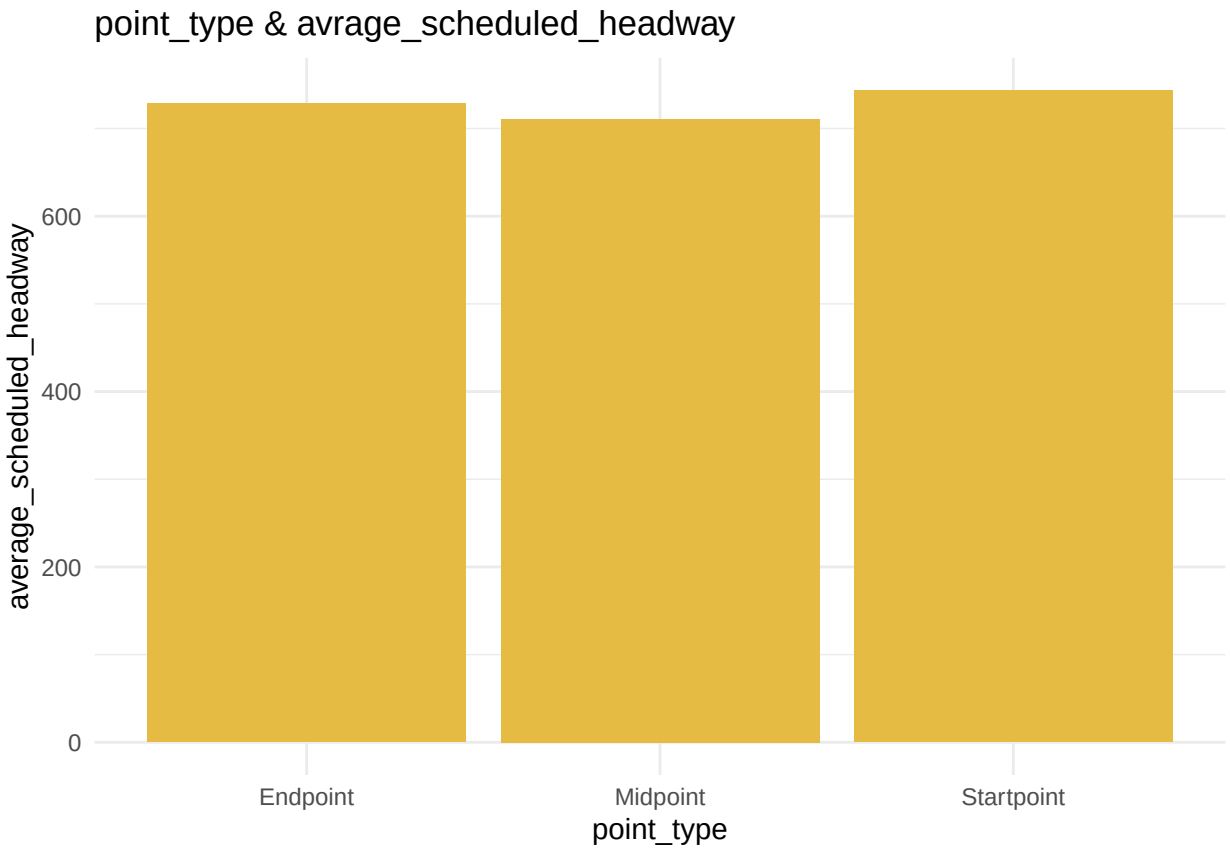

```
#The most point_type chosen is Midpoint, followed by similar Endpoint and Startpoint, and pullback and
```

```
newBus_Alldata<-Bus_Alldata[complete.cases(Bus_Alldata),]

data4_bus <- newBus_Alldata %>% group_by(point_type) %>% summarise(average_headway = mean(headway))

ggplot(data4_bus) +
  aes(x = point_type, y = average_headway) +
  geom_col(fill = "#C34C88") +
  labs(x = "point_type", y = "avrage_headway", title = "point_type & avrage_headway") +
  theme_minimal()
```



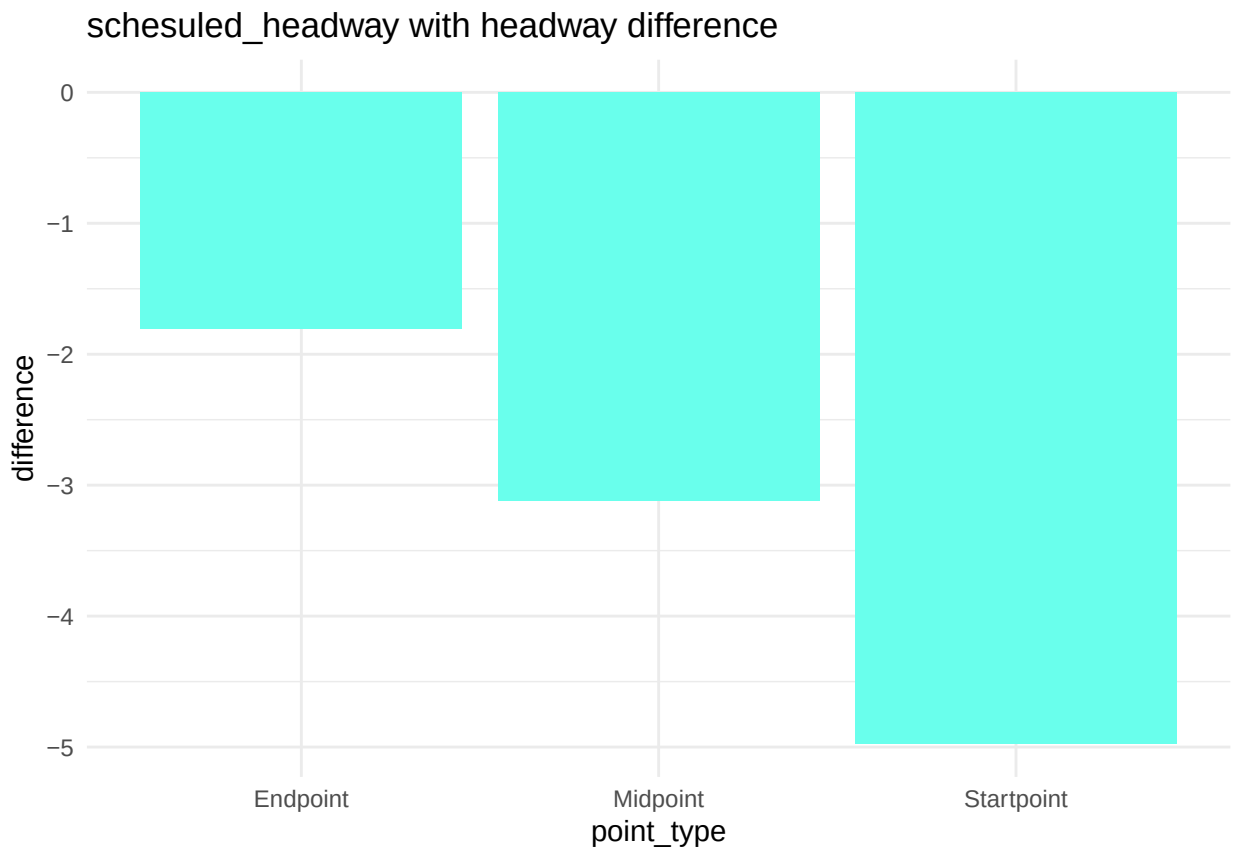

```
#The highest in the diagram is Starpoint, the second highest is Endpoint, and the lowest is Midpoint.
```

```
newBus_Alldata<-Bus_Alldata[complete.cases(Bus_Alldata),]

data5_bus <- newBus_Alldata %>% group_by(point_type) %>% summarise(avrage_scheduled_headway = mean(sche

ggplot(data5_bus) +
  aes(x = point_type, y = avrage_scheduled_headway) +
  geom_col(fill = "#E5BB43") +
  labs(x = "point_type", y = "average_scheduled_headway", title = "point_type & avrage_scheduled_headwa
  theme_minimal()
```




```
#The highest in the diagram is Starpoint, the second highest is Endpoint, and the lowest is Midpoint.
```

```
data5_bus$difference<-data5_bus$avrage_scheduled_headway - data4_bus$average_headway

ggplot(data5_bus) +
  aes(x = point_type, y = difference) +
  geom_col(fill = "#69FFEC") +
  labs(
    x = "point_type",
    y = "difference",
    title = "schesuled_headway with headway difference"
  ) +
  theme_minimal()
```




```
#The average difference between scheduled_headway and headway is the least in Endpoint, and the largest
```