MA615 FINAL

Xinpeng Hua

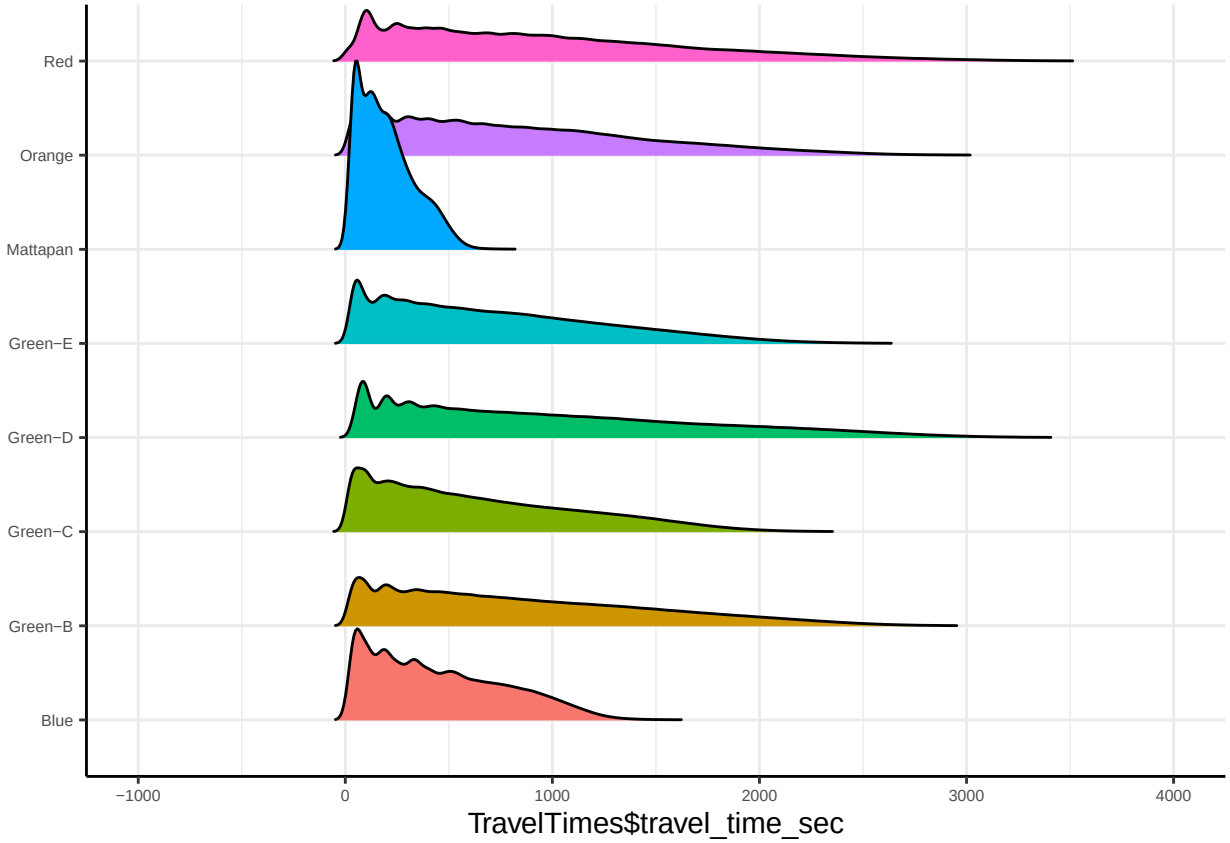
2022-12-16

Firstly, I use the subway data from 2021.10 to 2022.9. Because the data is too large, some plots cannot be run in my computer. Therefore, in this report there are two data sets I working for.

```
HR4_21 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/HRTravelTimesQ4_21.csv", header = T)
LR4_21 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/HRTravelTimesQ4_21.csv", header = T)
HR1_22 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/2022-Q1_HRTravelTimes.csv", header = T)
HR2_22 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/2022-Q2_HRTravelTimes.csv", header = T)
HR3_22 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/2022-Q3_HRTravelTimes.csv", header = T)
LR1_22 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/2022-Q1_LRTravelTimes.csv", header = T)
LR2_22 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/2022-Q2_LRTravelTimes.csv", header = T)
LR3_22 <- read.csv("C:/Users/HXP/Desktop/MA615 Final/2022-Q3_LRTravelTimes.csv", header = T)
```

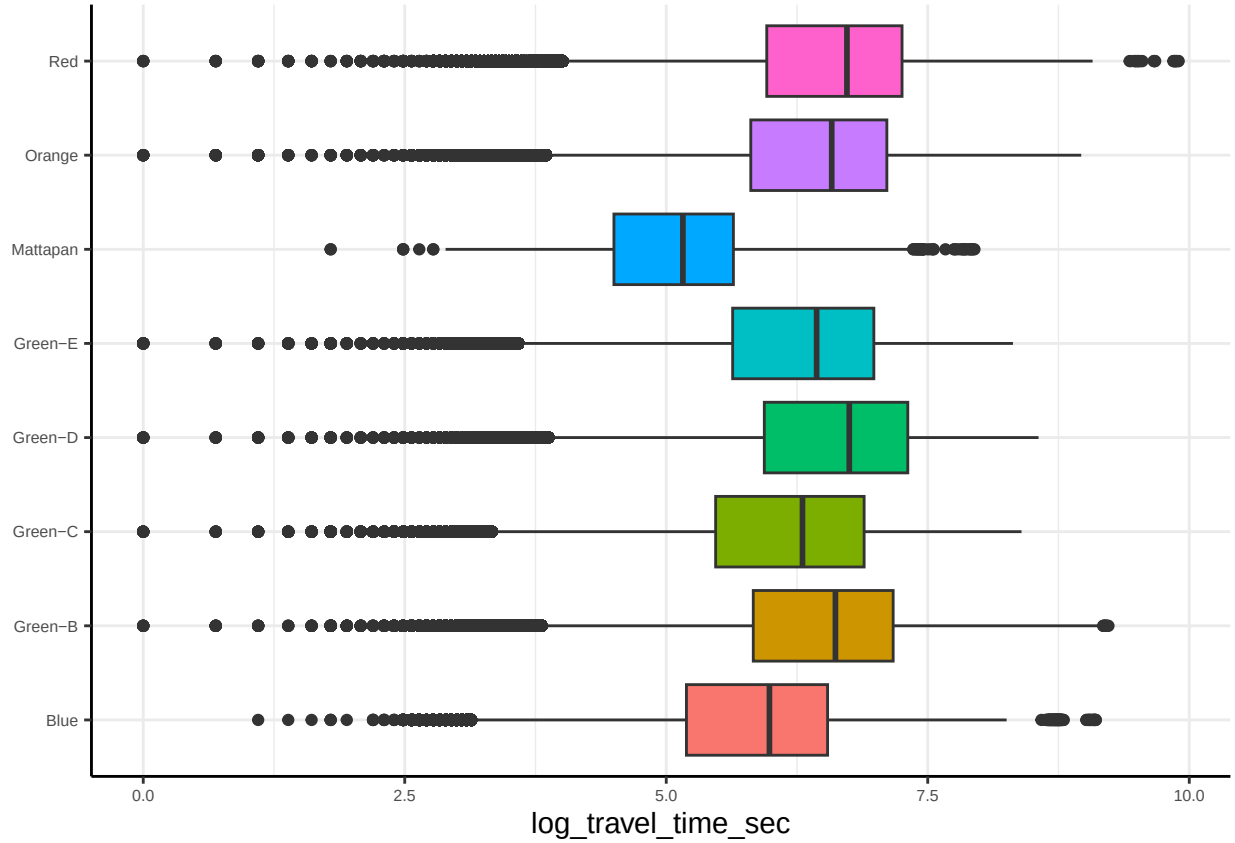
```
HR4_21 <- filter(HR4_21,HR4_21$service_date %in% c("2021-10-18","2021-10-19","2021-10-20","2021-10-21",
LR4_21 <- filter(LR4_21,LR4_21$service_date %in% c("2021-10-18","2021-10-19","2021-10-20","2021-10-21",
HR1_22 <- filter(HR1_22,HR1_22$service_date %in% c("2022-01-05","2022-01-06","2022-01-07","2022-01-08",
LR1_22 <- filter(LR1_22,LR1_22$service_date %in% c("2022-01-05","2022-01-06","2022-01-07","2022-01-08",
HR2_22 <- filter(HR2_22,HR2_22$service_date %in%
c("2022-04-23","2022-04-24","2022-04-25","2022-04-26","2022-04-27","2022-04-28","2022-04-29","2022-05-0
LR2_22 <- filter(LR2_22,LR2_22$service_date %in%
c("2022-04-23","2022-04-24","2022-04-25","2022-04-26","2022-04-27","2022-04-28","2022-04-29","2022-05-0
HR3_22 <- filter(HR3_22,HR3_22$service_date %in%
c("2022-07-13","2022-07-14","2022-07-15","2022-07-16","2022-07-17","2022-07-18","2022-07-19","2022-08-1
LR3_22 <- filter(LR3_22,LR3_22$service_date %in%
c("2022-07-13","2022-07-14","2022-07-15","2022-07-16","2022-07-17","2022-07-18","2022-07-19","2022-08-1
TravelTimes <-rbind(HR4_21,LR4_21,HR1_22,LR1_22,HR2_22,LR2_22,HR3_22,LR3_22)
TravelTimes <-cbind(log_travel_time_sec = log(TravelTimes$travel_time_sec),TravelTimes)
```

```
EDA1 <-ggplot(TravelTimes, aes(x = TravelTimes$travel_time_sec, y = route_id, fill = route_id)) +
   geom_density_ridges_gradient(scale = 2, rel_min_height = 0.001) +theme_bw() +theme(legend.position="

        panel.border = element_blank(),
        axis.line = element_line(colour = "black"))+
  theme(legend.position = "none")+theme(axis.title.y = element_blank(),axis.text=element_text(size=6))+
  theme(axis.title.x = element_text(hjust = "0.5"))+scale_x_continuous(limits = c(-1000,4000))
EDA1
```



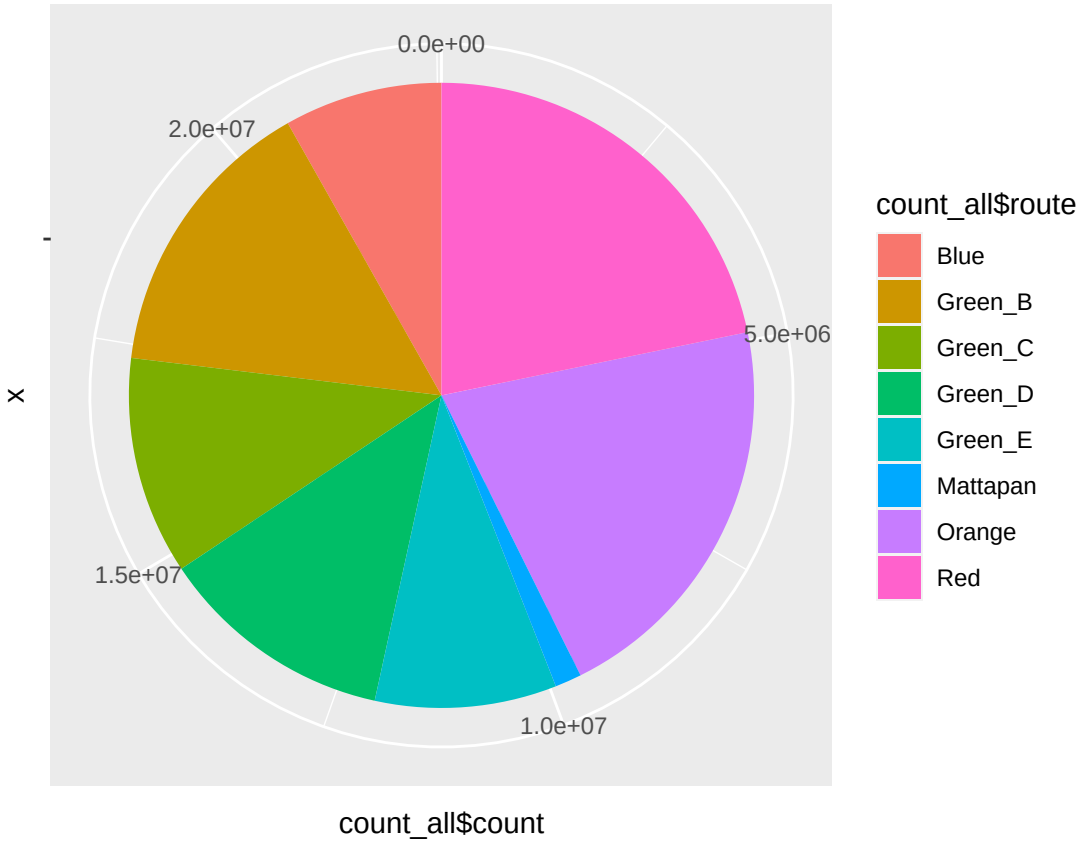
The ridge plot shows that the travel time in sec for every line.

```
EDA2 <-ggplot(TravelTimes, aes(x = log_travel_time_sec, y = route_id, fill = route_id)) +
  geom_boxplot() +theme_bw() +theme(legend.position="none",

        panel.border = element_blank(),
        axis.line = element_line(colour = "black"))+theme(axis.title.y = element_blank(),axis.text=eleme
EDA2
```



As the ridge plot is not good for reviewing, I decide to log the travel time in sec to improve the visual effect. And this box plot shows the travel time in sec for each line after log.

```
Orange <- nrow(TravelTimes[which(TravelTimes$route_id == 'Orange'),])
Blue <- nrow(TravelTimes[which(TravelTimes$route_id == 'Blue'),])
Red <- nrow(TravelTimes[which(TravelTimes$route_id == 'Red'),])
Green_B <- nrow(TravelTimes[which(TravelTimes$route_id == 'Green-B'),])
Green_C <- nrow(TravelTimes[which(TravelTimes$route_id == 'Green-C'),])
Green_D <- nrow(TravelTimes[which(TravelTimes$route_id == 'Green-D'),])
Green_E <- nrow(TravelTimes[which(TravelTimes$route_id == 'Green-E'),])
Mattapan <- nrow(TravelTimes[which(TravelTimes$route_id == 'Mattapan'),])
count_all <- data.frame(route = c("Orange","Blue","Red","Green_B","Green_C","Green_D","Green_E","Mattapa

EDA3 <-ggplot(count_all, aes(x="", y=count_all$count, fill=count_all$route)) +
  geom_bar(stat="identity", width=1) +
  coord_polar("y", start=0)
EDA3
```



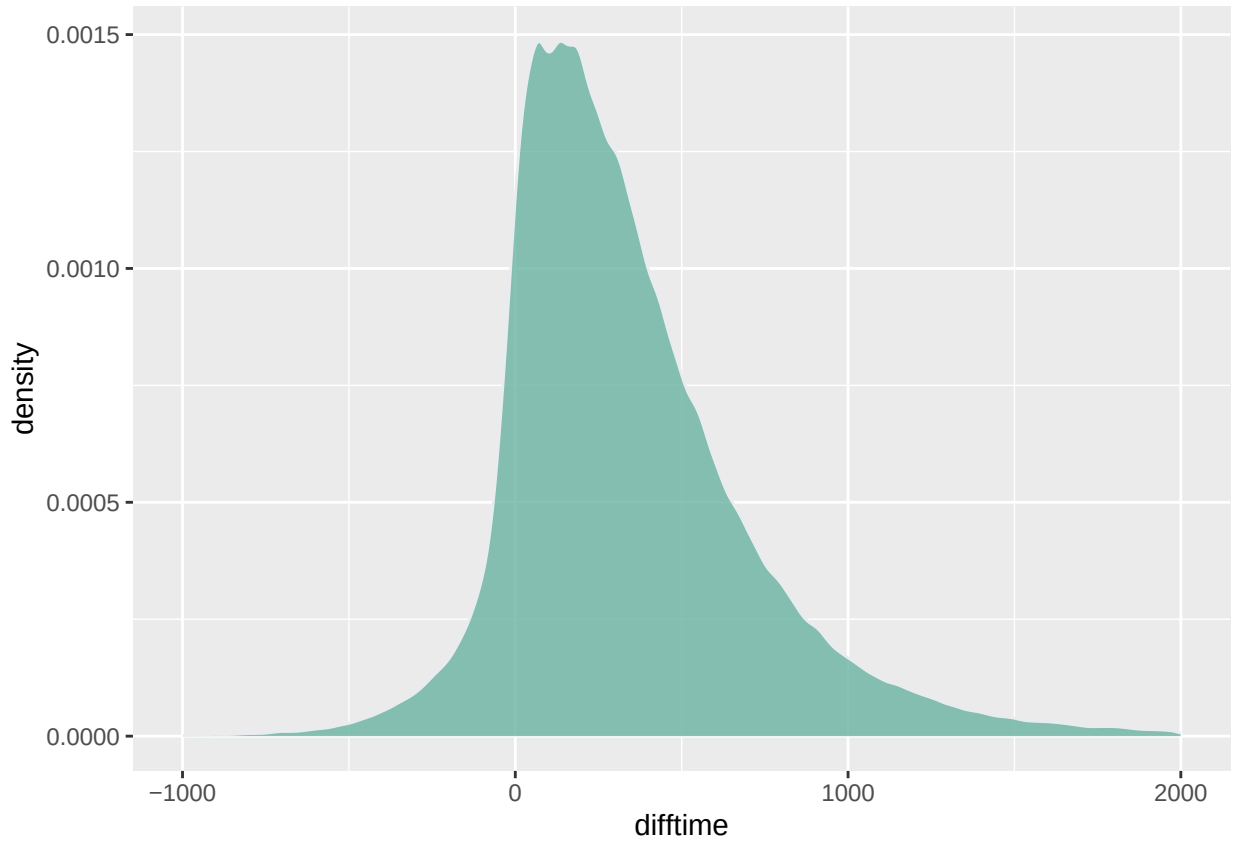
This pie chart reviews the number of data for each line, which means the larger the amount of data, the more and more frequent the stops of the line. It is clear that the red and the orange line have the largest data among them. And the Mattapan line have the least.

And I use the Bus Data to explore.

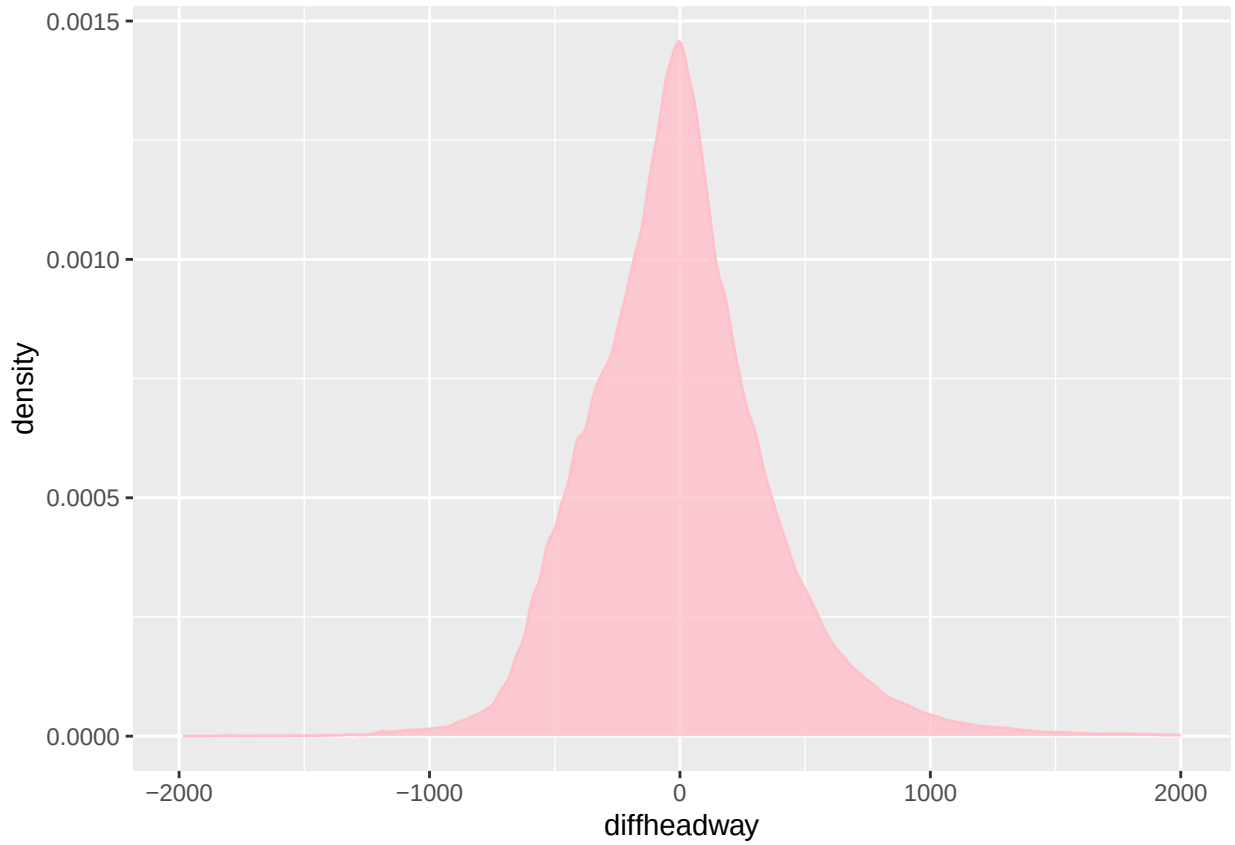
```
B10_21 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2021-10.csv")
B11_21 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2021-11.csv")
B12_21 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2021-12.csv")
B01_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-01.csv")
B02_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-02.csv")
B03_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-03.csv")
B04_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-04.csv")
B05_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-05.csv")
B06_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-06.csv")
B07_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-07.csv")
B08_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-08.csv")
B09_22 <-read.csv("C:/Users/HXP/Desktop/MA615 Final/MBTA-Bus-Arrival-Departure-Times_2022-09.csv")

B10_21 <- filter(B10_21,B10_21$service_date %in% c("2021-10-18","2021-10-19","2021-10-20","2021-10-21",

B11_21 <- filter(B11_21,B11_21$service_date %in% c("2021-11-07","2021-11-08","2021-11-09","2021-11-10",

B12_21 <- filter(B12_21,B12_21$service_date %in% c("2021-12-01","2021-12-02","2021-12-03","2021-12-04",

B01_22 <- filter(B01_22,B01_22$service_date %in% c("2022-01-05","2022-01-06","2022-01-07","2022-01-08",
```

```
B02_22 <- filter(B02_22,B02_22$service_date %in% c("2022-02-22","2022-02-23","2022-02-24","2022-02-25",

B03_22 <- filter(B03_22,B03_22$service_date %in% c("2022-03-04","2022-03-05","2022-03-06","2022-03-07",

B04_22 <- filter(B04_22,B04_22$service_date %in% c("2022-04-23","2022-04-24","2022-04-25","2022-04-26",

B05_22 <- filter(B05_22,B05_22$service_date %in% c("2022-05-06","2022-05-07","2022-05-08","2022-05-09",

B06_22 <- filter(B06_22,B06_22$service_date %in% c("2022-06-19","2022-06-20","2022-06-21","2022-06-22",

B07_22 <- filter(B07_22,B07_22$service_date %in% c("2022-07-13","2022-07-14","2022-07-15","2022-07-16",

B08_22 <- filter(B08_22,B08_22$service_date %in% c("2022-08-13","2022-08-14","2022-08-15","2022-08-16",

B09_22 <- filter(B09_22,B09_22$service_date %in% c("2022-09-23","2022-09-24","2022-09-25","2022-09-26",
B <- rbind(B10_21,B11_21,B12_21,B01_22,B02_22,B03_22,B04_22,B05_22,B06_22,B07_22,B08_22,B09_22)
```

```
B1<-B[complete.cases(B[,10:11]),]
B1$service_date <- substr(B1$service_date,1,7)
B1$difftime <- (as.numeric(substr(B1$actual,12,13)) - as.numeric(substr(B1$scheduled,12,13))) * 3600 +
(as.numeric(substr(B1$actual,18,19)) - as.numeric(substr(B1$scheduled,18,19)))
B1<-B1[complete.cases(B1[,12:14]),]
B2 <- filter(B1,route_id %in% c("57","66","64","47","8","60","65"))


B3 <-cbind(diffheadway = B2$headway-B2$scheduled_headway,B2)

EDA4<- B3 %>%
  filter( difftime<2000&difftime>(-1000) ) %>%
  ggplot( aes(x=difftime)) +
    geom_density(fill="#69b3a2", color="#e9ecef", alpha=0.8)


EDA5 <-B3 %>%
  filter( diffheadway<2000&diffheadway>(-2000) ) %>%
  ggplot( aes(x=diffheadway)) +
    geom_density(fill="pink", color="pink", alpha=0.8)

EDA4
```




```
EDA5
```



I select the bus route IDs which are around Boston University and my apartment in Allston. And through the density plots with difftime and diffheadway data, it reviews that although the diffheadway is like normal distribution which distributed around zero which means the mean of diffheadway is around 0. But the difftime is more like positive, which means the buses always delay among these lines. Actually, I always get the same feeling. I went to university by 57 bus, but it more than 50% likely to be delayed in my daily life.

It is pity that I have not completed well in the shiny app part. I cannot make sure whether or not it can work well as my computer run it so slowly and hard. The data is too large so that every time I run the data, it took me a bunch of minutes. But I will keep learning shiny to keep going.