Assignment_1

Abbas Zal - 2072054

2023-04-13

Exercise 1 - American Airlines Employees

In this exercise, we have four .txt files that are included of information about four American airlines from 1990 to 2023.

1) First we load the data.
2) Then,merge the four data tibble in a common tibble.
3) After that, We produce a plot of the behavior of the employees as a function of time for all four companies.
4) Then, we see when each company reach the minimum and maximum number of employees.
5) Again we plot the fraction of part-time worker over the total employees as a function of time.
6) At the end,We discuss about the COVID-19 pandemic and its influences in the employed workers of the airline companies.

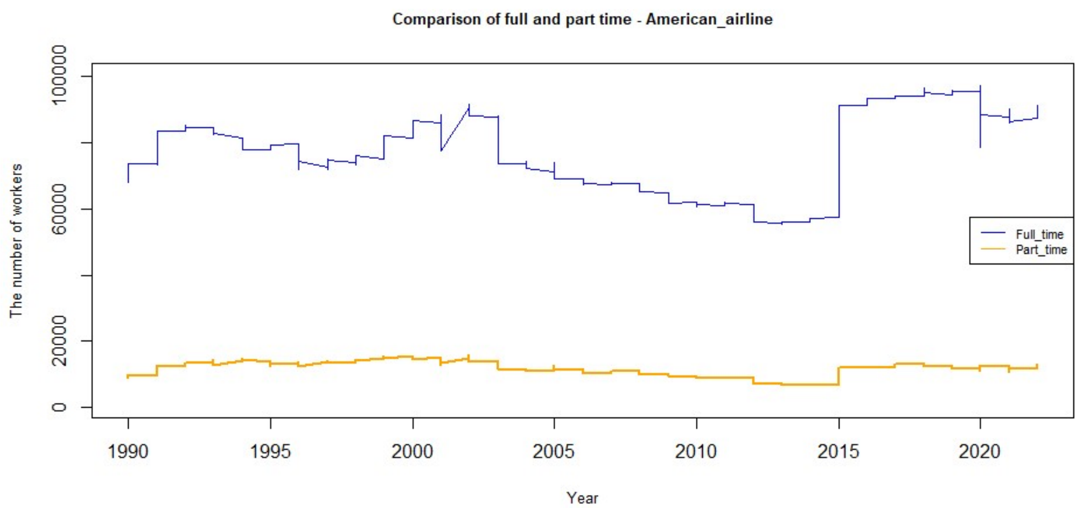
```r
#Loading Data
#exercise 1_1
dt_1<- read.delim("american_airline_empl.txt")
dt_2 <- read.delim("delta_airline_empl.txt")
dt_3 <- read.delim("federal_express_empl.txt")
dt_4 <- read.delim("united_airline_empl.txt")
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

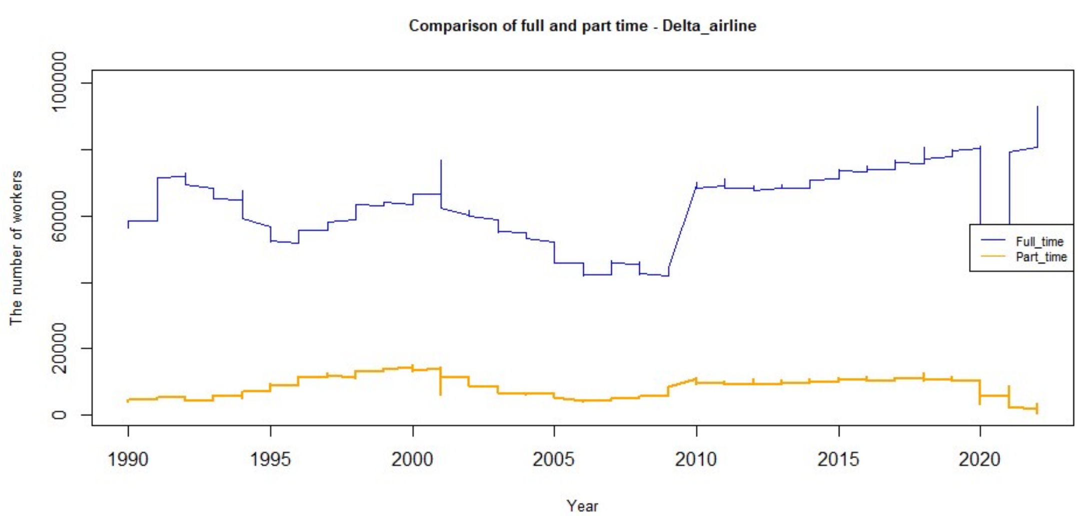
```r
#exercise 1_2
suppressMessages(library(dplyr))
tibble1 <- tibble(dt_1)
tibble2 <- tibble(dt_2)
tibble3 <- tibble(dt_3)
tibble4 <- tibble(dt_4)
common_tibble <- bind_rows(tibble1, tibble2, tibble3, tibble4)
common_tibble

## # A tibble: 1,588 × 5
##    Month  Year Full.time Part.time Grand.Total
##    <int> <int> <chr>     <chr>     <chr>
```

```
## 1     1  1990 68,137    9,039     77,176
## 2     2  1990 68,725    9,273     77,998
## 3     3  1990 69,509    9,376     78,885
## 4     4  1990 69,713    9,326     79,039
## 5     5  1990 70,376    9,309     79,685
## 6     6  1990 71,258    9,369     80,627
## 7     7  1990 72,018    9,651     81,669
## 8     8  1990 72,513    9,694     82,207
## 9     9  1990 72,776    9,788     82,564
## 10   10  1990 73,111    9,737     82,848
## # ℹ 1,578 more rows
```

```
#exercise 1_3
total <- rbind(data.frame(dt_1) , data.frame(dt_2) , data.frame(dt_3) ,
data.frame(dt_4) )
full <- gsub("[^[:digit:].]", "", total[,3])
full <- as.integer(full)
part <- gsub("[^[:digit:].]", "", total[,4])
part <- as.integer(part)
colors <- c("red", "blue", "green", "orange", "black")


x <- rep(c(1990:2022),each=12)
y1 <- c(full[1:396])
y2 <- c(part[1:396])
# create a line graph
plot(x, y1, type = "l", col = "blue",lwd=0.5, ylim = c(1000,100000),
     main = "Comparison of full and part time - American_airline", xlab =
"Year", ylab = "The number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y2, col = "orange",lwd=2)
legend("right", legend=c("Full_time", "Part_time"),
       col= c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```

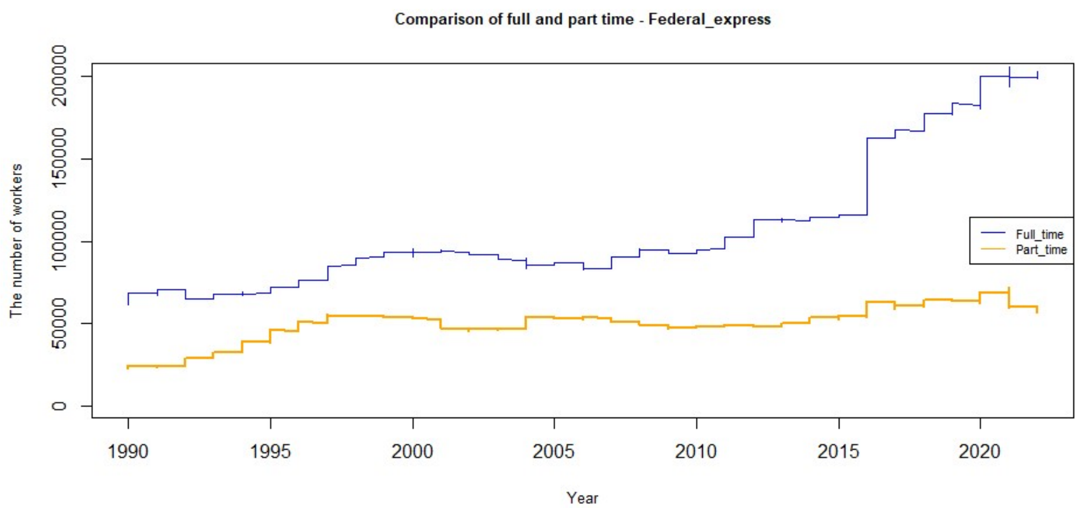



```
x <- rep(c(1990:2022),each=12)
y1 <- c(full[398:793])
y2 <- c(part[398:793])
# create a line graph
plot(x, y1, type = "l", col = "blue",lwd=0.5, ylim = c(1000,100000),
     main = "Comparison of full and part time - Delta_airline", xlab =
"Year", ylab = "The number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y2, col = "orange",lwd=2)
legend("right", legend=c("Full_time", "Part_time"),
       col= c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```

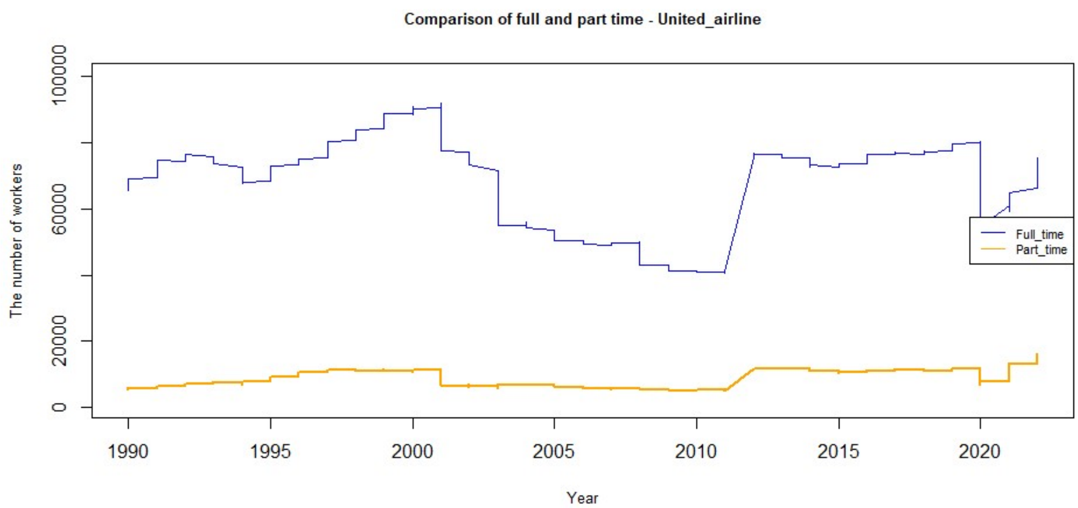



```
x <- rep(c(1990:2022),each=12)
y1 <- c(full[795:1190])
y2 <- c(part[795:1190])
```

```r
# create a line graph
plot(x, y1, type = "l", col = "blue",lwd=0.5, ylim = c(1000,200000),
     main = "Comparison of full and part time - Federal_express", xlab =
"Year", ylab = "The number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y2, col = "orange",lwd=2)
legend("right", legend=c("Full_time", "Part_time"),
       col= c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```




```r
x <- rep(c(1990:2022),each=12)
y1 <- c(full[1192:1587])
y2 <- c(part[1192:1587])
# create a line graph
plot(x, y1, type = "l", col = "blue",lwd=0.5, ylim = c(1000,100000),
     main = "Comparison of full and part time - United_airline", xlab =
"Year", ylab = "The number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y2, col = "orange",lwd=2)
legend("right", legend=c("Full_time", "Part_time"),
       col= c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```

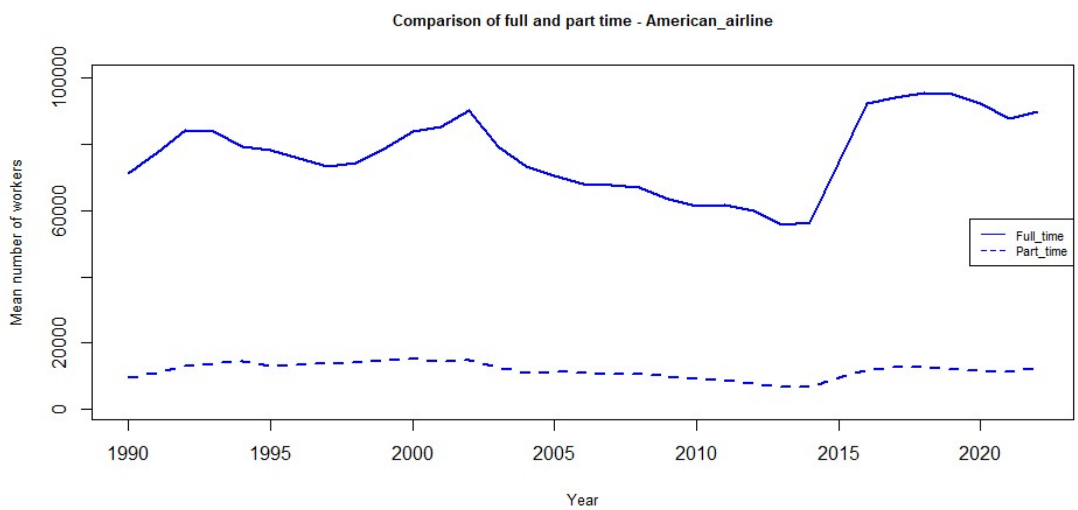


In this section, I did one optional task that compare all companies workers in part time and full time -one by one and together- . First of all i computed the mean number of workers for all years for each company:

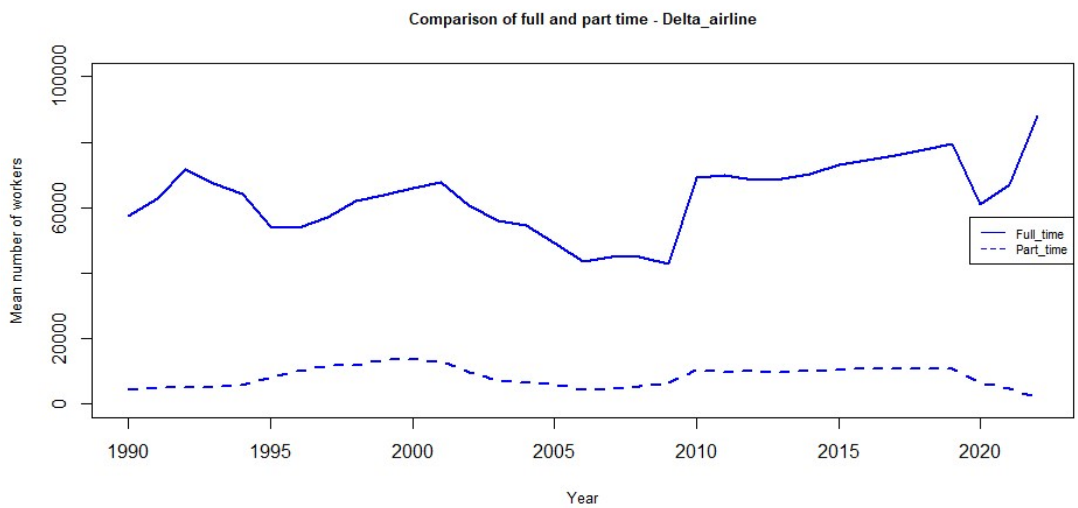
```
meanfull_1 <- numeric(1)
for (i in seq(from = 1, to = 396, by = 12)) {
  if(i<12 ){
    meanfull_1[length(meanfull_1)] <- mean(full[i:(i+11)])
  } else{
    meanfull_1[length(meanfull_1) + 1] <- mean(full[i:(i+11)])
  }
}
meanpart_1 <- numeric(1)
for (i in seq(from = 1, to = 396, by = 12)) {
  if(i<12 ){
    meanpart_1[length(meanpart_1)] <- mean(part[i:(i+11)])
  } else{
    meanpart_1[length(meanpart_1) + 1] <- mean(part[i:(i+11)])
  }
}
meanfull_2 <- numeric(1)
for (i in seq(from = 398, to = 793, by = 12)) {
  if(i<409 ){
    meanfull_2[length(meanfull_2)] <- mean(full[i:(i+11)])
  } else{
    meanfull_2[length(meanfull_2) + 1] <- mean(full[i:(i+11)])
  }
}
meanpart_2 <- numeric(1)
for (i in seq(from = 398, to = 793, by = 12)) {
  if(i<409 ){
    meanpart_2[length(meanpart_2)] <- mean(part[i:(i+11)])
  } else{
```

```
    meanpart_2[length(meanpart_2) + 1] <- mean(part[i :(i+11)])
  }
}


meanfull_3 <- numeric(1)
for (i in seq(from = 795, to = 1190, by = 12)) {
  if(i<806){
    meanfull_3[length(meanfull_3)] <- mean(full[i:(i+11)])
  } else{
    meanfull_3[length(meanfull_3) + 1] <- mean(full[i:(i+11)])
  }
}
meanpart_3 <- numeric(1)
for (i in seq(from = 795, to = 1190, by = 12)) {
  if(i<806 ){
    meanpart_3[length(meanpart_3)] <- mean(part[i:(i+11)])
  } else{
    meanpart_3[length(meanpart_3) + 1] <- mean(part[i:(i+11)])
  }
}


meanfull_4 <- numeric(1)
for (i in seq(from = 1192, to = 1587, by = 12)) {
  if(i<1203 ){
    meanfull_4[length(meanfull_4)] <- mean(full[i:(i+11)])
  } else{
    meanfull_4[length(meanfull_4) + 1] <- mean(full[i:(i+11)])
  }
}
meanpart_4 <- numeric(1)
for (i in seq(from = 1192, to = 1587, by = 12)) {
  if(i<1203 ){
    meanpart_4[length(meanpart_4)] <- mean(part[i:(i+11)])
  } else{
    meanpart_4[length(meanpart_4) + 1] <- mean(part[i:(i+11)])
  }
}
colors <- c("red", "blue", "green", "orange", "black","brown")

x <- c(1990:2022)
y1 <- c(meanfull_1)
y2 <- c(meanpart_1)


# create a line graph with one line
plot(x, y1, type = "l", col = "blue",lwd=2, ylim = c(1000,100000),
     main = "Comparison of full and part time - American_airline", xlab =
```

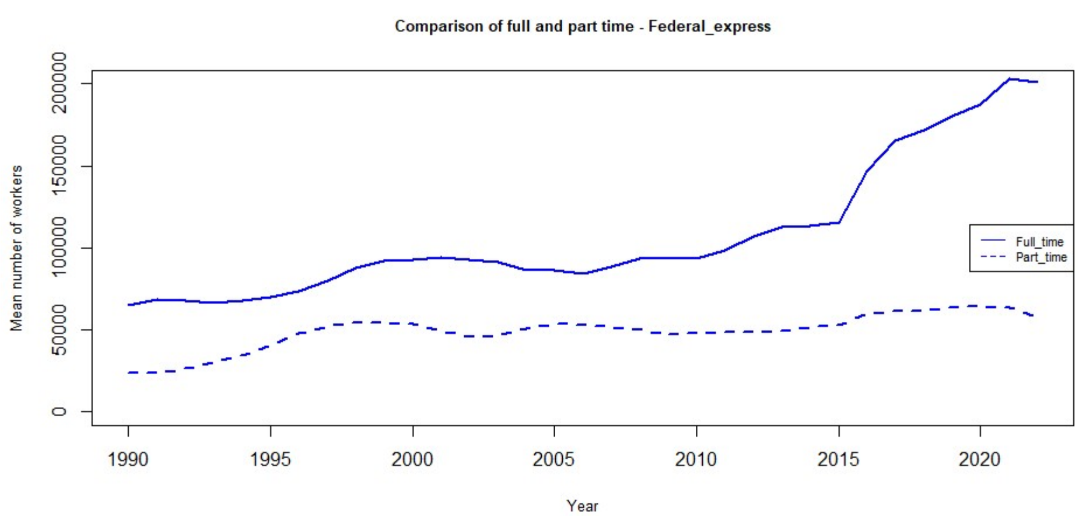
```
"Year", ylab = "Mean number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y2, col = "blue",lwd=2,lty=2)

legend("right", legend=c("Full_time", "Part_time"),
       col=colors[2], lty=c(1,2),cex = 0.7)
```




```
x <- c(1990:2022)
y3 <- c(meanfull_2)
y4 <- c(meanpart_2)
plot(x, y3, type = "l", col = "blue",lwd=2, ylim = c(0,100000),
     main = "Comparison of full and part time - Delta_airline", xlab =
"Year", ylab = "Mean number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y4, col = "blue",lwd=2,lty=2)

legend("right", legend=c("Full_time", "Part_time"),
       col=colors[2], lty=c(1,2),cex = 0.7)
```

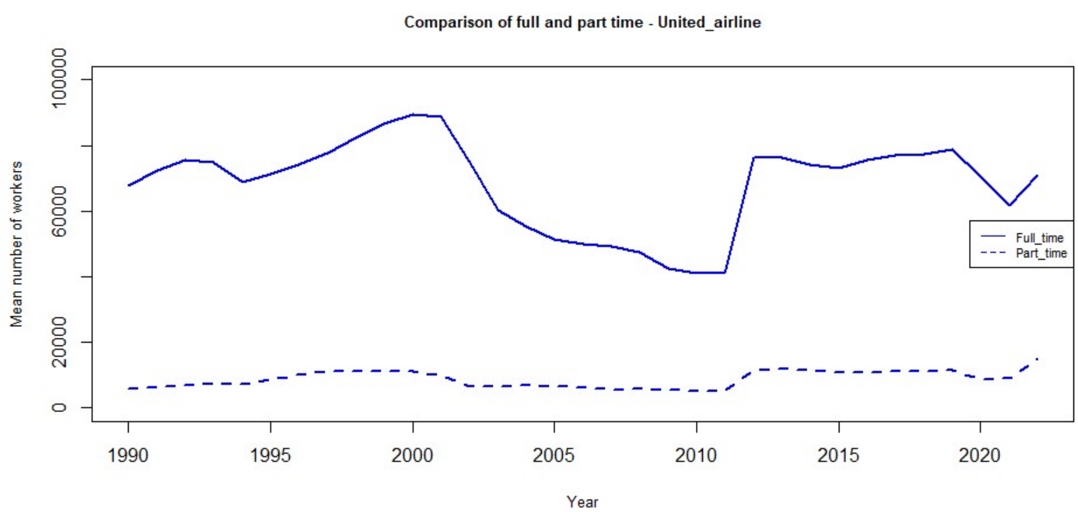



```
x <- c(1990:2022)
y5 <- c(meanfull_3)
y6 <- c(meanpart_3)

plot(x, y5, type = "l", col = "blue",lwd=2, ylim = c(0,200000),
     main = "Comparison of full and part time - Federal_express", xlab =
"Year", ylab = "Mean number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y6, col = "blue",lwd=2,lty=2)

legend("right", legend=c("Full_time", "Part_time"),
       col=colors[2], lty=c(1,2),cex = 0.7)
```



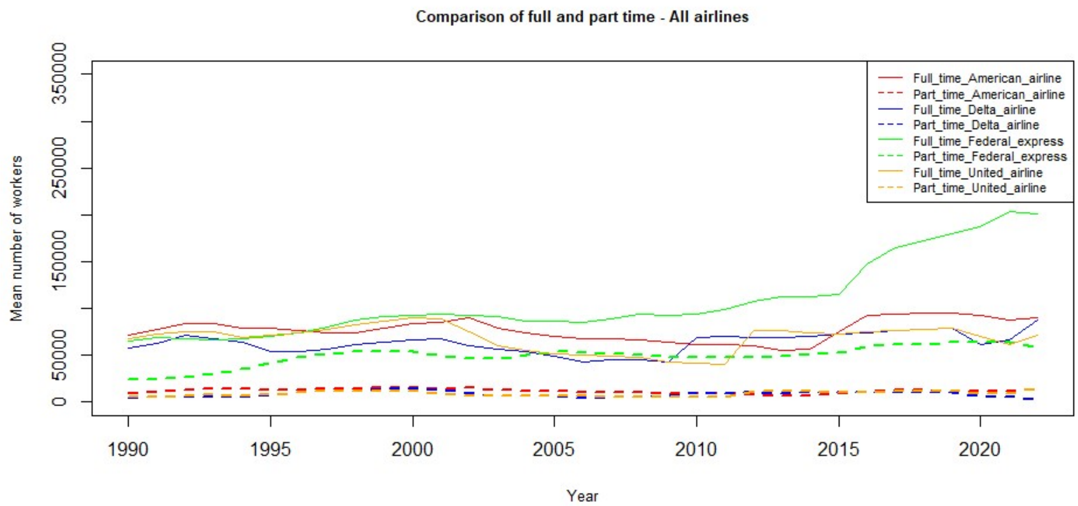

```
x <- c(1990:2022)
y7 <- c(meanfull_4)
```

```
y8 <- c(meanpart_4)

plot(x, y7, type = "l", col = "blue",lwd=2, ylim = c(0,100000),
     main = "Comparison of full and part time - United_airline", xlab =
"Year", ylab = "Mean number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
# add a second line to the plot
lines(x, y8, col = "blue",lwd=2,lty=2)

legend("right", legend=c("Full_time", "Part_time"),
       col=colors[2], lty=c(1,2),cex = 0.7)
```




```
x1 <- c(1990:2022)
y1 <- c(meanfull_1)
y2 <- c(meanpart_1)
y3 <- c(meanfull_2)
y4 <- c(meanpart_2)
y5 <- c(meanfull_3)
y6 <- c(meanpart_3)
y7 <- c(meanfull_4)
y8 <- c(meanpart_4)

plot(x1, y1, type = "l", col = colors[1], ylim = c(0,350000),
     main = "Comparison of full and part time - All airlines", xlab = "Year",
ylab = "Mean number of workers", cex.main=0.8 , cex.lab=0.8)
options(scipen = 10)
lines(x, y2, col=colors[1], lwd=2, lty=2)
lines(x, y3, col=colors[2])
lines(x, y4, col=colors[2], lwd=2, lty=2)
lines(x, y5, col=colors[3])
lines(x, y6, col=colors[3], lwd=2, lty=2)
lines(x, y7, col=colors[4])
lines(x, y8, col=colors[4], lwd=2, lty=2)
```

```
legend("topright", legend=c("Full_time_American_airline",
"Part_time_American_airline","Full_time_Delta_airline",
"Part_time_Delta_airline","Full_time_Federal_express",
"Part_time_Federal_express","Full_time_United_airline",
"Part_time_United_airline"),
       col=rep(c(colors[1:4]),each=2), lty=c(1,2,1,2,1,2,1,2),cex = 0.7)
```



For this exercise at the first step i did not focus column data and i did not know that we have the summation of all worker for all companies in “Grand total” column, so I wrote an appropriate code for calculating the number of workers i n each month of years for each company and i used this list foe 1_3 and 1_4.

```
full_1 <- gsub("[^[:digit:].]", "", dt_1[,3])
part_1 <- gsub("[^[:digit:].]", "", dt_1[,4])
full_1<- as.numeric(full_1)
part_1 <- as.numeric(part_1)
full_1<- as.integer(full_1)
part_1<- as.integer(part_1)


full_2 <- gsub("[^[:digit:].]", "", dt_2[,3])
part_2 <- gsub("[^[:digit:].]", "", dt_2[,4])
full_2 <- as.numeric(full_2)
part_2 <- as.numeric(part_2)
full_2<- as.integer(full_2)
part_2<- as.integer(part_2)


full_3 <- gsub("[^[:digit:].]", "", dt_3[,3])
part_3 <- gsub("[^[:digit:].]", "", dt_3[,4])
full_3 <- as.numeric(full_3)
part_3 <- as.numeric(part_3)
full_3<- as.integer(full_3)
```

```
part_3<- as.integer(part_3)


full_4 <- gsub("[^[:digit:].]", "", dt_4[,3])
part_4 <- gsub("[^[:digit:].]", "", dt_4[,4])
full_4 <- as.numeric(full_4)
part_4 <- as.numeric(part_4)
full_4<- as.integer(full_4)
part_4<- as.integer(part_4)


sum_1 <- numeric(length(full_1))
sum_2 <- numeric(length(full_1))
sum_3 <- numeric(length(full_1))
sum_4 <- numeric(length(full_1))

for (i in seq_along(full_1)) {
  sum_1[i]  <-  full_1[i] + part_1[i]
}


for (i in seq_along(full_1)) {
  sum_2[i]  <-  full_2[i] + part_2[i]
}


for (i in seq_along(full_1)) {
  sum_3[i]  <-  full_3[i] + part_3[i]
}


for (i in seq_along(full_1)) {
  sum_4[i]  <-  full_4[i] + part_4[i]
}
```

Now we want to know when did each company reach the minimum and maximum number of employess.

```
#exercise1_3


total <- rbind(data.frame(dt_1) , data.frame(dt_2) , data.frame(dt_3) ,
data.frame(dt_4) )


i <- data.frame(0, nrow(dt_1) , nrow(dt_2) + nrow(dt_1) , nrow(dt_3)+
nrow(dt_2) + nrow(dt_1) )

max_index <- numeric(4)
```

```r
max_i <- numeric(4)

for (j in 1:4) {
  full_j_name <- paste0("full_", j)
  part_j_name <- paste0("part_", j)

  full_j <- get(full_j_name)
  part_j <- get(part_j_name)

  sum_j <- full_j + part_j
  max_i[j] <- max(sum_j)
  max_index[j] <- which(sum_j == max_i[j])
}
maximums <- data.frame(matrix(0, nrow=4, ncol=4))
names(maximums) <- c("Month", "Year", "Full_time", "Part_time")
rownames(maximums) <- c("american_airline", "delta_airline",
"federal_express", "united_airline")
for(j in 1:4){
    maximums[j,1:4] <- total[max_index[j] + i[[j]], 1:4]
}
print(" Maximum number of employess")

## [1] " Maximum number of employess"

maximums

##                   Month Year Full_time Part_time
## american_airline      6 2018    96,543    12,628
## delta_airline         1 2023    94,236       439
## federal_express       3 2021   204,406    65,977
## united_airline        3 2001    91,041    11,005

min_index <- numeric(4)
min_i <- numeric(4)

# Calculate max_i and max_index for each sum_i
for (j in 1:4) {
  full_j_name <- paste0("full_", j)
  part_j_name <- paste0("part_", j)

  full_j <- get(full_j_name)
  part_j <- get(part_j_name)

  sum_j <- full_j + part_j
  min_i[j] <- min(sum_j)
  min_index[j] <- which(sum_j == min_i[j])
}

minimums <- data.frame(matrix(0, nrow=4, ncol=4))
names(minimums) <- c("Month", "Year", "Full_time", "Part_time")
```

```
rownames(minimums) <- c("american_airline", "delta_airline",
"federal_express", "united_airline")
for(j in 1:4){
  minimums[j,1:4] <- total[min_index[j] + i[[j]], 1:4]
}
print(" Minimum number of employess")
```

```
## [1] " Minimum number of employess"
```

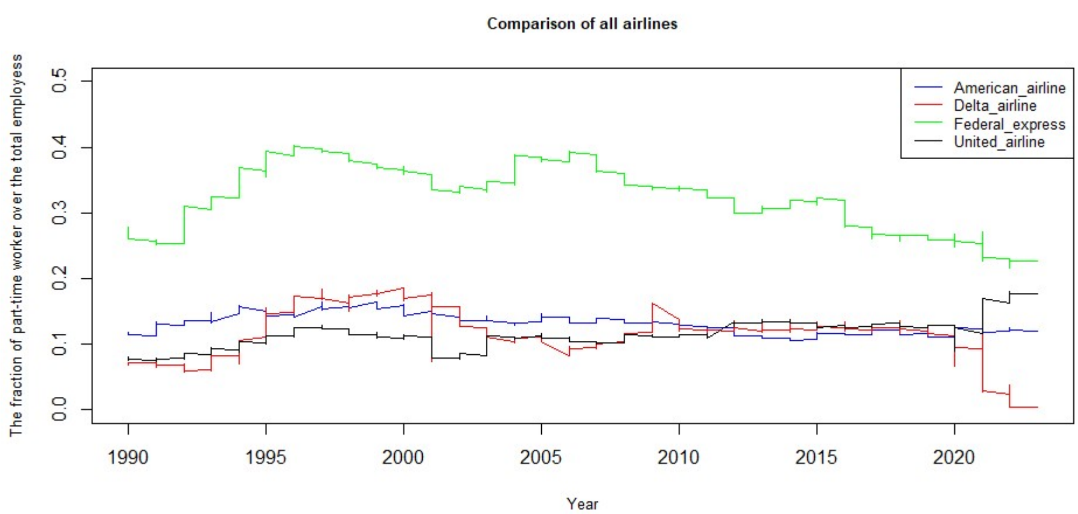
```
minimums
```

```
##                  Month Year Full_time Part_time
## american_airline     9 2013    55,462     6,828
## delta_airline       11 2006    41,948     4,462
## federal_express      1 1990    61,305    23,580
## united_airline       6 2011    40,522     5,259
```

In this part we want to plot the fraction of part-time worker over the total employess as a function of time. Again i used the elements of sum rather than “Grand total”.

```
#exercise 1_5

frac_1 <- numeric(length(part_1))
frac_2 <- numeric(length(part_1))
frac_3 <- numeric(length(part_1))
frac_4 <- numeric(length(part_1))

for (i in 1:length(part_1)){
  frac_1[i] <- part_1[i] / sum_1[i]
}
for (i in 1:length(part_2)){
  frac_2[i] <- part_2[i] / sum_2[i]
}
for (i in 1:length(part_3)){
  frac_3[i] <- part_3[i] / sum_3[i]
}
for (i in 1:length(part_1)){
  frac_4[i] <- part_4[i] / sum_4[i]
}



vec <- 1990:2023
vec <- vec[-34]
x <- rep(vec, each=12)
x <- c(x, 2023)

y1 <- c(frac_1)
y2 <- c(frac_2)
y3 <- c(frac_3)
```

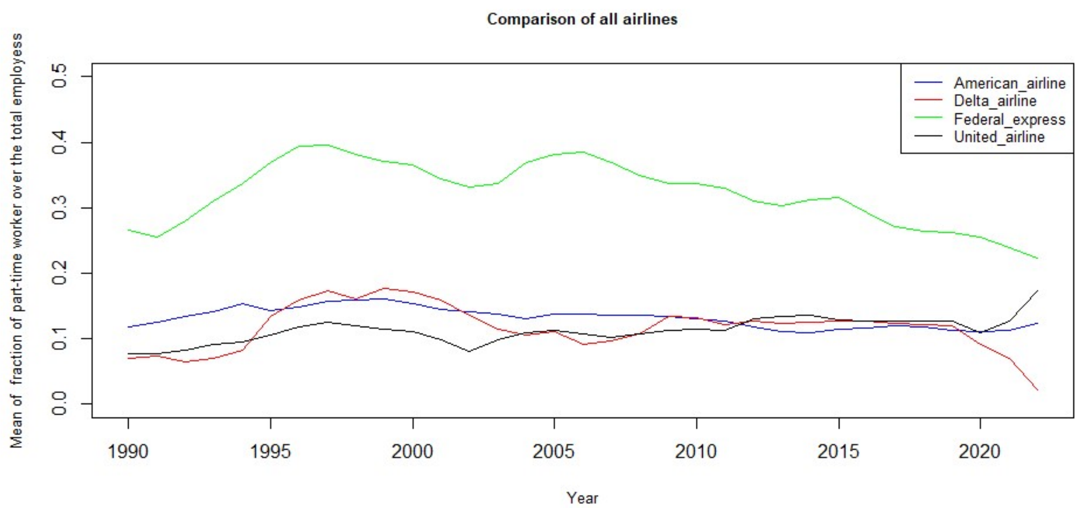
```
y4 <- c(frac_4)

# create a line graph
plot(x, y1, type = "l", col = "blue", ylim = c(0,0.5),
     main = "Comparison of all airlines", xlab = "Year", ylab = "The fraction
of part-time worker over the total employess", cex.main=0.8 , cex.lab=0.8)

# add a second line to the plot
lines(x, y2, col = "red")

lines(x, y3, col = "green")

lines(x, y4, col = "black")
legend("topright", legend=c("American_airline","Delta_airline",
"Federal_express", "United_airline"),
       col=rep(c(colors[2],colors[1],colors[3],colors[5])),
lty=c(1,1,1,1),cex = 0.8)
```



Now we compare also the mean of this values:(this is optional)

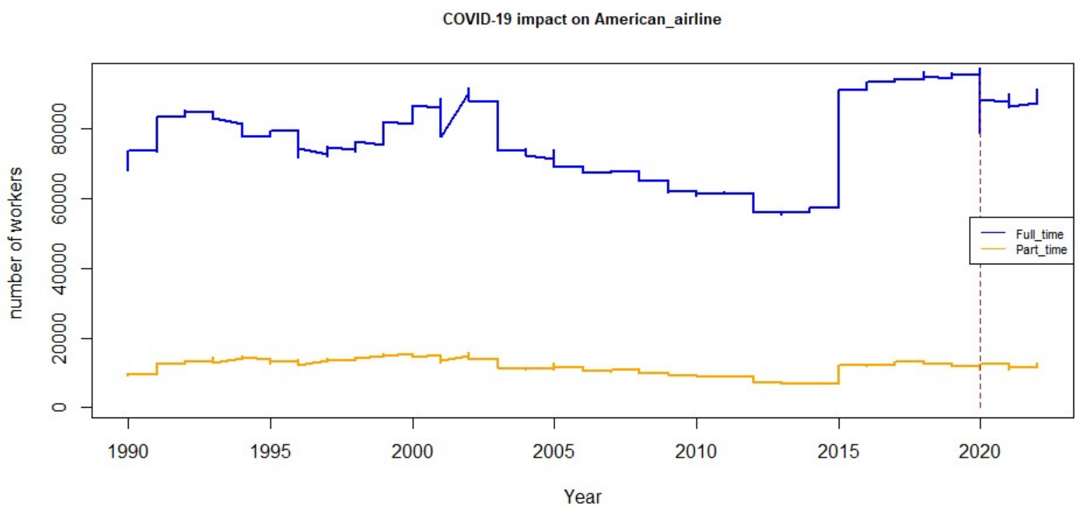
```
meanfrac_1 <- numeric(1)
meanfrac_2 <- numeric(1)
meanfrac_3 <- numeric(1)
meanfrac_4 <- numeric(1)
for (i in seq(from = 1, to = 396, by = 12)) {
  if(i<12 ){
    meanfrac_1[length(meanfrac_1)] <- mean(frac_1[i:(i+11)])
  } else{
    meanfrac_1[length(meanfrac_1) + 1] <- mean(frac_1[i:(i+11)])
  }
}
for (i in seq(from = 1, to = 396, by = 12)) {
  if(i<12 ){
```

```
    meanfrac_2[length(meanfrac_2)] <- mean(frac_2[i:(i+11)])
  } else{
    meanfrac_2[length(meanfrac_2) + 1] <- mean(frac_2[i:(i+11)])
  }
}
for (i in seq(from = 1, to = 396, by = 12)) {
  if(i<12 ){
    meanfrac_3[length(meanfrac_3)] <- mean(frac_3[i:(i+11)])
  } else{
    meanfrac_3[length(meanfrac_3) + 1] <- mean(frac_3[i:(i+11)])
  }
}
for (i in seq(from = 1, to = 396, by = 12)) {
  if(i<12 ){
    meanfrac_4[length(meanfrac_4)] <- mean(frac_4[i:(i+11)])
  } else{
    meanfrac_4[length(meanfrac_4) + 1] <- mean(frac_4[i:(i+11)])
  }
}




x <- c(1990:2022)
y1 <- c(meanfrac_1)
y2 <- c(meanfrac_2)
y3 <- c(meanfrac_3)
y4 <- c(meanfrac_4)

# create a line graph
plot(x, y1, type = "l", col = "blue", ylim = c(0,0.5),
     main = "Comparison of all airlines", xlab = "Year", ylab = "Mean of
fraction of part-time worker over the total employess", cex.main=0.8 ,
cex.lab=0.8)

# add a second line to the plot
lines(x, y2, col = "red")

lines(x, y3, col = "green")

lines(x, y4, col = "black")
legend("topright", legend=c("American_airline","Delta_airline",
"Federal_express", "United_airline"),
       col=rep(c(colors[2],colors[1],colors[3],colors[5])),
lty=c(1,1,1,1),cex = 0.8)
```



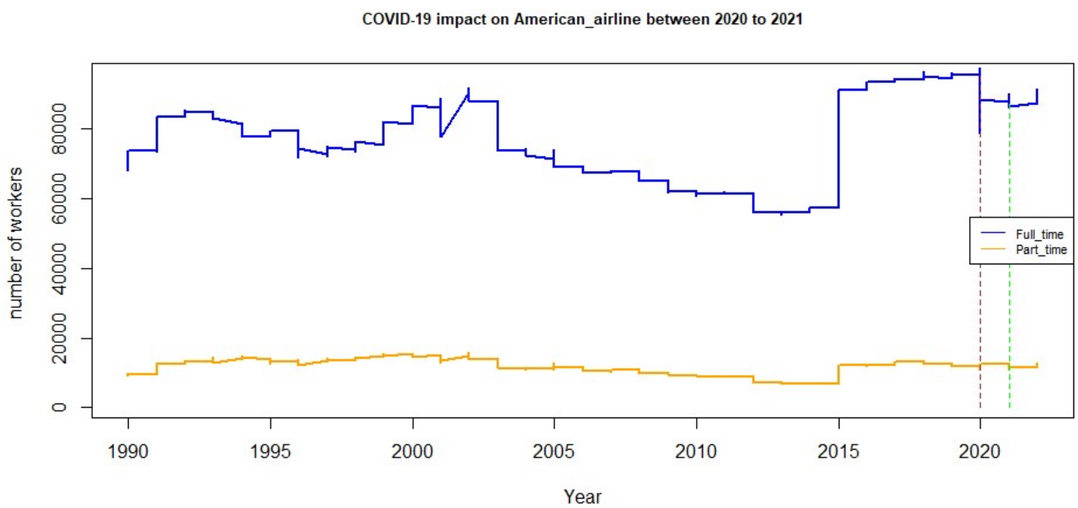
Now it is a question for all of us that did the COVID-19 pandemic have any influence in the employed workers of the airline companies?

```
#exercise1_6
x <- rep(c(1990:2022),each=12)
y1 <- c(full[1:396])
y2 <- c(part[1:396])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 95000),
     main = "COVID-19 impact on American_airline", xlab = "Year", ylab =
"number of workers ", cex.main=0.8)
options(scipen = 10)


lines(x, y2, col = "orange", lwd = 2)
#2020
point_to_label <- 361
x_coord <- x[point_to_label]
y_coord <- y1[point_to_label]
#after 12 months 2021
# point_to_label <- 373
# x_coord1 <- x[point_to_label]
# y_coord1 <- y1[point_to_label]

segments(x_coord, 0, x_coord, y_coord, col = "brown", lty = "dashed")
legend("right", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



As we can see for Americana airlines we saw the start time for outbreak of Covid_19 with brown dashed line. The official date of first first recorded instance of person-to-person spread is 2020 in the U.S.A . So as it expected we do not see any especial consequence in part time and full time workers for American airline, but the result from 2020 to 2021 was strange. This is highlighted with green dashed line in next plot. This company had experienced big fall in number of full time workers.

```
x <- rep(c(1990:2022),each=12)
y1 <- c(full[1:396])
y2 <- c(part[1:396])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 95000),
     main = "COVID-19 impact on American_airline between 2020 to 2021", xlab
= "Year", ylab = "number of workers ", cex.main=0.8)
options(scipen = 10)


lines(x, y2, col = "orange", lwd = 2)
#2020
point_to_label <- 361
x_coord <- x[point_to_label]
y_coord <- y1[point_to_label]
#after 12 months 2021
point_to_label <- 373
x_coord1 <- x[point_to_label]
y_coord1 <- y1[point_to_label]

segments(x_coord, 0, x_coord, y_coord, col = "brown", lty = "dashed")
segments(x_coord1, 0, x_coord1, y_coord1, col = "green", lty = "dashed")
legend("right", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



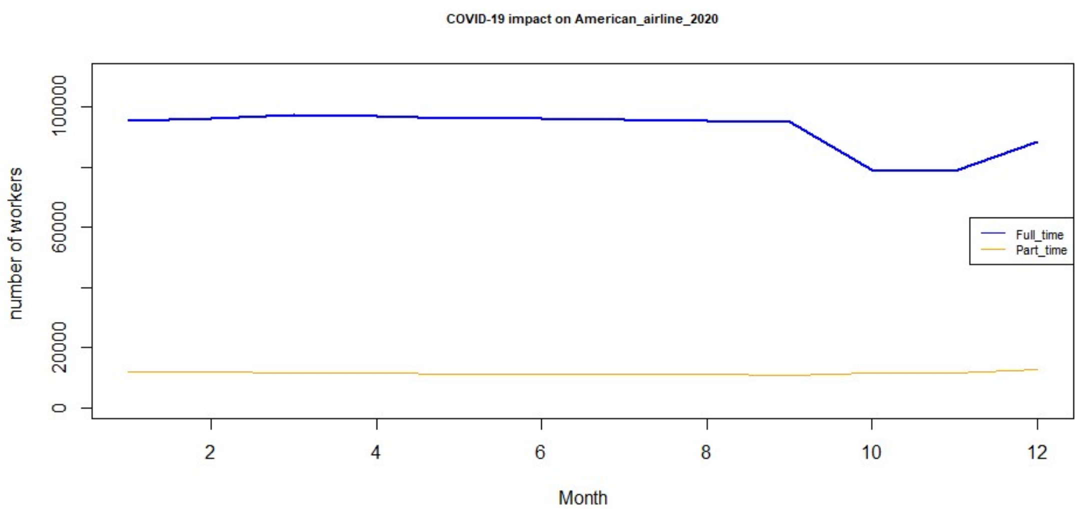
Generally, All airlines experienced huge fall in the number of employees, But we will conclude that there was a important relation between the peak of covid_19 outbreak and this rate of employees. Such as American Airline that this is one of the largest airline companies in the United States, this have been significantly impacted by the COVID-19 pandemic from 2020 to 2022. As we can see in the previous plot, this company experienced a fall in the number of employees frrom 2020 to 2021. For example:

```
#The number of American airline workers in 2020
dt_1[361:372,]

##      Month Year Full.time Part.time Grand.Total
## 361      1 2020    95,612    11,840     107,452
## 362      2 2020    96,021    11,884     107,905
## 363      3 2020    97,373    11,735     109,108
## 364      4 2020    97,083    11,496     108,579
## 365      5 2020    96,062    11,284     107,346
## 366      6 2020    96,061    11,191     107,252
## 367      7 2020    95,792    11,236     107,028
## 368      8 2020    95,312    11,232     106,544
## 369      9 2020    95,197    10,973     106,170
## 370     10 2020    79,281    11,565      90,846
## 371     11 2020    78,761    11,613      90,374
## 372     12 2020    88,418    12,587     101,005
```

we can see this trend more obvious in this plot:

```
x <-  c(1:12)
y1 <- c(full[361:372])
y2 <- c(part[361:372])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 110000),
     main = "COVID-19 impact on American_airline_2020", xlab = "Month", ylab
= "number of workers ", cex.main=0.7)
options(scipen = 1)
```

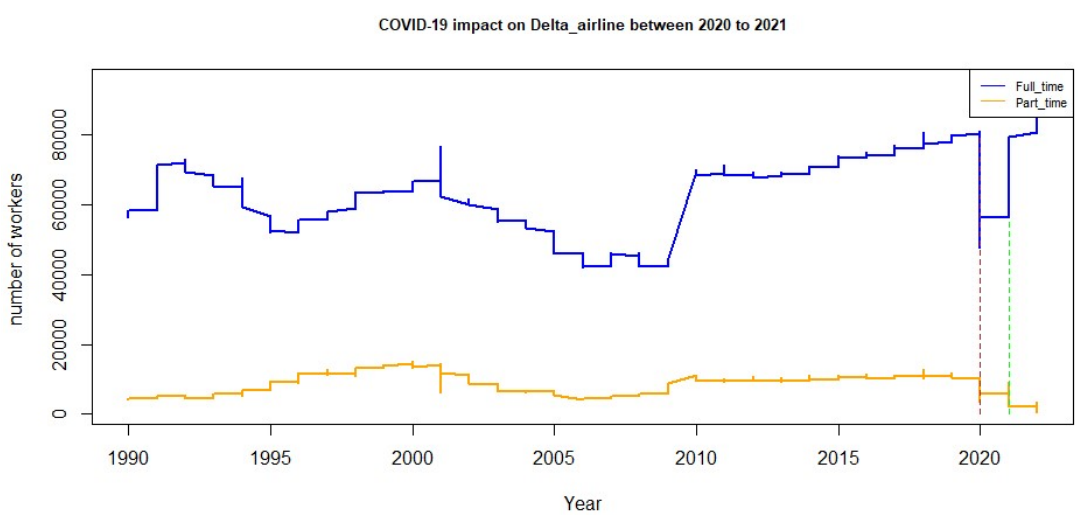
```r
lines(x, y2, col = "orange")
legend("right", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



I think this fact is obvious as clear as black and white that the preference of companies had been changed in pandemic duration because the rate of demand was decreasing and consequently the incomes of them were falling so they preferred to have less full time workers. We did not see any noticeable change in part time workers. Also we can mention healthy issues, because most of the people preferred to work part time or at home in pandemic duration. This fact is attractive that Fed-ex experienced good situation at that moment. we will discuss about this fact. Another noticeable point is that they were a strong relations between these trends and the number of cases for Covid_19.For example in 2021 that U.S.A experienced durations without peak of covid these trend were changed to progressive for all companies. So now we want to compare this trend for other company except Fed-ex.

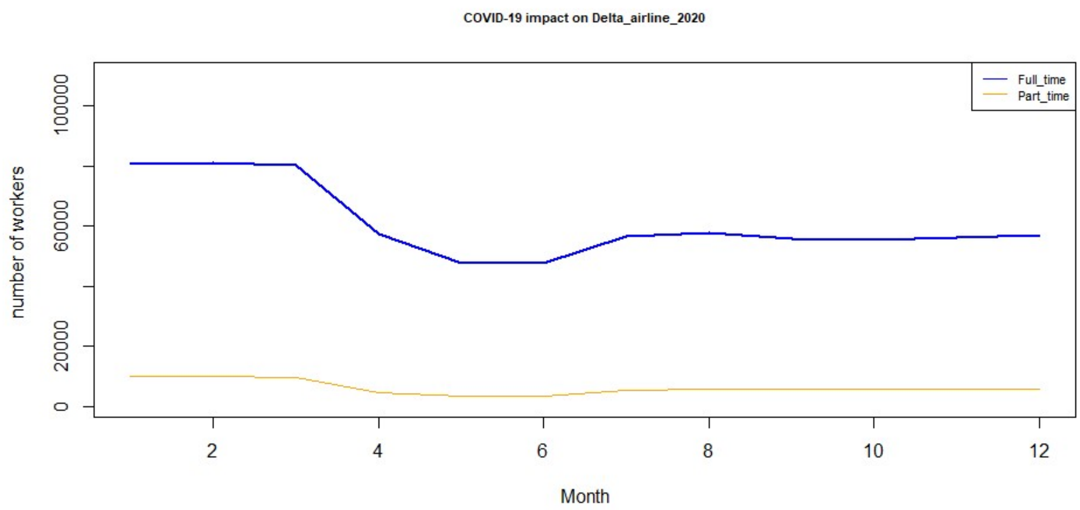
```r
#The number of Delta airline workers in 2020
dt_2[361:372,]

##     Month Year Full.time Part.time Grand.Total
## 361     1 2020    80,582    10,245      90,827
## 362     2 2020    81,177    10,239      91,416
## 363     3 2020    80,179     9,801      89,980
## 364     4 2020    57,498     4,512      62,010
## 365     5 2020    47,581     3,403      50,984
## 366     6 2020    47,877     3,591      51,468
## 367     7 2020    56,513     5,563      62,076
## 368     8 2020    57,630     5,716      63,346
## 369     9 2020    55,767     5,743      61,510
## 370    10 2020    55,384     5,845      61,229
## 371    11 2020    56,393     5,885      62,278
## 372    12 2020    56,827     5,959      62,786
```

```r
x <- rep(c(1990:2022),each=12)
y1 <- c(full_2[1:396])
y2 <- c(part_2[1:396])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 95000),
     main = "COVID-19 impact on Delta_airline between 2020 to 2021", xlab =
"Year", ylab = "number of workers ", cex.main=0.8)
options(scipen = 10)


lines(x, y2, col = "orange", lwd = 2)
#2020
point_to_label <- 361
x_coord <- x[point_to_label]
y_coord <- y1[point_to_label]
#after 12 months 2021
point_to_label <- 373
x_coord1 <- x[point_to_label]
y_coord1 <- y1[point_to_label]

segments(x_coord, 0, x_coord, y_coord, col = "brown", lty = "dashed")
segments(x_coord1, 0, x_coord1, y_coord1, col = "green", lty = "dashed")
legend("topright", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



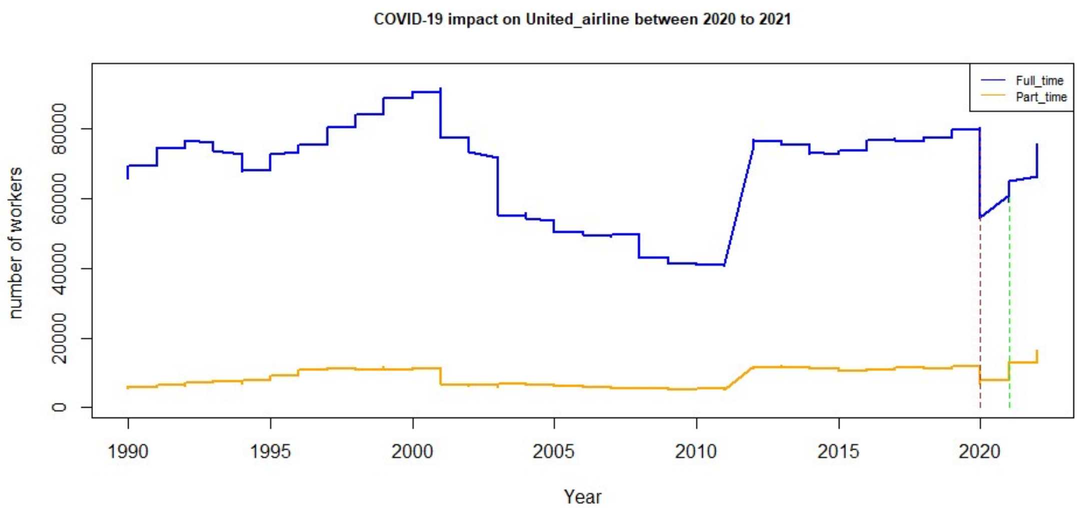

```r
x <-  c(1:12)
y1 <- c(full[758:769])
y2 <- c(part[758:769])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 110000),
     main = "COVID-19 impact on Delta_airline_2020", xlab = "Month", ylab =
"number of workers ", cex.main=0.7)
options(scipen = 1)
lines(x, y2, col = "orange")
```

```
legend("topright", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



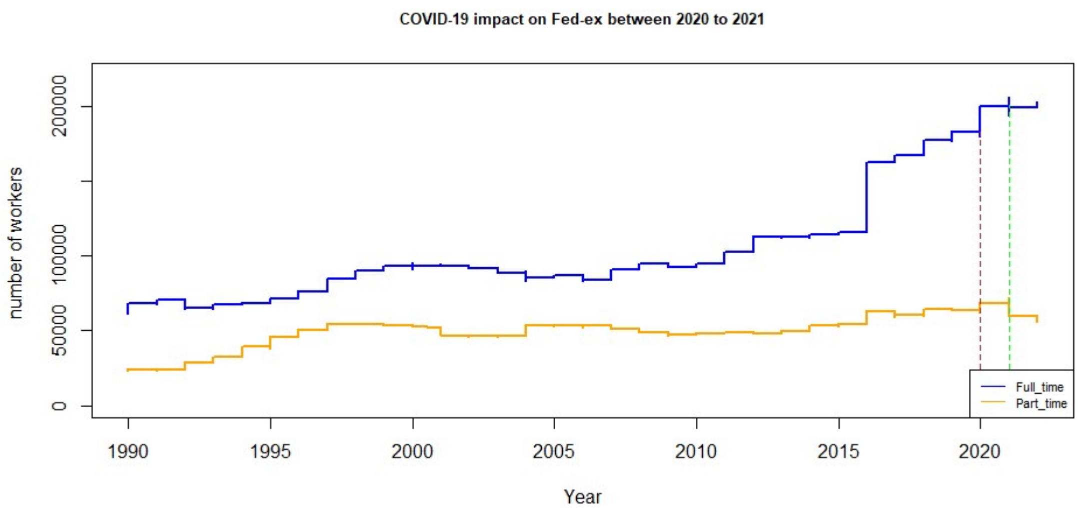
We see same result for United_Airline:

```
x <- rep(c(1990:2022),each=12)
y1 <- c(full_4[1:396])
y2 <- c(part_4[1:396])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 95000),
     main = "COVID-19 impact on United_airline between 2020 to 2021", xlab =
"Year", ylab = "number of workers ", cex.main=0.8)
options(scipen = 10)


lines(x, y2, col = "orange", lwd = 2)
#2020
point_to_label <- 361
x_coord <- x[point_to_label]
y_coord <- y1[point_to_label]
#after 12 months 2021
point_to_label <- 373
x_coord1 <- x[point_to_label]
y_coord1 <- y1[point_to_label]

segments(x_coord, 0, x_coord, y_coord, col = "brown", lty = "dashed")
segments(x_coord1, 0, x_coord1, y_coord1, col = "green", lty = "dashed")
legend("topright", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



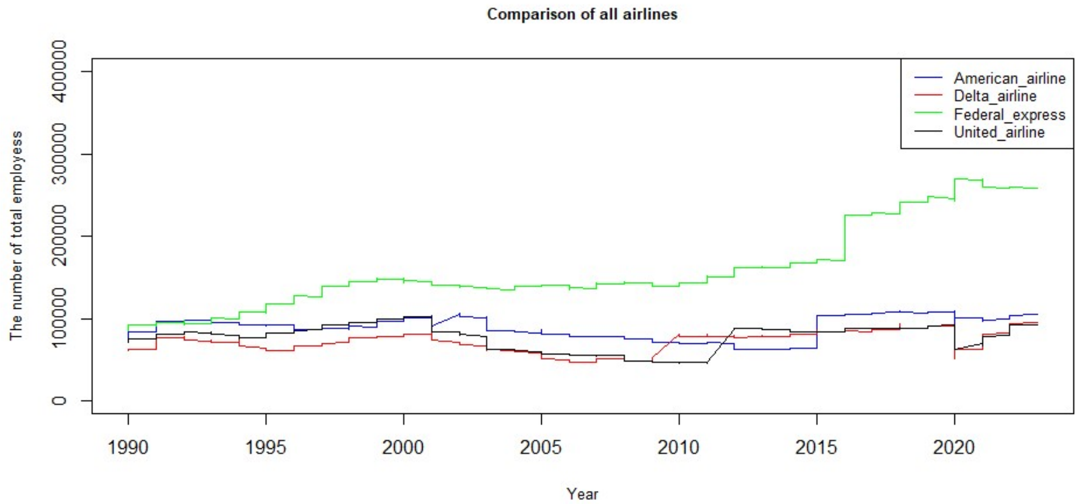
As w mentioned in the Fed-ex this trend was different as we will see this company faced a increasing trend in full time and part time:

```
x <- rep(c(1990:2022),each=12)
y1 <- c(full_3[1:396])
y2 <- c(part_3[1:396])
plot(x, y1, type = "l", col = "blue", lwd = 2, ylim = c(1000, 220000),
     main = "COVID-19 impact on Fed-ex between 2020 to 2021", xlab = "Year",
ylab = "number of workers ", cex.main=0.8)
options(scipen = 10)


lines(x, y2, col = "orange", lwd = 2)
#2020
point_to_label <- 361
x_coord <- x[point_to_label]
y_coord <- y1[point_to_label]
#after 12 months 2021
point_to_label <- 373
x_coord1 <- x[point_to_label]
y_coord1 <- y1[point_to_label]

segments(x_coord, 0, x_coord, y_coord, col = "brown", lty = "dashed")
segments(x_coord1, 0, x_coord1, y_coord1, col = "green", lty = "dashed")
legend("bottomright", legend=c("Full_time", "Part_time"),
       col=c(colors[2],colors[4]), lty=c(1,1),cex = 0.7)
```



Now we have a Comparison of all airlines and companies:

```
vec <- 1990:2023
vec <- vec[-34]
x <- rep(vec, each=12)
x <- c(x, 2023)

y1 <- c(sum_1)
y2 <- c(sum_2)
y3 <- c(sum_3)
y4 <- c(sum_4)


plot(x, y1, type = "l", col = "blue", ylim = c(1000,400000),
     main = "Comparison of all airlines", xlab = "Year", ylab = "The number
of total employess", cex.main=0.8 , cex.lab=0.8)


lines(x, y2, col = "red")

lines(x, y3, col = "green")

lines(x, y4, col = "black")
legend("topright", legend=c("American_airline","Delta_airline",
"Federal_express", "United_airline"),
       col=rep(c(colors[2],colors[1],colors[3],colors[5])),
lty=c(1,1,1,1),cex = 0.8)
```



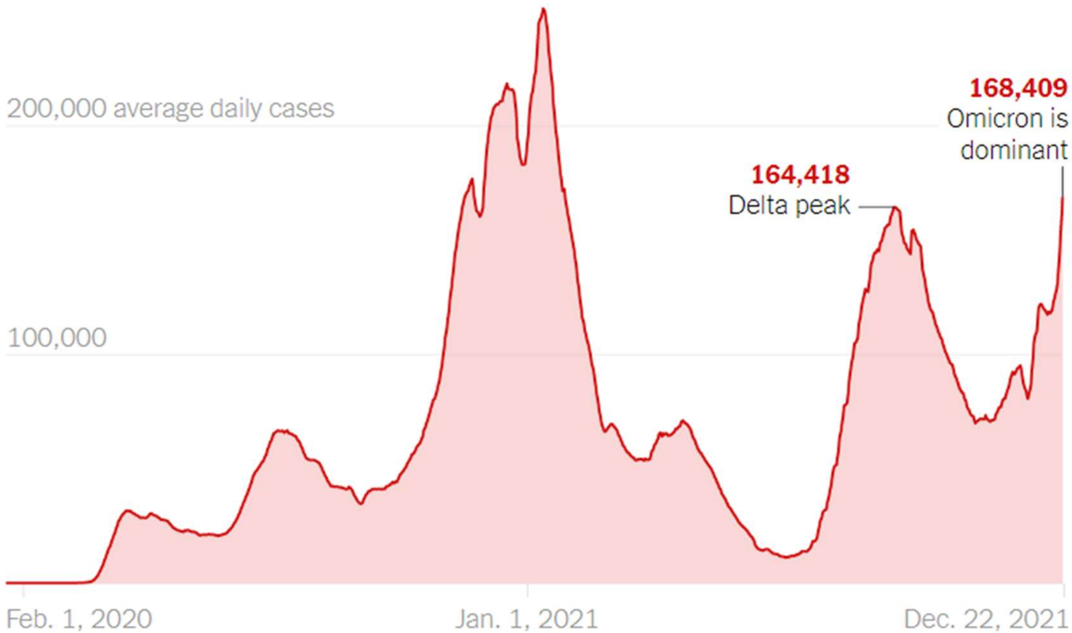
As we see most of them was depended to the peak of covid_19. foe example in 2021, United state faced with the most number of people that suffered of covid_19. so it is logic and make sense that most of air lines had a huge fall in the number of workers. For more informations,American Airlines, Delta Air Lines and United Airlines are three of the largest airline companies in the United States, and all four have been significantly impacted by the COVID-19 pandemic.

American Airlines and Delta Air Lines, like many other airlines, were forced to implement cost-cutting measures in response to the decline in air travel demand caused by the pandemic. In 2020, American Airlines announced plans to furlough 19,000 employees, while Delta Air Lines furloughed more than 1,700 pilots. Both airlines also implemented pay cuts for many employees and reduced their flight schedules in response to the decline in demand.

United Airlines also faced similar challenges, announcing plans to furlough 13,000 employees in 2020. In addition to furloughs, the company also implemented pay cuts and reduced its flight schedules in response to the pandemic.

In the other hand the Fed-ex had a good situation in this pandemic. because this company had a lot of request and experienced more demand vice verse other airlines. This fact was rooted in people preferences at that moment because they preferred to received their own stuffs with these kind of company rather than participating in society.so we did not see decreasing trend in this company.

As we can see in the New York times data, The falling and increasing of companies were related to peak of Covid_19:



Exercise 2 - Data Frames and Tibble

the nycflights13 R package contains data on all flights departing from New Your City airports in 2013. All available data is organized into four tibbles:

airlines: contains metadata on airlines names and corresponding carrier codes

airports: contains metadata on all airports connected to NYC

flights: has data of all flights departing from the three NYC airports (JFK, LGA and EWR) in 2013

planes: Plane metadata for all plane numbers found in the FAA aircraft registry.

1.First of all we Plot the total number of flights departed from each of the three NYC airports as a function of time.

2.secondly, Plot the average number of flights computed over the first five working days of each week as a function of the week number of the year.

3.Then, For each flight in the data frame we compute the departure delay and extract the following pieces of information.

4.We plot of the average plane speed as a function of departure day of the year.

5.After that,we analyze the flights offered by each airline company and determine.

```
# Load the packages
library(nycflights13)
```

```
library(dplyr)
library(ggplot2)

str(flights)

## tibble [336,776 × 19] (S3: tbl_df/tbl/data.frame)
##  $ year          : int [1:336776] 2013 2013 2013 2013 2013 2013 2013 2013
2013 2013 ...
##  $ month         : int [1:336776] 1 1 1 1 1 1 1 1 1 1 ...
##  $ day           : int [1:336776] 1 1 1 1 1 1 1 1 1 1 ...
##  $ dep_time      : int [1:336776] 517 533 542 544 554 554 555 557 557 558
...
##  $ sched_dep_time: int [1:336776] 515 529 540 545 600 558 600 600 600 600
...
##  $ dep_delay     : num [1:336776] 2 4 2 -1 -6 -4 -5 -3 -3 -2 ...
##  $ arr_time      : int [1:336776] 830 850 923 1004 812 740 913 709 838 753
...
##  $ sched_arr_time: int [1:336776] 819 830 850 1022 837 728 854 723 846 745
...
##  $ arr_delay     : num [1:336776] 11 20 33 -18 -25 12 19 -14 -8 8 ...
##  $ carrier       : chr [1:336776] "UA" "UA" "AA" "B6" ...
##  $ flight        : int [1:336776] 1545 1714 1141 725 461 1696 507 5708 79
301 ...
##  $ tailnum       : chr [1:336776] "N14228" "N24211" "N619AA" "N804JB" ...
##  $ origin        : chr [1:336776] "EWR" "LGA" "JFK" "JFK" ...
##  $ dest          : chr [1:336776] "IAH" "IAH" "MIA" "BQN" ...
##  $ air_time      : num [1:336776] 227 227 160 183 116 150 158 53 140 138
...
##  $ distance      : num [1:336776] 1400 1416 1089 1576 762 ...
##  $ hour          : num [1:336776] 5 5 5 5 6 5 6 6 6 6 ...
##  $ minute        : num [1:336776] 15 29 40 45 0 58 0 0 0 0 ...
##  $ time_hour     : POSIXct[1:336776], format: "2013-01-01 05:00:00" "2013-
01-01 05:00:00" ...
```

In this part first of all we create a new column for date then we want to group flights by date and origin airport, and count the number of flights for each group.

```
#exercise2-1

flights$date <- as.Date(paste(flights$year, flights$month, flights$day), "%Y
%m %d")
str(flights$date)

##  Date[1:336776], format: "2013-01-01" "2013-01-01" "2013-01-01" "2013-01-
01" "2013-01-01" ...

# group flights by date and origin airport, and count the number of flights
for each group
flights_by_date_origin <- flights %>%
  group_by(date, origin) %>%
  summarize(num_flights = n())
```

```
## `summarise()` has grouped output by 'date'. You can override using the
## `.groups` argument.
```

```
flights_by_date_origin
```

```
## # A tibble: 1,095 × 3
## # Groups:   date [365]
##    date       origin num_flights
##    <date>     <chr>        <int>
##  1 2013-01-01 EWR            305
##  2 2013-01-01 JFK            297
##  3 2013-01-01 LGA            240
##  4 2013-01-02 EWR            350
##  5 2013-01-02 JFK            321
##  6 2013-01-02 LGA            272
##  7 2013-01-03 EWR            336
##  8 2013-01-03 JFK            318
##  9 2013-01-03 LGA            260
## 10 2013-01-04 EWR            339
## # ℹ 1,085 more rows
```

For example the number of flights from 'LGA' at the fourth day of 2013 was:

```
flights_by_date_origin[12,]
```

```
## # A tibble: 1 × 3
## # Groups:   date [1]
##   date       origin num_flights
##   <date>     <chr>        <int>
## 1 2013-01-04 LGA            258
```

Now we Plot the total number of flights departed from each of the three NYC airports:

```
# plot the data
ggplot(flights_by_date_origin, aes(x = date, y = num_flights, color =
origin)) +
  geom_line() +
  scale_x_date(date_labels = "%b %d", date_breaks = "1 day") +
  labs(title = "Total Number of Flights Departed by Origin Airport",
       x = "Date", y = "Number of Flights", color = "Origin") +
  theme_minimal()
```

Firstly we add a new column called “day_of_week” to the “flights” data frame that contains the day of the week for each flight date using the “weekdays()” function.

Then, we create two new data frames called “flights_by_week” and “flights_by_weekend” that group the flights by week number, day of the week, and origin airport and calculate the average number of flights for each week and origin on working days and weekends, respectively.

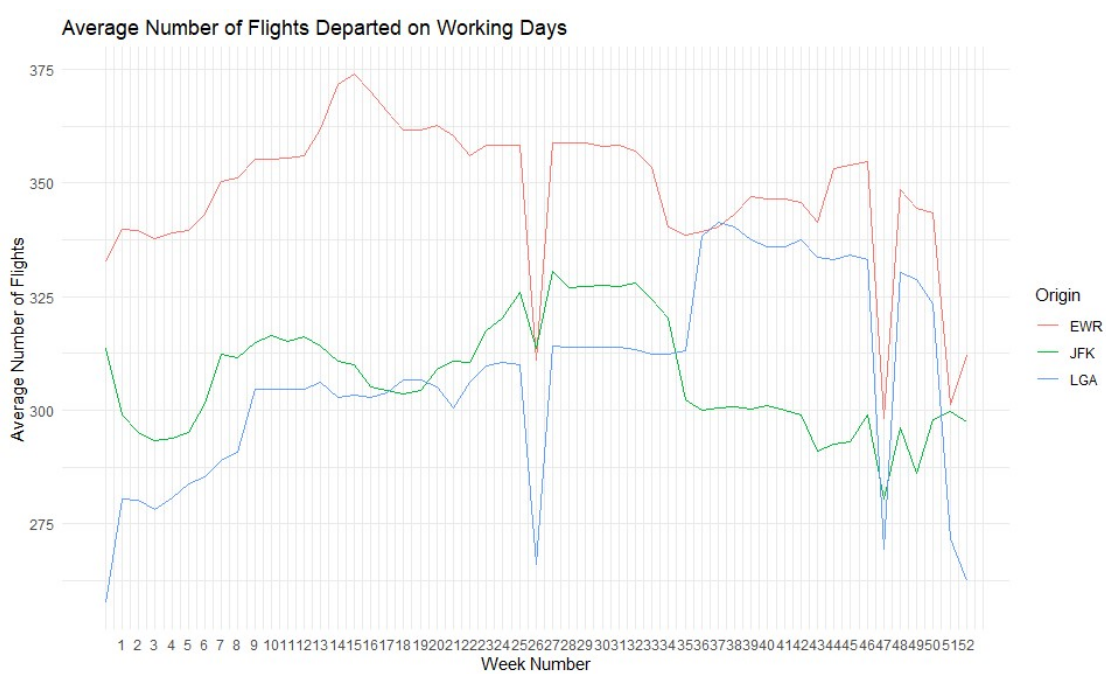
```
#exercise2-2
flights$day_of_week <- weekdays(flights$date)
str(flights$day_of_week)

##  chr [1:336776] "Tuesday" "Tuesday" "Tuesday" "Tuesday" "Tuesday"
"Tuesday" ...

# define working days from Monday to Friday
working_days <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")

# group flights by week number, day of the week, and origin airport
flights_by_week <- flights %>%
  mutate(week_num = format(date, "%U")) %>%
  filter(day_of_week %in% working_days) %>%
  group_by(week_num, day_of_week, origin) %>%
  summarize(num_flights = n()) %>%
  group_by(week_num, origin) %>%
  summarize(avg_flights = mean(num_flights))

## `summarise()` has grouped output by 'week_num', 'day_of_week'. You can
override
## using the `.groups` argument.
```

```
## `summarise()` has grouped output by 'week_num'. You can override using the
## `.groups` argument.
```

```
flights_by_weekend <- flights %>%
  mutate(week_num = format(date, "%U")) %>%
  filter(day_of_week %in% c("Saturday", "Sunday")) %>%
  group_by(week_num, origin) %>%
  summarize(num_flights = n()) %>%
  group_by(week_num, origin) %>%
  summarize(avg_flights = mean(num_flights))
```

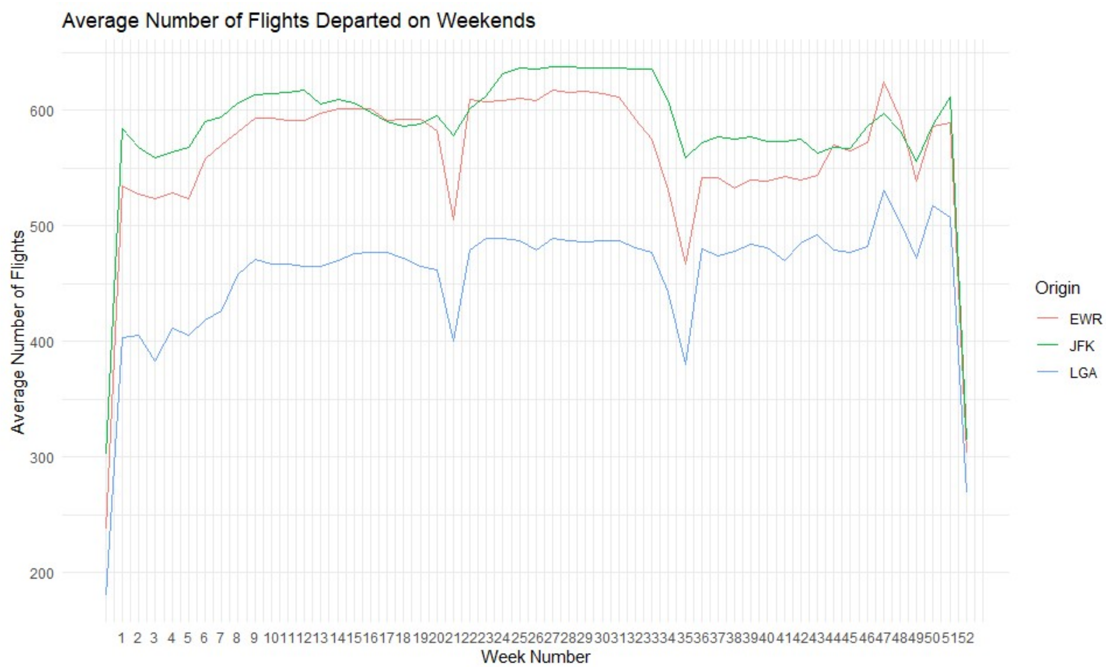
```
## `summarise()` has grouped output by 'week_num'. You can override using the
## `.groups` argument.
## `summarise()` has grouped output by 'week_num'. You can override using the
## `.groups` argument.
```

```
# plot for working days
ggplot(flights_by_week, aes(x = as.numeric(week_num), y = avg_flights, color
= origin)) +
  geom_line() +
  scale_x_continuous(breaks = seq(1, 52, by = 1)) +
  labs(title = "Average Number of Flights Departed on Working Days",
       x = "Week Number", y = "Average Number of Flights", color = "Origin")
+
  theme_minimal()
```



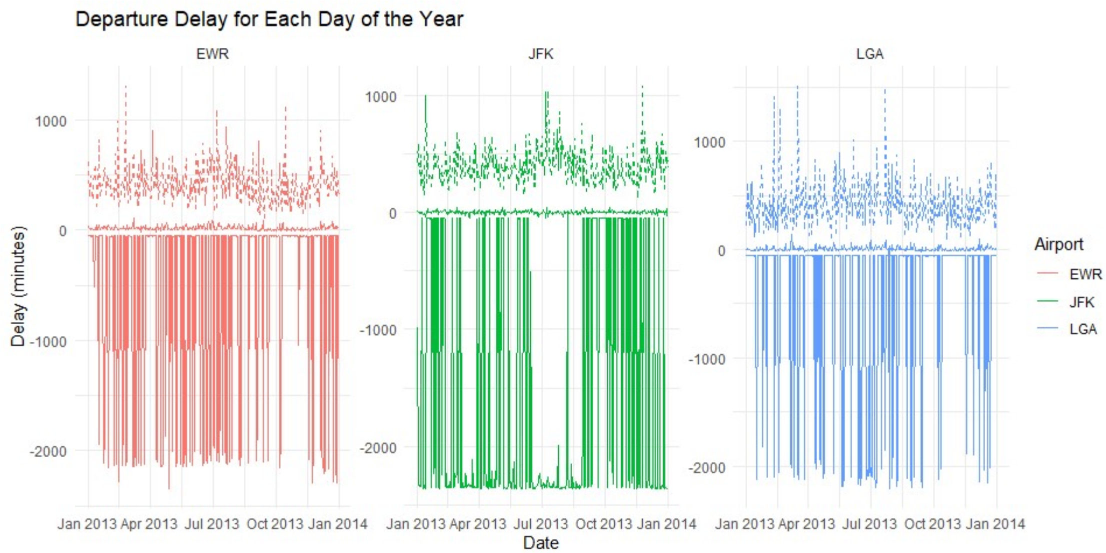

```
# plot for weekends
ggplot(flights_by_weekend, aes(x = as.numeric(week_num), y = avg_flights,
color = origin)) +
  geom_line() +
```

```
  scale_x_continuous(breaks = seq(1, 52, by = 1)) +
  labs(title = "Average Number of Flights Departed on Weekends",
       x = "Week Number", y = "Average Number of Flights", color = "Origin")
+
  theme_minimal()
```

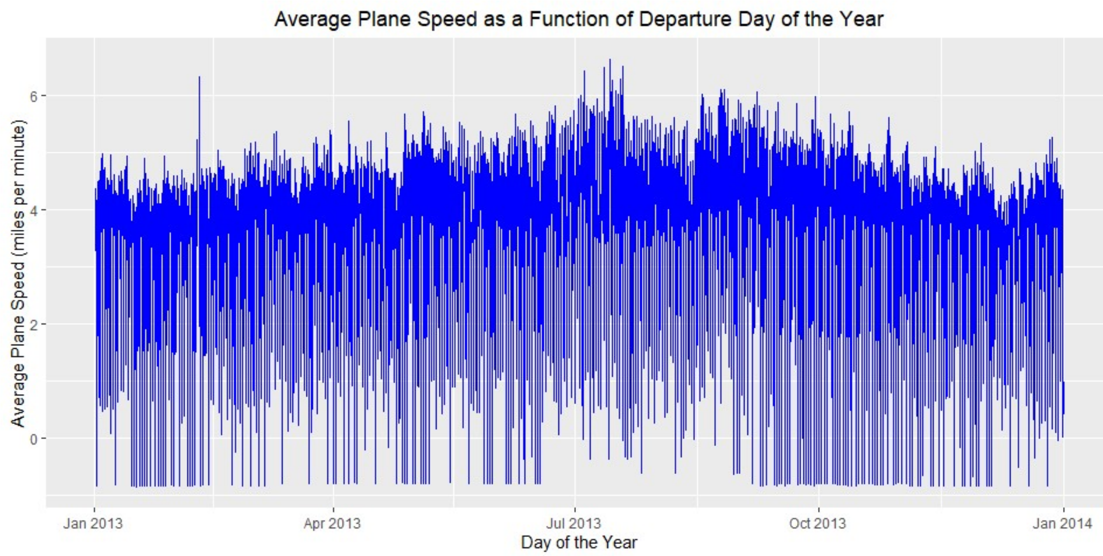



```
#exercise2-3
# Compute departure delay for each flight
flights <- flights %>%
  mutate(dep_delay = dep_time - sched_dep_time)
flights

## # A tibble: 336,776 × 21
##      year month   day dep_time sched_dep_time dep_delay arr_time
sched_arr_time
##     <int> <int> <int>    <int>          <int>     <int>    <int>
<int>
##   1  2013     1     1      517            515         2      830
819
##   2  2013     1     1      533            529         4      850
830
##   3  2013     1     1      542            540         2      923
850
##   4  2013     1     1      544            545        -1     1004
1022
##   5  2013     1     1      554            600       -46      812
837
##   6  2013     1     1      554            558        -4      740
728
##   7  2013     1     1      555            600       -45      913
```

```
854
##  8  2013     1     1      557            600       -43      709
723
##  9  2013     1     1      557            600       -43      838
846
## 10  2013     1     1      558            600       -42      753
745
## # ℹ 336,766 more rows
## # ℹ 13 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
## #   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance
<dbl>,
## #   hour <dbl>, minute <dbl>, time_hour <dttm>, date <date>, day_of_week
<chr>

# Compute min, max, and average delay for each day of the year for each
airport
delay_summary <- flights %>%
  group_by(year, month, day, origin) %>%
  summarize(min_delay = min(dep_delay, na.rm = TRUE),
            max_delay = max(dep_delay, na.rm = TRUE),
            avg_delay = mean(dep_delay, na.rm = TRUE))

## `summarise()` has grouped output by 'year', 'month', 'day'. You can
override
## using the `.groups` argument.

# Plot
ggplot(delay_summary, aes(x = as.Date(paste(year, month, day, sep = "-")), y
= avg_delay, color = origin)) +
  geom_line(aes(y = min_delay), linetype = "solid") +
  geom_line(aes(y = max_delay), linetype = "dashed") +
  geom_line() +
  labs(title = "Departure Delay for Each Day of the Year",
       x = "Date", y = "Delay (minutes)", color = "Airport") +
  facet_wrap(~ origin, scales = "free_y") +
  theme_minimal()
```




```r
#exercise2-4
# Calculate flight duration in minutes
flights$duration <- as.numeric(flights$arr_time - flights$dep_time)

# Calculate average plane speed in miles per minute (v = x/t)
flights$avg_speed <- flights$distance / flights$duration

# Extract the day of the year from the departure time
flights$day_of_year <- (flights$time_hour)

# Calculate the average plane speed for each day of the year
avg_speed_by_day <- flights %>%
  group_by(day_of_year) %>%
  summarise(avg_speed = mean(avg_speed, na.rm = TRUE))

# Plot average plane speed
ggplot(avg_speed_by_day, aes(x = day_of_year, y = avg_speed)) +
  geom_line(color = "blue") +
  labs(x = "Day of the Year", y = "Average Plane Speed (miles per minute)") +
  ggtitle("Average Plane Speed as a Function of Departure Day of the Year") +
  theme(plot.title = element_text(hjust = 0.5))
```




```
#exercise2-5
# Group flights by airlines and the number of flights per day
flights_per_day <- flights %>%
  group_by(carrier, year, month, day) %>%
  summarize(n = n())

## `summarise()` has grouped output by 'carrier', 'year', 'month'. You can
## override using the `.groups` argument.
```

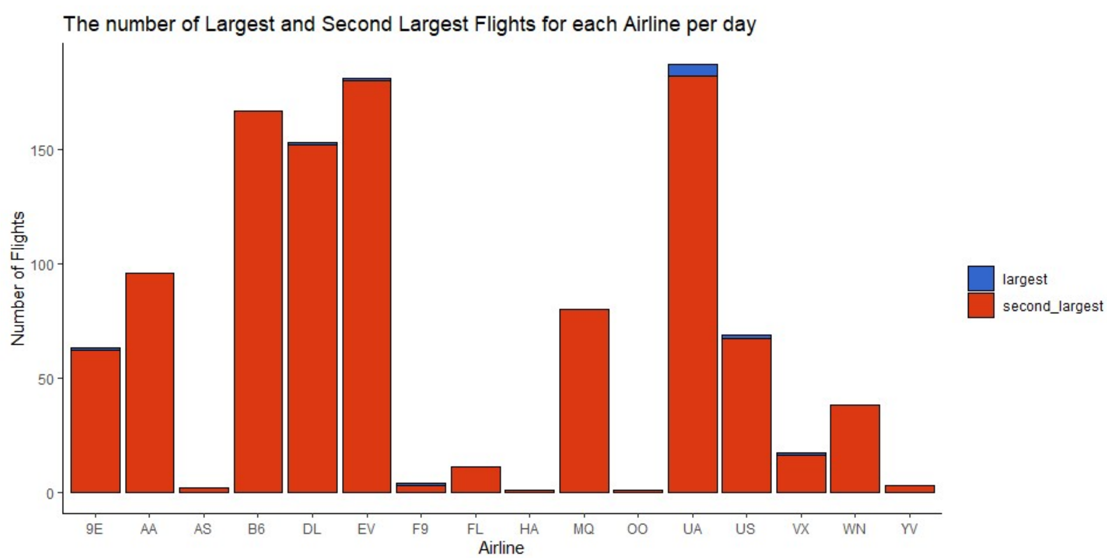
we can use this code to find first and second largest number of flights per day for each airline but it is time_consuming so we can use one suitable loop for this work.

```
# row_num_largest <- which.max(flights_per_day[[5]])
# # Exclude the row with the largest value
# flights_per_day_excl_largest <- flights_per_day[-row_num_largest,]
#
# # Find the row with the second-largest value
# row_num_second_largest <- which.max(flights_per_day_excl_largest[[5]])
#
# large_1 <- data.frame(matrix(0, nrow=2, ncol=5))
# names(large_1) <- c("carrier" ,   "year",  "month"  , "day" ,    "n")
# rownames(large_1) <- c("largest", "second_largest")
# large_1 <- data.frame(
#   flights_per_day[row_num_largest,],
# flights_per_day[row_num_second_largest ,])


# Initialize an empty data frame to store the largest and second largest
number of flights per day for each airline
large_1 <- data.frame(matrix(0, nrow=0, ncol=6))
names(large_1) <- c("carrier", "year", "largest_num_flights_per_day",
"second_largest_num_flights_per_day", "largest_date", "second_largest_date")
```

```
# Loop over each unique airline
for (airline in unique(flights_per_day$carrier)) {
  # Subset the data for the current airline
  airline_data <- filter(flights_per_day, carrier == airline)

  # Find the row with the largest value
  row_num_largest <- which.max(airline_data[[5]])

  # Exclude the row with the largest value
  airline_data_excl_largest <- airline_data[-row_num_largest,]

  # Find the row with the second-largest value
  row_num_second_largest <- which.max(airline_data_excl_largest[[5]])

  # Create a data frame with the largest and second-largest values for the
current airline
  airline_large_1 <- data.frame(
    carrier = airline,
    year = airline_data[row_num_largest, "year"],
    largest_num_flights_per_day = airline_data[row_num_largest,5],
    second_largest_num_flights_per_day =
airline_data_excl_largest[row_num_second_largest,5],
    largest_date = as.Date(paste(airline_data[row_num_largest, "year"],
airline_data[row_num_largest, "month"], airline_data[row_num_largest, "day"],
sep="-")),
    second_largest_date =
as.Date(paste(airline_data_excl_largest[row_num_second_largest, "year"],
airline_data_excl_largest[row_num_second_largest, "month"],
airline_data_excl_largest[row_num_second_largest, "day"], sep="-"))
  )

  # Add the data for the current airline to the overall data frame
  large_1 <- rbind(large_1, airline_large_1)
}


#it is sorted
sorted_large_1 <- large_1[order(large_1[,3], decreasing = TRUE),]
sorted_large_1

##    carrier year   n n.1 largest_date second_largest_date
## 12      UA 2013 187 182   2013-11-27          2013-06-11
## 6       EV 2013 181 180   2013-09-23          2013-09-16
## 4       B6 2013 167 167   2013-12-23          2013-12-27
## 5       DL 2013 153 152   2013-12-27          2013-01-02
## 2       AA 2013  96  96   2013-06-13          2013-06-14
## 10      MQ 2013  80  80   2013-08-29          2013-09-05
## 13      US 2013  69  67   2013-11-26          2013-11-01
## 1       9E 2013  63  62   2013-12-05          2013-12-04
```

```
## 15       WN 2013   38  38   2013-11-04          2013-11-05
## 14       VX 2013   17  16   2013-07-11          2013-04-03
## 8        FL 2013   11  11   2013-01-02          2013-01-03
## 7        F9 2013    4   3   2013-09-13          2013-05-01
## 16       YV 2013    3   3   2013-02-17          2013-02-24
## 3        AS 2013    2   2   2013-01-01          2013-01-02
## 9        HA 2013    1   1   2013-01-01          2013-01-02
## 11       OO 2013    1   1   2013-01-30          2013-06-15
```

```
# Create a data frame for the data you provided
data <- data.frame(n = c(63, 96, 2, 167, 153, 181, 4, 11, 1, 80, 1, 187, 69,
17, 38, 3),
                   n_1 = c(62, 96, 2, 167, 152, 180, 3, 11, 1, 80, 1, 182,
67, 16, 38, 3),
                   carrier = c(large_1[,1]))

# Define the colors for the bars
colors <- c("#3366CC", "#DC3912")

# Create a bar plot with two bars for each airline
ggplot(data, aes(x = factor(carrier), y = n, fill = "largest")) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  geom_bar(aes(y = n_1, fill = "second_largest"), stat = "identity", position
= "dodge", color = "black") +
  scale_fill_manual(values = colors, name = "") +
  theme_classic() +
  labs(title = "The number of Largest and Second Largest Flights for each
Airline per day", x = "Airline", y = "Number of Flights") +
  guides(fill = guide_legend(title = ""))
```



Now we work like previous task and find the first and second lsrgest nymber of flight per week for each airline:

```r
# Calculate the number of flights per week for each airline
# Group flights by carrier, year, and week
flights_per_week <- flights_per_day %>%
  group_by(carrier, year, week = format(as.Date(paste(year, month, day, sep =
"-")), "%U")) %>%
  summarize(n = sum(n))
```

```
## `summarise()` has grouped output by 'carrier', 'year'. You can override
using
## the `.groups` argument.
```

```r
# Initialize data frame
large_1_week <- data.frame(matrix(0, nrow=0, ncol=4))
names(large_1_week) <- c("carrier", "year",  "largest_num_flights_per_week",
"second_largest_num_flights_per_week")

# Loop over each unique airline
for (airline in unique(flights_per_week$carrier)) {
  # Subset the data for the current airline
  airline_data <- filter(flights_per_week, carrier == airline)

  # Find the row with the largest value
  row_num_largest <- which.max(airline_data[[4]])

  # Exclude the row with the largest value
  airline_data_excl_largest <- airline_data[-row_num_largest,]

  # Find the row with the second-largest value
  row_num_second_largest <- which.max(airline_data_excl_largest[[4]])

  # Create a data frame with the largest and second-largest values for the
current airline
  airline_large_1_week <- data.frame(
    carrier = airline,
    year = airline_data[row_num_largest, "year"],
    largest_num_flights_per_week = airline_data[row_num_largest, 4],
    second_largest_num_flights_per_week =
airline_data_excl_largest[row_num_second_largest, 4]
  )

  # Add the data for the current airline to the overall data frame
  large_1_week <- rbind(large_1_week, airline_large_1_week)
}

#it is sorted
sorted_large_1_week <- large_1_week[order(large_1_week[,3], decreasing =
TRUE),]
sorted_large_1_week
```

```
##    carrier year    n  n.1
## 12      UA 2013 1189 1188
## 4       B6 2013 1147 1126
## 6       EV 2013 1145 1137
## 5       DL 2013  982  982
## 2       AA 2013  655  655
## 10      MQ 2013  518  518
## 13      US 2013  419  419
## 1       9E 2013  408  397
## 15      WN 2013  257  253
## 14      VX 2013  113  112
## 8       FL 2013   74   74
## 16      YV 2013   18   18
## 7       F9 2013   15   14
## 3       AS 2013   14   14
## 9       HA 2013    7    7
## 11      OO 2013    6    6
```

```
flights_per_month <- flights %>%
  group_by(carrier, year, month) %>%
  summarize(n = n())
```

```
## `summarise()` has grouped output by 'carrier', 'year'. You can override
using
## the `.groups` argument.
```

```
smallest_flights_per_month <- flights_per_month %>%
  group_by(carrier) %>%
  summarize(smallest_n = min(n)) %>%
  arrange(smallest_n)
smallest_flights_per_month
```

```
## # A tibble: 16 × 2
##    carrier smallest_n
##    <chr>        <int>
##  1 OO               1
##  2 YV              18
##  3 HA              21
##  4 F9              49
##  5 AS              52
##  6 FL             202
##  7 VX             271
##  8 WN             911
##  9 9E            1437
## 10 US            1552
## 11 MQ            2044
## 12 AA            2517
## 13 DL            3444
## 14 EV            3827
## 15 B6            4103
## 16 UA            4346
```

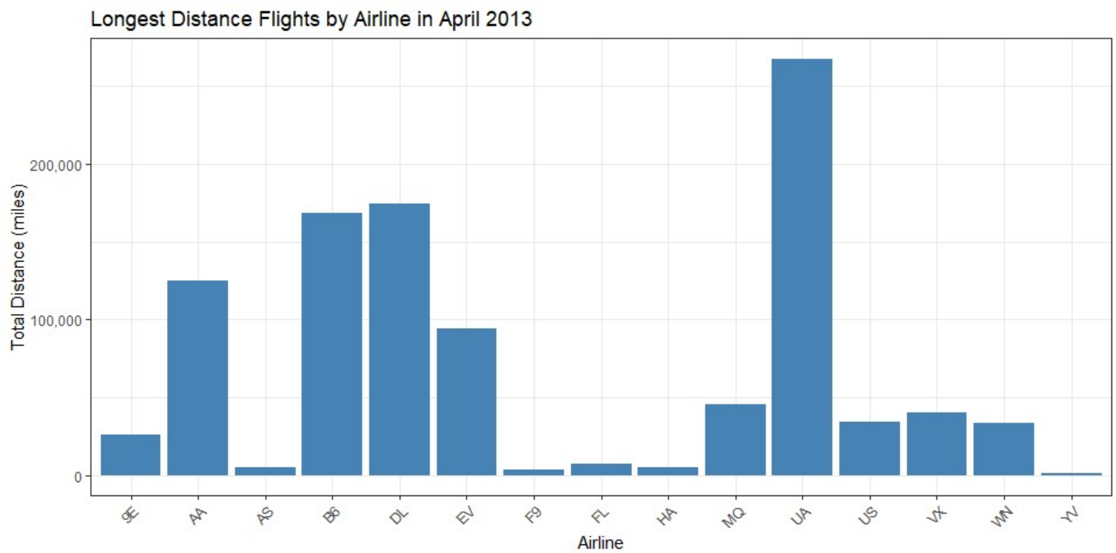
```
distance_per_day <- flights %>%
  group_by(carrier, year, month, day) %>%
  summarize(total_distance = sum(distance))
```

```
## `summarise()` has grouped output by 'carrier', 'year', 'month'. You can
## override using the `.groups` argument.
```

```
longest_distance_per_month <- distance_per_day %>%
  group_by(carrier, year, month) %>%
  arrange(desc(total_distance)) %>%
  slice(1)
#for example:To see the longest distance flights for all airlines in the
fourth month
longest_distance_april <- longest_distance_per_month %>%
  filter(month == 4) %>%
  arrange(desc(total_distance))
longest_distance_april
```

```
## # A tibble: 15 × 5
## # Groups:   carrier, year, month [15]
##    carrier  year month   day total_distance
##    <chr>   <int> <int> <int>          <dbl>
##  1 UA       2013     4    15         267591
##  2 DL       2013     4    15         174752
##  3 B6       2013     4     1         168783
##  4 AA       2013     4     2         125238
##  5 EV       2013     4    11          94017
##  6 MQ       2013     4     4          45437
##  7 VX       2013     4     3          40024
##  8 US       2013     4    15          34361
##  9 WN       2013     4     1          33599
## 10 9E       2013     4    28          26195
## 11 FL       2013     4     1           7287
## 12 HA       2013     4     1           4983
## 13 AS       2013     4     1           4804
## 14 F9       2013     4     1           3240
## 15 YV       2013     4     9           1088
```

```
ggplot(longest_distance_april, aes(x = carrier, y = total_distance)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  ggtitle("Longest Distance Flights by Airline in April 2013") +
  xlab("Airline") +
  ylab("Total Distance (miles)") +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_y_continuous(labels = scales::comma)
```


Longest Distance Flights by Airline in April 2013
Total Distance (miles)
200,000
100,000
0
9E
AA
AS
B6
DL
EV
F9
FL
HA
MQ
UA
US
VX
WN
YV
Airline