Air Passenger Demand Forecast

Mengxuan Lu

12/11/2020

```r
library(easypackages)
libraries("readr","tidyverse","ggplot2","lubridate","ggpubr","forecast","seasonal")
```

General Overview

Goal

Forecast the U.S. air passenger demand.

Source of the data

I downloaded the data from https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=259&DB_Short_Name=Air%20Carriers.

Overview the data

```r
US_carrier <- read_csv("US_carrier.csv")
glimpse(US_carrier)
```

```
## Rows: 3,840,585
## Columns: 23
## $ DEPARTURES_SCHEDULED <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...
## $ DEPARTURES_PERFORMED <dbl> 11, 1, 1, 14, 56, 1, 4, 1, 2, 37, 7, 2, 3, 28,...
## $ PAYLOAD              <dbl> 383600, 12500, 12500, 746000, 67200, 1200, 500...
## $ SEATS                <dbl> 0, 50, 50, 25, 336, 6, 0, 0, 0, 222, 63, 18, 1...
## $ PASSENGERS           <dbl> 0, 46, 46, 0, 109, 1, 0, 0, 0, 83, 15, 9, 6, 3...
## $ FREIGHT              <dbl> 33915, 0, 0, 221471, 4, 0, 0, 0, 0, 0, 0, 0, 0...
## $ MAIL                 <dbl> 4598, 0, 0, 0, 194, 0, 0, 0, 0, 0, 0, 0, 0, 55...
## $ DISTANCE             <dbl> 571, 507, 501, 507, 40, 40, 40, 40, 40, 40, 40...
## $ RAMP_TO_RAMP         <dbl> 1149, 115, 121, 1503, 1379, 19, 100, 19, 44, 7...
## $ AIR_TIME             <dbl> 928, 89, 88, 1069, 1267, 17, 80, 14, 34, 629, ...
## $ UNIQUE_CARRIER       <chr> "AS", "OH (1)", "OH (1)", "PT (1)", "4Y", "4Y"...
## $ CARRIER_NAME         <chr> "Alaska Airlines Inc.", "Comair Inc.", "Comair...
## $ CARRIER_GROUP_NEW    <dbl> 3, 3, 3, 1, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5...
## $ ORIGIN               <chr> "JNU", "CVG", "RDU", "CVG", "EEK", "EEK", "EEK...
## $ ORIGIN_CITY_NAME     <chr> "Juneau, AK", "Cincinnati, OH", "Raleigh/Durha...
## $ ORIGIN_STATE_ABR     <chr> "AK", "KY", "NC", "KY", "AK", "AK", "AK", "AK"...
## $ DEST                 <chr> "ANC", "PHL", "DTW", "PHL", "BET", "BET", "BET...
```

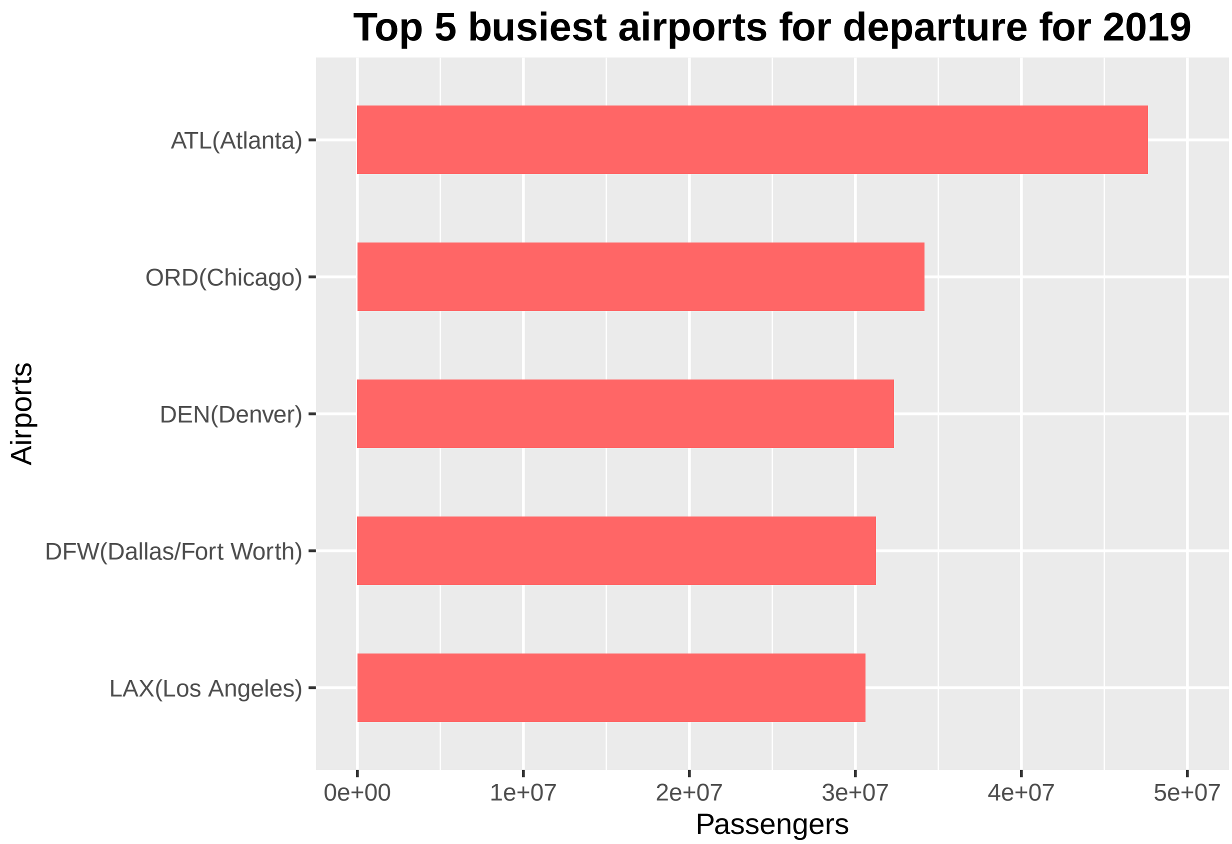
```
## $ DEST_CITY_NAME       <chr> "Anchorage, AK", "Philadelphia, PA", "Detroit,...
## $ DEST_STATE_ABR       <chr> "AK", "PA", "MI", "PA", "AK", "AK", "AK", "AK"...
## $ YEAR                 <dbl> 2010, 2010, 2010, 2010, 2010, 2010, 2010, 2010...
## $ QUARTER              <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2...
## $ MONTH                <dbl> 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 5...
## $ CLASS                <chr> "G", "F", "F", "P", "F", "L", "G", "G", "G", "...
```

EDA

Questions I am interested

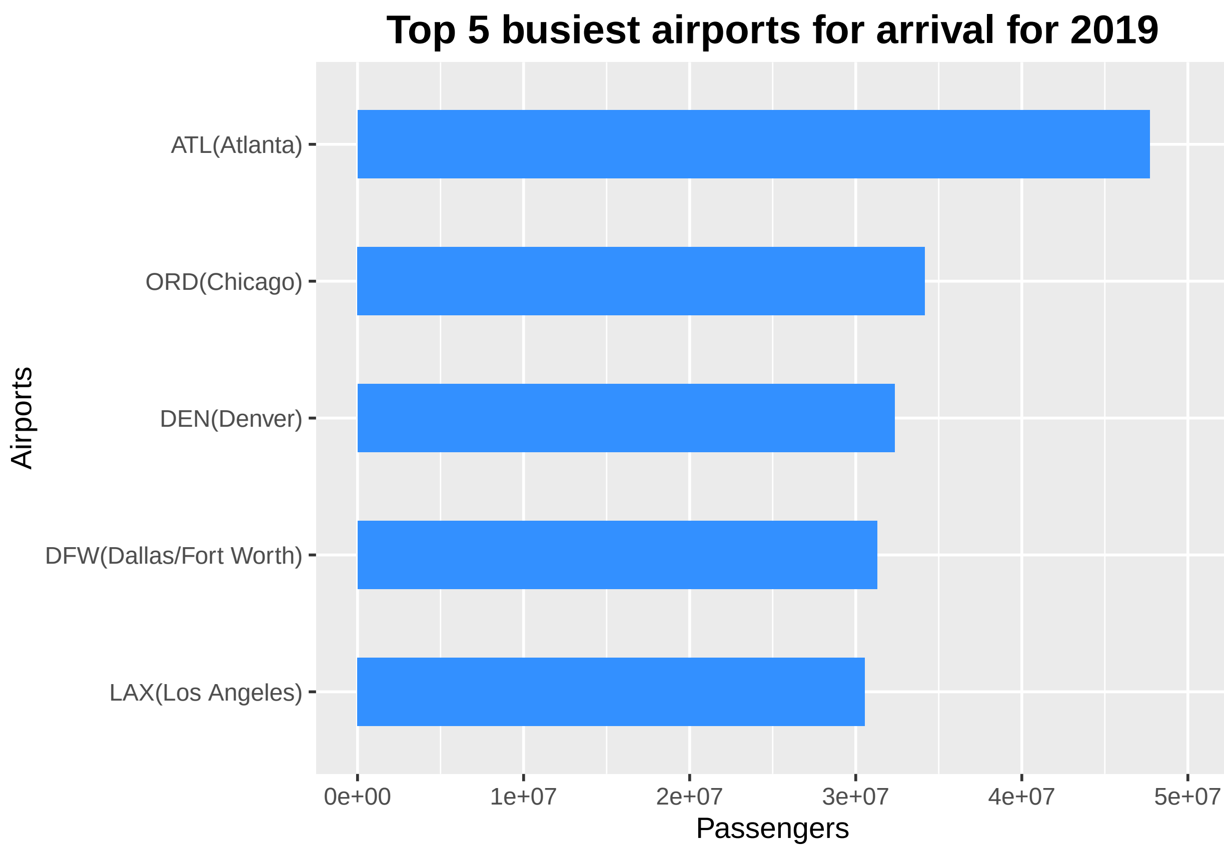
1) What are the top 5 busiests airport for departure in 2019?

```r
top_airport_depart <- US_carrier %>%
  filter(YEAR == 2019) %>%
  select(ORIGIN, ORIGIN_CITY_NAME, PASSENGERS) %>%
  group_by(ORIGIN, ORIGIN_CITY_NAME) %>%
  summarise(Passengers = sum(PASSENGERS)) %>%
  arrange(desc(Passengers))

top_airport_depart <- head(top_airport_depart, n = 5)
top_airport_depart <- top_airport_depart %>%
  separate(ORIGIN_CITY_NAME, c("City", "State"), sep = ",") %>%
  mutate(Origin_Airport = paste0(ORIGIN,"(", City, ")" )) %>%
  select(Origin_Airport, Passengers)

p <- ggplot(data = top_airport_depart,
          aes(x = reorder(Origin_Airport,Passengers), Passengers)) +
  geom_bar(stat = "identity", width = 0.5, fill = "#FF6666") +
  scale_y_continuous(limits = c(0,50000000)) +
  coord_flip()

p + labs(title = "Top 5 busiest airports for departure for 2019",
        y = "Passengers", x = "Airports") + theme(plot.title = element_text(
          size = 15,face = "bold", hjust = 0.5, vjust = 0.3))
```



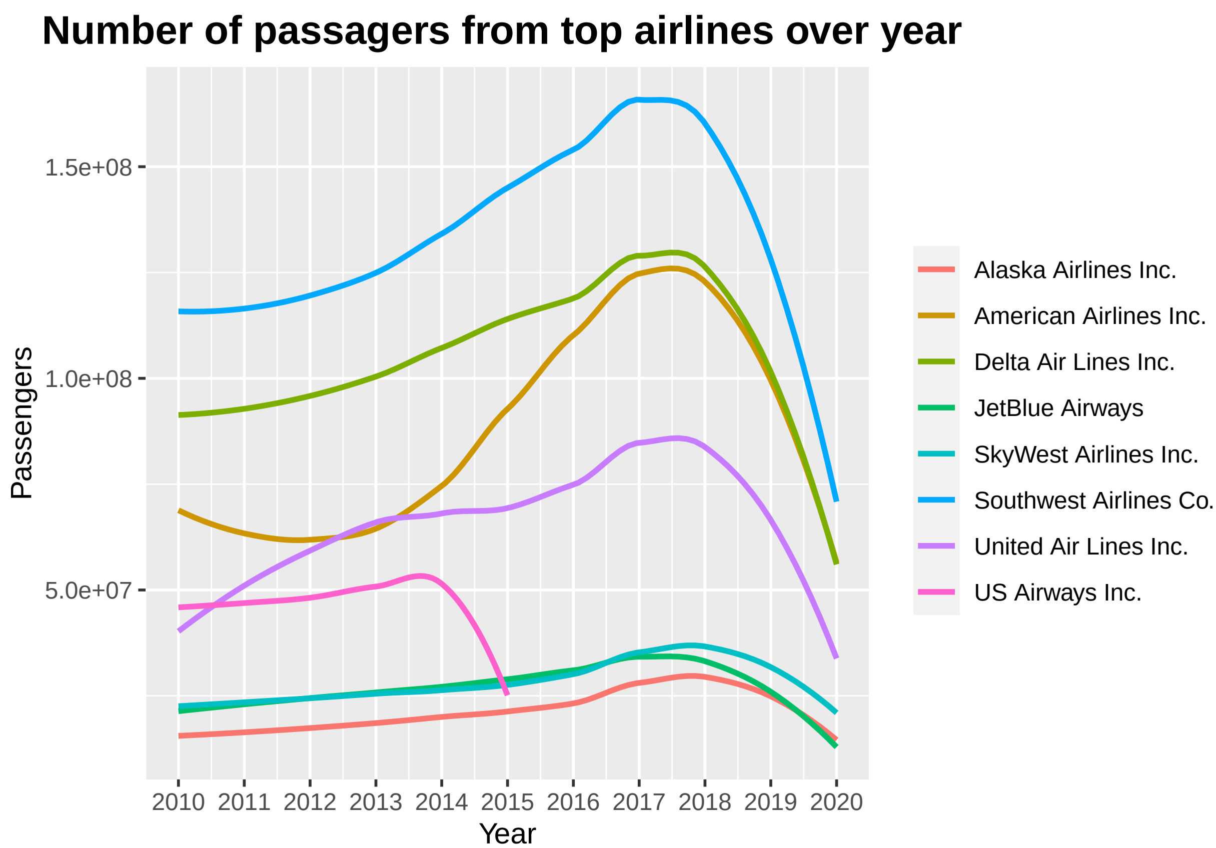
2) What are the top 5 busiest airports for arrival in 2019?

```
top_airport_arr <- US_carrier %>%
  filter(YEAR == "2019") %>%
  select(DEST, DEST_CITY_NAME, PASSENGERS) %>%
  group_by(DEST, DEST_CITY_NAME) %>%
  summarise(Passengers = sum(PASSENGERS)) %>%
  arrange(desc(Passengers))

top_airport_arr <- head(top_airport_arr, n = 5)

top_airport_arr <- top_airport_arr %>%
  separate(DEST_CITY_NAME, c("City","State"), sep = ",") %>%
  mutate(Dest_airport = paste0(DEST, "(", City, ")"))


q <- ggplot(data = top_airport_arr, aes(x = reorder(Dest_airport, Passengers),
                                   y = Passengers)) +
  geom_bar(stat = "identity", width = 0.5, fill = "#3390FF") +
  scale_y_continuous(limits = c(0, 50000000)) +
  coord_flip()

q + labs(title = "Top 5 busiest airports for arrival for 2019", x = "Airports") +
 theme(plot.title = element_text(size = 15, face = "bold", hjust = 0.5,
                                 vjust = 0.5))
```



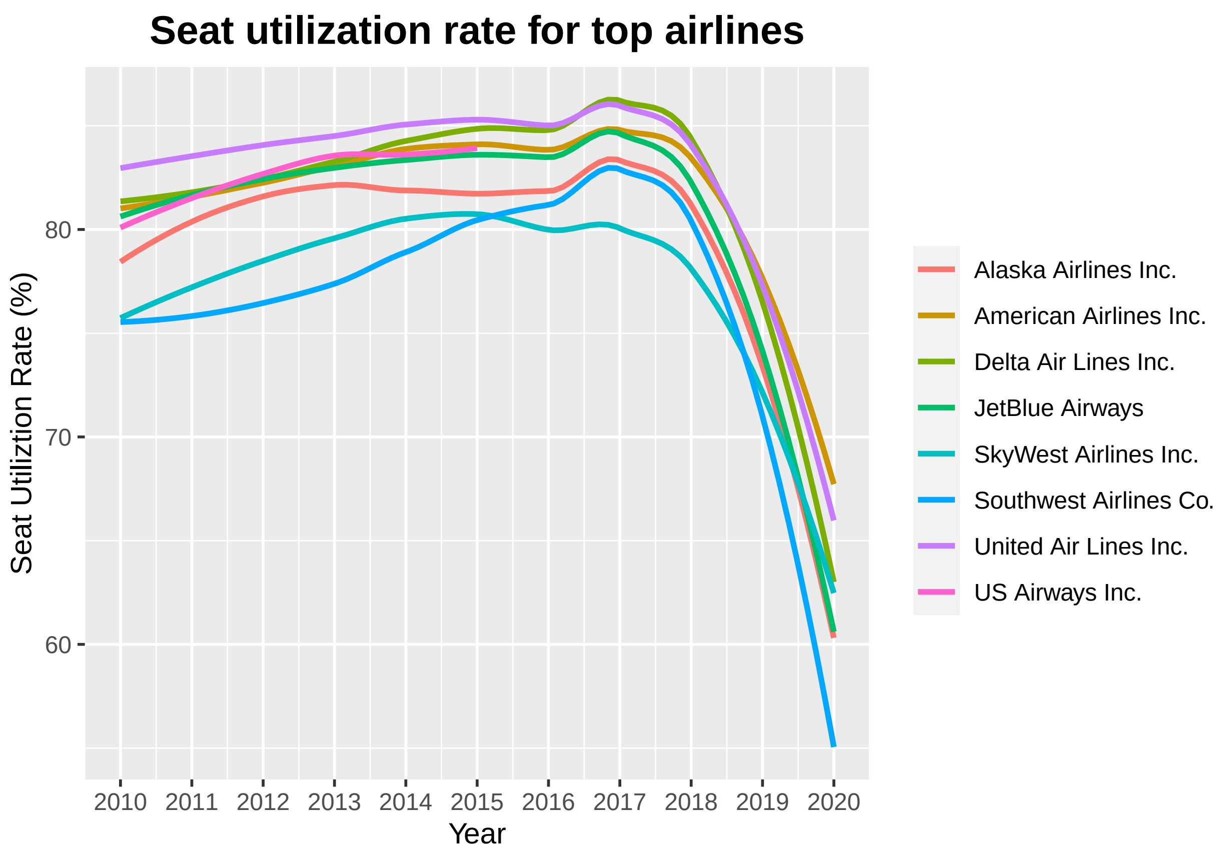
3) Who has the most passengers year by year?

```
top_airline_year <- US_carrier %>%
  select(PASSENGERS, UNIQUE_CARRIER, CARRIER_NAME, YEAR) %>%
  group_by(YEAR, UNIQUE_CARRIER, CARRIER_NAME) %>%
  summarise(Passengers = sum(PASSENGERS)) %>%
  arrange(desc(Passengers), UNIQUE_CARRIER) %>%
  filter(UNIQUE_CARRIER %in% c("WN","DL","AA","UA","OO","US","AS","B6"))

m <- ggplot(data = top_airline_year) +
  scale_x_continuous(breaks = seq(2010, 2020, by = 1)) +
  geom_smooth(aes(x = YEAR, y = Passengers, color = CARRIER_NAME), se = FALSE)

m + labs(title = "Number of passagers from top airlines over year ",
         x = "Year") +
  theme(plot.title=(element_text(size = 15, face = "bold", hjust = 0.5)),
        legend.title = element_blank())
```



4) Who has the highest seat utilization rate among top airlines?

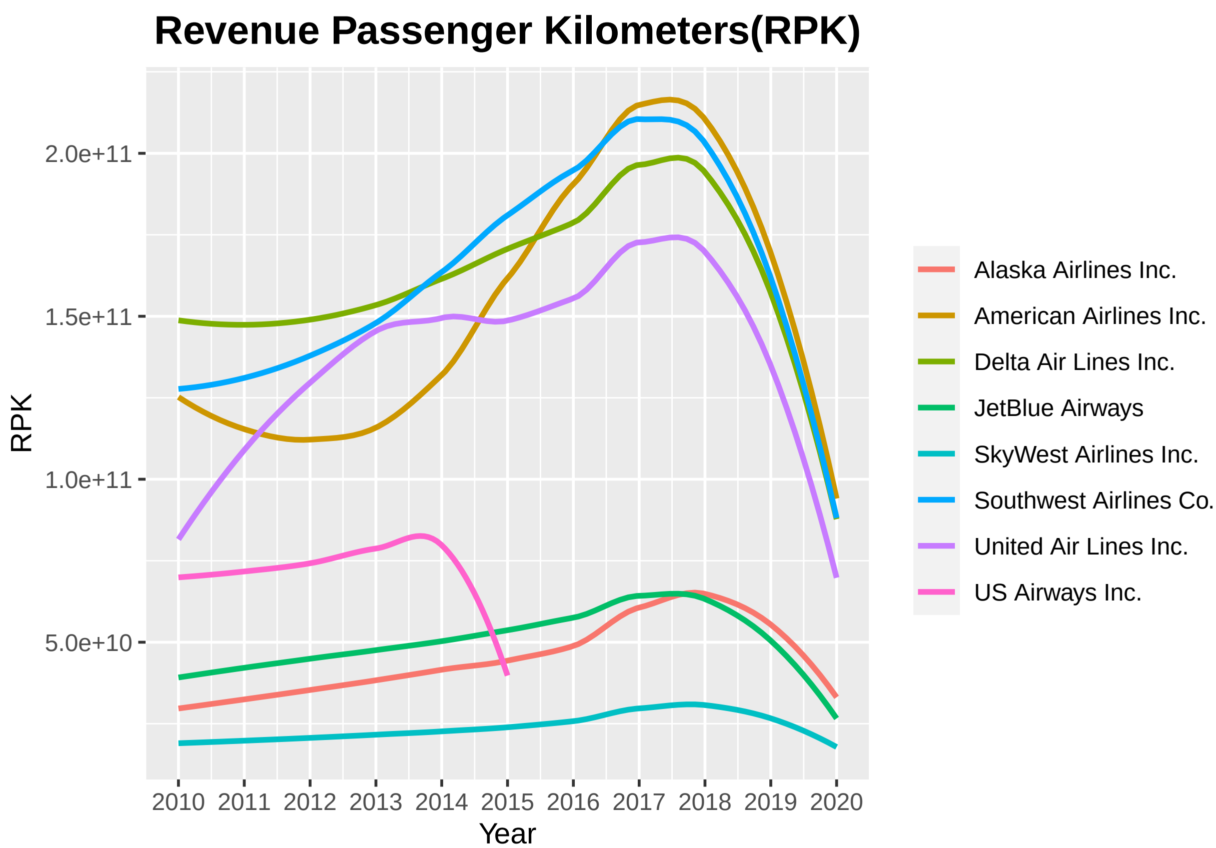
```
top_seat_uti <- US_carrier %>%
  mutate(EMPTY_SEATS = SEATS - PASSENGERS) %>%
  filter(SEATS > 0) %>%
  select(UNIQUE_CARRIER, CARRIER_NAME, EMPTY_SEATS, PASSENGERS, SEATS, YEAR) %>%
  group_by(YEAR, UNIQUE_CARRIER, CARRIER_NAME) %>%
  summarise(Total_empty_seats = sum(EMPTY_SEATS), Total_seats = sum(SEATS),
            Passengers = sum(PASSENGERS)) %>%
  mutate(Seat_utilization =
                  round(Passengers/Total_seats *100, 2)) %>%
  arrange(desc(YEAR), desc(Seat_utilization)) %>%
  select(YEAR, UNIQUE_CARRIER, CARRIER_NAME, Seat_utilization)

seats <- top_seat_uti %>%
  filter(UNIQUE_CARRIER %in% c("WN","DL","AA","UA","OO","US","AS","B6"))

l <- ggplot(data = seats) +
  geom_smooth(aes(x = YEAR, y = Seat_utilization,
                  color = CARRIER_NAME), se = FALSE) +
  scale_x_continuous(breaks = seq(2010, 2020, by = 1))

l + labs(title = "Seat utilization rate for top airlines", x = "Year",
         y = "Seat Utiliztion Rate (%)") +
  theme(plot.title = element_text(size = 15, face = "bold", hjust = 0.5),
        legend.title = element_blank())
```



5) Who has the highest RPK rate among top airlines?

RPK = the number of paying passengers x total distance traveled

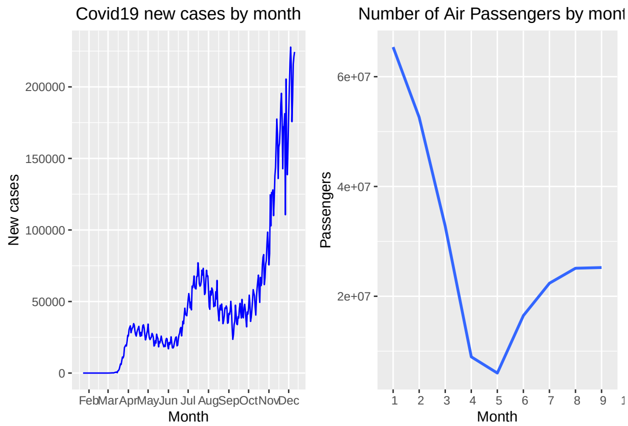
```
# calculate RPK per airline over year, the DISTANCE is in mile, needs to convert to kilometers
Total_RPK <- US_carrier %>% mutate(RPK = PASSENGERS * DISTANCE * 1.61) %>%
  select(YEAR, UNIQUE_CARRIER, CARRIER_NAME, RPK) %>%
  group_by(YEAR,UNIQUE_CARRIER, CARRIER_NAME) %>%
  summarise(TOTAL_RPK = sum(RPK)) %>%
  arrange(desc(TOTAL_RPK))

RPK <- Total_RPK %>%
  filter(UNIQUE_CARRIER %in% c("WN","DL","AA","UA","OO","US","AS","B6"))

a <- ggplot(RPK) +
  geom_smooth(aes(x = YEAR, y = TOTAL_RPK, color = CARRIER_NAME),se=F) +
  scale_x_continuous(breaks = seq(2010, 2020, by = 1))

a + labs(title = "Revenue Passenger Kilometers(RPK)", x = "Year", y = "RPK") +
  theme(plot.title = element_text(size = 15, hjust = 0.5, face = "bold"),
        legend.title = element_blank())
```



6) Does the trend of passengers amount have relation with Covid19 cases?

```
# import covid19 cases from ourworldindata.org
covid19 <- read.csv("covid-data.csv")

# filter just United States
covid19$date <- ymd(covid19$date, tz = Sys.timezone())
US_covid19 <- covid19 %>% filter(location == "United States") %>%
  select(date, new_cases) %>%
  mutate(Year = year(date), Month = month(date), Date = day(date))

# plot the new cases by month
covid_plot <- ggplot(data = US_covid19, aes(x = date, y =new_cases)) +
  geom_line(color = "blue") +
  scale_x_datetime(date_labels = "%b", date_breaks = "1 month") +
  ggtitle("Covid19 new cases by month") + xlab("Month") + ylab("New cases") +
  theme(plot.title = element_text(hjust = 0.5))

# plot the number of passengers in total in 2020
US_carrier$MONTH <- as.integer(US_carrier$MONTH)
pass_plot <- US_carrier %>% filter(YEAR == 2020) %>%
  select(MONTH, PASSENGERS) %>%
  arrange(MONTH) %>%
  group_by(MONTH) %>%
  summarise(Passengers = sum(PASSENGERS)) %>%
  ggplot(aes(x = MONTH, y = Passengers)) +
  geom_smooth(se = F) + scale_x_discrete(limits = seq(1, 12, by = 1)) +
```

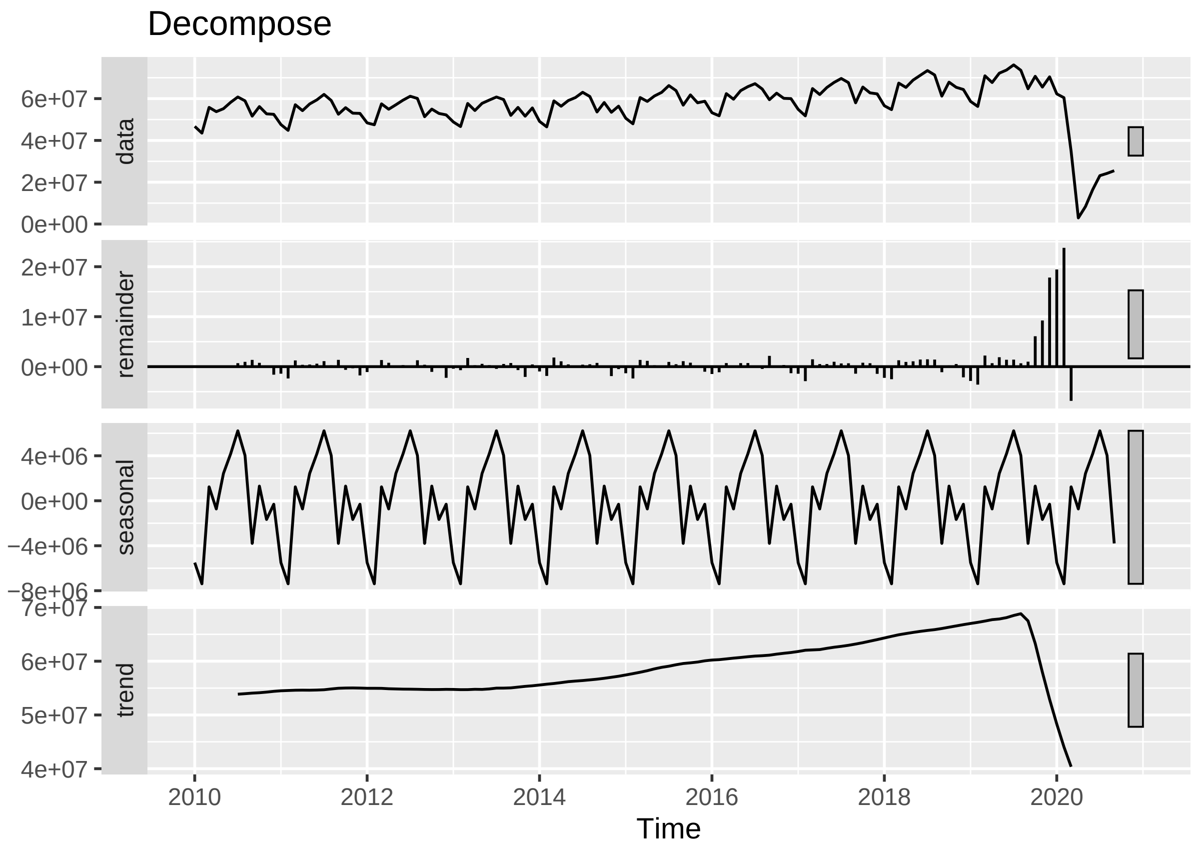
```r
  ggtitle("Number of Air Passengers by month") + xlab("Month") +
  theme(plot.title = element_text(hjust = 0.5))

ggarrange(covid_plot, pass_plot, ncol = 2, nrow = 1)
```



Forecast

Decomposition

```r
count_passenger <- US_carrier %>% select(PASSENGERS,YEAR,MONTH) %>%
  arrange(YEAR, MONTH) %>%
  group_by(YEAR, MONTH) %>%
  summarise(passengers = sum(PASSENGERS))

pass <- c(count_passenger$passengers)

# convert to time-series dataset in order to analyze it
passenger = ts(pass, start = 2010, frequency = 12)
fit_dec <- decompose(passenger)
autoplot(fit_dec) + ggtitle("Decompose")
```



It is very obviously that the dataset has seasonality and trend.

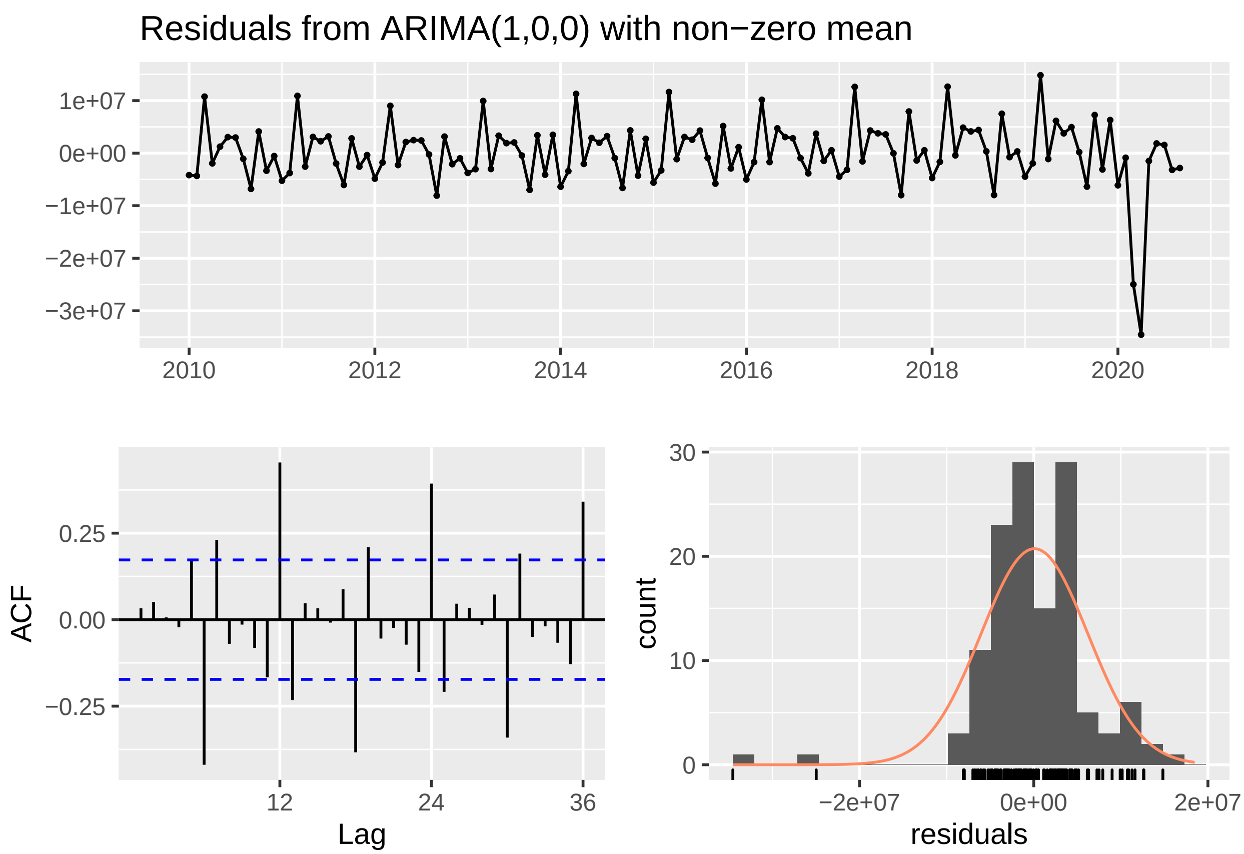
ARIMA

```
(fit.arima <- auto.arima(passenger))
```

auto.arima()

```
## Series: passenger
## ARIMA(1,0,0) with non-zero mean
##
## Coefficients:
##          ar1      mean
##       0.8667  55098143
## s.e.  0.0468   3736667
##
## sigma^2 estimated as 3.788e+13:  log likelihood=-2199.35
## AIC=4404.69   AICc=4404.88   BIC=4413.27
```

```
checkresiduals(fit.arima)
```

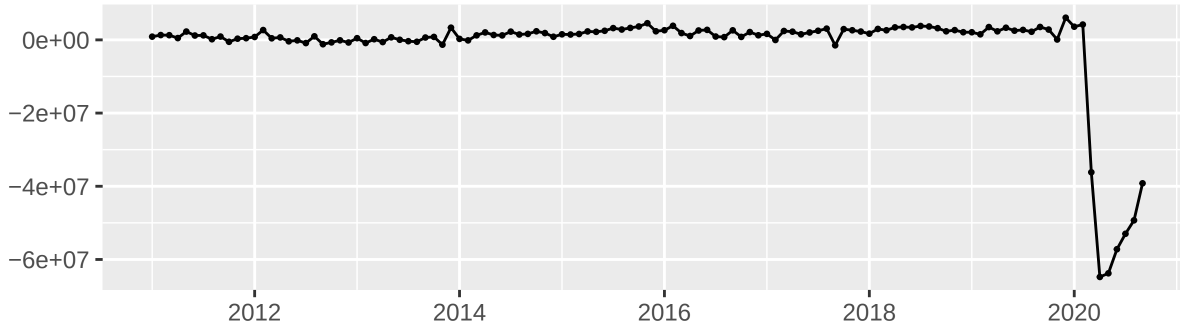


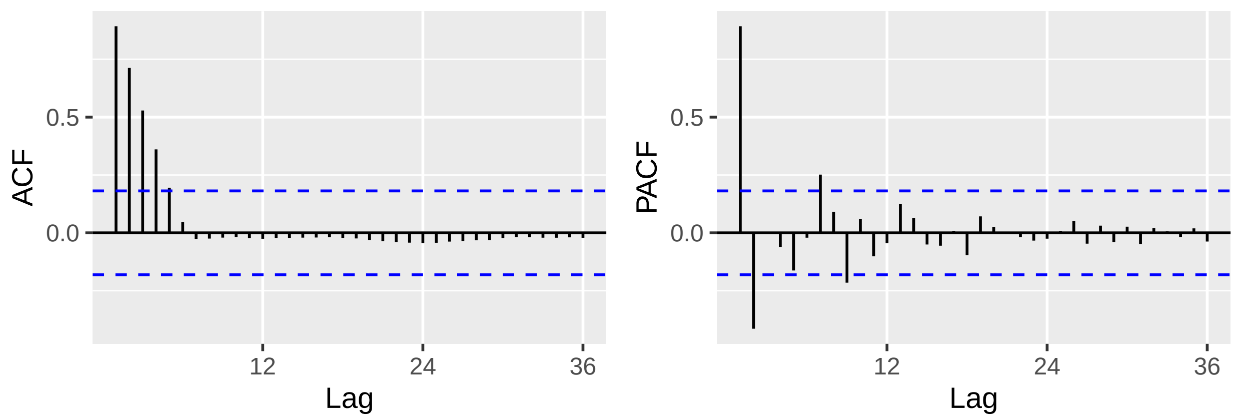

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,0,0) with non-zero mean
## Q* = 140.21, df = 22, p-value < 2.2e-16
##
## Model df: 2.   Total lags used: 24
```

The build-in auto ARIMA doesn't give a better performance from the ACF chart.

Manual ARIMA parameter selection

```
# manual ARIMA model parameter selection
diff_pas <- diff(passenger, lag = 12)
diff_pas %>% ggtsdisplay()
```





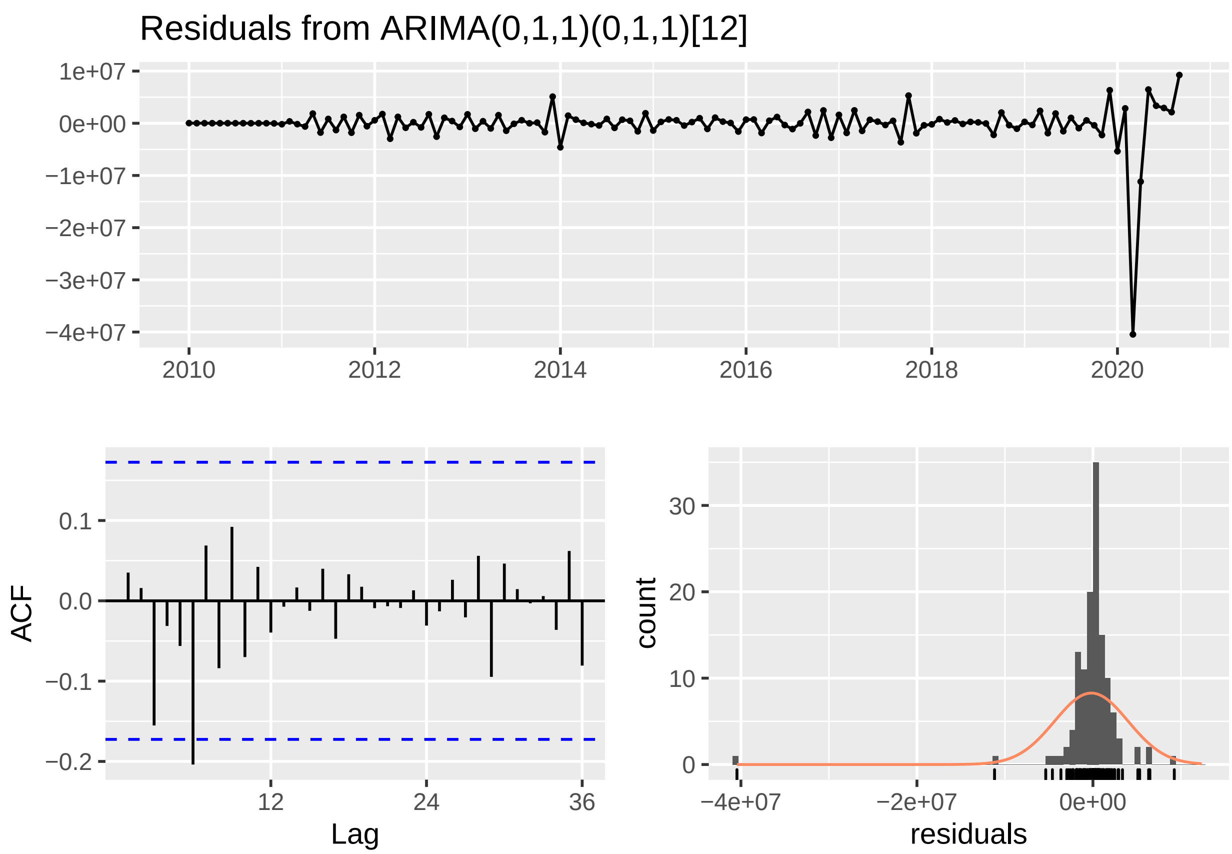
Differencing

```
(myarima <- Arima(passenger, order = c(0,1,1),seasonal = c(0,1,1)))
```

Parameter selection, choose the lowest AICc

```
## Series: passenger
## ARIMA(0,1,1)(0,1,1)[12]
##
## Coefficients:
##          ma1      sma1
##       0.4415   -0.5195
## s.e.  0.0792    0.2143
##
## sigma^2 estimated as 1.965e+13:  log likelihood=-1940.91
## AIC=3887.82   AICc=3888.04   BIC=3896.08
```

```r
checkresiduals(myarima)
```




```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,1)(0,1,1)[12]
## Q* = 14.782, df = 22, p-value = 0.8714
##
## Model df: 2.   Total lags used: 24
```

Compare models ARIMA vs. ETS

```r
mytrain <- window(passenger, start = 2010, end = 2017)
mytest <- window(passenger, start = 2018)

(myarima <- Arima(mytrain, order = c(0,1,1),seasonal = c(0,1,1)))
```

```
## Series: mytrain
## ARIMA(0,1,1)(0,1,1)[12]
##
## Coefficients:
##           ma1     sma1
```

```
##       -0.6396  -0.8494
## s.e.   0.0779   0.4405
##
## sigma^2 estimated as 6.943e+11:  log likelihood=-1090
## AIC=2185.99   AICc=2186.34   BIC=2192.82
```

```r
(fit.ets <- hw(mytrain, seasonal = "additive", damped = TRUE))
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Feb 2017       53172173 52142348 54201998 51597192 54747155
## Mar 2017       64693714 63618100 65769327 63048704 66338723
## Apr 2017       62070241 60941127 63199355 60343410 63797071
## May 2017       65125480 63935490 66315470 63305547 66945413
## Jun 2017       66938046 65680254 68195837 65014420 68861672
## Jul 2017       69180834 67848831 70512837 67143711 71217957
## Aug 2017       67360744 65948654 68772835 65201139 69520350
## Sep 2017       59931910 58434386 61429435 57641644 62222177
## Oct 2017       64060352 62472549 65648155 61632017 66488688
## Nov 2017       60613526 58931065 62295986 58040425 63186627
## Dec 2017       61681262 59900193 63462331 58957352 64405172
## Jan 2018       56557482 54674239 58440726 53677310 59437654
## Feb 2018       54498055 52509385 56486726 51456646 57539464
## Mar 2018       65993074 63896103 68090046 62786033 69200115
## Apr 2018       63343610 61135707 65551514 59966914 66720307
## May 2018       66373379 64052162 68694596 62823384 69923374
## Jun 2018       68160983 65724293 70597673 64434387 71887579
## Jul 2018       70379309 67825187 72933431 66473117 74285502
## Aug 2018       68535246 65861914 71208578 64446738 72623755
## Sep 2018       61082919 58288762 63877077 56809624 65356214
## Oct 2018       65188337 62271886 68104788 60728010 69648664
## Nov 2018       61718948 58678869 64759027 57069549 66368347
## Dec 2018       62764573 59599654 65929491 57924248 67604898
## Jan 2019       57619123 54328265 60909982 52586189 62652058
```

```r
fct_ets <- forecast(fit.ets, h = 24) %>% accuracy(passenger)
fct_arima <- forecast(myarima, h = 24) %>% accuracy(passenger)


print("HW ETS model")
```

```
## [1] "HW ETS model"
```

```r
fct_ets
```

```
##                     ME      RMSE       MAE       MPE     MAPE      MASE
## Training set   66744.4  718740.9  581216.9 0.1037647 1.046750 0.4108188
## Test set     1115283.9 1787882.3 1405316.6 1.6134316 2.137016 0.9933133
##                     ACF1 Theil's U
## Training set -0.04190049        NA
## Test set      0.45369848 0.2750317
```

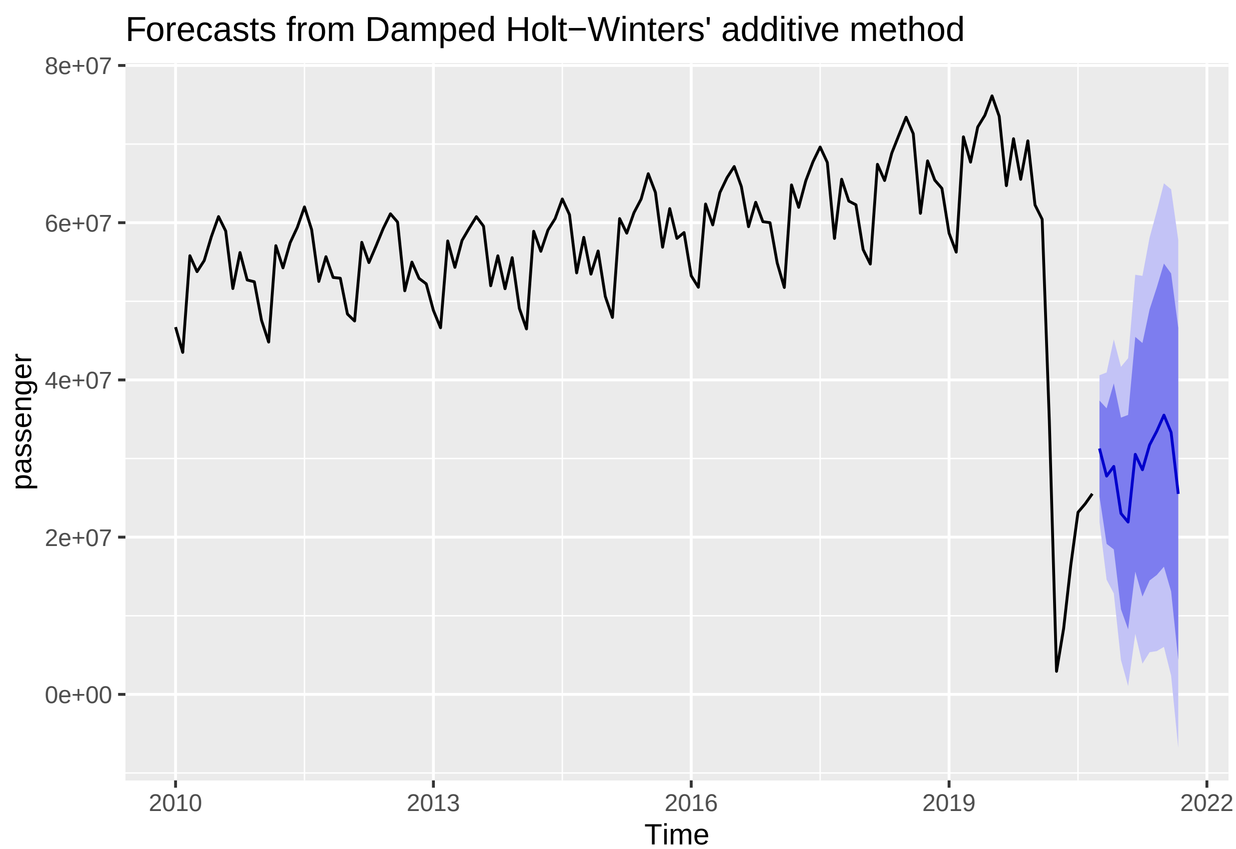
```r
print("ARIMA model")
```

```
## [1] "ARIMA model"
```

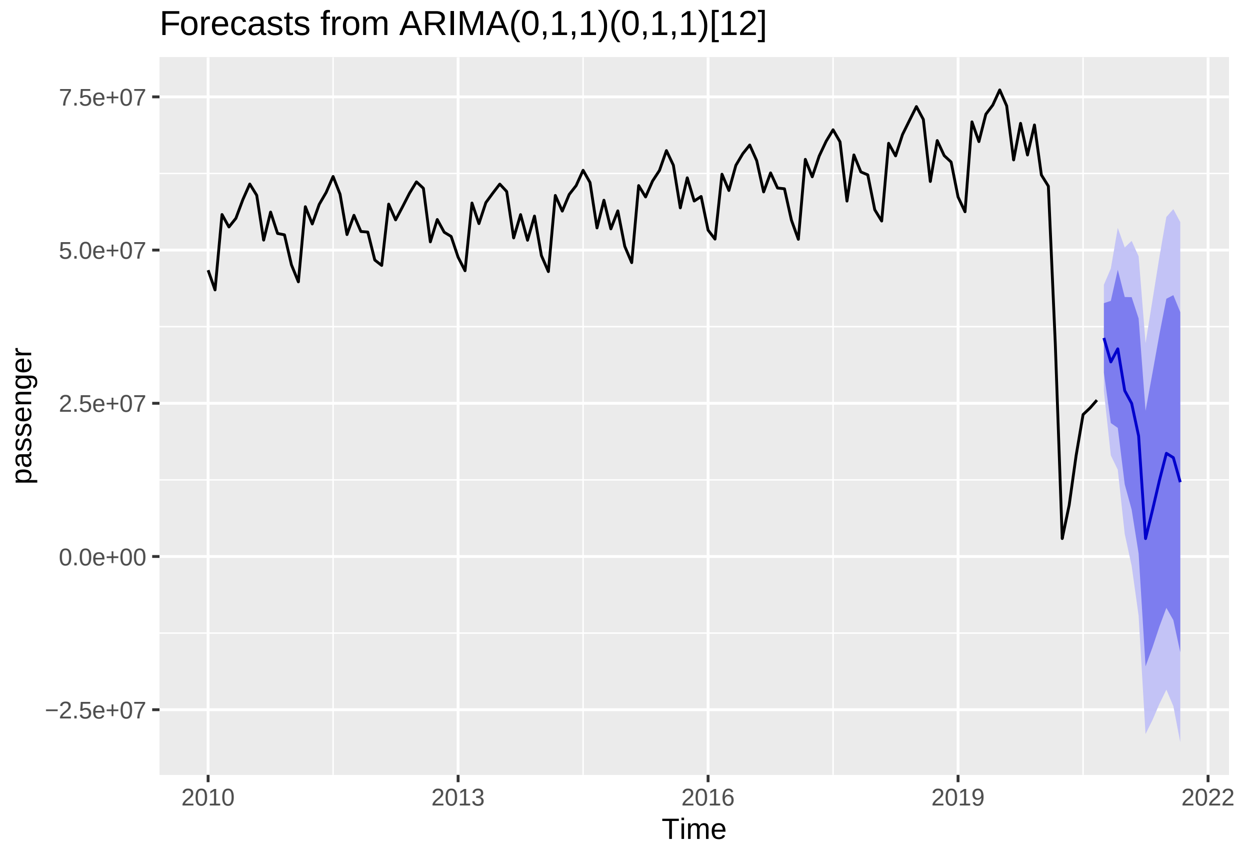
```r
fct_arima
```

```
##                        ME      RMSE       MAE       MPE      MAPE      MASE
## Training set   66915.93  756164.7  539701.5 0.1152882 0.9531873 0.3814747
## Test set     1201740.17 1814867.2 1475603.9 1.7482451 2.2395563 1.0429942
##                    ACF1 Theil's U
## Training set -0.1011249        NA
## Test set      0.3724236 0.2807994
```

```r
##Comparison plot for two models

forecast_ets <- forecast(hw(passenger,damped = TRUE,seasonal = "additive"), h = 12)
forecast_arima <- forecast(Arima(passenger, order = c(0,1,1),seasonal = c(0,1,1)),
                           h = 12)

autoplot(forecast_ets)
```




```r
autoplot(forecast_arima)
```



ETS and ARIMA model gives us a quite different forecast plot. ARIMA model shows a drop in 2021 basically because of the previous drop pattern in 2020. The pandemic in 2020 is an unusual event, which puts uncertainty and uncontrolled in any time series models. We definitely can't rely on the forecast based on that event, but we can take that as a reference for our future decision-making.