Air Freight Demand Prediction - Robert Wieckowski

Robert Wieckowski

08/03/2022

Contents

Introduction

In my project I will be analyzing data provided by US Bureau of Transportation Statistics. I ill focus on their data related to movement of freight across USA, and with deep analysis I will try to prove which indicators allow for better predictions of possible volume airlines needs to face.

##Data Overview Data for this project is available at webiste https://www.transtats.bts.gov/DL_SelectFields.aspx?gnoyr_VQ=FIL&QO_fu146_anzr=Nv4%20Pn44vr45 , and will small cleanup is ready to be used. Files are containing information about passengers, freight and mail volume for each airport pairs thye did operate within USA from 2010 until 2019. Raw files for analysis are available in repo https://github.com/RobertWieckowski/R

Data preparation

As we do have our data split per year, we need to merge it all into one dataset. Further analysis revealed that there might be missing records, however around 300 missing lines, comparing to roughly 2 millions records should not have influence over final result. Method used for cleanup can be observed below.

```
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(dplyr)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")

library(tidyverse)
library(caret)
library(dplyr)

#  Source of data
#  https://www.transtats.bts.gov/DL_SelectFields.asp?gnoyr_VQ=FIL&QO_fu146_anzr=Nv4%20Pn44vr45

# Files avilable in Github repo
# https://github.com/RobertWieckowski/R



#  Unzip Flights data and Passenger data and save it in single data frame
  raw_data <-
            list.files(pattern = "*.zip") %>%
            map_df(~read_csv(.))

#  Find NA values
 sapply(raw_data, function(x) sum(is.na(x)))

#  Remove column 37 which is entirely NA ans find NA again
  raw_data = select(raw_data,-37)
  sapply(raw_data, function(x) sum(is.na(x)))

# After one more check we have got 358 rows at most with NA data, so we can remove it as well.
  raw_data <- raw_data[rowSums(is.na(raw_data)) == 0,]

  raw_data = raw_data %>%
  group_by(AIRLINE_ID) %>%
  mutate(total_for_airline = sum(FREIGHT)) %>%
  filter(total_for_airline > 0 )
```

Insight

We will now examine our data and try to get insight into US airline market. As focus of our project is directed on freight market, we will get biggest freight airlines operating at it. In below top 10 airlines list parameter X represents volume of US market shared by specific airline in total between 2010-2019.

```
##                         Airline         x
## 1  Federal Express Corporation 52.058606
## 2        United Parcel Service 28.839598
## 3                Atlas Air Inc.  4.055272
## 4                   ABX Air Inc  2.152429
## 5      Polar Air Cargo Airways  1.715740
## 6  Air Transport International  1.334875
## 7          Delta Air Lines Inc.  1.217482
## 8       Southwest Airlines Co.  1.174474
## 9        United Air Lines Inc.  1.106679
## 10             Kalitta Air LLC  1.033875
```

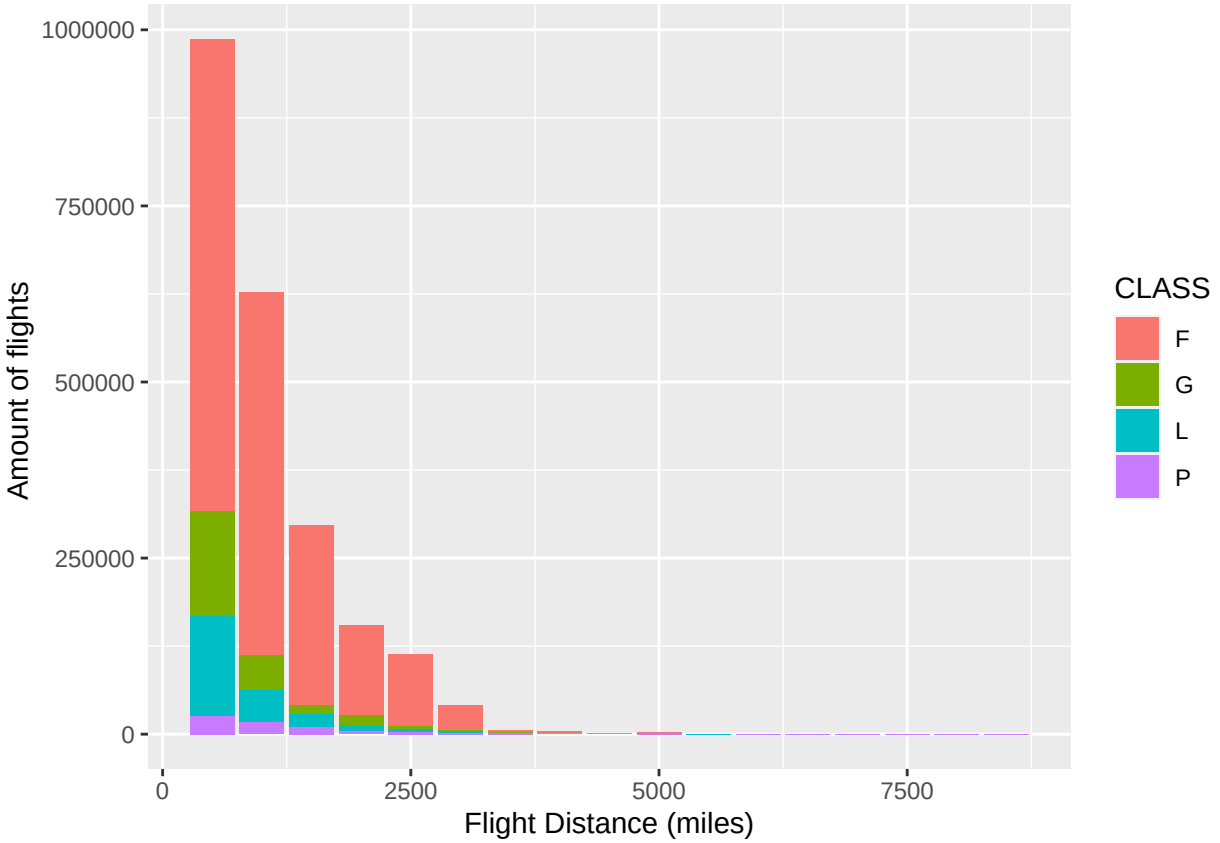
Not surprisingly FedEx and UPS are on top of the list. If you might be concerned that there is no DHL to be found, something which we might expact. Answer is simple, DHL airplanes are not flying under DHL name, they are subcontracted from independent airlines, providing service to DHL. Differently to FedEx and UPS, which are managing their own fleet.

We can also check what are most poplar US routes.

```
## # A tibble: 30 x 3
## # Groups:   ORIGIN_CITY_NAME [15]
##    ORIGIN_CITY_NAME DEST_CITY_NAME  Count
##    <chr>            <chr>           <int>
##  1 Minneapolis, MN  Chicago, IL      1726
##  2 Chicago, IL      Minneapolis, MN  1706
##  3 Detroit, MI      Chicago, IL      1544
##  4 Chicago, IL      Detroit, MI      1493
##  5 Washington, DC   New York, NY     1445
##  6 New York, NY     Washington, DC   1395
##  7 Chicago, IL      Atlanta, GA      1311
##  8 Chicago, IL      New York, NY     1295
##  9 Atlanta, GA      Chicago, IL      1292
## 10 Detroit, MI      Washington, DC   1267
## # ... with 20 more rows
```

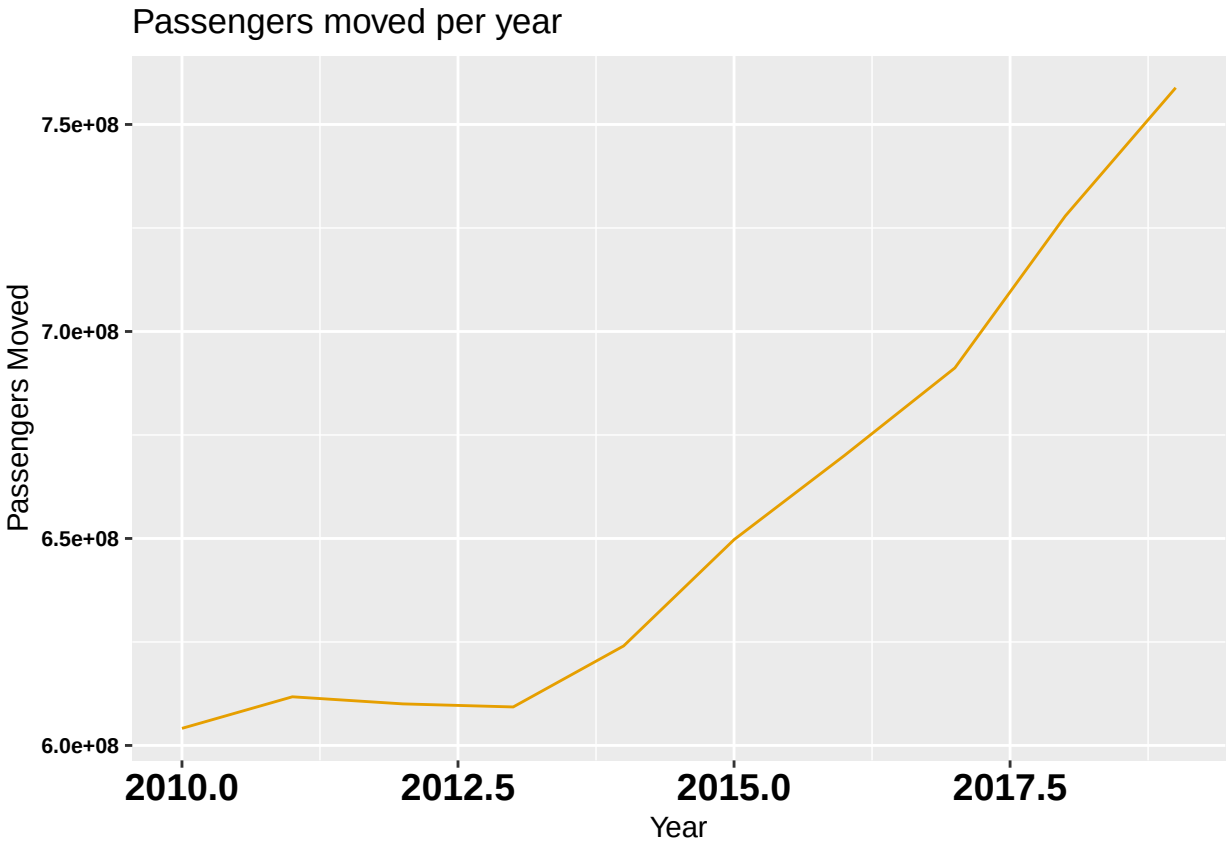
Answer meets our predictions, where we see that most popular routes are between biggest airlines hubs.

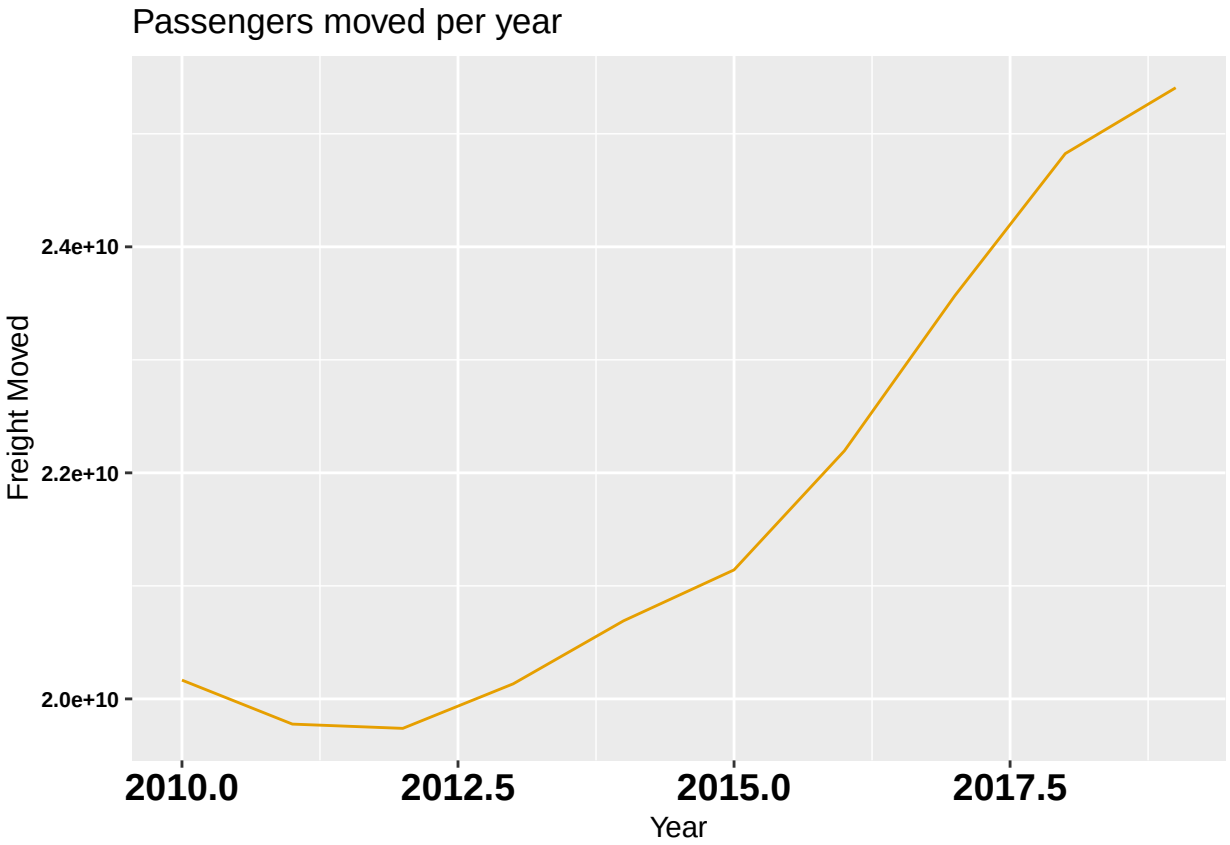
We can also check what is the relationship between flight distance and number of flights.

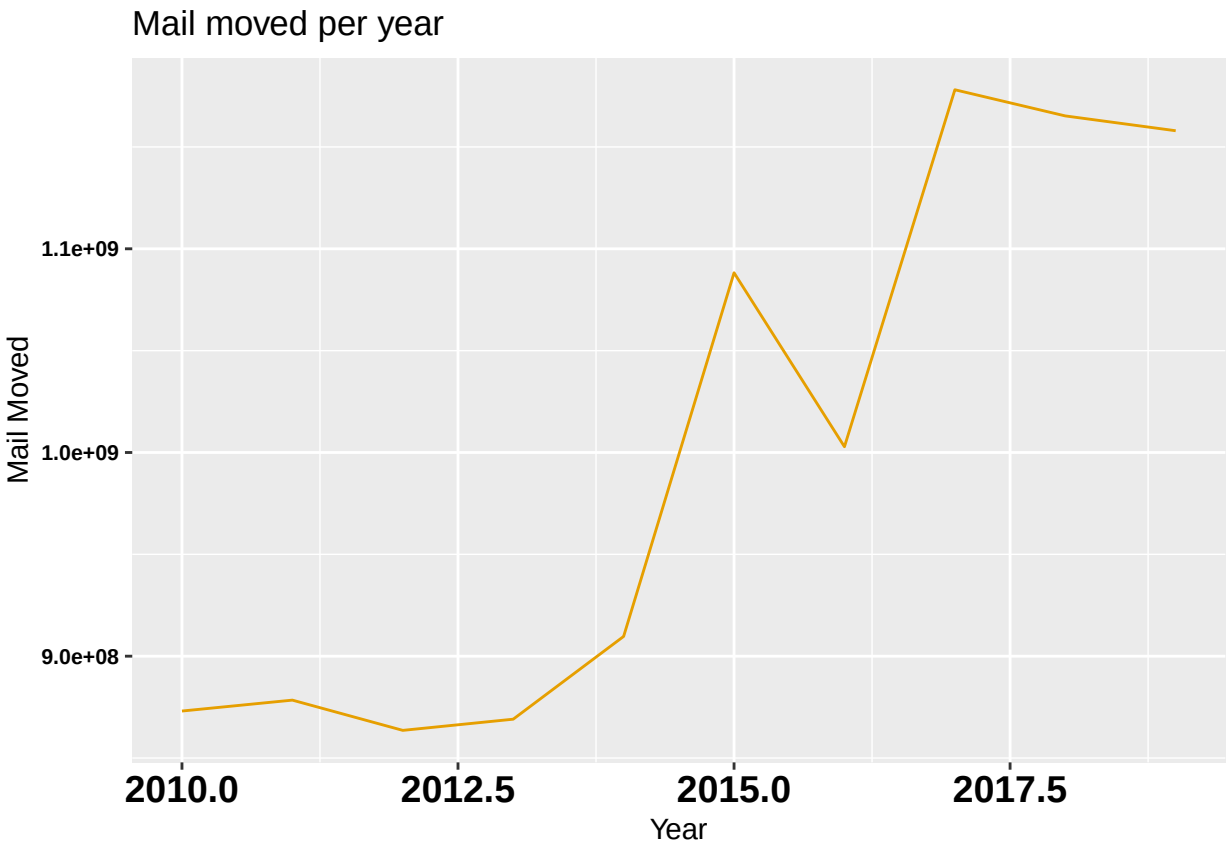


Not surprisingly number of flights is decreasing with the distance. On a longer routes airlines are flying between the hubs/biggest airport, which planes having biggest capacity. Whether greatest amount of flights are between hubs and regional airports.

We can also check how passenger, freight and mail ,market developed between 2010-2019 with below charts.


Passengers moved per year
Passengers Moved
7.5e+08
7.0e+08
6.5e+08
6.0e+08
2010.0
2012.5
2015.0
2017.5
Year


Passengers moved per year
Freight Moved
2.4e+10
2.2e+10
2.0e+10
2010.0
2012.5
2015.0
2017.5
Year



We should notice that Y Axis is not starting from 0, meaning charts are showing growth from starting point - year 2010. I general we can see steady market increase, with only one bump for mail service having bumpy 2016 year.

Methods

Now with the insights into airline market we can focus on our task, which is developing algorithm predicting freight volume. To do it, we will split our data into test and training set. Test set will be 30 percent of whole data. Reason for such big chunk is fact that we will be comparing many airport pairs, and we would expect this factor might be very important in our analysis. Having 30 percent of data available for test will allow us to check algorithm with as many possible combinations as available. Split method can be found below.

We also need to verify our algorithm, and when Root Mean Square Error (RMSE) function might be our natural selection, I decided to use Scatter Index. It is very similar function described as RMSE divided by mean of whole dataset. It shows us what percentage of mean dataset value is our model.

Model development

First Model

Or first model will be as simple mean of freight volume transported between 2010 and 2019. Our calculation will be saved for further analysis at the end.

```
#First model - Mean
```

```
    #Calculation
    average_freight <- mean(train_set$FREIGHT)
    #Model test
    first_model <- SI(test_set$FREIGHT, average_freight)
    #Saving results
    si_results = tibble(method = "First Model - Mean", SI = first_model)
```

Second Model

Second Model is previous one with added airline as a factor in our calculations. Similar to above model result will be saved for further examination.

```
#Second model - Airline

    #Calculation
    average_per_airline <- train_set %>%
      group_by(AIRLINE_ID) %>%
      summarize(airline = mean(FREIGHT - average_freight))

    predicted <- test_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      mutate(pred = average_freight + airline   ) %>%
      .$pred
    #Model test
    second_model <- SI(test_set$FREIGHT, predicted)
    #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                              "Second Model - Added Airline", SI = second_model ))
```

Third Model

Third model is considering departure airport to previous one.

```
#Third Model - Origin

    #Calculation
    average_with_distance_group <- train_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      group_by(ORIGIN) %>%
      summarize( origin = mean(average_freight - airline))

    predicted <- test_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      mutate(pred = average_freight + airline + origin  ) %>%
      .$pred

    #Model test
    third_model <- SI(test_set$FREIGHT,predicted)
    #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                              "Third Model - Added Origin", SI = third_model ))
```

Fourth Model

Fourth model is considering destination.

```
#Fourth Model - Destination

    #Calculation
    average_with_class <- train_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group, by='ORIGIN') %>%
      group_by(DEST) %>%
      summarize( destination = mean(average_freight - airline -  origin))

    predicted <- test_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      mutate(pred = average_freight +airline + origin + destination  ) %>%
      .$pred
    #Model test
    fourth_model <- SI(test_set$FREIGHT,predicted)
    #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                        "Fourth Model - Added Destination", SI = fourth_model ))
```

Fifth Model

Fifth one evaluates using year when cargo was moved.

```
#Fifth Model - Year

    #Calculation
    average_with_year <- train_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      group_by(YEAR) %>%
      summarize( year = mean(average_freight - airline - origin - destination))

    predicted <- test_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      left_join(average_with_year , by='YEAR') %>%
      mutate(pred = average_freight + airline + origin + destination + year  ) %>%
      .$pred
    #Model test
    fifth_model <- SI(test_set$FREIGHT,predicted)
    #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                                  "Fifth Model - Added Year", SI = fifth_model ))
```

Sixth Model

Sixth one takes into consideration month flight happened, asuming volumes might differ depends on time it was moved.

```
# Sixth Model - Month

     #Calculation
     average_with_month <- train_set %>%
       left_join(average_per_airline, by='AIRLINE_ID') %>%
       left_join(average_with_distance_group , by='ORIGIN') %>%
       left_join(average_with_class , by='DEST') %>%
       left_join(average_with_year , by='YEAR') %>%
       group_by(MONTH) %>%
       summarize( month = mean(average_freight - airline - origin -
                                 destination- year))

     predicted <- test_set %>%
       left_join(average_per_airline, by='AIRLINE_ID') %>%
       left_join(average_with_distance_group , by='ORIGIN') %>%
       left_join(average_with_class , by='DEST') %>%
       left_join(average_with_year , by='YEAR') %>%
       left_join(average_with_month , by='MONTH') %>%
       mutate(pred = average_freight + airline + origin +
                destination + year-month  ) %>%
       .$pred
     #Model test
     sixth_model <- SI(test_set$FREIGHT,predicted)
     #Saving results
     si_results <- bind_rows(si_results,tibble(method=
                                  "Sixth Model - Added Month", SI = sixth_model ))
```

Seventh Model

Seventh model check whether group to which carrier is assigned makes an impact on volume they move.

```
# Seventh Model - Carrier Group

     #Calculation
     average_with_group <- train_set %>%
       left_join(average_per_airline, by='AIRLINE_ID') %>%
       left_join(average_with_distance_group , by='ORIGIN') %>%
       left_join(average_with_class , by='DEST') %>%
       left_join(average_with_year , by='YEAR') %>%
       left_join(average_with_month , by='MONTH') %>%
       group_by(CARRIER_GROUP) %>%
       summarize( group = mean(average_freight - airline - origin -
                                 destination- year - month))

     predicted <- test_set %>%
       left_join(average_per_airline, by='AIRLINE_ID') %>%
       left_join(average_with_distance_group , by='ORIGIN') %>%
       left_join(average_with_class , by='DEST') %>%
```

```
      left_join(average_with_year , by='YEAR') %>%
      left_join(average_with_month , by='MONTH') %>%
      left_join(average_with_group , by='CARRIER_GROUP') %>%
      mutate(pred = average_freight + airline + origin +
               destination + year+month+ group ) %>%
      .$pred
    #Model test
    seventh_model <- SI(test_set$FREIGHT,predicted)
    #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                   "Seventh Model - Added Carrier Group", SI = seventh_model ))
```

Eigth Model

Eights one predicts using distance at which flights take place. It is done by considering distance group already calculated by data provider.

```
# Eight Model - Distance Group

    #Calculation
    average_with_or_state <- train_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      left_join(average_with_year , by='YEAR') %>%
      left_join(average_with_month , by='MONTH') %>%
      left_join(average_with_group , by='CARRIER_GROUP') %>%
      group_by(DISTANCE_GROUP) %>%
      summarize( distance = mean(average_freight - airline - origin +
                                   destination- year - month-group))

    predicted <- test_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      left_join(average_with_year , by='YEAR') %>%
      left_join(average_with_month , by='MONTH') %>%
      left_join(average_with_group , by='CARRIER_GROUP') %>%
      left_join(average_with_or_state , by='DISTANCE_GROUP') %>%
      mutate(pred = average_freight + airline + origin +
               destination + year+month+ group+distance ) %>%
      .$pred
    #Model test
    eigth_model <- SI(test_set$FREIGHT,predicted)
    #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                        "Eigth Model - Added Distance Group", SI = eigth_model ))
```

Ninth Model

Ninth model compares class to which airline is assigned by data provider.

```r
# Ninth Model - Airline Class

    #Calculation
    average_with_dest_state <- train_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      left_join(average_with_year , by='YEAR') %>%
      left_join(average_with_month , by='MONTH') %>%
      left_join(average_with_group , by='CARRIER_GROUP') %>%
      left_join(average_with_or_state , by='DISTANCE_GROUP') %>%
      group_by(CARRIER_GROUP) %>%
      summarize( carrier_group = mean(average_freight - airline - origin +
                                        destination- year - month-group-distance))

    predicted <- test_set %>%
      left_join(average_per_airline, by='AIRLINE_ID') %>%
      left_join(average_with_distance_group , by='ORIGIN') %>%
      left_join(average_with_class , by='DEST') %>%
      left_join(average_with_year , by='YEAR') %>%
      left_join(average_with_month , by='MONTH') %>%
      left_join(average_with_group , by='CARRIER_GROUP') %>%
      left_join(average_with_or_state , by='DISTANCE_GROUP') %>%
      left_join(average_with_dest_state , by='CARRIER_GROUP') %>%
      mutate(pred = average_freight + airline + origin +
               destination + year+month+ group+distance+carrier_group ) %>%
      .$pred
   #Model test
    nineth_model <- SI(test_set$FREIGHT,predicted)
  #Saving results
    si_results <- bind_rows(si_results,tibble(method=
                     "Nineth Model - Added Airline Class", SI = nineth_model ))
```

Now when we have all models tested we can compare how successful they are, and if factors chosen for model provide success.

```
# Overall results
      options(pillar.sigfig=5)
      print(si_results)
```

```
## # A tibble: 9 x 2
##   method                                SI
##   <chr>                              <dbl>
## 1 First Model - Mean                 6.4473
## 2 Second Model - Added Airline       5.6985
## 3 Third Model - Added Origin         5.9829
## 4 Fourth Model - Added Destination   6.1461
## 5 Fifth Model - Added Year           6.1471
## 6 Sixth Model - Added Month          6.1474
## 7 Seventh Model - Added Carrier Group 6.1755
## 8 Eigth Model - Added Distance Group  6.1910
## 9 Nineth Model - Added Airline Class  6.2141
```

Conclusions

From final results we could see that second model considering only mean of a freight volume and airline was most successful. Other factors considered unfortunately created noise within algorithm, and made overfitting. Without examining data we could assume that origin or destination might be factor into volume predictions, but these two contributed greatly into disrupting it. Other factors also did not help in getting better results, however did not contributed as much into it.

Knowledge of a air freight market might able to get answer why. Each routes are served by specific aircraft types, with limited capacity. Meaning that if there would be higher demand, there might be not enough supply. Cargo would wait for space available on another aircraft.

Another factor might be the fact that majority of flight within USA are passenger flights with cargo being moved inside aicraft's belly. In this situation supply for freight is driven by passengers volume. With these flight cargo is not major revenue contributor, meaning passengers got priority. If there is low demand for passengers, but high for cargo, there will be no bigger plane provided to accommodate it. Cargo will wait, as this is usual rules when shipping freight. Airlines can delay arrival of it.

We might also consider freight carriers, and ask ourself why these airlines are not adjusting to floating cargo demand. Answer would be fact, that these are working within its own designed networks. They have got certain routes, and are not changing them so easily. In case when there would be high demand for cargo from New York to Seattle, freight will fly on usual lanes New York - Los Angeles - Seattle, and rest of it might fly New York - San Francisco - Seattle. There would be different middle points between final origin and destination.

Dataset is providing numbers for individual legs, however people, cargo and mail is flying more than one leg. There destination is not necessarily ending at their first destination airport. We cannot be sure what is actual origin and destination for each individual journey. If we would like to improve model, this would be data needed. Actual origin and destination, not only specific legs.