Name: FIT 5147 Visualization Project
Title: Flight Delay Analysis
Author: Yizheng Fang
Student number: 29407699
Tutor: Debbie
Tutorial: Friday 12pm

Introduction

The flight delay is becoming a more and more common phenomenon to the traveller as well as to airlines companies. However, travellers spend the amount of their time on waiting for delayed or cancelled flights and result of wasting their time and may delay their schedules even miss the next flight. The main purpose of this project is to help our audience, frequent travellers who travel in domestic of United State and use air flight as their main traffic tools, to avoid possible delays or cancellations during the journey. The Project will provide general ideas using visualizations to avoid flight delays or cancellations by selecting airlines, optimal seasons, and suitable locations. Besides, same flight data is used as previous exploration project, as long as the project is maturely developed, there may be more data included.

Design

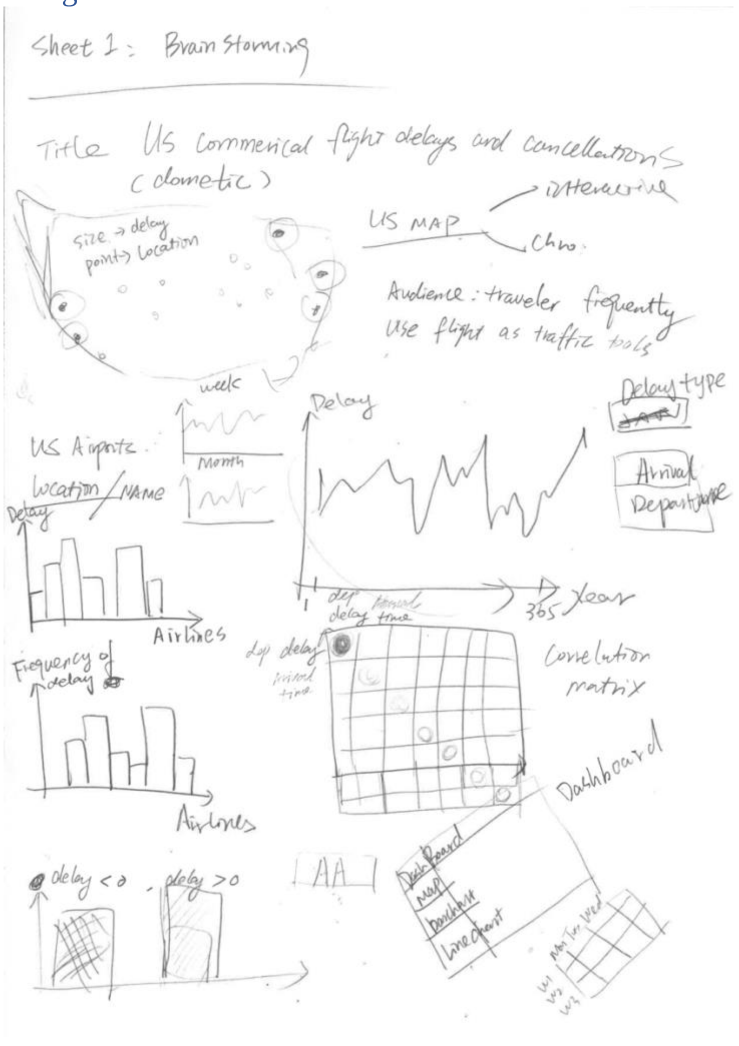


Figure 1: The original stage of ideas is generated by the previous project and considered becoming more interactive. The plots are assumed to be done in shiny, which is an interactive package in R to build web apps. The project should include various kinds of plots, such as bar chart, to compare airline companies; line chart, to show the seasonality may exist during the whole year. Besides, a dashboard layout should be used to navigate different plots. Hence, some of the analytical plots like correlation matrix may not give our audience explicit ideas and may not be included in the visualizations.

In addition, in order to make our audience more impressive, map plots are generated in both interactive and static but with flight paths on the map. Each airport becomes a point and when the user hovers their mouse, there will be details about delay present. The flight route path can be created by connecting points and great circles on the map. Users can choose different airports to show the flight path on the map by a search bar on the side of the map.

Moreover, there are line charts with the axis that can be controlled. The user can choose the period to present and see the predicted delay in another chart. Moreover, the different delay types such as departure and arrival delays can be compared in the same plot.

Last but not least, the bubble chart can be shown as it can have multiple variables within one plot and the relationship between data is also shown. The legend of colour and size is easy to use in the plot.

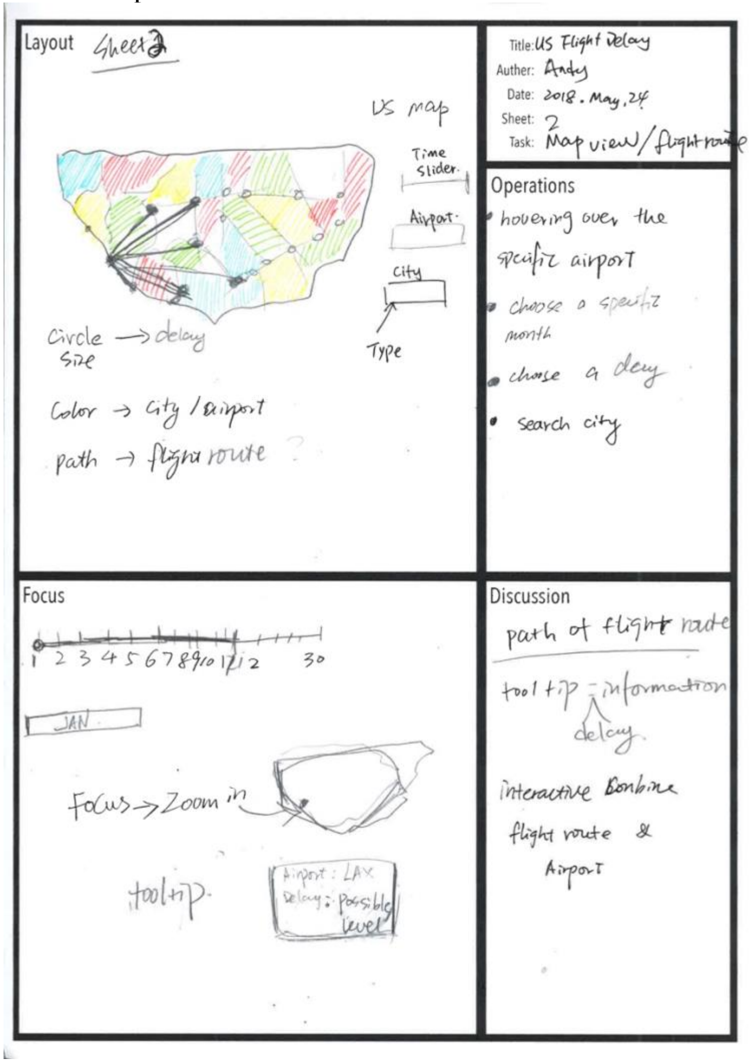
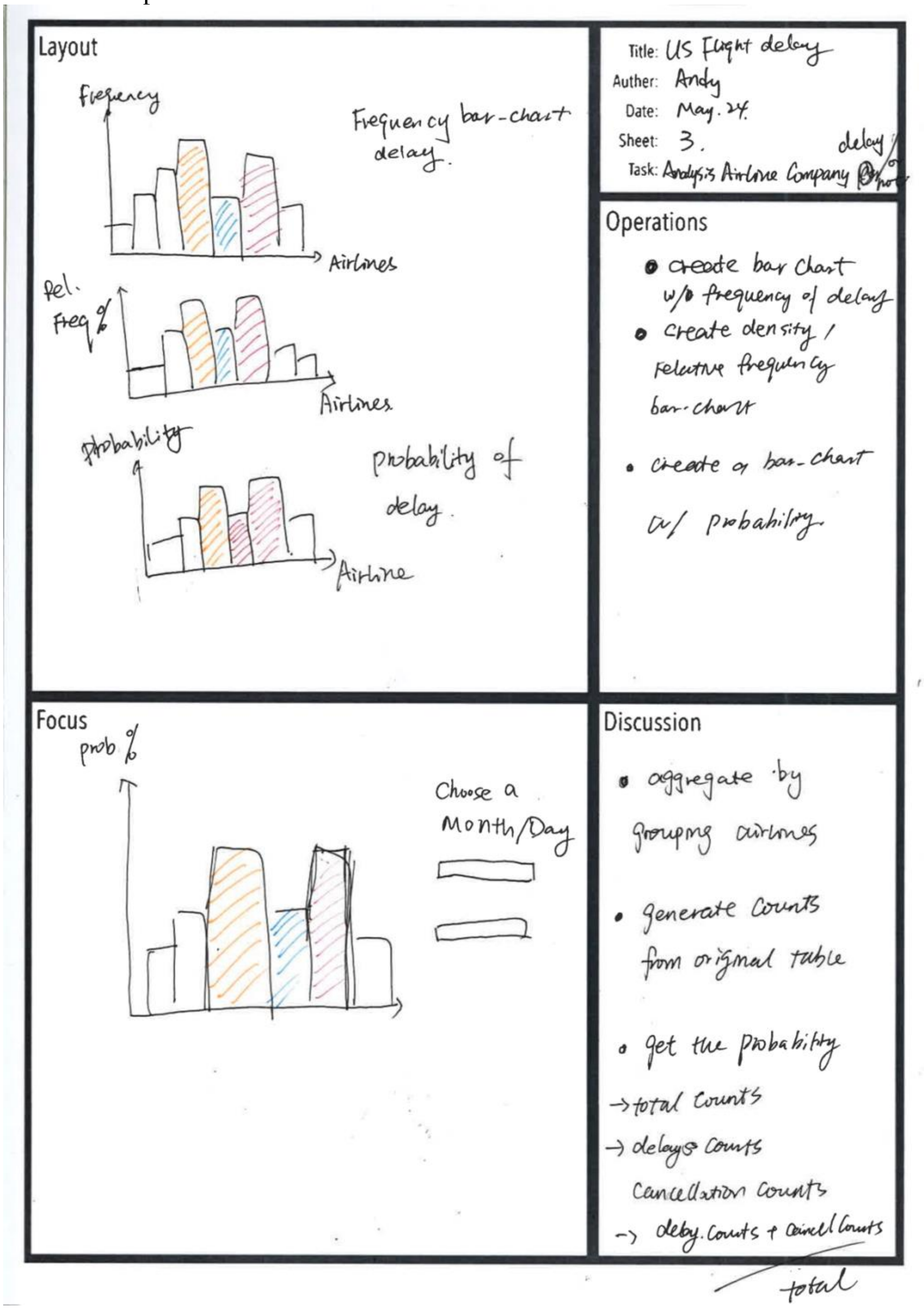
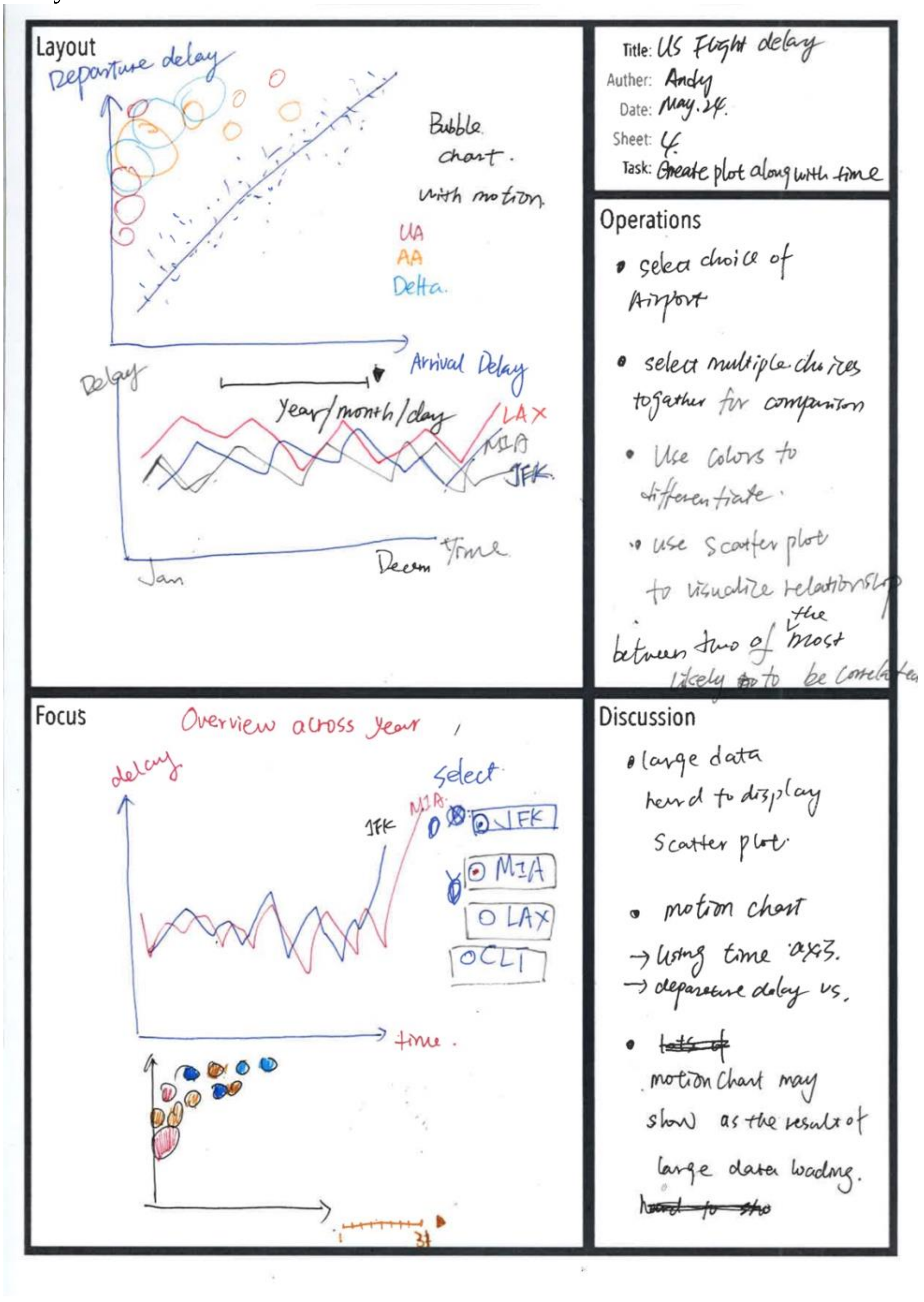


Figure 2: The map plots may have different types to present and should be used in different ways. There will be one interactive map plot built by leaflet package, and with useful labels and marks, users can be hovering on the map to see the delay details in specific location; Another map is a static map but shows flight path, so users are able to select the code of airport in a selection bar to see the flight route on the map. The map is can be built in different r map packages but depending on the type of data, we need to select them carefully as some of the functions are not easy to use. There may be a search bar for users to select the city they preferred. The map plot will use geographic data hence missing necessary data may cause some problems.



Layout
Frequency
Frequency bar-chart delay.
Airlines
Rel. Freq %
Airlines
probability
probability of delay.
Airline
Title: US Flight delay
Auther: Andy
Date: May. 24.
Sheet: 3.
Task: Analysis Airline Company delay/
Operations
create bar chart w/o frequency of delay
create density / relative frequency bar-chart
create a bar-chart w/ probability.
Focus
prob %
Choose a Month/Day
Discussion
aggregate by grouping airlines
generate counts from original table
get the probability
→ total counts
→ delays counts
cancellation counts
→ delay.counts + cancel counts
total

Figure 3: The bar chart is a commonly used plot in visualization, but the static one is not attractive, so we need to make the bar chart more fancy and interactive. The bar chart is built by comparing airlines in colours, different months on the x-axis, and the y-axis shows the frequency which is the number of delays in each month. The plot has advantages that it is easy to compare categorical data, but it cannot show the information about non-tabular data in this project. The frequency bar chart can show the total delay times and a bar chart with probabilities may provide users with a hint which airline may be more likely to have flight delays.



Title: US Flight delay
Auther: Andy
Date: May. 24.
Sheet: 4.
Task: Create plot along with time

Operations

- select choice of Airport
- select multiple choices togather for comparison
- Use colors to differentiate.
- use scatter plot to visualize relationship between two of the most likely to to be correlated

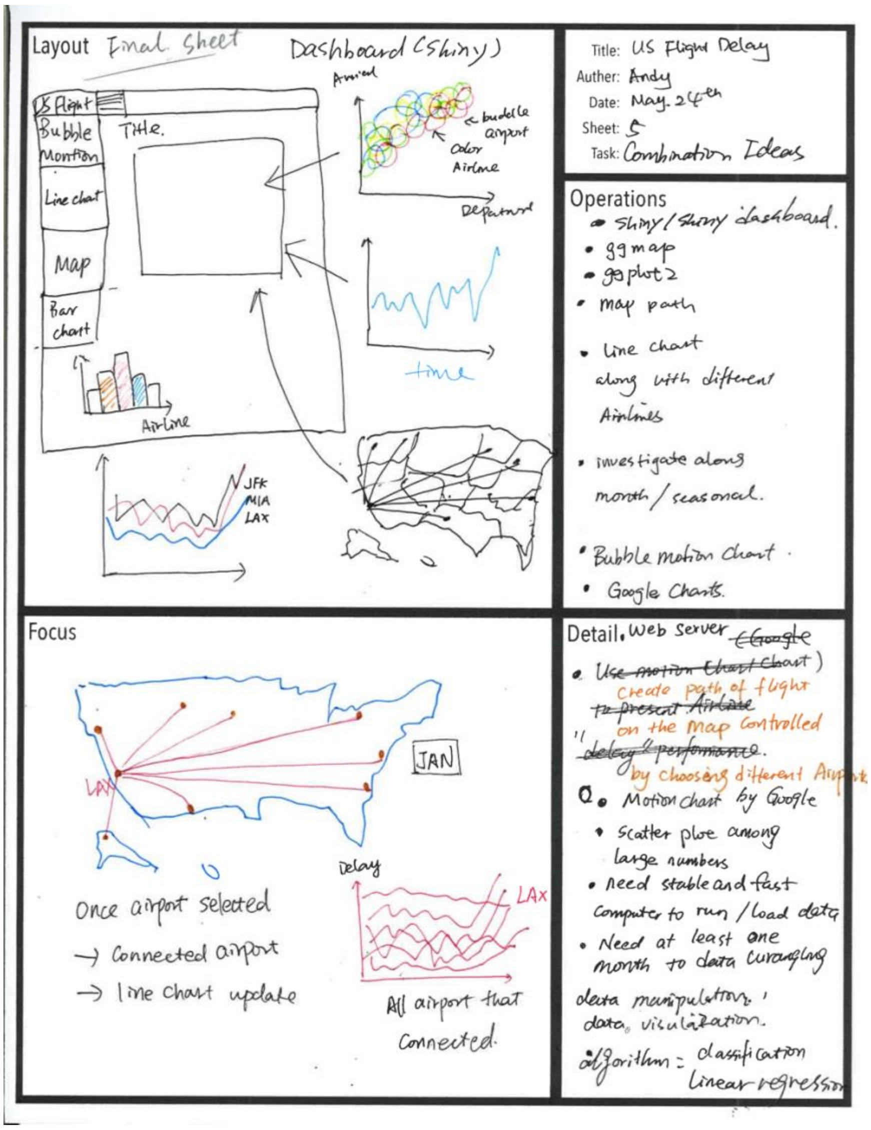
Discussion

- large data hard to display scatter plot.
- motion chart
  → using time axis.
  → departure delay vs.
- ~~lots of~~ motion chart may slow as the result of large data loading.
  ~~hard to sho~~

Figure 4: The line chart is always a good idea to show a trend in time and time series data performance. In this project, we use a line plot that has interactive features, so users can

control the range of time to present. The comparison of airlines using line chart may not give better insights.

Also, a bubble chart is useful to show the relationships by adding more factors with colour and size of bubbles. The x and y-axis are the predictor variables that we are interested in, it is easy to see whether there is a linear trend in between. Besides, a slider control using time data, so it is becoming a motion chart and bubbles will changes over time automatically as click start button. However, as the result of the data is too large and too complicated, it is not easy to show the whole five million data points at once. The big dataset will make the app extremely slow. Therefore, I may not use bubble chart in my final version but will try it further.



Title: US Flight Delay
Auther: Andy
Date: May. 24th
Sheet: 5
Task: Combination Ideas

Operations
- shiny / shiny dashboard.
- ggmap
- ggplot2
- map path
- line chart along with different Airlines
- investigate along month / seasonal.
- Bubble motion chart.
- Google Charts.

Detail. Web server ~~(Google~~
- Use ~~motion Chart Chart~~) create path of flight ~~to present Airline~~ on the map controlled " ~~delay " performance~~. by choosing different Airport
- Motion chart by Google
  - scatter plot among large numbers
- need stable and fast computer to run / load data
- Need at least one month to data curangling

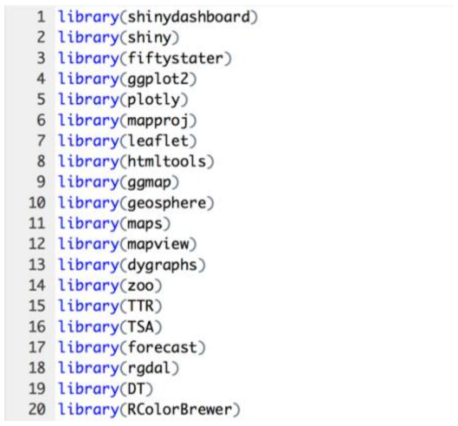
data manipulation, data visulization.

algorithm: classification
linear regression

Figure 5: Finally, we need to think about which plot or visualization makes the most sense to our audience. The final version should contain a map that has the points of airport according to their longitude and latitude, users can choose their location and see the details in flight delays. Besides, to avoid flight delay, choosing a good airline that has a good reputation for flight delays times is quite important. Users can view the difference in times of flight delay

using bar chart in a year. Finally, even we cannot promise an accurate prediction of flight delays, getting more ideas of flight delay trend in a year, or in specific period will help our users to make better decisions. There is also flight map contains flight paths for different airports, the performance of a static map may be not as well as interactive one, the project can still be optimized by add and manipulate more data to match the functions used for the flight path. Finally, we may need to add a data table in the app, so users can view the data directly and search the data by keywords.

Implementation



```
library(shinydashboard)
library(shiny)
library(fiftystater)
library(ggplot2)
library(plotly)
library(mapproj)
library(leaflet)
library(htmltools)
library(ggmap)
library(geosphere)
library(maps)
library(mapview)
library(dygraphs)
library(zoo)
library(TTR)
library(TSA)
library(forecast)
library(rgdal)
library(DT)
library(RColorBrewer)
```

This project is mainly based on shiny in R and using many other R packages to implement different layouts and various plots. There are over 10 libraries such as ggplot, ggmap, plotly and shiny used in the project. Since these packages contain useful functions, so we can save amounts of time to write codes by ourselves. In addition, in library plotly, does a great work on creating the bar chart with fancy colours and interactive ideas.

In shinydashboard package, we use this library to create a shiny dashboard, with the name of the tab and users can navigate different plots. By changing the argument inside the dashboard function, we can adjust the size of plots and background colour. Hence, the dashboard can combine different plots together.

Furthermore, some other libraries related to data manipulation should be used, such as TSA, dplyr, and lubridate etc. The TSA library can convert related data to time series data, so we can put it into given functions and produce necessary diagrams. The dplyr is a very popular library to merge or reshape the data. In this project, it is used to aggregate some of variables and group according to the condition. Furthermore, we use lubridate library to create a new date column that concatenated year, month and day variables in the original data. Then creating a time series data using ts function in TSA. And using dygraphs library to visualize it in shiny.

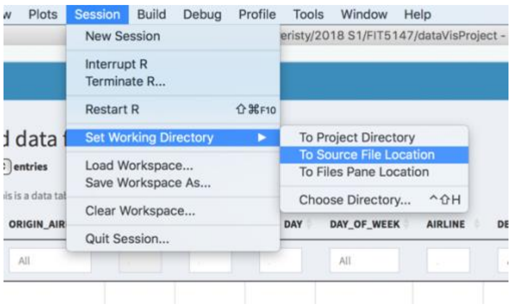
Some other useful libraries like maps and geosphere should also be used in this project. The map function can create a map template according to the map type specified. It is easy to set up the colour and background of the map and we can add points of airports by their geographic location on the map then connect points by lines. And an interactive map library leaflet, which does a great job of creating interactive maps. Besides, since the maps are easy to interact, and we can add labels and more information in the tooltip.

Besides, some of the libraries may be not quite frequently be used such as packages with colour patterns. The Rgdal package is used to produce colours depends on given arguments, however, it is not easy to use as the name of the colour pattern may be varied in different functions. There are some of the packages may not be used in the final version of the project but have been tried on for adding more features.

Finally, we add the data table in the app, so users can view the data directly. The DT package does a great job on that. In this package, we can change the style of the tables and provide different functions to provide more functionalities, such as adding a data filter. We disable the editing function in the project, so users have right to read-only.

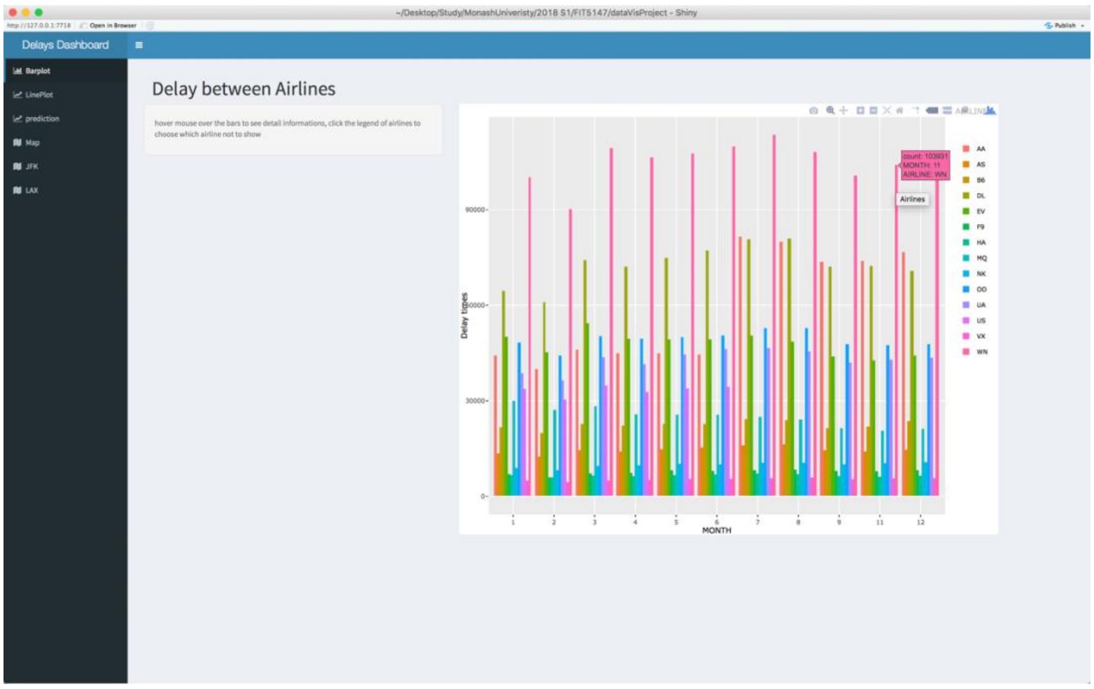
User guide

The user can use the web app based on shiny in R by running with given data in same folders. The data including the original data of flights, airports and airlines and the data manipulated by r functions. Therefore, loading the data before running any r functions is necessary and the working directory should be checked. Besides, as the data is very large, we separately compress the data and upload. The user should unzip the achieve file and move all the data file into the same folder of the app program. To set working directory, users can select session at the toolbar and select set working directory and set as file source location.



In order to make our data and visualizations efficient and effective, we save our data in the workplace as RData format so users can load them together on their devices. The loading data will make sure there are not missing dataset and avoid unexpected errors.

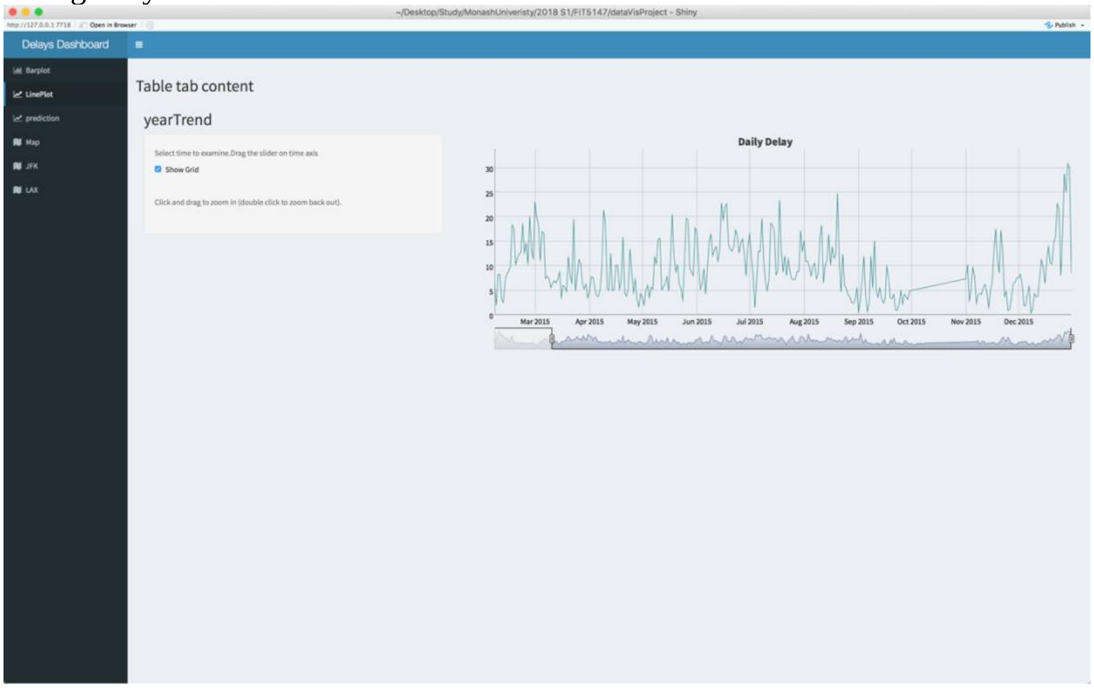
One more thing before running shiny app is that users need to run the supplement code in r and it may take a while.When the user running the app, there will be several seconds loading time, and since a large number of data, the loading time may be a little bit longer than usual. The user can see a shiny dashboard with two parts. On the left part, there is a menu bar and users can navigate to different plots; On the right side, the plots will be present in this part. There may also be some slider bar for users to change or control the plots and interact.



hover mouse over the bars to see detail informations, click the legend of airlines to choose which airline not to show

Figure 6: The first plot is created by the plotly package, and also used ggplot2. In this bar chart, users can view the bar in different colours as different airlines. The x-axis is the different months and the y-axis is the frequency of flight delays. As long as users click their mouse on the legend on the side of the chart, the clicked airline is unchosen. So, users do not need to concern about ambiguity in mixture colours at the bottom of the bar chart. For example, the airline WN has the highest number of delays in each month, travellers may prefer to choose other airlines with less number of delays

Moreover, users can hover their mouse over the bins, so they can see the details on the which month and airlines as well as the numbers of delays for each airline company in each month during the year.

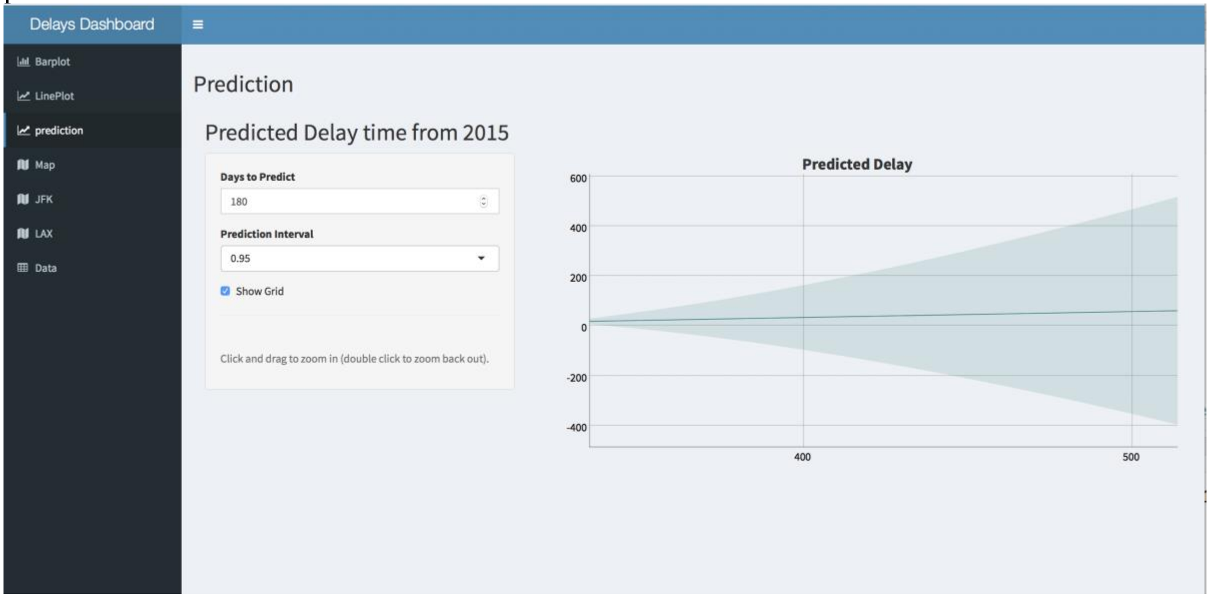


Select time to examine.Drag the slider on time axis

☑ Show Grid

Click and drag to zoom in (double click to zoom back out).

Figure 7: This is a line chart with some interactive features so users can control them by some of the controllers in the chart. Since the line chart shows an overview in the whole year, users would like to investigate in the specific period in the year, or in the range of different months. Hence, users can use the slider bar at the bottom of the plot and drag it left or right to choose which month to see and users also can double-click to return the original version. There is a grid on the plot, so users can choose to click or unclick the checkbox. If users want to plan a trip in June, they may want to see which day is less likely to have the delay and have fewer delay values by drag and hold the mouse in the plot, so the details of trends in the selected period are shown.



Prediction

Predicted Delay time from 2015

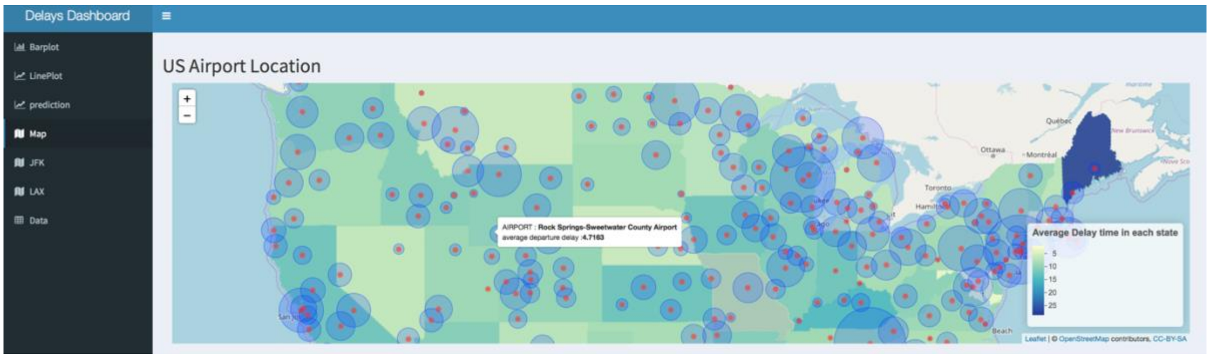
Days to Predict: 180

Prediction Interval: 0.95

✓ Show Grid

Click and drag to zoom in (double click to zoom back out).

Figure 8: In this prediction diagram, users can view the prediction value by changing the values on the left sidebars. The sidebar has two option, one is controlling how many days ahead to predict; another one is the prediction level which provides different levels of confidence in prediction for given data. Users can view the prediction by moving the mouse over the plot and a predicted average delay will be shown on the top. Since the data is not containing date data, convert to a date data type may have minor problems that cannot show exactly date on the x-axis as it is not included in original data, so only time shown as numerical numbers, and predict average delay ahead from 366.



AIRPORT : Rock Springs-Sweetwater County Airport
average departure delay :4.7163

Figure 9: The leaflet package can create an interactive map with various markers and labels. Users can zoom in or out by slide their mouse up or down. There is also a +/- toolbar on the top left corner, users can click + or – to zoom in or zoom out. When users hover the mouse over the state in the map, there will be a highlight on state and state name will be shown. In this project, we met the problem of inconsistency of state names. As the longitude and

latitude are for airports, we cannot correctly show the names of the states, however, if we can collect more geographic data of each state, we can certainly fix that problem.

In addition, after zooming in the map, if we move the mouse on the circles and the points in the circles, there will be a tooltip pops up, details about the code of the airport and the full name of that airport as well as the average delay time will be shown. Since the size of the transparent circles is depend on delay values, we can see the result of flight delay by checking the how large the circle is. The legend on the right corner is the average delay time in each state, as the colour is darker, there will more delay time on average. Users can have a general idea how does the delay distribution across the country.

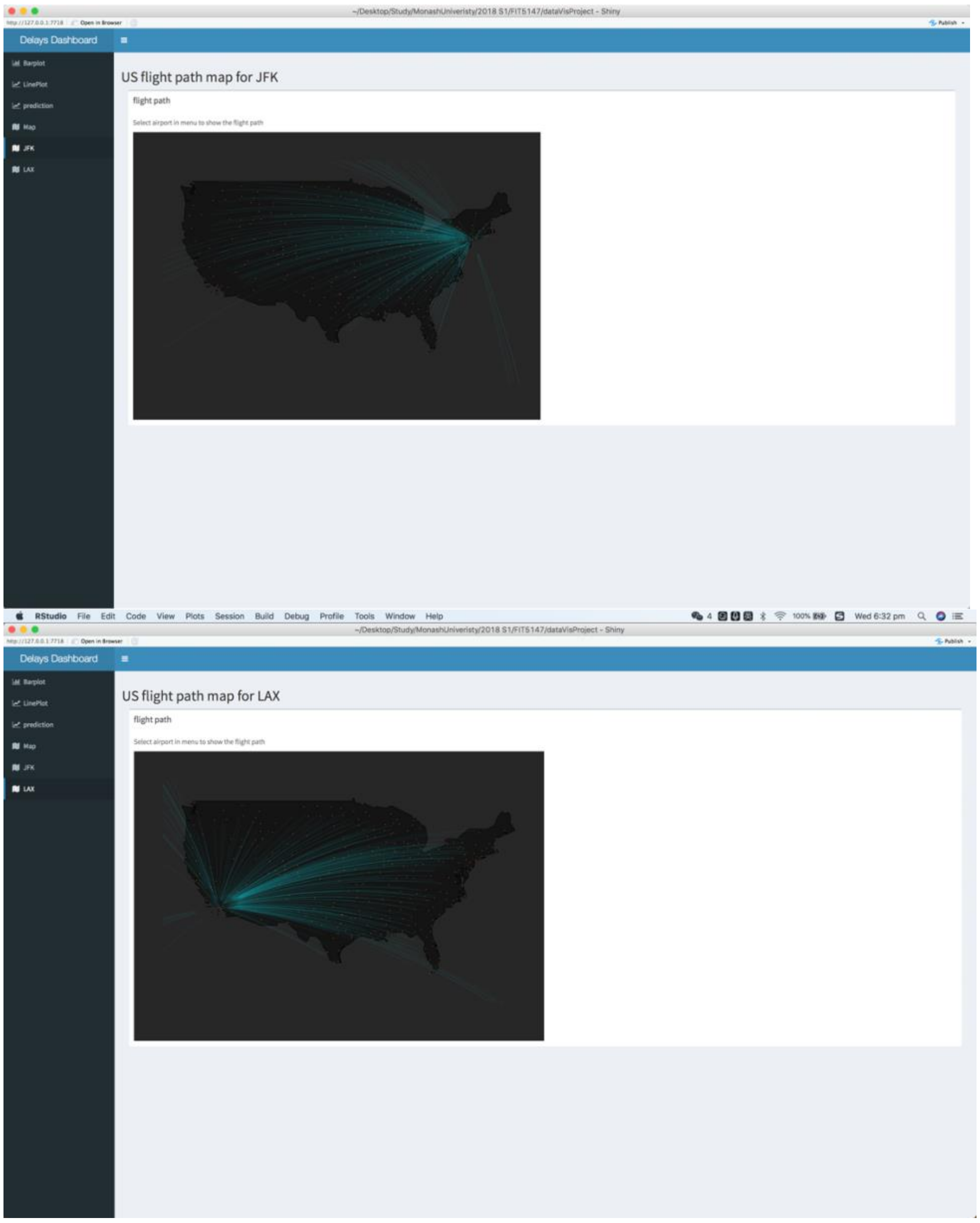
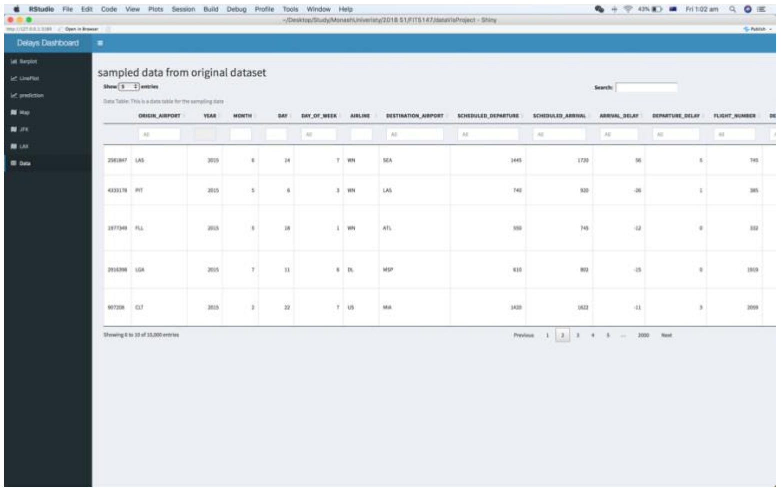

Delays Dashboard
Barplot
LinePlot
prediction
Map
JFK
LAX
US flight path map for JFK
flight path
Select airport in menu to show the flight path
US flight path map for LAX
flight path
Select airport in menu to show the flight path

Figure 10: There are two plots in this part. As the map is created in a static version, users can only select them by choosing in a menu bar. The name of the tab is the name of the airport. Each map shows the airports in orange points and blue lines of flight routes for that airport. In this project, we cannot show all of the airports in the shiny, and the airport points on the map cannot be connected because the data type in our data is not quite consistent. Therefore, we create a static flight route path to show the ideas in the app. If we can connect exactly two

points as original airport and destination airport, the flight route should be shown for each flight route and the stroke of the line will be delay values. Users can select different airport on the sidebar and switch in the same tab.


RStudio File Edit Code View Plots Session Build Debug Profile Tools Window Help
Delays Dashboard
Barplot
LinePlot
prediction
Map
JFK
LAX
Data
Publish

sampled data from original dataset

Show 5 entries

Search:

Data Table: This is a data table for the sampling data

	ORIGIN_AIRPORT	YEAR	MONTH	DAY	DAY_OF_WEEK	AIRLINE	DESTINATION_AIRPORT	SCHEDULED_DEPARTURE	SCHEDULED_ARRIVAL	ARRIVAL_DELAY	DEPARTURE_DELAY	FLIGHT_NUMBER
2581847	LAS	2015	6	14	7	WN	SEA	1445	1720	56	5	745
4333178	PIT	2015	5	6	3	WN	LAS	740	920	-26	1	385
1977349	FLL	2015	5	18	1	WN	ATL	550	745	-12	0	332
2916398	LGA	2015	7	11	6	DL	MSP	610	802	-15	0	1919
907208	CLT	2015	2	22	7	US	MIA	1420	1622	-11	3	2059

Showing 6 to 10 of 10,000 entries

Previous 1 2 3 4 5 … 2000 Next

Figure 11: This a data explorer table embedded into the shiny dashboard, and also because there are over 20 columns, users can adjust the size of the windows to see full data. In this data table, users can see the details of the data and filter the variables at the top of the table.

Some features like highlight the row when hovering is added to the data table.

Besides, the search bar supports users to search any specific values or attributes that interested in and control the number of entries shown in a page from 5 to 100 at the top left the select bar and change the number of the page at the bottom of the table.

To implement the data explorer table, we have to sample 10000 of observations in the original dataset to avoid crashes of the app. The app should improve its speed and accessibility to make it compatible with the large size of data.

Conclusion

To sum up, the app built in shiny has several plots to present. First is a bar chart, which use colour to compare the average flight delay. Second, there are time series plots with the interactive x-axis and a prediction plot in time with side toolbar to select different numbers of days to predict. Furthermore, there is an interactive map with different markers and legends in colours; the parameters used in the map will control the polygons and circles, so users can interact with those labels. Moreover, in order to create a flight route, we connect different points of airports on the map, and to avoid the errors may crash the app, we use the static map to show the ideas of flight paths. Finally, we also support users to view the data directly and interact with the data table.

In order to make the map optimal, we may need cloud database to store the large sample size of data and improve the speed of devices to run the app. In shiny, because R is not good at saving the memory of computers, users may need some devices meet minimum requirements.

Appendix and References

https://www.kaggle.com/usdot/flight-delays (US Department of Transportation, 2015)

https://en.wikipedia.org/wiki/List_of_the_busiest_airports_in_the_United_States

(Passenger Boarding (Enplanement) and All-Cargo Data for U.S. Airports - Previous Years, 2016)

https://www.transtats.bts.gov/homedrillchart.asp
(US Department of Transportation, n.d.)

US Department of Transportation. (2015). 2015 Flight Delays and Cancellations: filghts.csv [Data file]. Retrieved from https://www.kaggle.com/usdot/flight-delays

US Department of Transportation. (2015). 2015 Flight Delays and Cancellations: airlines.csv [Data file]. Retrieved from https://www.kaggle.com/usdot/flight-delays

US Department of Transportation. (2015). 2015 Flight Delays and Cancellations: airports.csv [Data file]. Retrieved from https://www.kaggle.com/usdot/flight-delays