Airfare Analyser - Group 8

Harsh Murdeshwar, 210641, harsharunun21@iitk.ac.in
Shubham Patel, 210709, devang21@iitk.ac.in

CS685: Data Mining

Abstract

Due to distance and time constraints, many students prefer to travel to their homes via flights. Dynamic pricing algorithms make it confusing to understand when to book a ticket. To better understand the trends in price fluctuations we work towards our project. We collected data by scraping Google Flights. We present insightful observations regarding the variation of prices due to factors such as day, festivals, etc. As a part of the project, we also develop a tool to help better visualize India's domestic flight network.

1 Links

GitHub Repo
Connectivity Dashboard (Please view in full screen [Presentation Mode])

2 Motivation of the Problem

Travelling is an integral part of our life. Be it special festivals or the end sem recess, every one of us loves to go home and enjoy these moments with our family and relatives. Many of the students here at IIT Kanpur live very far from their homes, which makes going home by train both time-consuming and tiring. Due to this, these students prefer to travel home via flights, but the complex dynamic pricing algorithms employed by the airlines make the task a bit tricky as it is very confusing to decide the best time to purchase the tickets. We begin our project with the aim of better understanding the underlying trends and deriving helpful insights. While working on a project, we also realized the need for visualization of the connectivity of domestic flights, which we have also tried to address via this effort.

3 Data Used

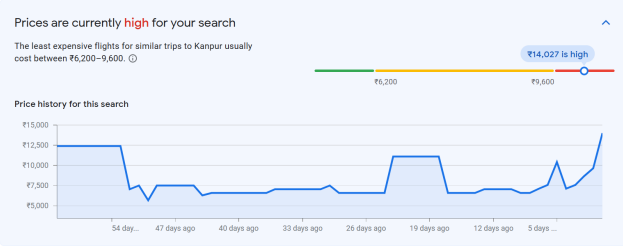
In order to collect data to be used in our project, we first checked whether there were ready-made datasets available online. After some searching, we found that there were datasets available; however, they were all related to flights that did not connect Indian cities, and almost all of them were paid. Since we wanted to analyze the prices of Indian flights, we decided to obtain the data by ourselves.

We then turned our attention to websites that provide flight booking services to obtain data. We found that most of these websites do not store the history of flight prices and have a very limited amount of data.

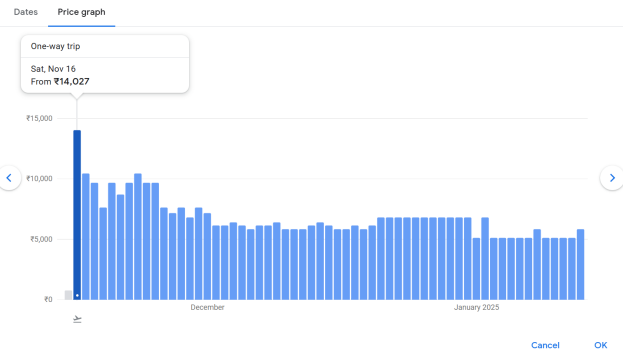
Finally, we ended up on Google Flights. Google Flights records a 60-day history of prices as well as a more structured representation of flight prices on a given day. Hence, we decided to go ahead with Google Flights as the source of our data.

One more aspect that we realised would be helpful to know about for general knowledge and the purpose of planning a trip would be the connectivity of various airports in India. While searching for individual source and destination airports, we can determine whether a flight exists or not, but this process is relatively cumbersome. To help improve this, we decided to utilise the data from Google Flights itself. During this process, we realised that having an interactive tool to visualise this beautiful network would be really insightful. Also, this data would help to determine which airports are active with large number of arrivals and departures while those which are less active. We used the following data from Google Flights:

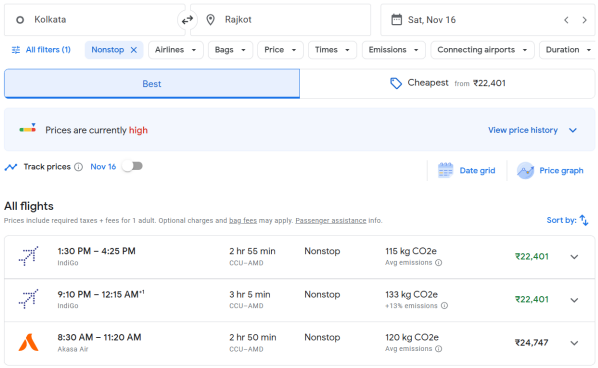
1. 60-day price history for each travel date:



2. The prices of flights on a particular day:



3. The number of direct flights on a day between a pair of airports in India



4 Methodology

4.1 Scraping

4.1.1 Approach

Google Flights is a website tailored for users and not meant to act as a data source / API for data. Due to this, for viewing the graphs mentioned above, there exist specific user flows. E.g. Click the ‘Show Price History‘ button, followed by hovering the mouse on a specific point on the graph to view the price for the corresponding date. To mimic this user flow, we utilise the ‘Selenium‘ library in Python with ‘chrome-web drivers‘ and perform the necessary actions.

4.1.2 Automation via Cron

The script for extracting histories for upcoming travel dates extracts the histories for all the upcoming travel dates. Running this for accessing every new data point would be slow and inefficient. Also if we run this script after 30 days, the data for 30 days will be lost since they would then be previous days. Due to this, it would be more efficient to extract the prices on a given booking date for which we create a script. Manually running this script daily would again be a challenge since it is highly possible to miss taking readings on certain days. To resolve this, we create a cron job in Linux which automatically runs the scripts daily at the scheduled time.

4.1.3 Multithreading

India has nearly 100 airports. The number of flights arriving and departing from airports varies based on the number of days. Getting the data for the connectivity graph of India would require roughly 100 * 100 * 7 requests. At a rate of 7 requests per minute by one session, collecting all this data would take a significant amount of time. To speed up the process we spawn multiple threads. After tuning for the rate and failed requests (discussed below), we find the optimal number of threads to be 8.

4.1.4 Challenges

There were several challenges of different kinds we faced which helped us to learn tactics to employ while scraping data.

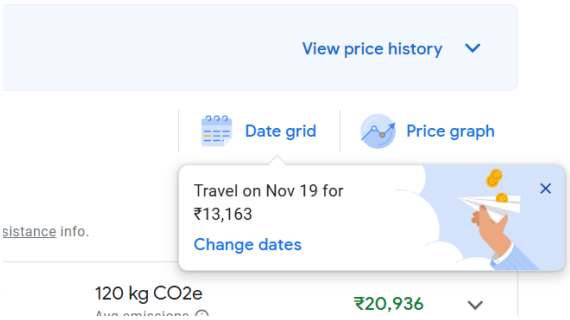
Anonymised URLs and Randomised Source Code:

```
https://www.google.com/travel/flights/search?tfs=CBwQAhomEgoyMDI0LTEyL
TA2KABqBwgBEgNCT01yDQgCEgkvbS8wMjJ0cTRAAUgBcAGCAQsI_____________AZgBAg&tfu=EgYIAhAAGAA
```

```
<div class="eaO3rb qs41qe yRaoXe" id="Oacf4b" jsname="IpRbQ" role="tab"
   aria-selected="true"
   jslog="212305;ved:2ahUKEwi308SKrt6JAxWli2YCHVzUAkUQ0foMegQIAxAh;track:click"
   tabindex="0" jscontroller="RGYs6" jsaction="KjsqPd">
   <div class="sdWl2d">Best</div>
</div>
```

The URLs are also encoded in an unknown format. This prevents us from directly querying the required URL based on the source and destination airport. To visit a given page, entering the travel details has to be simulated. Additionally to reduce the interpretability of the html pages and to make scraping harder, the source code of the website has been anonymised. This made it hard for us to directly scrape the contents of elements. To tackle this, we began by hard-coding the class names and were partially successful. But soon, we realised that some of these random names also varied across sessions, complicating the task. Finally, we utilised 'aria-labels' on the site, meant for supporting accessibility tools, for extracting relevant information from the pages.

Pop-Up Suggestions:



For scraping data, we simulated the navigation the same as a normal user by altering the date to see and record the data for the corresponding date. Due to this, the website considered the code as a normal user and gave various types of pop suggestions. Due to the appearance of these unexpected pop-up elements our code had broken down sometimes after executing for hours. To resolve these, we had to explicitly add a function for closing pop-ups.

Rate Limitations:

The website from which the data is being also took a few seconds to load the response. This leads to a bottleneck in the amount of data that can be scrapped in a given amount of time. There was one more aspect important especially for the connectivity graph. When the rate of requests exceeded a certain threshold, the website started to give `Oops and error occurred.` and refrained from displaying the data. To maintain a balance between the time spent by the thread sleeping and the number of failed requests, we tune the frequency of requests and also we repeat the procedure multiple times.

4.2 Data Pre-processing

The data that we scraped with selenium could not be directly used for analysis as it was in the form of strings such as:

- ”61 days ago - ₹5,798”
- ”One-way trip
  Tue, Oct 8
  From ₹8,871”

We used a Python script to convert this data into an organized format by extracting the price as an integer, finding the travel date based on the position of the string in the file or from the string itself, and calculating the booking date from the number of days ago (from the data collection date).

We then recorded this data into a CSV file with columns: `src, dest, booking_date, travel_date, price`. This CSV file would now act as our dataset for all the analysis that we would perform.

We also had to deal with missing data because, on some days, the cron job would run into some error due to either power issues or network issues.

Given the data from two sources, i.e. history and cron, we combined and handled two different prices on the same day by using the mean of the price. For some days, there were 2 different values of prices for a given date of booking, travel because the readings were taken at different times of the day and flight prices sometimes are different at different times of the day.

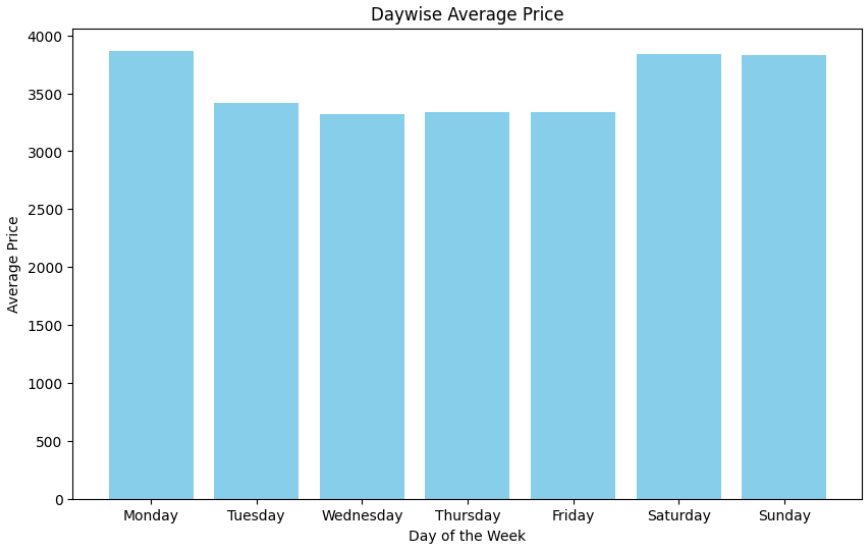
5 Results

5.1 General trends

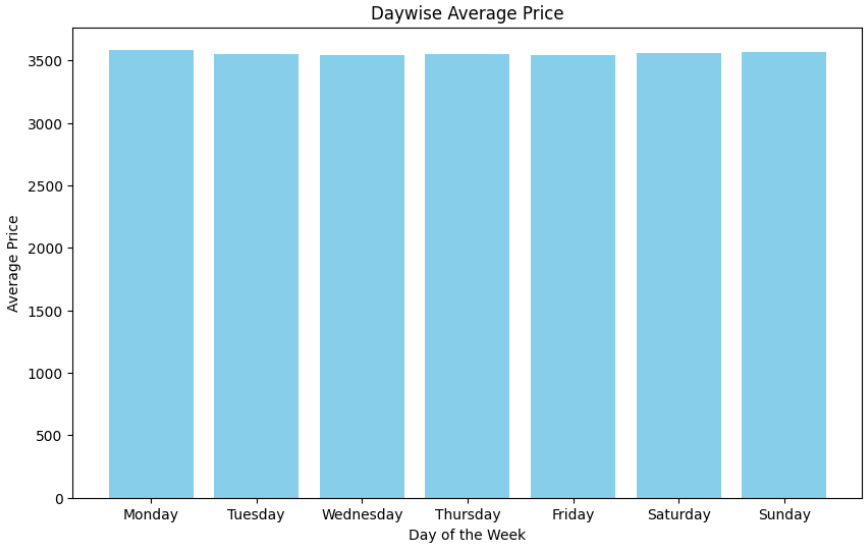
5.1.1 Average price based on the day of week

We consider day of the week for:

- **Travel Date**: From the plots below, we can infer that on Mondays and on weekends, the prices tend to be a bit higher than the other days



- **Booking Date**: From the plots below, we can infer that the day of the week of the booking date has minimal effect on flight prices



5.1.2 Price heatmap over travel and booking dates

In the heatmap shown below, the vertical axis represents the booking date, the horizontal axis represents the travel date, yellow represents lower prices, and blue represents higher prices. The heatmap makes it clear that prices substantially increase for travel during Diwali, even when the booking date is months before the travel date. Also, for Kanpur to Delhi, the prices are higher after 1st Nov, while for Delhi to Kanpur, prices are higher before 1st Nov. This suggests that people from Delhi are traveling to Kanpur at the start of Diwali and returning after Diwali.

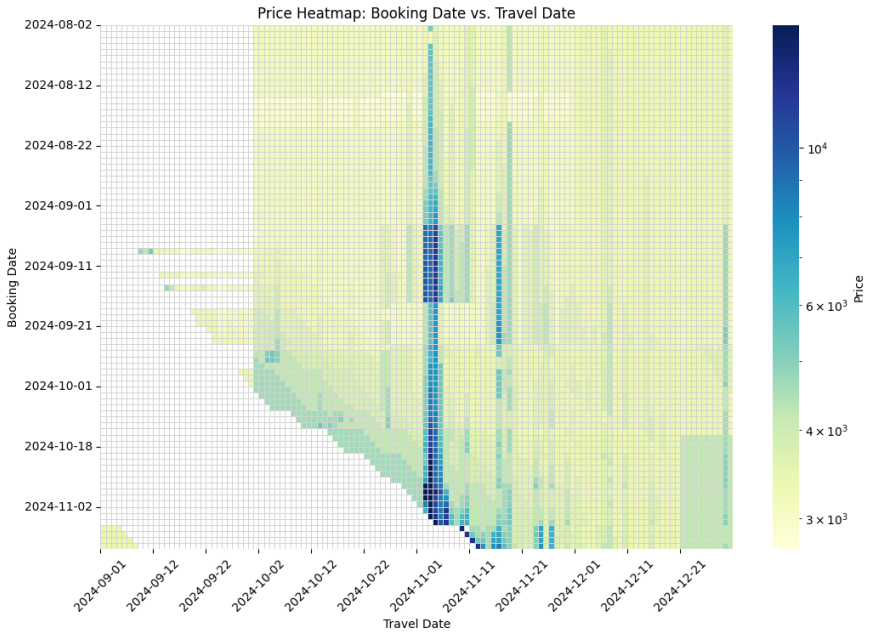


Figure 1: Kanpur to Delhi

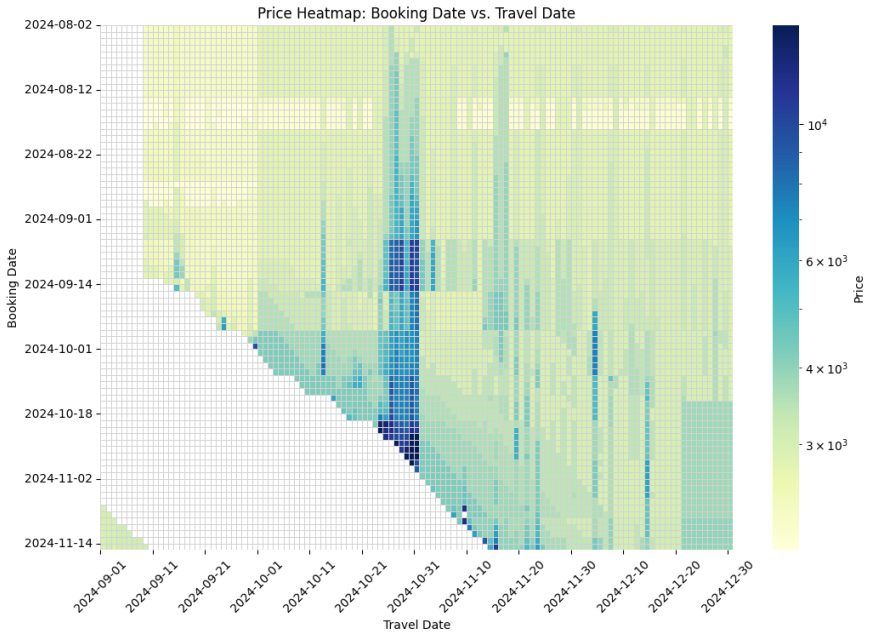
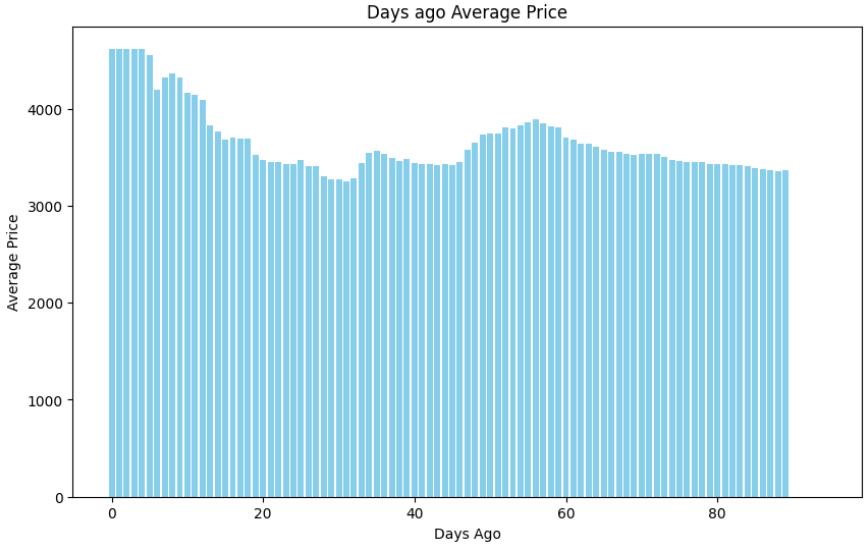


Figure 2: Delhi to Kanpur

5.1.3 Average Price vs Days before travel

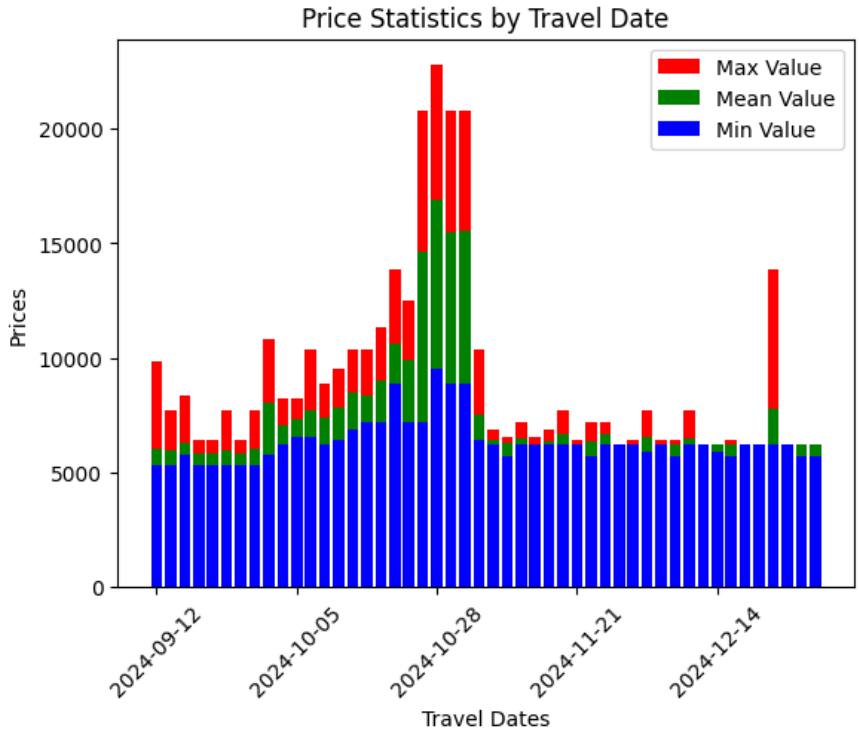
From the plots, we can notice that there is a minimum in the flight prices around 20-30 days before the travel date. Contrary to the belief that the earlier you book, the better, it seems that the best time to book domestic

flights is actually around 1 month before the travel date. This is because not sufficient seats on the flights have been sold, and there is only one month left, so they need to quickly sell the seats. For this, they drop prices.



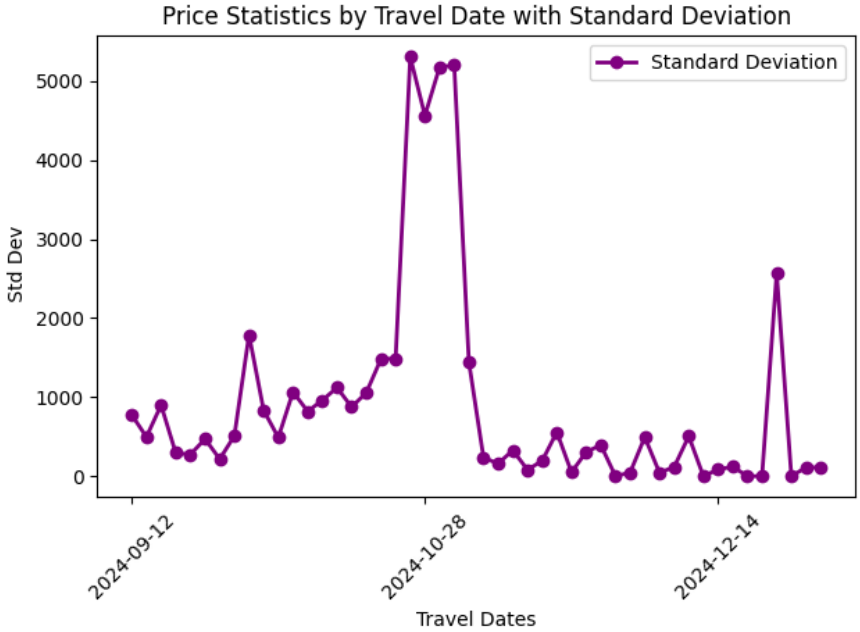
5.1.4 Min-Mean-Max bar graph over travel dates

We can observe the minimum mean and maximum all increase during the Diwali period, which is expected. Additionally, we observe one more interesting aspect. Before the Diwali period, the prices are steadily increasing. After the Diwali period, instead of smoothly decreasing, they abruptly drop. Possibilities could be that children's vacation usually begins one or two weeks before Diwali. So during this period, slowly families start travelling to their home towns/tourist places at different times based on their parent's lifestyle (working / non-working). So there is a gradual increase. But finally, when schools start on Monday, everyone has to reach home since a child's studies are important, which is why there is an abrupt decrease in demand having a similar effect on prices.



5.1.5 Standard Deviation Graph

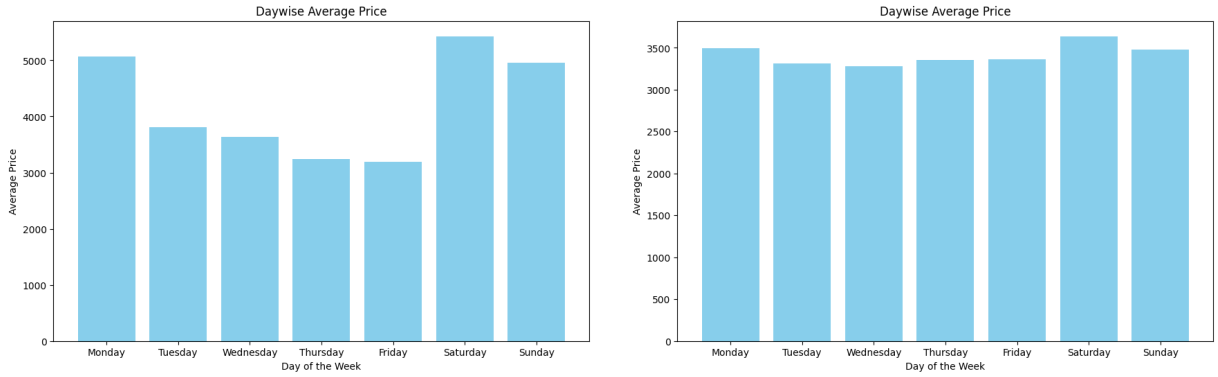
We can observe that during the period of Diwali festivities, the standard deviation in prices increase significantly, which is caused by greater fluctuations in prices. Additionally, similar to in previous graphs, we can also observe a spike for 31st December which is New Year day.



5.2 Diwali vs Non-Diwali days

5.2.1 Effect of travel day of week

In case of Diwali period the expenses incurred in travel can be reduced significantly by adjusting the day of travel. If it is possible by taking one/two more leaves from college/office and travelling on low price days. In the case of a non-Diwali period, there is an impact, but it is not that pronounced.



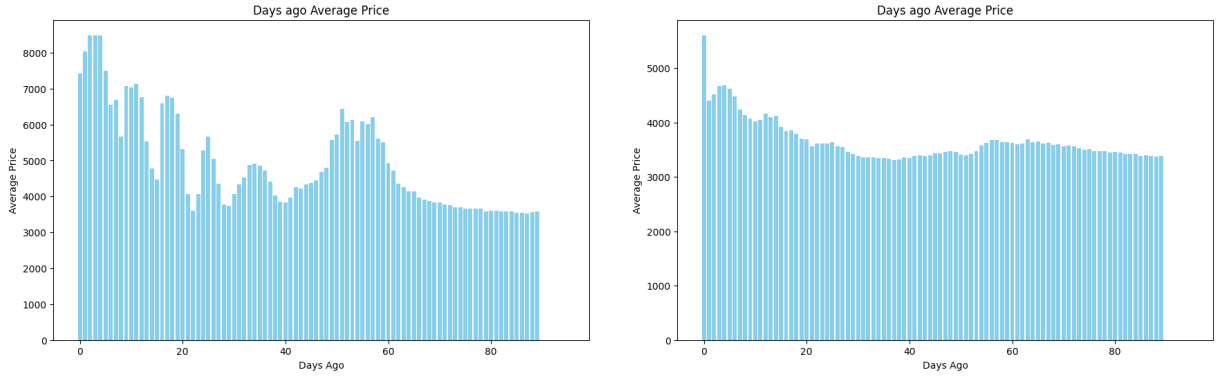
(a) Diwali Period

(b) Non-Diwali Period

Figure 3: Effect of day of travel.

5.2.2 Average Price vs Days before travel

As we can clearly see from the figure below, the fluctuations in price are more volatile during the festive period as compared to smoother trends during the normal time period. This makes it more difficult to decide on a good time to book the tickets since one is unsure whether the fare would go lower than its current value in future or no.

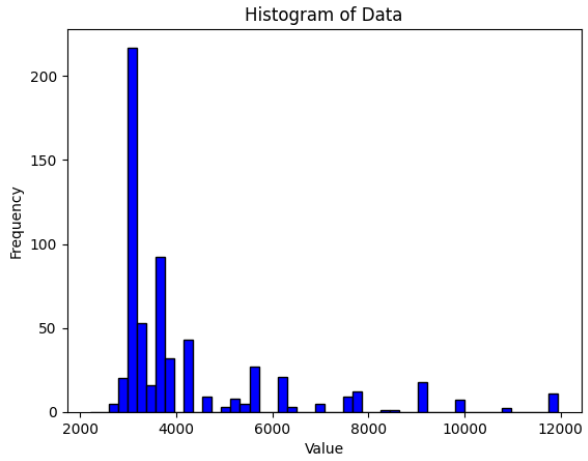


(a) Diwali Period

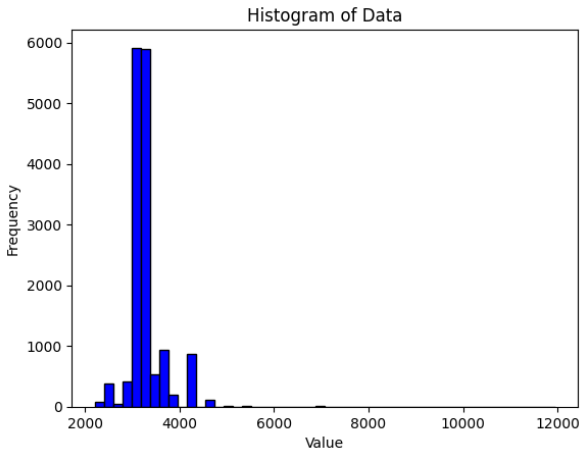
(b) Non-Diwali Period

5.2.3 Price Histogram

Over here we can notice that for the non seasonal period we have a less broad spread of prices in history for the same route while it is more spread in case of Diwali period. Also the non seasonal price distribution appears to be closer in shape to a gaussian distribution as compared to the other.

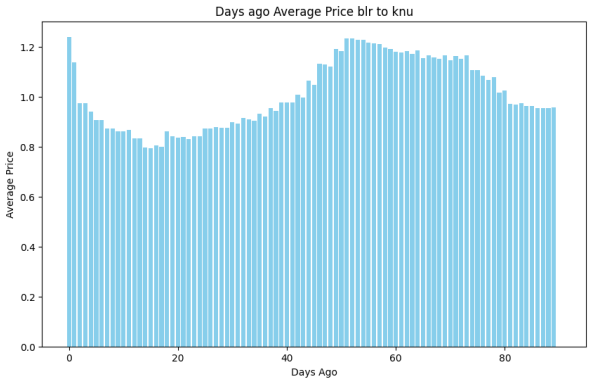


(a) Diwali Period

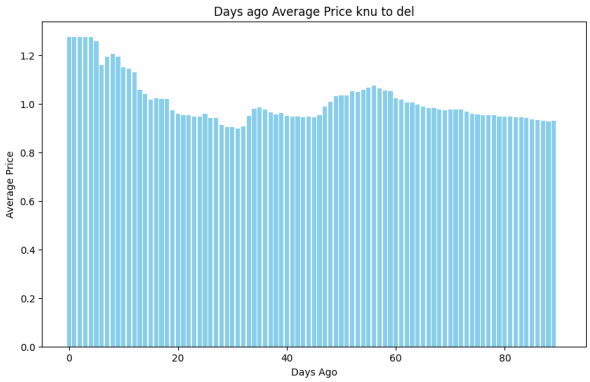


(b) Non-Diwali Period

5.3 Comparison Between Routes



(a) Kanpur to Bangalore



(b) Kanpur to Delhi

Figure 6: Variation in average price history across routes (Normalised by Mean).

We notice that even for the same source airport, the price histories of two different destination airports vary significantly. In the case of Kanpur to Bangalore, the best time to book would be 15 days before, while for Delhi, it is 30 days in advance. While there is a rise and fall in both of the cases, the distribution is very different. Hence, we would recommend that it would be better to do an analysis specific to the route that one is planning to travel on and avoid making extrapolations from one route to another.

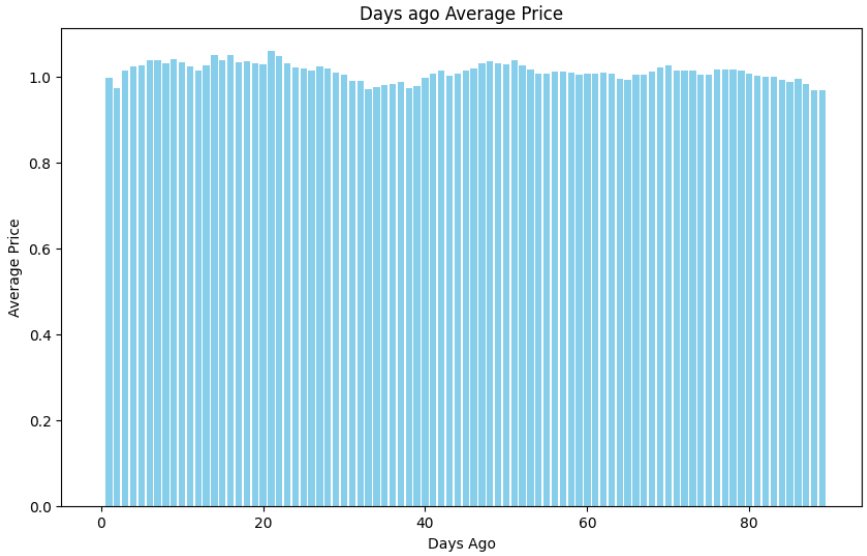


Figure 7: Price history for Mumbai to Delhi

As compared to the less active route of Kanpur, over here while we can see variations in the prices of flights, they are relatively much smaller. This could possibly be because there are many different options passengers have to choose from, so if one of the prices varies is high, then the passenger can select another flight. So varying the price too much might be negative for the airlines.

Similarly the variation of price is lesser in here with the day of travel as compared to before.

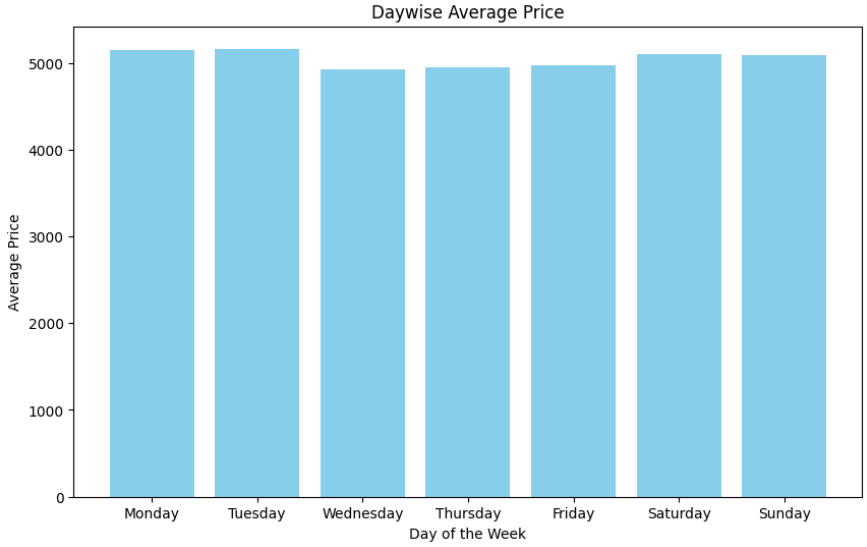


Figure 8: Average price on different days of week for Mumbai to Delhi

5.4 Connectivity Graph

We develop an interactive dashboard to enhance the visualisation of the connectivity graph of domestic flights in India. In the connectivity distribution, we display how the connected airports are distributed across the country (weighed by frequency). In the middle, we display the paths connecting airports. The edges corresponding to the paths are displayed in proportion to the relative frequency of flights per week.
Link: Connectivity Graph Dashboard (Please view in full screen [Presentation Mode])

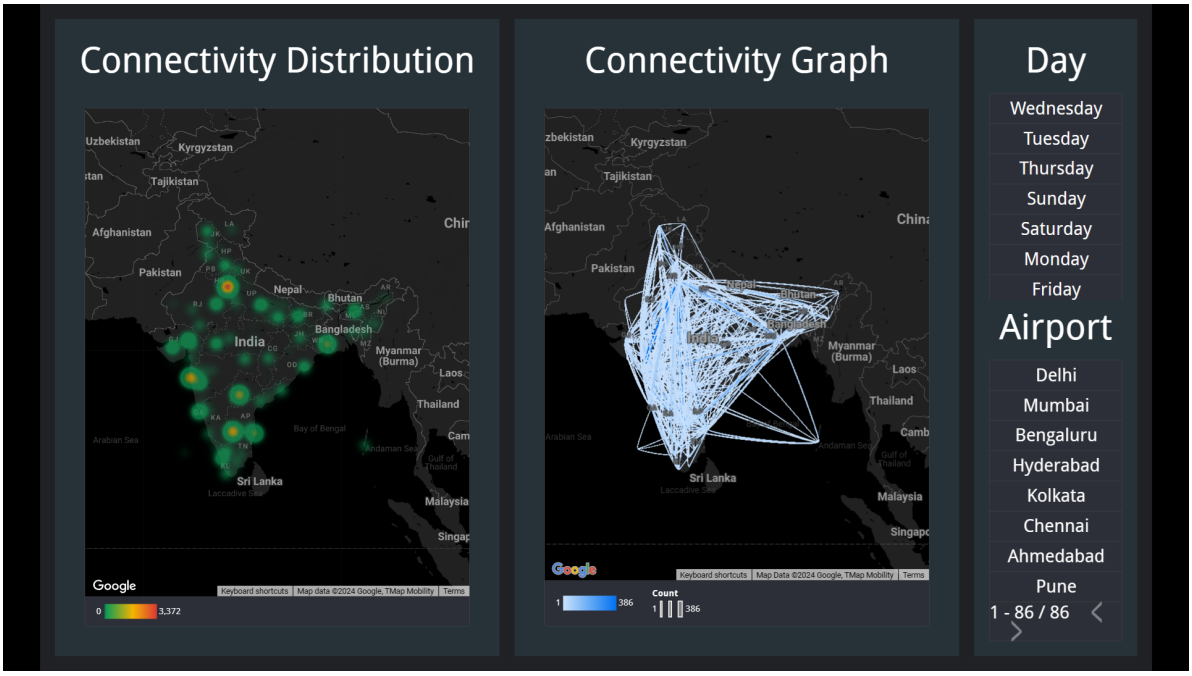
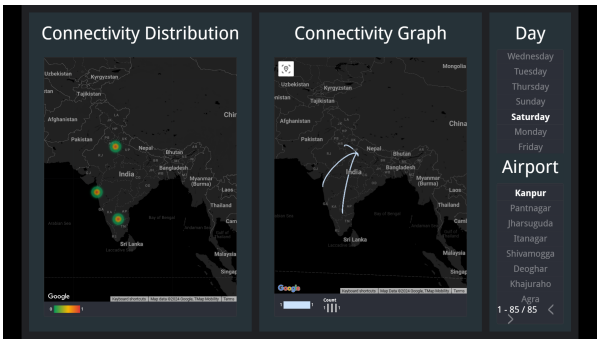
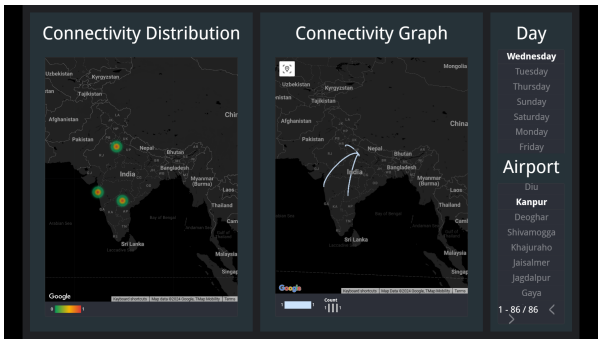


Figure 9: Connectivity Graph Dashboard

Initially, the entire connectivity graph for India is displayed. On the right, we enhance visualisation by allowing the user to select a specific airport and even a specific day of the week.



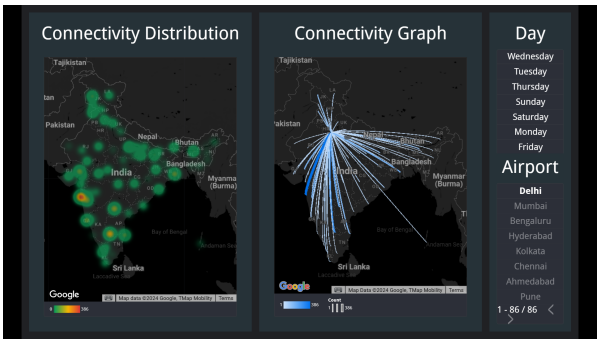
(a) Kanpur (Saturday)



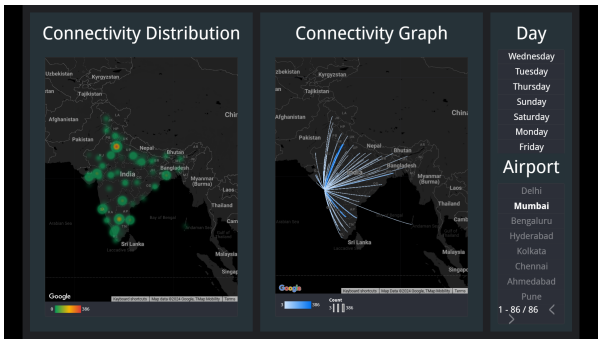
(b) Kanpur (Wednesday)

Figure 10: Variation across different days of the week. There is flight from Kanpur to Bangalore on Saturday and to Hyderabad on Wednesday.

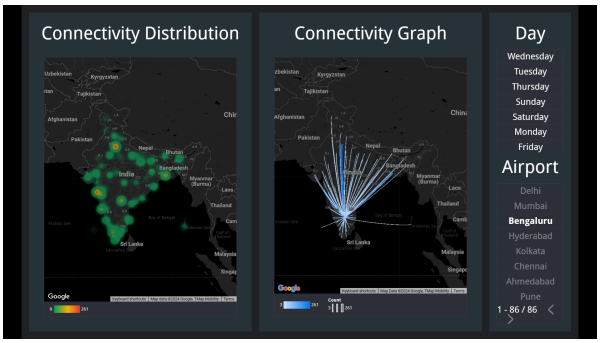
Following are the visualisations corresponding to the Top 3 most active and least active airports. As we can see the most active airports not only connected with high frequenct to each other bu are connected to most of the airports across the country. The least active airports are connected to some of the major airports making it possible to travel to different locations across the country from these airports via flight changes.



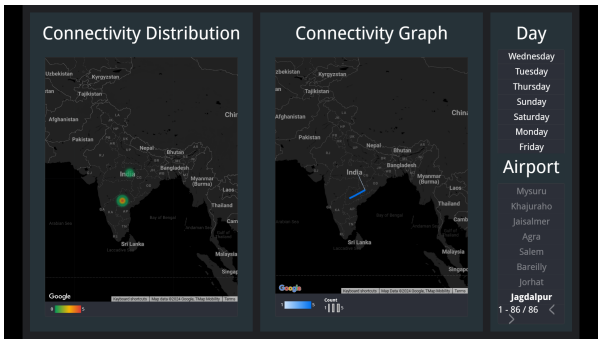
(a) Delhi (Most Active)



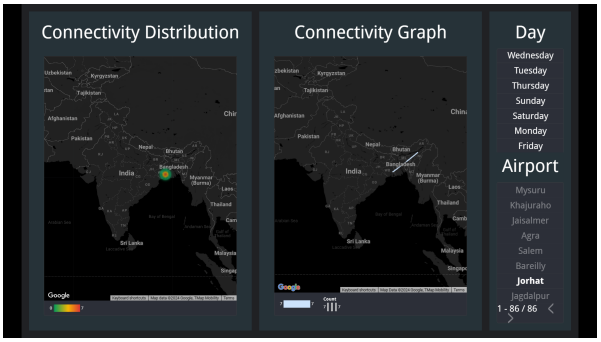
(b) Mumbai (2nd Active)



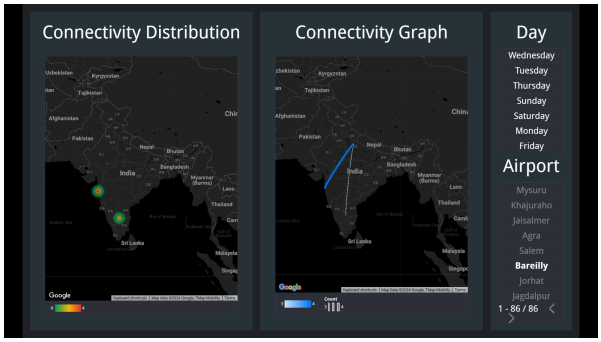
(c) Bangalore (3rd Active)



(d) Jagdalpur (Least Active)



(e) Jorhat (2nd Least Active)



(f) Bareilly (3rd Least Active)

We also analyse the variation in the frequency of flights across days of the week. We collected the data for this graph a few weeks ago for the week from 11 Nov 2024 to 18 Nov 2024. We notice a really interesting aspect. It is not only that the prices of flights but also the frequency of the flights is influenced by the holiday pattern. Since 15 Nov 2024 is a holiday on the occasion of Guru Nanak Jayanti, this becomes a long weekend. As we can see, there is a high number of flights on Thursday, possibly because on Thursday evening, many people are leaving with their families over the long weekend. Finally, there is a spike on Sunday since

people would likely be returning to their homes and back to school and work. This possibly indicates that the strategies of the airline companies are not only limited to prices but also relate to scheduling of special seasonal flights to make maximum revenue.

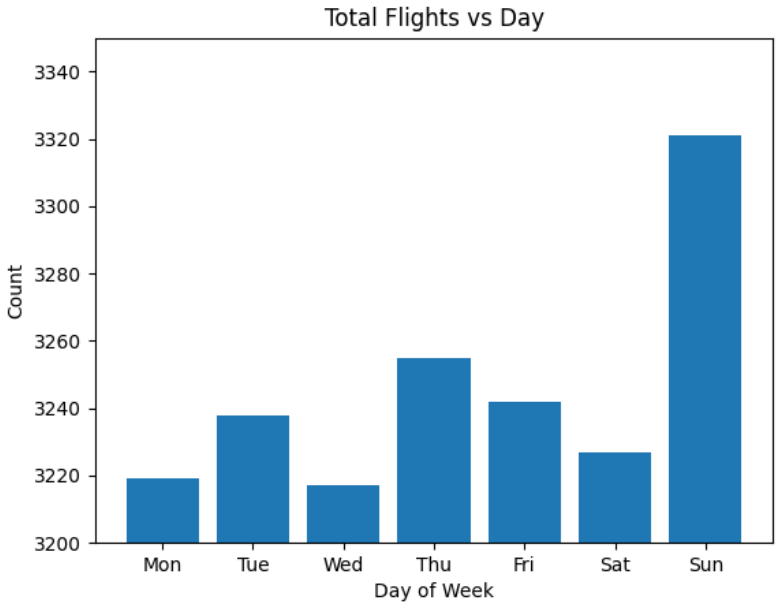
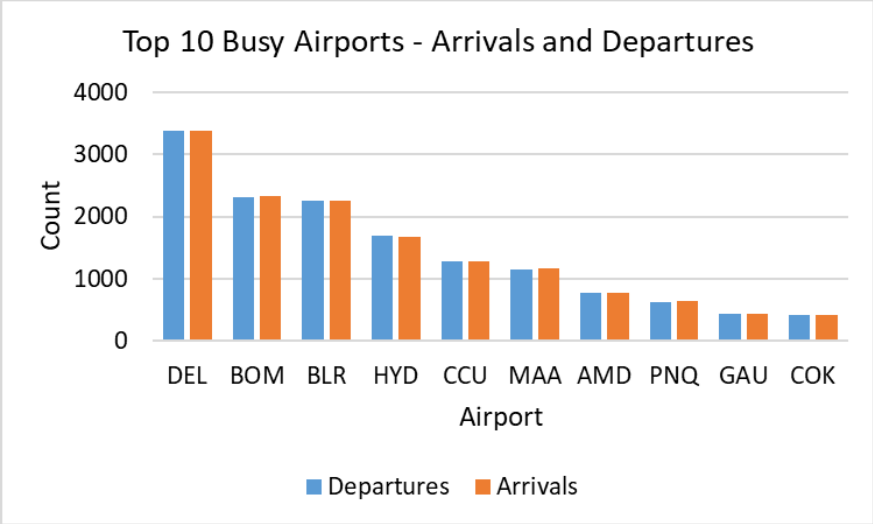


Figure 12: Distribution of travels across days of week

Following is the trend of arrivals and departures for the 10 most active airports. Initially it might seem a bit strange that the departures are slightly less than the number of arrivals. This could be possible due to maintenance aspects of flights. Also, one more possible reason is that the data we have collected only consists of domestic travel by flight. It is likely that some of the domestic flights could be planned to proceed to international routes.



6 Conclusions and Future Directions

This project gave us a really exciting opportunity to dive deeper into the problems many of the students in our community face which is regarding the dynamic pricing of airplane tickets. We began by scraping different kinds of data from the Google Flights website tackling different challenges in it. We then pre-process the data and extract insightful observations from it. Our analysis primarily focuses on the following domains: 1) Analysis of variation in flight prices; 2) Impact of festive seasons on flight prices; 3) Variation across airports; and 4) Visualization of the domestic flight network. A similar analysis can be extended to different sources and destinations cities. While our analyses currently focus on non-stop flights, a more detailed analysis could also focus on flights including stops to study patterns in relation to prices with the number and types of stops. We hope to have helped users gain more knowledge to better understand the trends in Airfares.

7 Team Contributions

We have decided to keep our contribution weightings as equal - 50%, 50%.
Both team members were involved in all aspects; the following are the aspects that were led by one of the members.

- Harsh Murdeshwar
  - Scraping history of prices for days
  - Cron Job Data
  - Graph Plots Visualisation
- Shubham Patel
  - Scraping future prices on a given day
  - Connectivity Data
  - Connectivity Graph Visualisation