Web Scraping - Airline Price Analysis

Sri R Sankaranarayanan
DSC680 | Applied Data Science | Project 2 | Milestone 3

https://github.com/rengsankar1986/DSC680

Business Problem :-

Airline companies use complex algorithms to calculate flight prices given various conditions present at that particular time. These methods take financial, marketing, and various social factors into account to predict flight prices. It is difficult for airlines to maintain prices since prices change dynamically due to different conditions. That's why we will try to use machine learning to solve this problem. This can help airlines by predicting what prices they can maintain. It can also help customers to predict future flight prices and plan their journey accordingly.

Background/History :

This project is combining both my passion for travelling and Data Science to find cheap ticket price. Flight price analysis in Airline Transportation is a complex phenomenon and requires frequently checking through travel websites for deals based on preferences. The Objective of this project is to perform Web Scraping on Kayak.com to predict flight prices and send an email alert to my personal gmail. My target variable will be the price (continuous numeric value).

Data Explanation :-

I will be scraping through kayak.com websites for current pricing data in real-time (web scraping technique explained below), below will be the datasets that I am going to be procuring in real-time.

Airline: This column will have all the types of airlines like Delta, Emirates, Jet Airways, Air India, and many more operating through my input routes.

Date_of_Journey: This column will let us know about the date on which the passenger's journey will start.

Source: This column holds the name of the place from where the passenger's journey will start.

Destination: This column holds the name of the place to where passengers wanted to travel.

Route: Here we can know about that what is the route through which passengers have opted to travel from his/her source to their destination.

Arrival_Time: Arrival time is when the passenger will reach his/her destination.

Duration: Duration is the whole period that a flight will take to complete its journey from source to destination.

Total_Stops: This will let us know in how many places flights will stop there for the flight in the whole journey.

Additional_Info: In this column, we will get information about food, kind of food, and other amenities.

Price: Price of the flight for a complete journey including all the expenses before onboarding.

Data Science Methods :-

a. I would be scraping data using Python Selenium. Below are the steps for procuring real-time JSON dataset from Expedia.com

   1. Construct the URL of the search results from Expedia- Here is one for the available flights listed from Dallas to Chennai, India – https://www.expedia.com/Flights-Search?flight-type=on&mode=search&trip=oneway&leg1=from%3ADallas+%28DFW+-+Dallas-Fort+Worth+Intl.%29%2Cto%3AChennai+%28MAA+-+Chennai+Intl.%29%2Cdeparture%3A8%2F16%2F2022TANYT&options=cabinclass%3Aeconomy&passengers=children%3A2%5B6%3B2%5D%2Cadults%3A2%2Cseniors%3A0%2Cinfantinlap%3AY&fromDate=8%2F16%2F2022&d1=2022-08-16
   2. Download HTML of the search result page using Python Requests.
   3. Parse the page using LXML – LXML lets you navigate the HTML Tree Structure using Xpaths. We have predefined the XPaths for the details we need in the code.
   4. Save the data to a JSON file.

b. Importing the training data
c. Data pre-processing
d. Importing test data and data pre-processing
e. Creating Independent and Dependent variables
f. Features Selections
g. Model Building
h. Hyperparameter tuning

Analysis :

I will be scraping data from Kayak.com and performing time series analysis of the output data which is price. I am planning to find trends in the past observations and outline the best price between certain time of the search. I would like to perform EDA on the web scraped dataset just to get good idea on how the pricing is affected on different search variables and conclude what are the variables affecting the pricing.

Conclusion:

The idea is to learn and perform web scraping using Python – this humble attempt of finding the best possible Flight deals is personal milestone and I would like to leverage this to other areas like home search sites, web scrape review sites and then perform sentimental analysis.

Ethical Considerations:-

I am presenting my user-agent string in the Expedia or Kayak website when procuring datasets so the website administrator understands the intention of the Python requests. We will not import or use any personal data and have already confirmed with the websites that it is allowed for Web Scraping. The price data is publicly available data, hence we will use that as our target variable. The rate of Web scraping allowed will be taken into consideration when running the requests.

Assumptions :-

There are no pop-ups or Capcha's on the website that I will be scraping – Of course, there are ways to avoid them, but I am not taking it that deeper yet.

Limitations :-

I will have to perform this webscraping using VPN or VPS network since kayak.com blocks the service calls at random times. I also have challenges in sending an email with the results from the VPN network since certain firewall blocks have to bypassed from the websites.

Challenges/Issues:-

Answering bot captcha challenge questions may become challenge in using sites like Expedia or Kayak. I have other alternative sites like Skyscanner.com or momondo.com if the primary sites do not allow Web Scraping.

Future Uses:-

I would like to integrate with Twilio to send text messages instead of emails, improvise the search using multiple inputs, schedule the program using bots or other schedulers for advanced more sophisticated results. I would also like to expand the web scraping to other similar sites with minimal changes to the Python program. I would also like to perform the best possible suggestions on the flight price using RIPPER and Q-Learning algorithms.

References :-

https://towardsdatascience.com/how-to-scrape-flight-prices-with-python-using-selenium-a8382a70d5d6 --> Web Scraping with Python using Selenium.

https://www.analyticsvidhya.com/blog/2021/06/flight-price-prediction-a-regression-analysis-using-lazy-prediction/ → Lazy prediction Regression Analysis

https://www.analyticsvidhya.com/blog/2022/01/flight-fare-prediction-using-machine-learning/ → I will be performing EDA for getting data insights, feature engineering, Data Visualization and then followed by ML algorithms.

https://medium.com/mlearning-ai/flight-fare-predictions-b1150fdc45a3 --> Good comparative case study similar to my project.

https://www.kaggle.com/datasets/nikhilmittal/flight-fare-prediction-mh --> Sample Dataset will be used from Kaggle.com for performing initial Data Visualization.

https://medium.com/@fneves/if-you-like-to-travel-let-python-help-you-scrape-the-best-fares-5a1f26213086 --> Good Case study on how Web Scraping is done.

https://www.altexsoft.com/blog/flight-price-predictor/ → Training models to find the best priced flight details per itinerary.

https://www.eff.org/document/preliminary-injunction-american-airlines-v-farechase-inc --> Ethical considerations

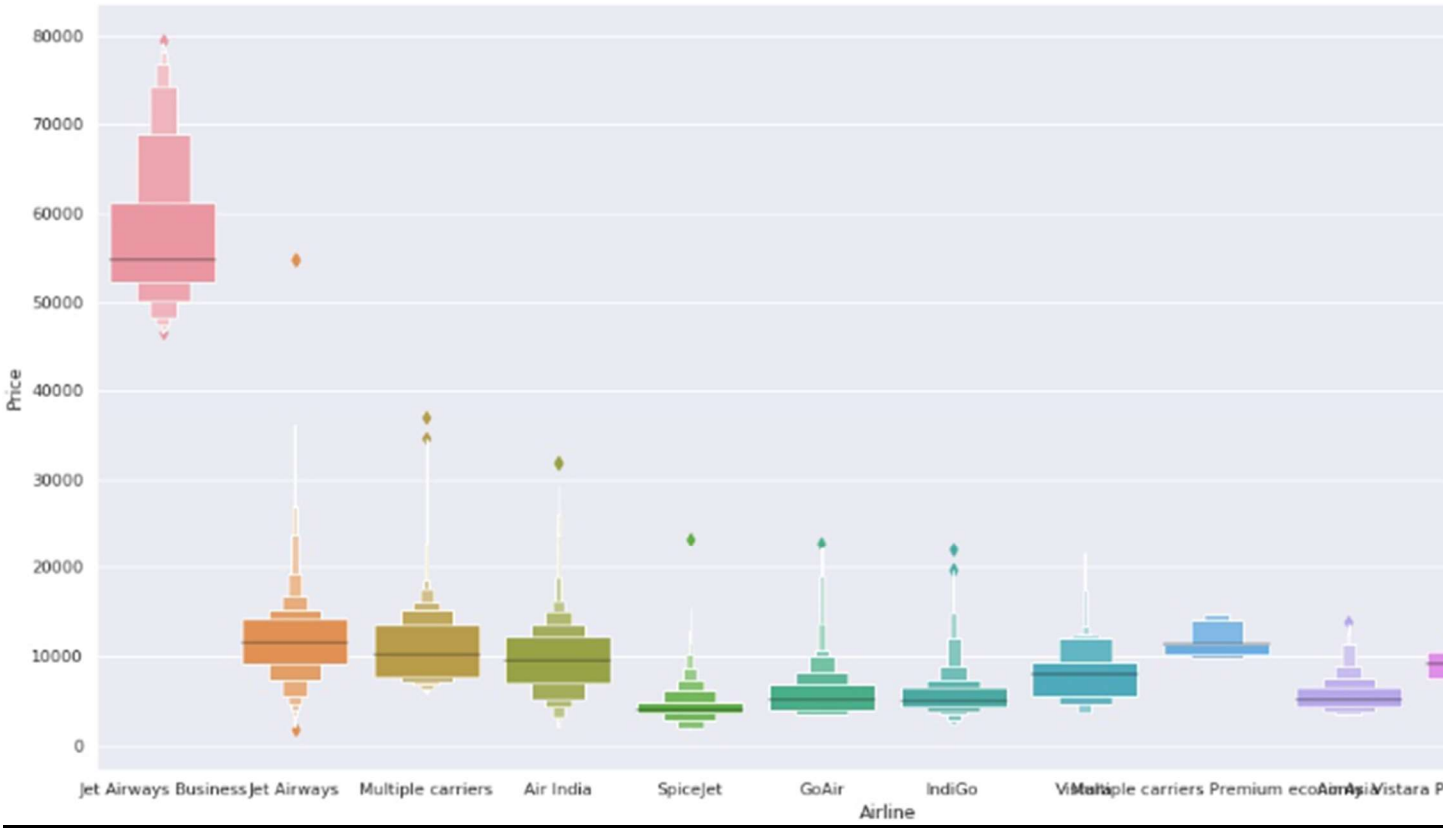
https://www.zyte.com/learn/web-scraping-best-practices/ → Ethical usage using Web Scraping best practices.

https://www.iwebscraping.com/how-python-and-selenium-are-used-to-scrape-flight-prices.php --> rendition of actual web scraping application.

https://www.todaysoftmag.com/article/3288/ethical-and-legal-considerations-in-web-scraping-at-scale --> Ethical challenges in Web Scraping understood.

https://www.youtube.com/watch?v=nN0OD6HLDJk → Video on Web Scraping and how It works

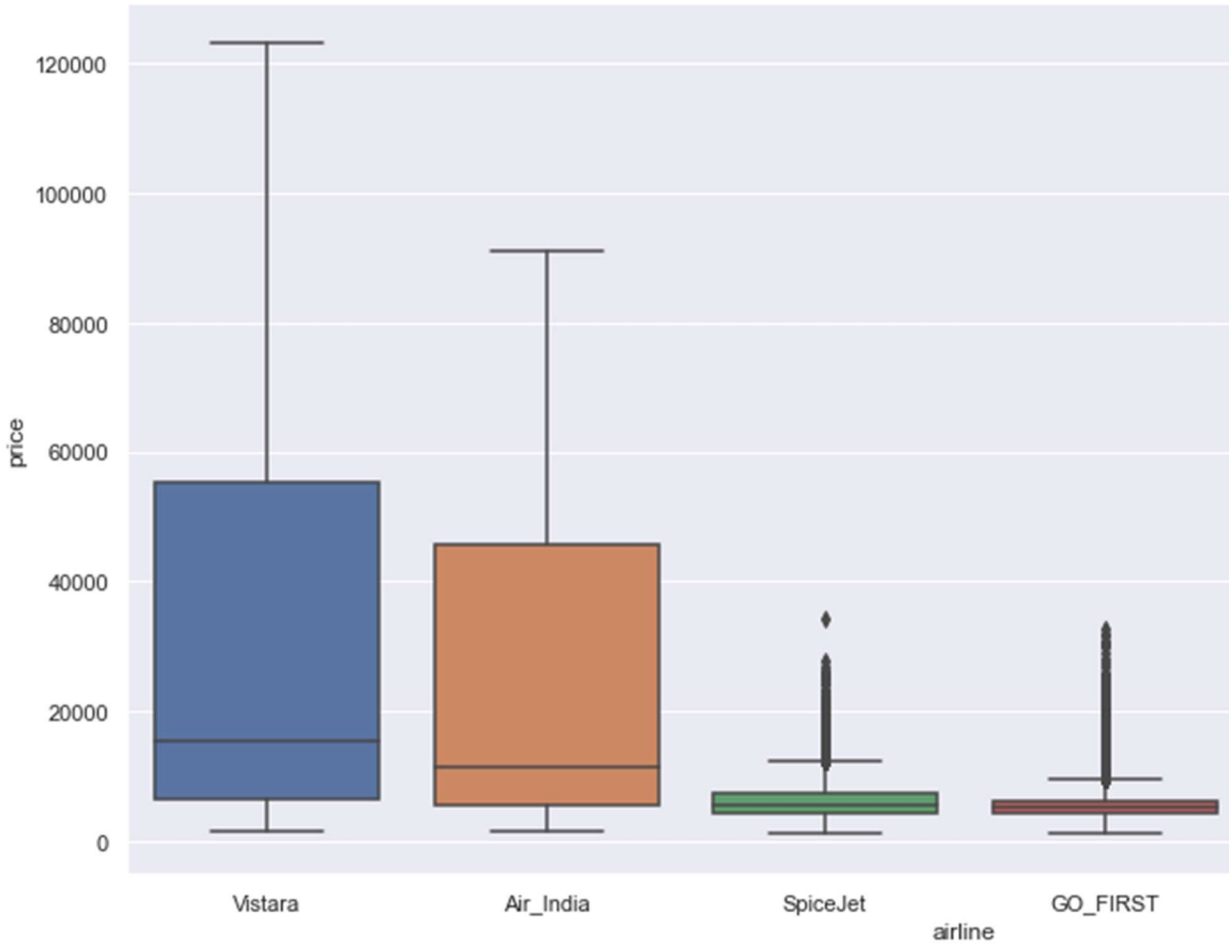
Appendix :-


80000
70000
60000
50000
40000
30000
20000
10000
0
Price
Jet Airways Business
Jet Airways
Multiple carriers
Air India
SpiceJet
GoAir
IndiGo
Multiple carriers Premium eco
Vistara P
Airline

In [13]:

```
1 plt.figure(figsize=(15,8))
2 sns.boxplot(x='airline',y='price',data=df.sort_values('price',ascending=Fals
```

Out[13]: <AxesSubplot:xlabel='airline', ylabel='price'>



In [15]:

```
1 plt.figure(figsize=(15,15))
2 sns.catplot(x='source_city',y='price',data=df.sort_values('price',ascendin
```

Out[15]:

```
<seaborn.axisgrid.FacetGrid at 0x15a474fb940>

<Figure size 1080x1080 with 0 Axes>
```