Introduction

Azure

Microsoft Azure, originally known as Windows Azure, is Microsoft's public cloud computing platform. It provides a wide range of cloud services, including computation, analytics, storage, and networking. Users can select from these services while building and growing new apps or using the public cloud to run already-existing apps.

All of the services made available through the Azure website are accessible to customers that have an Azure subscription. Subscribers can create cloud-based resources like virtual machines (VMs) and databases using these services. We use Azure as a cloud storage for this evaluation work, where we dump all of the data, which consists of Parquet Files. We also copy data from Azure to Databricks and vice versa.

Databricks

Databricks is a single, cloud-based platform that can meet all of your data demands also means that your complete data team can work together there. In addition to unifying and streamlining your data systems, Databricks is quick, affordable, and naturally scales to very massive volumes. Whether it's Amazon Web Services (AWS), Microsoft Azure, Google Cloud, or even a multi-cloud combination of those, Databricks is available on top of your current cloud.

In order to analyze data that arrives in pre-scheduled batches or real-time streams, many organizations presently run a complicated combination of data lakes and data warehouses with parallel "pipelines." They then add several more tools for analytics, business intelligence, or data science on top of that. You no longer need all of that with Databricks. Simply employ Databricks. With the aid of Databricks, you can:

- Gather all of your data in one location;
- Easily manage both batched data and real-time data streams.

Data can be transformed, organised, calculated on, queried, analysed, and used for machine learning and artificial intelligence. Reports can then be produced to show the results to your company.

Databricks is used by big businesses, little businesses, and everything in between. Databricks is used by some of the most well-known businesses in Australia and around the world, including Coles, Shell, Microsoft, Atlassian, Apple, Disney, and HSBC.

Introduction to the Dataset

Since 1971, the New York City Taxi and Limousine Commission (TLC) has been in charge of issuing licences and enforcing regulations for the city's taxi cabs. TLC made millions of trip data from both yellow and green taxi cabs available to the public. Each entry contains fields that record the pick-up and drop-off times and locations, trip distances, itemised rates, rate kinds, payment methods, and driver-reported passenger counts. Yellow taxi cabs are authorised to pick up passengers who hail them on the street anywhere in New York. To enhance taxi service and availability in the boroughs, green cabs were introduced only recently in August 2013.

Part 1: Data Cleaning and setup

The information was obtained and uploaded to the Azure Cloud Platform from the New York City government's official website. After setting up the link between Azure and Databricks. We then turned the Yelow taxi’s April 2022 Parquet File to CSV and loaded it into Azure.

Parquets File have multiple benefits over CSV file

- Firstly, the size of parquet files is very less compared to CSV ones. The April-2022 parquet file had a file size of around 50 Mb. Whereas the CSV version of it had a file size of about 300 MB. Hence making parquet files very economical, as they occupy very less hard disk space, hence they will cost less
- Secondly, Parquet files seem to be very secure, as CSV files are easily accessible and anyone can open and see or read the data.
- Parquet files have a faster query run time compared to CSV

Moreover, upon counting we found that total number of rows or records for green taxis were 9,390,483 and total count of rows of yellow taxis stood at 152,823,008.

We then began cleaning up the data, eliminating any entries where a trip finished before its starting time and we also removed any trips with negative speeds, as there was no speed variable. Speed was determined using the formula Speed=Distance/Time.

We also excluded trips that travelled at extremely high speeds because most streets in NYC have a 25 mph speed limit. therefore, we removed any rows that had a speed greater than that (0.007 miles per second). We eliminated trips that had too high or too low travel time, we eliminated trips with travel time less than 30 seconds and greater than 1 hour. Similarly, we also eliminated trips with too high or too low travel distance. As we are not allowed to remove missing values, we imputed them.

Both the cleaned yellow and green taxi dataset were combined and was mounted back on Azure. Data was copied from Azure into Databricks, where we ran some queries on it to answer the business questions.

Part 2: Answering Business Questions

Question 1: Calculate the following information for each year and month:

a) Total number of Trips made in each year and month
In Table A, total number of journeys in each month and year are listed.

b) Which day In the week had the most trips
In Table A, the day of the week with most trips has been listed.

c) Which hour had the most trips during the day?
In Table A, the hour that had the most trips is listed.

d) How many passengers were there on average?
In Table A, the Average number of passengers for that month and year is shown.

e) What was the average amount paid per trip?
In Table A, the average amount per trip is mentioned

f) Average fee paid by a passenger?
in Table A, the average amount paid is mentioned

Year	Month	Total Number of trips Q1.A	Day with most Trips Q1.B	Hour with most trips Q1.C	Average passenger count Q1.D	Average amount per trip Q1.E	Average amount per passenger Q1.F
2019	1	2820308	Wednesday	18	1.55	20.48	13.42
2019	2	2695387	Thursday	18	1.55	23.31	15.27
2019	3	3068823	Thursday	21	1.55	23.34	15.22
2019	4	2919659	Monday	22	1.56	23.45	15.31
2019	5	2953592	Wednesday	21	1.56	23.71	15.45
2019	6	2745934	Friday	22	1.55	23.54	15.34
2019	7	2471711	Tuesday	21	1.56	23.15	15.07
2019	8	2417467	Wednesday	21	1.57	23.21	15.12
2019	9	2559996	Saturday	21	1.54	23.67	15.68
2019	10	2769000	Wednesday	21	1.53	23.83	16.01
2019	11	2585809	Friday	21	1.54	23.51	15.69
2019	12	2582923	Monday	21	1.55	23.63	15.77
2020	1	2385439	Thursday	18	1.51	22.59	15.64
2020	2	2345298	Friday	18	1.5	22.87	15.73
2020	3	1177156	Monday	18	1.46	22.49	16.02

2020	4	100408	Tuesday	17	1.28	18.98	17.28
2020	5	161137	Thursday	16	1.31	20.04	20.61
2020	6	241551	Monday	17	1.35	21.42	18.66
2020	7	337990	Wednesday	17	1.38	21.91	18.42
2020	8	416274	Sunday	15	1.41	21.84	17.79
2020	9	551559	Tuesday	17	1.42	21.71	17.50
2020	10	670519	Wednesday	18	1.43	21.76	17.26
2020	11	600158	Sunday	14	1.41	21.63	17.42
2020	12	559328	Monday	15	1.42	21.70	17.56
2021	1	531463	Thursday	15	1.4	21.17	17.37
2021	2	537799	Thursday	18	1.41	21.66	17.54
2021	3	768175	Tuesday	17	1.4	21.65	17.49
2021	4	873376	Thursday	18	1.41	22.46	17.80
2021	5	1017687	Friday	18	1.42	22.14	16.99
2021	6	1163562	Tuesday	18	1.44	22.61	16.88
2021	7	1176542	Wednesday	18	1.47	22.44	16.53
2021	8	1166158	Monday	18	1.44	22.50	16.92
2021	9	1253457	Wednesday	18	1.43	23.95	18.16
2021	10	1446669	Friday	18	1.43	23.45	17.60
2021	11	1415785	Monday	18	1.42	23.64	17.72
2021	12	1294554	Wednesday	18	1.44	23.55	17.25
2022	1	927176	Friday	18	1.39	21.95	16.52
2022	2	1179399	Friday	18	1.39	22.54	17.07
2022	3	1455069	Wednesday	18	1.39	23.16	17.47
2022	4	1493546	Friday	18	1.41	23.42	17.44

TABLE A

Question 2- Calculate the following For Green and Yellow Taxi:

A. What were the avg, median, min, and max trip duration in minutes?
   Table B shows us these details. The trip's duration had to be divided by 60 because it was recorded in seconds.

B. What were avg, median, min, and max trip distance in Kilometers?
   Table B shows us these details. Trip distance variable had to multiplied by 1.6 because the data was measured in miles.

C. In kilometres per hour, what were the avg, median, min, and max speeds?
   Table B shows us these details. The time in seconds has been divided by 3600 seconds, and the distance in miles has been multiplied by 1.6.

	Taxi Type	**Average**	**Maximum**	**Minimum**	**Median**
Trip Duration	Yellow	19.81	59.98	4.77	17.85
	Green	21.48	59.98	4.77	19
Trip Distance	Yellow	5.93	19.2	3.2	4.8
	Green	6.68	19.2	3.2	3
Trip Speed	Yellow	19.2	40.27	17.95	16.36
	Green	19.2	40.27	18.67	17.96

TABLE B

Question 3 - What percentage of trips where the driver got a tip?

The percent of trips were driver got tips is 69.96%

Question 4 - What percentage of trips when tips were given to the driver resulted in tips of at least $10?

0.857% percent of tips were at least $10.

Question 5 - Organizing the trip duration into bins, then answering the to the following questions:

(a): Each bin's average speed
(b) per dollar average distance

Bin conversion was done for the trips durations. Table C below contains information on the Average speed in km/hr and distance per dollar.

Trip Duration Bins	Average speed in Km/hr	Average distance per dollar$
Under 5 minutes	39.21	0.24
Between 5 min – 10 min	23.99	0.23
Between 10 min – 20 min	17.74	0.23
Between 20 min – 30 min	17.87	0.27
Between 30 min -60 min	17.82	0.29

TABLE C

Question 6 - Which bin duration should a taxi driver choose to maximise his earnings?

A driver should aim for journeys that are under 5 minutes since the distance per dollar is lowest for shorter excursions. He will be making more money on shorter durations.

Part 3: Machine learning

To create the machine learning model i tried to use Linear Regression and Random Forest. Linear regression is a linear model, which means it works really nicely when the data has a linear shape. But, when the data has a non-linear shape, then a linear model cannot capture the non-linear features.

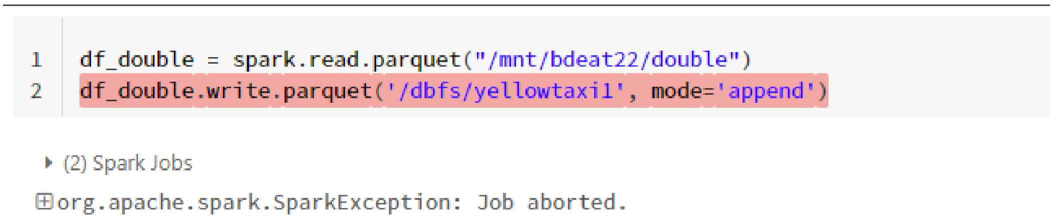
So, in this case, you can use the decision trees, which do a better job at capturing the non-linearity in the data by dividing the space into smaller sub-spaces depending on the questions asked. Both classification and regression tasks can be accomplished by Random Forest. A random forest generates accurate predictions that are simple to comprehend. Large datasets can be handled effectively. In comparison to the decision tree method, the random forest algorithm offers a higher level of accuracy in outcome prediction.

The dataset combines several numerical values with categorical values. The total fee is significantly influenced by categorical parameters such as pickup & drop-off location IDs, RateCodeID, & Payment Type. The total fee is greatly influenced by numerical values such as journey distance & trip time. Random Forest gave the RMSE score of 2.)9

Challenges faced

I faced multiple challenges in doing this assessment

- Databricks used to shutdown after 2 hours of idling so the progress had to be continually saved.
- Despite of using parquet files the processing time was a lot, moreover the Machine Learning training was a very big challenge.
- As there was an issue with ‘airport fee’ variable we were told to load yellow taxi’s data into two folders and we were told to append the variables format. But whenever I tried to do thI face the Error below:

  ```
  df_double = spark.read.parquet("/mnt/bdeat22/double")
  df_double.write.parquet('/dbfs/yellowtaxi1', mode='append')
  ```

  ```
  ⊞org.apache.spark.SparkException: Job aborted.
  ```

  I tried to look for solution everywhere, namely stack overflow, github and YouTube. But I continued to face this error. I tried to reconfiguring everything, restarted the cluster, cleared browser cache, I even tried to use different accounts on DataBricks. I still don’t understand why did I face this error. Due to this error, I had to do take completely different approach to answer the business questions.
- I faced the same error ‘Job Aborted’ multiple times. I believe this error used to come up on any command that required very heavy processing. Every so often, this error popped up on some commands like Count, ML dataset training or on the Business question queries. Note: this error used to not always come up on the same commands. It used to randomly pop up on the commands mentioned above.