|  |  |
| --- | --- |
| ***CHALLENGES AND BUSINESS RECOMMENDATIONS IN AVIATION INDUSTRY*** | A close up of a sign  Description automatically generated  Praveen Kumar Marichamy, SaiKarthik Kandikonda  Big-Data Analytics |

A close up of a sign

Description automatically generated

***Introduction:***

In this fast and ever-changing world, everyone knows the value of time and money moreover people started travelling from one place to another for various reasons like business, leisure, sports etc., In this case analysing the past airline services data became in-evitable for most of the companies to give better services for the people in terms of no delay in operating the airlines, providing best price for all the various class passengers and overall satisfaction of the customers as well. Lot of countries trying to build their airports as a hub for transits to attract people for tourism also. In bigger picture it indeed helps to boost the country’s economy as well. Countries such as Singapore and Dubai are the most important players in terms of airline industry and tourism related services.

***Motivation for Choosing the Project:***

Myself (Praveen) worked in a logistics software-based domain as a performance and test engineering analyst where I dealt with clients like Virgin Atlantic and KLM in Airline Industry as well as Maersk in shipping Industry. Karthik has worked as a data analyst and Business intelligence strategist for Changi Singapore airport. His project was based on analyzing the customer satisfaction index, overall rating and comparing the performances across all the employees working in the airport.

Both were pretty familiar with the airline industry domain and we know it’s one of the un-avoidable and profitable industries in the world. According to the current scenario we can even split the business into two categories post covid and pre covid. We are expecting many business outcomes like merger of small airlines, fewer number of travelers across the world, flight price changes etc.,

***Research problem and Dataset Details:***

Internet provides us a lot of datasets for airline domain either structured or semi-structured. We have chosen Airline on time performance dataset from[***http://stat-computing.org/dataexpo/2009/***](http://stat-computing.org/dataexpo/2009/) a collection of all the logs of domestic flights from the period of October 1987 to April 2008 in the United States of America. The records represent the individual flights where various flight related details such as the time and the date of arrival were mentioned. Moreover, the time taken for the taxi to runway were provided in additional.

*We are going to address some of the inferences from the dataset as well as the recommendations for the passengers to choose the flights wisely.*

1. What would be the best day of the week the day the passenger can choose to fly so that he/she can avoid delays in their journeys?

2. Predicting the top 20 busiest routes in USA and the number of flights travelled in a year

3. To predict the peak hours of the particular airport which will help the individual to book the flight on non-peak hours of the day.

4. Predicting the best Airline one can choose between the 2 Places based on the past data. This will help the passenger to get rid of delays and last-minute flight cancellations.

***List of Activities***

1. Data Gathering
2. Defining Problem Statement
3. Architecture and Data Pipeline
4. Establishing connections across various services
5. Loading the data and designing the schema
6. Performing Analysis
7. Plotting appropriate graphs
8. Storing the data back to No-Sql DB from Spark
9. Report and PPT

|  |  |
| --- | --- |
| ***Activities*** | ***Contributors*** |
| Data Gathering | SaiKarthik |
| Defining Problem Statement | Praveen |
| Architecture and Data Pipeline | Praveen, SaiKarthik |
| Establishing connections across various services | Praveen, SaiKarthik |
| Loading the data and designing the schema | Praveen, SaiKarthik |
| Performing Analysis | Praveen(2 ProblemStatements) Karthik (2 Problem Statements) |
| Plotting appropriate graphs | Praveen(2 ProblemStatements) Karthik (2 Problem Statements) |
| Storing the data back to No-Sql DB from Spark | SaiKarthik |
| Report and PPT | Praveen |

***Approach Overview:***

* As the data being big enough and semi-structured, we choose the MongoDB as the source for storing the output from the PySpark data frame.
* The initial approach being the data from the data source (***dataexpo 2009***) is sent to the spark RDD container using pyspark using ipython Jupyter notebook.
* Data being analyzed by using spark sqlcontext due to the variety of operations can be carried out inside the data.
* Once the data was cleaned and transformed, we have sent the data in to the MongoDB to store and can view the result for the future.

*\*Firstly, this approach was carried out in the Horton works hdp using oracle virtual machine but due to the limitations of RAM and ROM occupied during the process made us to install the software’s in our local system and interconnected them accordingly.*

***Architecture Diagram for Airline Project:***



\**Diagram has drawn using* ***drawio*** *architecture diagram software*

***Pyspark:***

Spark is widely used as a computing engine after MapReduce framework. The main advantage of using the spark it supports stream processing whereas Hadoop supports batch processing. The default language of spark is Scala and in our project, we have installed pyspark on top of spark for programming.

1. Pyspark is a great language for performing exploratory data analysis and building machine learning pipelines too.
2. Easy to integrate and work with RDD’s in python programming language.

***PySpark SQL Context:***

SQL Context is used to enable and run applications using SQL queries and running SQL functions and returns all the outputs/ results as a data frame.

1. Spark SQL is a spark module for structured data processing and also acts as a distributed SQL query engine.

***Spark and Mongo Installed in HDP:***

A screenshot of a computer

Description automatically generated

\**As mentioned, due to the limitations we switched back to our local system and software’s for analyzing the data*

***MongoDB and Server Screenshot:***

***A screenshot of a cell phone

Description automatically generated***

***A screenshot of a cell phone

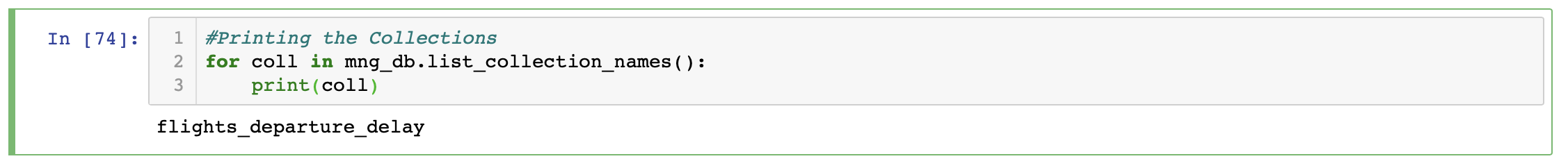
Description automatically generated***

***MongoDB Connector in Ipython Notebook:***

***A screenshot of a cell phone

Description automatically generated***

******

******

* Our Database dataexpo was created successfully.
* Our Database collection flights\_departure\_delay was successfully created.

***Spark Installation and Pyspark Shell:***

***A screenshot of a cell phone

Description automatically generated*** ***A screenshot of a cell phone

Description automatically generated***

***Spark, Pyspark Connector and SQLContext in Ipython Notebook:***

***A screenshot of a social media post

Description automatically generated***

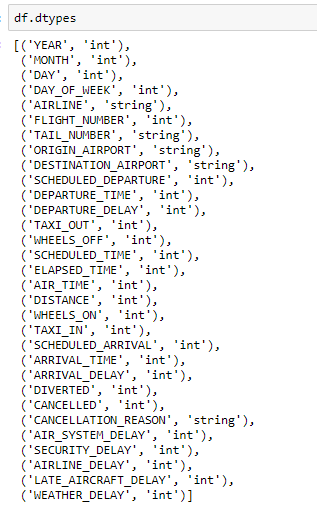
***Business Recommendations and Inferences from the Dataset:***

1. *Sample head of airline dataset with columns and rows printed.*

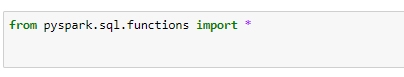
*A screenshot of a social media post

Description automatically generated*

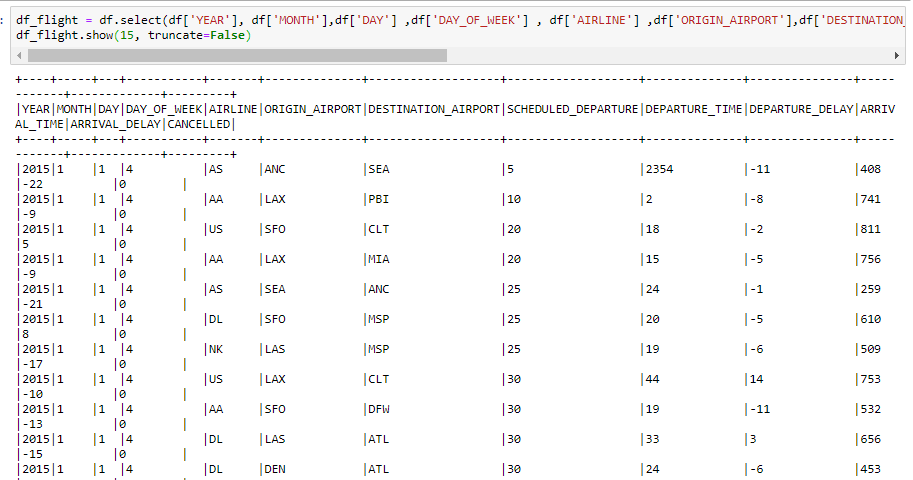
*Next we tried to analyze the data types of all the variables in the dataset*



*We need to import pyspark sql package in order to perform operations in PYSPARK SQL.*



*Created a new Pyspark data frame with all the required columns so that we can reduce the unnecessary variables.*

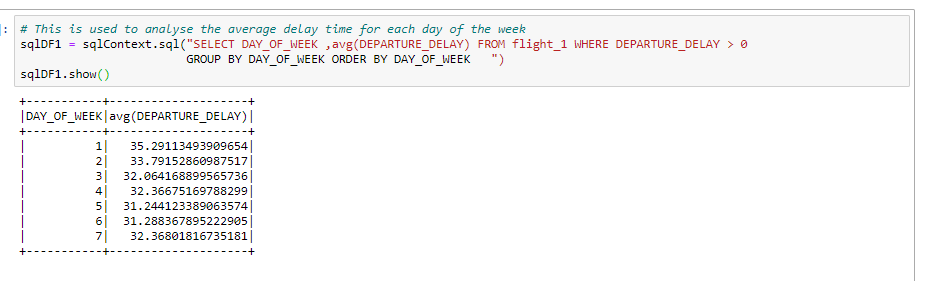


*In order to perform Spark Sql queries we need a Table upon which we can perform analysis, data frames can not be used to perform analysis directly.so we have created a temporary view upon which we can perform analysis.*

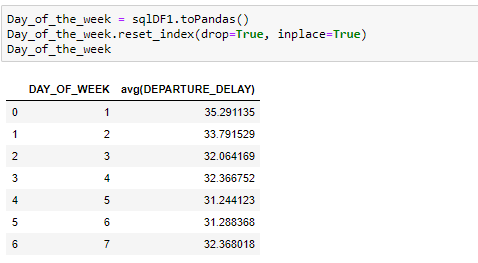


***Analysis 1:***

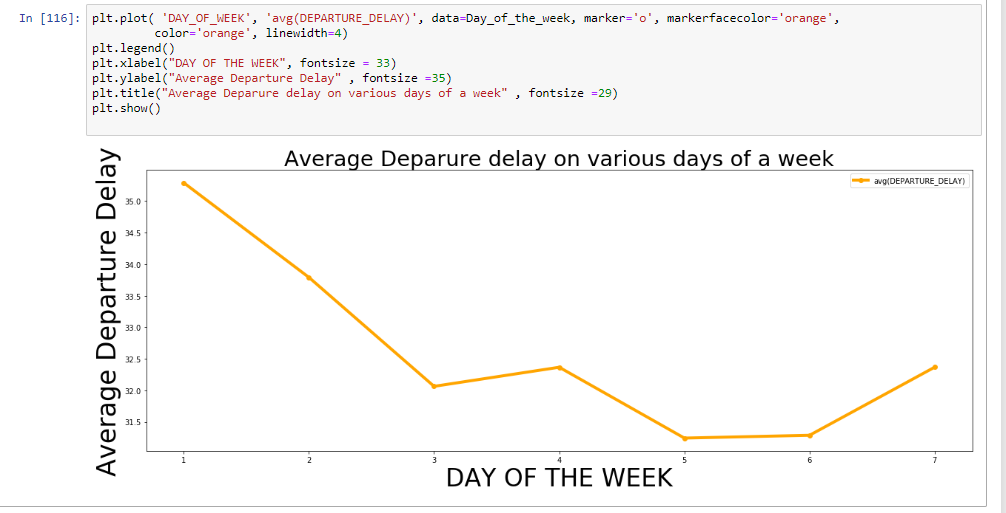
***What would be the best day of the week the day the passenger can choose to fly so that he/she can avoid delays in their journeys?***



*As shown in the above image the output data is not in the much presentable way, we can not plots graphs directly on the spark data frame, in order to overcome this we have converted it to a pandas data frame as shown in the figure below.*



***Graph:***

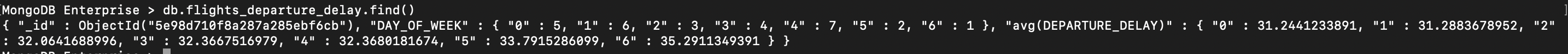


***Inference:*** *From the above graph we can clearly say that the average flight delays are more on day 1, 2 of the week.*

***Inserting the data into MongoDB:***

***A screenshot of a cell phone

Description automatically generated***

******

***Analysis 2:***

***Analysing the most busiest routes in USA***

# 

# *Inference: From the above grid we can capture the busiest routes and this will help us to arrange more number of flights in case required.*

# *MongoDB Insertion:*

# *A close up of a window Description automatically generated*

# *A screenshot of a social media post Description automatically generated*

# *Analysis 3:*

# *Analysing the best Airline one can choose based on the past data. This will help the passenger to get rid of delays and last minute flight cancellations.*

# 

# *Converting it to pandas data frame*

# 

# *Graph:*

# 

# *Inference : The above analysis is useful when someone wants to be strategic in terms of choosing a Airline to avoid last minute cancellations and delays in journey. This also helps the companies to improve their services based on the data, which will eventually help them in improving the business eventually.*

# *MongoDB Insertion:*

# *A screenshot of a social media post Description automatically generated*

# *A picture containing computer Description automatically generated*

# *Analysis 4:*

# *Analysis of which month has highest number of cancellations, which helps the passangers in planning their Journey accordingly*

# 

# *Converting the Spark DF to Pandas data frame for further analysis.*

# 

# 

# *Graph:*

# 

# *Inference: From the above graph we can infer that the probability of the flights getting cancel are more in the months of January, February.*

# *MongoDB Insertion:*

# *A screenshot of a social media post Description automatically generated*

# *A picture containing computer Description automatically generated*