Air Ticket Price Prediction

Using PySpark

Saiteja Namani

INTRODUCTION

In today's interconnected world, air travel has become an essential mode of transportation for both business and leisure travellers. The airline industry is a dynamic sector where ticket prices fluctuate based on various factors including Distance, Day of week, Month of travel, Holiday, Season, Demand, Promotional type, Fuel type, Number of stops. Predicting airfare can provide multiple benefits, ranging from helping travellers make cost-effective decisions to enabling airlines in revenue management.

This project aims to develop a model that can accurately predict air ticket prices. The primary objective is to provide insights into the factors influencing ticket prices and offer a tool that can aid in forecasting future airfares, ultimately assisting consumers.

DATA DESCRIPTION

The collection of data is the very first step in every project. There are various sources of data available on numerous websites, but we have gathered our data from a hackathon by ProjectPro. The dataset contains 45000 rows of data and 19 features. The features present in this dataset are Flight ID for unique flight identification, Airline name, cities of Departure and Arrival, along with Distance in kilometres. It also details Departure and Arrival times, Flight Duration, Aircraft Type, Number of Stops, and the Day of Week and Month of Travel. Additional factors such as Holiday Season, Demand, Weather Conditions, Passenger Count, Promotion Type, Fuel Price, and the Flight Price itself are provided, offering a comprehensive view of the variables influencing flight operations and pricing.

DATASET LINK : https://www.projectpro.io/hackathon/title/hackathon-datascience-regression-rewards-prize

EXPLORATORY DATA ANALYSIS

In the initial phase of my data exploration, we began by extracting the pertinent dataset and initializing a Spark session to facilitate the handling of large-scale data processing. Once the data was loaded into a dataframe, we have seen there are a significant number of null entries. Rather than eliminating these potentially insightful data points, we chose to categorize them under a 'missing' label, thus preserving the integrity of the. For numerical columns, the data distribution appeared to be normal so, we decided to impute the missing values with the mean to maintain the dataset's natural variance.

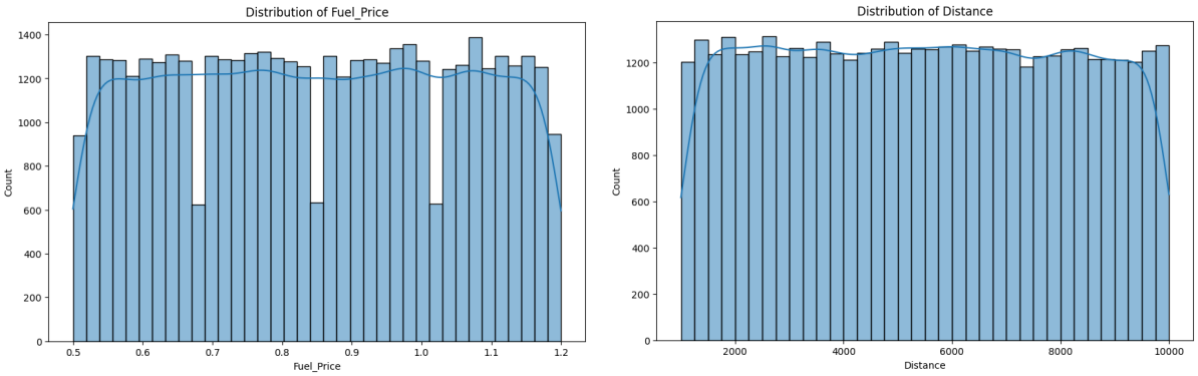

Fig 1: Distribution of Numerical features Fuel price and Distance

We Plotted a scatter plot for distance, fuel price with flight price and noticed that individually there is no relationship was founded So, we created a new column that the interaction between the distance of the flight and fuel price which gives us fuel efficiency of flight. This data produced a right-skewed distribution. To address this skewness and normalize the distribution, we applied a square root transformation to this newly created column.

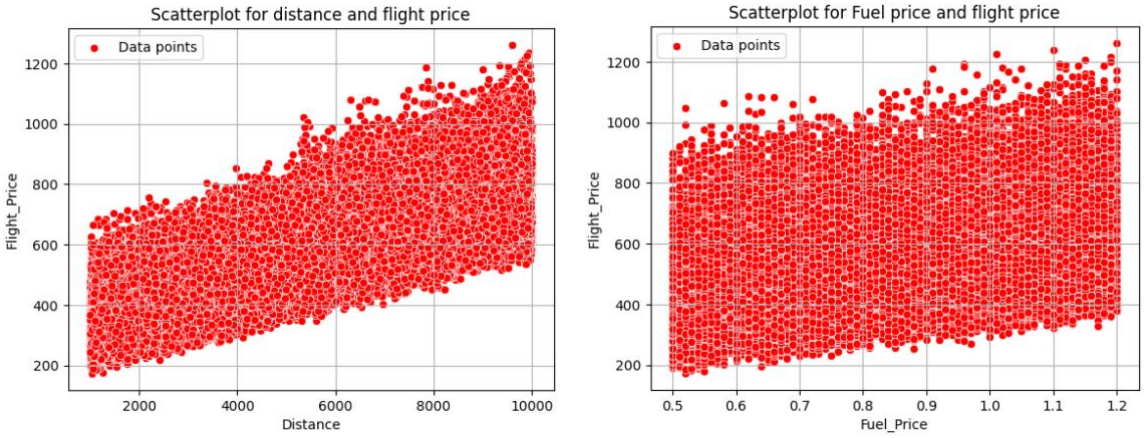

Fig 2: Scatter plots for Distance, Fuel Price with Flight Price

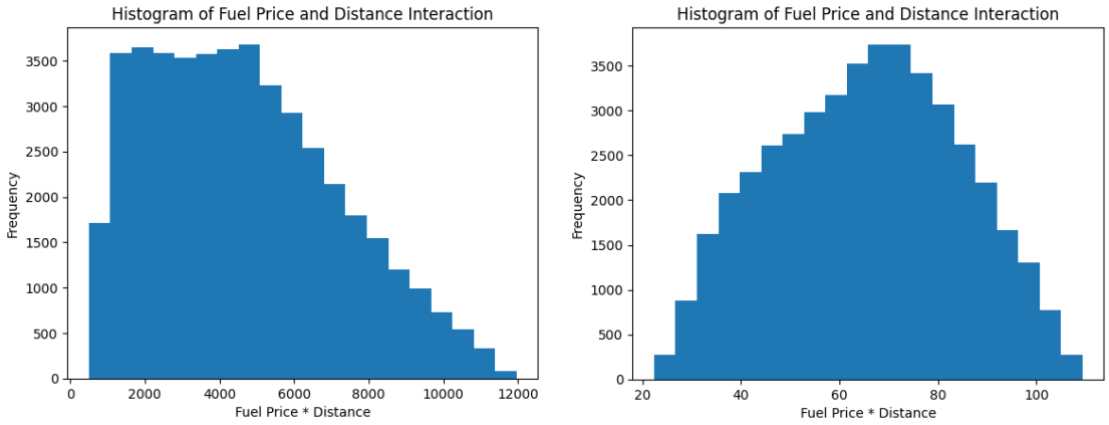

Fig 3: Histograms of Fuel price distance, on the left before transformation, on the right after square root transformation.

Additionally, recognizing the potential impact of the number of stops and flight duration, we created an additional column named 'flight category.' This column classifies flights into distinct categories based on the number of stops and the total duration of the flight. We also constructed some boxplots to check how the features impacts on flight price. The demand shows that higher the demand higher the flight price. In holidays it is seen larger flight prices are in summer. We have seen some interesting thing in weather conditions where cloudy, clear conditions are high compared to snow and rainy condition.

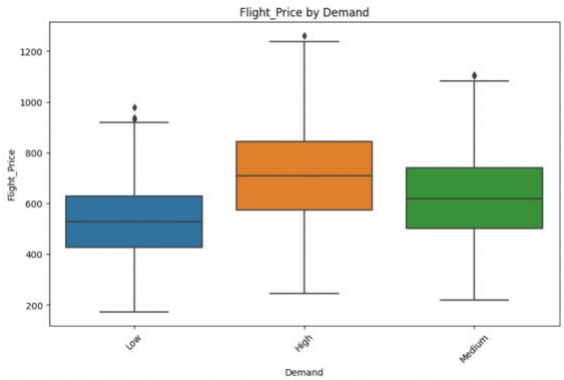

Fig 4: Boxplots of Demand.

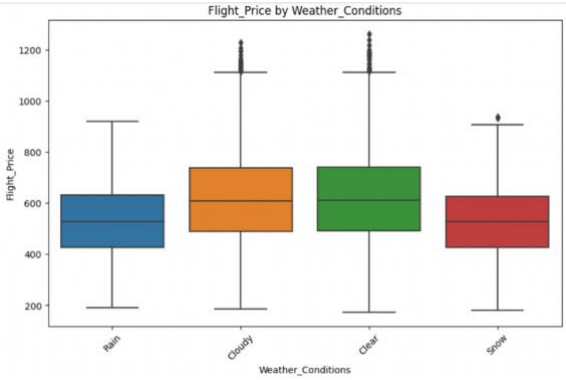

Fig 5: Boxplots of Weather Conditions.

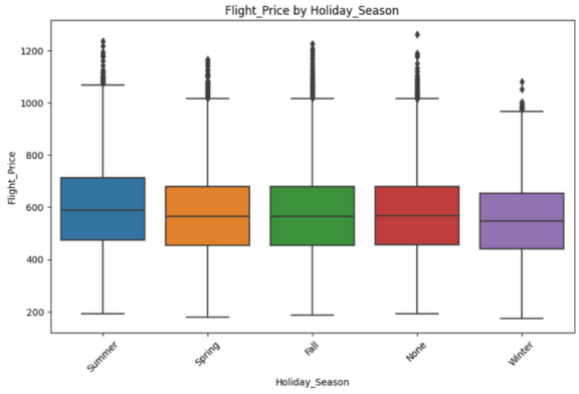

Fig 6: Boxplots of Holiday seasons.

DATA PREPARATION

Based on all Exploratory data analysis we created a new data frame with most significant features and this data frame consists of Demand, Aircraft Type, Day of Week, Month of Travel, Holiday Season, Weather Conditions, Promotion Type, Departure City, Arrival City, Fuel price distance and Flight Category. Now based on these features we predicted the flight price. Then categorical columns were processed through String Indexing and One Hot Encoding to convert them into numerical format suitable for machine learning algorithms, while the numerical 'sqrt_fuel_price_distance' was directly utilized. We did a Vector Assembler to merge these processed features into a single vector, which serves as the input for the model. The entire pre-processing workflow was structured into a Pipeline for

streamlined execution. We also joined the transformed data with the original flight prices, ensuring a coherent structure by synchronizing them with a monotonically increasing identifier and casting the target variable, 'Flight_Price', to a double type to align with model requirements. Then we performed machine learning regression algorithms to predict flight price.

MODELING

Linear Regression

Linear Regression is a fundamental algorithm in the machine learning, designed to understand and quantify the relationship between an output variable and one or more input variables. It accomplishes this through linear predictor functions, which form the basis for a model that can illustrate these relationships graphically with a straight line, hence the term 'linear' in its name. The prediction for linear regression looks like the below Fig 7.

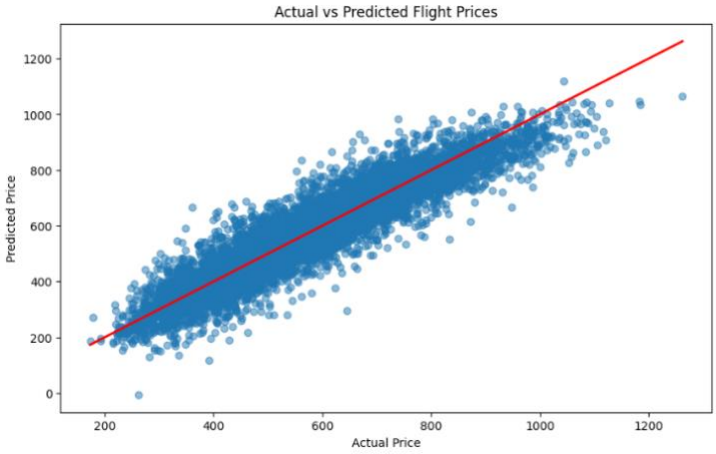

Fig 7: Linear regression predictions.

The above scatter plot shows a positive correlation between actual and predicted flight prices, with the line of best fit reflecting a generally accurate prediction model. Some variance is observed, particularly at higher price points, suggesting the model's precision may diminish with more expensive flights.

Decision Tree Regressor

A Decision Tree Regressor is an algorithm that predicts a target numerical value using a tree-like model of decisions. It segments the input space into distinct regions by recursively partitioning the data based on the input features that lead to the most significant reduction in variance for the target variable. The final prediction is made based on the mean value of the target variable in the terminal leaves of the tree. The model is represented as a series of decision rules branching out from the root to the leaves, where each leaf node corresponds to a numerical prediction. The predictions of decision tree regressor is in the below Fig 8.

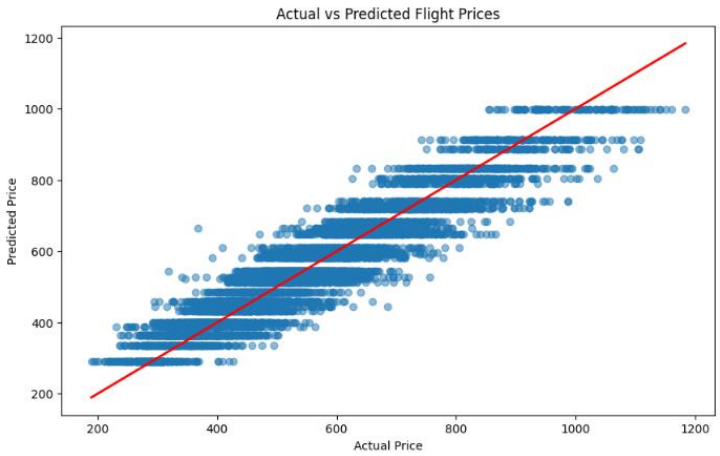

Fig 8: Decision Tree Regressor.

The scatter plot shows an overall positive trend, as denoted by the red line, suggests the model can predict flight prices with a degree of reliability. Nonetheless, the spread of predictions around the actual prices, particularly in the higher price segments, indicates inconsistencies that could be addressed to enhance the model's performance.

Gradient Boosting Regressor

Gradient Boosting Regressor operates on the principle of ensemble learning, where multiple weak predictive models (typically decision trees) are combined to form a strong predictor. This algorithm improves its predictions iteratively by optimizing an objective function. Each tree in the sequence is built to correct the errors made by the previous one, with predictions updated by adding the output of the new tree scaled by a learning rate. The final model compiles these trees to make predictions by summing their outputs, hence taking a 'boosted' gradient step towards reducing the model's error. The predictions of Gradient Boosting Regressor is in the below Fig 9.

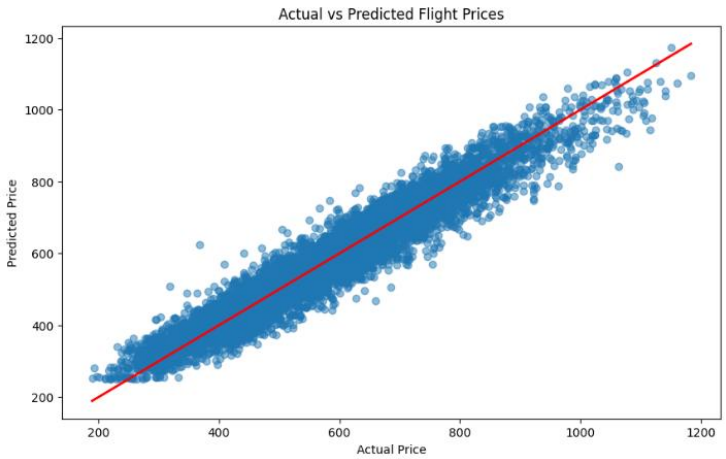

Fig 9: Gradient Boosting Regressor.

In the above scatter plot the prediction accuracy appears consistent across the range of prices, with a tighter clustering of points around the line at lower prices and slightly more dispersion at higher prices. This consistency suggests that the predictive model is robust and performs well across various price levels.

Factorization Machine Regressor

A Factorization Machine Regressor is a general-purpose supervised learning algorithm that captures interactions between features in a high-dimensional sparse dataset. It combines the advantages of support vector machines with factorization models, allowing it to estimate reliable parameters even with a vast feature space. The model is characterized by an equation that includes linear terms for each input feature, and factorized interaction terms that capture the influence of feature interactions on the output variable. It is particularly effective in scenarios like recommendation systems where the data has many categorical variables that can be encoded into binary features. The predictions of Factorization Machine Regressor is in the below figure.

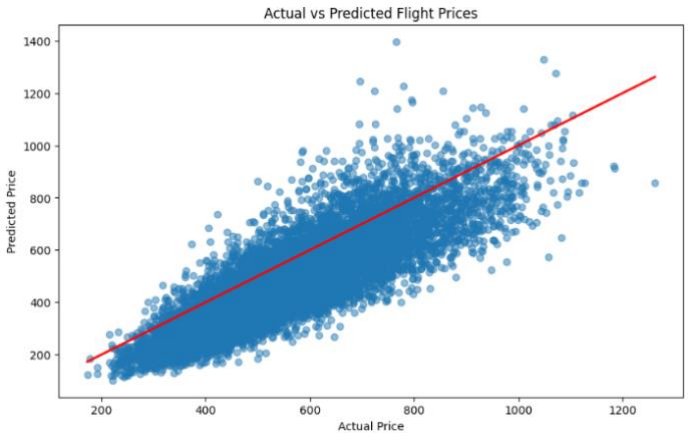

Fig 10: Factorization Machine Regressor.

While predictions largely correspond to actual prices, there is noticeable variability, especially in the mid to high price range. The density of points near the trendline suggests the model is more accurate for lower-priced flights, with increased prediction error as prices rise.

Isotonic Regression

Isotonic Regression is utilized when the model needs to predict a target variable where there is an assumed order or progression. It works by fitting a freeform line (a step function) to the variables in a way that the fitted values are non-decreasing along with the inputs. This method ensures that the model is monotonically increasing, aligning with the prior knowledge that the output should increase with the input. The final prediction is a piecewise constant function, where each segment is fit to parts of the data, maintaining the order. This model

didn’t work well with our problem compared to the rest of models and the predictions are as in the Fig 11 .

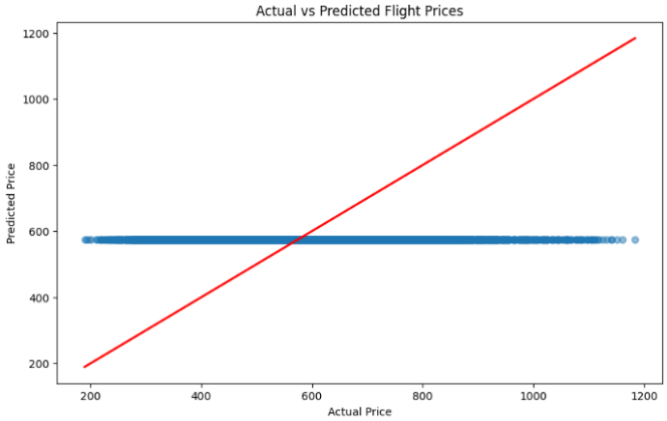

Fig 11: Isotonic Regression.

This pattern of predictions suggests a significant model limitation, failing to account for the variability in actual flight prices. The discrepancy between the predictions and the trendline implies a need for substantial model recalibration or revaluation of the prediction algorithm.

MODEL EVALUATION

RMSE

RMSE is a tool that helps in determining how accurately the model is making the predictions. It calculates how much error the model creates while making these predictions. It measures the standard of predictions. Mathematically, it is defined as the square root of the average of the squares of all the errors. Error is defined as the difference between the actual and predicted value. Less the RMSE, the better the performance of the model.

Machine learning model	**RMSE**
Linear Regression	44.17793352241557
Decision Tree Regressor	55.35188919971214
Gradient Boosting Regressor	40.10117003679778
Factorization Machine Regressor	124.65341639243252
Isotonic Regressor	163.0730391149828

Table 1 : RMSE comparison table

The evaluation of various machine learning models for flight price prediction reveals that the Gradient Boosting Regressor yielded the lowest RMSE of 40.10, indicating the highest predictive accuracy among the tested models. Linear Regression also performed well with an RMSE of 44.18, suggesting a competent baseline model.

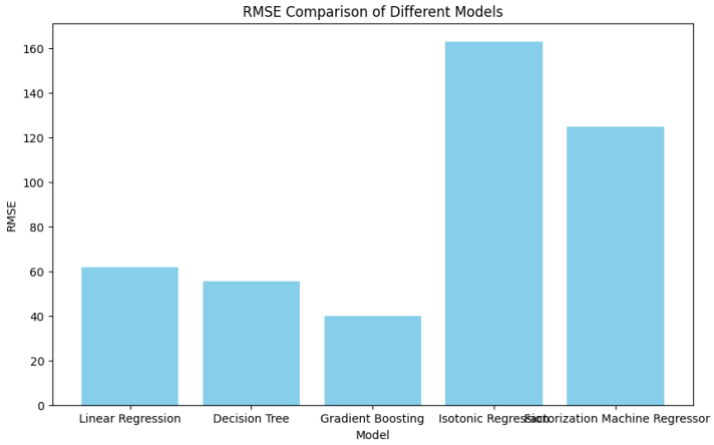

Fig 12: graph that shows the model performance.

CHALLENGES

In our project, we faced several challenges related to data handling, feature engineering, and computational limitations. Initially, we dealt with a significant amount of missing data; instead of eliminating these, we categorized them as 'missing', a strategy that helped maintain the dataset's completeness. Furthermore, we encountered an unclear relationship between the number of stops in a flight and its pricing. To clarify this, we considered the number of stops alongside flight distance, leading to the creation of a new 'Flight Category' feature that classifies flights as short haul, medium haul, or long haul. We also struggled to establish a direct relationship between flight distance, fuel price, and flight pricing. To address this, we introduced a 'Fuel Estimation for Total Distance' metric, calculated by multiplying the distance by the fuel price and basing it on a simplified consumption rate of 1 litre per kilometre. While this helped in our analysis, we were also constrained by computational power; for instance, running a decision tree regressor took nearly 190 minutes, which made it impractical to explore deep learning models. These computational limitations significantly impacted our ability to experiment with more complex algorithms, shaping the scope and methodology of our project.

CONCLUSION

In summary, this project set out to make airfare prices predictable and succeeded by identifying a model that forecasts with impressive accuracy. The Gradient Boosting Regressor proved especially effective, achieving our main objective of delivering a trustworthy forecasting tool. This work offers practical benefits, helping travellers plan cost-effectively and airlines to better understand pricing strategies. While there's potential for even more sophisticated models in the future, the project marks a significant step towards demystifying airfare pricing.

REFERENCES

[1] "Decision Tree Regression Model." *Apache Spark*, https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.ml.regression.DecisionTreeRegressionModel.html#decisiontreeregressionmodel

[2] "FM Regressor." *Apache Spark*, https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.ml.regression.FMRegressor.html#pyspark.ml.regression.FMRegressor

[3] "GBT Regressor." *Apache Spark*, https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.ml.regression.GBTRegressor.html#pyspark.ml.regression.GBTRegressor

[4] "Isotonic Regression Model." *Apache Spark*, https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.ml.regression.IsotonicRegressionModel.html#pyspark.ml.regression.IsotonicRegressionModel

[5] "Linear Regression." *Apache Spark*, https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.ml.regression.LinearRegression.html#pyspark.ml.regression.LinearRegression

[6] Theofis, Kalampokas, et al. " A Holistic Approach on Airfare Price Prediction Using Machine Learning Techniques." *APPLIED RESEARCH, vol. 11, May 2023, IEEE Access*, http://dx.doi.org/10.1109/ACCESS.2023.3274669.