# MSc Project Report

School of Engineering and Computer Science

University Of Hertfordshire

**Flight Price Prediction using machine learning based on key features such as Airline, flight, Source City, Departure Time, Stops, Arrival Time and Duration.**

Report By

Vaishnavi Ajay Kadam

Student ID

21041993

Supervisor

Minghua Zheng

# MSc Final Project Declaration

This report is submitted in partial fulfilment of the requirement for the degree of Master of Science in Artificial Intelligence and Robotics at University of Hertfordshire (UH).

It is my own work except were indicated in the report.

I did not use human participants in my MSc Project.

I hereby give permission for the report to be made available on the university website provided the source is acknowledged.

Abstract

In the realm of modern global connectivity, the aviation industry is a cornerstone, continually striving for efficiency and effectiveness in flight services. A pivotal component of this industry is the dynamic pricing of airline tickets, which can undergo substantial fluctuations due to various factors like flight schedules, routes, duration, and special occasions such as holidays. For travellers, gaining insights into flight prices prior to embarking on a journey holds the potential for significant cost savings and enhanced travel planning.

In response to this demand, the Flight Price Prediction Using Multiple ML Models project aims to elevate the precision of flight cost predictions through the application of advanced machine learning techniques. Leveraging historical flight data and sophisticated modelling algorithms, this initiative seeks to empower travellers with the knowledge necessary for making informed decisions regarding flight bookings, ultimately optimising their travel experiences.

This comprehensive endeavour involves the meticulous evaluation of five distinct machine learning models: Decision Tree Regressor, Linear Regression, Bagging Regressor, Random Forest Regressor, and XGB Regressor. Each model undergoes rigorous assessment employing an array of metrics including Mean Absolute Error (MAE), Percent Absolute Error (MAPE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R). These metrics collectively provide a holistic view of the models' predictive capabilities and overall performance.

The outcomes of this project are poised to deliver invaluable insights into the selection of the most suitable model for forecasting flight costs. Furthermore, these results unveil the intricacies of employing machine learning within the tourism industry, laying the groundwork for more precise and efficient travel planning and decision-making. As the aviation sector continues to expand, the ability to harness data-driven predictions becomes increasingly essential, and this project stands as a significant contribution to the realms of tourism and transportation.

Acknowledgements

Behind the success of any good work, there will be contribution of many who have rendered their services through useful timely suggestions either directly or indirectly. It is my bounded responsibility to convey my word of thanks to all those in a nutshell.

I am highly grateful to Mr. Minghua Zheng for his support, encouragement and valuable suggestion received throughout my research work. I also thank all my friends and well-wishers, who are directly or indirectly involved, for their inspiration, encouragement and support throughout.

Table of Contents

# MSc Final Project Declaration 2

# Abstract 3

# Acknowledgements 4

# Chapter 1: Introduction 7

## Primary Factor Behind Flight Price Fluctuations 7

## Background Research 7

## Problem Statement 8

## Aims and Objectives 9

## Research Questions 9

## Legal and Ethical Issues 9

# Chapter 2: Literature Review 11

# Chapter 3: Methodology 15

## Linear Regression 15

## Decision Tree Regression 15

## Random Forest Regressor 16

## Bagging regressor 17

## XGB Regressor 18

# Chapter 4: Data Analysis and Evaluation 19

## Dataset Description 19

## Importing The Dataset 20

## Data Pre-Processing 20

## Data Cleaning 21

## Data Transformation 21

## Feature Engineering 22

## Data Splitting 24

## Data Normalization 25

## Data Metrics used For Evaluation 25

## Mean Absolute Error 25

## Mean Absolute Percentage Error 26

## Mean Squared Error 26

## Root Mean Square Error 27

## R-Squared Value 27

# Chapter 5: Result 28

## Analysis 28

## Conclusion 30

Chapter 6: Future Enhancement 31

References 33

Code 35

# **Chapter 1: Introduction**

The cost of air travel is shaped by an intricate interplay of diverse elements, producing a pricing landscape that is both dynamic and occasionally uncertain.

The interplay of supply and demand holds a crucial role in determining flight prices. On highly sought-after routes or during peak travel periods, increased demand often results in higher fares. Conversely, travellers might find lower fares during off-peak times or on less popular routes. Airlines employ strategic revenue management techniques to optimize pricing and seat allocation in response to these demand patterns.

The timing of booking significantly influences prices. Closer to the departure date, ticket costs tend to rise due to limited seat availability, motivating passengers to secure their seats well in advance. However, there are instances where last-minute deals can surface if airlines have unsold seats.

Fuel costs wield a substantial impact on airline expenditures, leading fluctuations in oil prices to directly affect ticket prices. Additional determinants such as operational expenses, aircraft upkeep, labour charges, competitive pressures, and even geopolitical events collectively impact pricing decisions.

Airlines frequently utilize dynamic pricing algorithms that factor in real-time data and customer behaviour to adjust fares. The inclusion of personalization, loyalty programs, and bundled offerings further intricately shape the pricing structure.

Grasping these multifaceted components empowers travellers to navigate the intricate realm of flight pricing, enabling them to make well-informed choices about when and how to book flights for the most advantageous deals.

# 1.1 Primary Factor Behind Flight Price Fluctuations

The primary driver behind changes in flight prices lies in the intricate balance between traveller demand and the supply of available seats. Variations in passenger interest prompt airlines to raise prices during periods of high demand, while decreased demand results in price reductions. Other influential factors encompass seasonal patterns, the dynamics of early booking, shifts in fuel costs, operational expenditures, competitive influences, and external events such as geopolitical disturbances. This complex interplay of elements continually shapes the fluid panorama of flight prices.

# Background Research

Conducting background research for flight price prediction constitutes a foundational stride in comprehending the intricate factors governing airfare fluctuations and in crafting precise forecasting models. This research encompasses a multitude of dimensions contributing to the multifaceted nature of flight pricing.

To begin, historical data analysis assumes paramount importance. Researchers delve into prior airfare trends to decipher patterns and connections with variables like travel seasons, holidays, and economic circumstances. This historical perspective lends insight into the price responses under varied circumstances. Seasonal trends and the interplay between demand and supply equally occupy a pivotal role in this background research. Explore existing research to gain domain insight and identify key determinants of airfares, including elements such as seasonal patterns, booking times, market competition and customer preferences. Prepare data, create appropriate specifications, determine data validation, choose modelling methods, strategize for anomaly detection and validation, and keep ethical and regulatory considerations in mind. Use domain expertise, benchmarks against industry benchmarks, and comprehensive supporting documentation to build robust airline fare forecasting models and gain valuable insight into complex pricing dynamics within airlines. Scrutinising pricing alterations during peak and off-peak intervals aids in comprehending the influence of traveller demand. Moreover, investigating the repercussions of external occurrences, such as major sporting events or geopolitical perturbations, fosters an understanding of their potential to impact pricing unexpectedly.

Given the substantial role of fuel prices in airline operational costs, investigating trends in oil prices becomes indispensable. Additionally, delving into the competitive landscape proves pertinent, as airlines adjust pricing strategies based on their rivals' manoeuvres. The technical facets of prediction models equally constitute a facet of background research. Grasping the array of machine learning algorithms, time series analyses, and regression techniques employed in prediction aids in selecting the most fitting approach for precise forecasts.

The comprehensive background research for flight price prediction encompasses historical data analysis, the comprehension of seasonal and external influences, scrutiny of fuel price trends, exploration of competitive dynamics, and familiarity with prediction methodologies. Such research stands pivotal in constructing dependable models that assist travellers in prudent decision-making and aid airlines in optimising pricing tactics.

# Problem Statement

The challenge of creating accurate and reliable models that forecast airfare fluctuations. This problem stems from the intricate and multifaceted nature of flight pricing, influenced by a myriad of variables. The primary issue is the lack of a crystal-clear formula to predict price changes due to the dynamic interplay of demand, supply, and various external factors.

Existing prediction models often struggle to capture the complexity of real-world scenarios, leading to inconsistencies in forecasts. Moreover, the volatility of external factors such as fuel prices, geopolitical events, and economic fluctuations adds a layer of uncertainty that complicates accurate predictions.

The problem statement also acknowledges the significance of striking a balance between model complexity and efficiency. Developing models that accurately incorporate all influential factors while remaining manageable is a substantial challenge. The problem encompasses the need to address the temporal aspect of flight pricing – predicting not just the direction of price changes but also the timing of these changes. This involves understanding when to book flights to secure the best deals amidst constantly shifting variables.

# Aims and Objectives

The aims and objectives of flight price prediction delineate the purpose and direction of this field in the realm of air travel. The primary aim is to create robust predictive models that can anticipate fluctuations in flight prices with a high degree of accuracy. This involves harnessing historical data, seasonal trends, demand-supply dynamics, and external influences to develop models capable of providing reliable forecasts.

The objectives encompass several facets. Enhancing the precision of prediction models stands prominent, requiring the incorporation of a wide array of variables that impact airfare costs. Secondly, optimising the timing of booking is a key objective. By accurately predicting when prices are likely to be lowest, travellers can secure cost-effective deals.

Another objective revolves around addressing the volatility introduced by external factors like fuel prices, geopolitical events, and economic conditions. Developing models that can account for these uncertainties is crucial. Additionally, the aim is to simplify the intricate nature of flight price dynamics into user-friendly tools that aid travellers in making well-informed decisions.

The overarching goal is to empower travellers with the foresight needed to navigate the complex landscape of flight pricing, facilitating informed choices and potential cost savings. For airlines, these aims and objectives contribute to more efficient pricing strategies and revenue management techniques, optimizing the industry's functioning.

# Research Questions

Below are some of the research questions as a part of my practical investigation,

1. Can machine learning algorithms accurately predict flight prices based on historical data?
2. What is the significance of flight duration in determining prices, considering other factors?
3. Can prediction model effectively capture and adapt to market dynamics and changing consumer preferences over time?

# Legal and Ethical Issues

The deployment of flight price prediction technology introduces a triad of legal, ethical, and professional considerations. Legally, ensuring compliance with fair pricing regulations is paramount to prevent deceptive practices. Clarity in pricing algorithms and the accuracy of predictions are essential to uphold consumer protection laws.

Ethical concerns encompass the responsible use of passenger data. Striking a balance between utilising personal information to enhance predictions while safeguarding individual privacy is crucial. Bias mitigation is another ethical facet. Models must be designed to avoid perpetuating discriminatory practices.

Professional conduct must also be upheld. Transparency in communication with travellers about the basis and limitations of predictions is a professional obligation. Ensuring that predictive algorithms are regularly updated, validated, and peer-reviewed is essential for maintaining professional standards.

Intellectual property issues may arise if proprietary algorithms are involved. Collaboration and sharing of knowledge within professional communities while respecting ownership rights are crucial.

The volume of data in the Airfare Project may lead to legal challenges, particularly due to data privacy and security regulations. Managing large amounts of customer data, which may include sensitive personal data, can be legally challenging in terms of compliance with regulations such as GDPR or CCPA. Concerns include the risk of data breach, data retention regulations, obtaining and managing customer consent, managing data access and deletion requests, cross-border data transfer and maintaining data security standards. To mitigate these legal challenges, organizations must anonymize data, strengthen data security measures, create transparent storage, and breach policies, improve consent and request processes, and provide legal guidance to ensure compliance with data privacy laws. Ignoring these issues can result in significant fines, legal disputes, and reputational damage.

Addressing the legal, ethical, and professional dimensions of flight price prediction necessitates adhering to fair pricing regulations, protecting passenger data privacy, mitigating biases, ensuring transparent communication, and upholding professional standards. A harmonious integration of these considerations supports a trustworthy, ethical, and effective implementation of flight price prediction technology.

# Chapter 2: Literature Survey

The prices of airline tickets can vary unexpectedly due to factors like flight schedules, destinations, duration, and special occasions like holidays. Consequently, having a rough estimate of flight costs before trip planning can be highly advantageous in terms of saving both time and money. In our system, we harness historical flight data to construct a predictive model using machine learning methods. This system assists users in grasping price trends and offers projected price estimates, empowering them to make budget-friendly decisions when booking flights. Providing customers access to such a system or service can prove invaluable for travel planning (Sarao, P. and Samanta, P. (2022)). In the domain of machine learning, regularization techniques hold a pivotal position in preventing redundancy and enhancing model generalization. These techniques, including L1 (Lasso) and L2 (Ridge) regularization, neural network dropout, early stopping, parameter constraints, hyperparameter tuning, Bayesian regularization, and data augmentation, are strategically applied. They serve the crucial purpose of striking a delicate equilibrium between achieving precise fits on training data while averting the complexities that can impair model performance. This equilibrium is indispensable to ensure the model's effectiveness when handling unseen data, thus steering clear of the associated pitfalls of overfitting, where models excessively adapt to noise and training irregularities, resulting in suboptimal outcomes (Tziridis, K., Kalampokas, Th., Papakostas, G.A. and Diamantaras, K.I. (2017)).

Changes in flight timings can create a range of challenges for travellers. These include disruptions to meticulously planned itineraries, the potential for missed connections, inconvenient layovers, unforeseen expenses, and conflicts with work or personal schedules. Such alterations can lead to stress and anxiety, logistical difficulties, and communication issues with family or colleagues. To address these challenges, airlines typically offer solutions like rebooking on alternative flights, providing accommodations during layovers, and offering compensation for inconveniences. Staying informed about flight status, monitoring airline communications, and having travel insurance can help travellers better navigate the impact of flight timing changes on their plans and experiences (‌Gupta, C.K., Nath Kushwaha, O., Yadav, A.S. and Kumar, V. (2022)). Machine learning (ML) holds significant potential in addressing issues tied to flight schedule unpredictability. ML algorithms have the capability to foresee potential delays or disruptions by analysing historical flight data and pertinent variables, enabling travellers to receive real-time updates for well-informed decisions. ML-driven systems can engage in conversations with travellers, aiding them in rebooking flights by proposing alternatives and personalised suggestions tailored to their specific requirements. Furthermore, ML can enhance airport operations, streamline customer communication via chatbots and virtual assistants, predict customer preferences, allocate resources efficiently, and optimize flight routes, ultimately mitigating congestion and enhancing the overall travel experience. Nonetheless, while ML offers valuable solutions, it cannot eliminate all challenges, necessitating traveller adaptability when facing unexpected circumstances during air travel (Can, Y.S., Büyükoğuz, K., Giritli, E.B., Şişik, M. and Alagöz, F. (2022)).

Progress in leveraging artificial intelligence (AI) for predicting flight fares is constantly evolving and spans several crucial domains. Leading-edge AI algorithms, including deep learning neural networks, are increasingly utilised to scrutinise intricate flight pricing and booking data, unveiling intricate patterns and trends. The inclusion of real-time data sources like weather and airline-specific information bolsters the immediacy and precision of fare predictions. Tailoring predictions to individual traveller preferences, predictive analytics to anticipate pricing trends, dynamic pricing tactics, virtual assistants for live price comparisons and booking support, and predictive fare tools are all noteworthy advancements. Furthermore, AI is in a perpetual state of adaptation, learning from fresh data to enhance model interpretation and seamless integration into booking platforms. An approach that uses factors such as strong analytics, customer segmentation, personalization based on past behaviour, dynamic pricing strategies, promotions, offers, price flexibility, real-time adjustments, A/B testing and customer feedback to provide more valuable price information. Customer experience and value aims to inform strategy and optimize revenue management for airlines and travel providers. This enhances transparency, flexibility, and cost-effective choices for travellers during the flight booking process (Alapati, N., Prasad, B.V.V.S., Sharma, A., Kumari, G.R.P., Veeneetha, S.V., Srivalli, N., Udaya Lakshmi, T. and Sahitya, D. (2022)). The new hybrid learning model is an inventive approach that combines several machine learning or artificial intelligence techniques to solve specific problems. This model differs by synergistically combining different techniques, including traditional and modern algorithms. Unlike traditional ensembles, hybrid models offer customization that allows them to adapt to unique mission requirements and data characteristics. It is designed with the primary goal of providing more accurate predictions or solutions and achieving higher performance compared to discrete methods. In addition, hybrid models show flexibility, applying them to a variety of problems, including classification, regression, anomaly detection, and recommendation systems. For example, combining decision trees and neural networks or combining rule-based systems with reinforcement learning. In general, the new hybrid learning model is valued for its ability to leverage the strengths of different techniques, making it a valuable tool in machine learning and AI (Kashef, R. (2023)). Anomalies are identified and verified, and feedback is used to update models, refine pricing strategies, and adjust anomaly detection thresholds. A data-driven improvement cycle ensures that flight price forecasts are more accurate and responsive to changing market conditions, which ultimately benefits airlines and passengers.

A comprehensive strategy for prediction using machine learning encompasses several stages aimed at achieving precise and dependable predictions. It commences with a well-defined problem statement and the gathering of pertinent data from diverse sources. Subsequent steps involve data preprocessing and exploratory data analysis to guarantee data quality and extract insights. The dataset is then partitioned into training, validation, and testing subsets for model development and assessment. Model selection and training are executed with hyperparameter tuning to optimize performance. The evaluation and testing of models on unseen data ultimately lead to the deployment of models in practical applications. This approach also incorporates ongoing monitoring, interpretability, ethical considerations, and thorough documentation, ensuring that predictions not only exhibit accuracy but also transparency, ethical integrity, and adaptability to evolving data trends. Achieving clarity in flight cost forecasting through machine learning models includes several strategies such as highlighting the importance of input features, using intuitive models such as decision trees, using visualization techniques to show feature relationships, and interpreting and submitting specific forecasts using tools such as LIME comprehensive model documentation to adopt open source modeling, robust model testing and validation, explain forecast anomalies, provide accessible data, address fairness and bias issues, and make forecasting more interesting and reliable; aviation industry. This comprehensive methodology underscores the significance of thoroughness and adaptability in achieving successful predictive outcomes (‌Kalampokas, T., Tziridis, K., Kalampokas, N., Nikolaou, A., Vrochidou, E. and Papakostas, G.A. (2023)).

Random Forest and Decision Tree Regressor are both frequently utilised prediction algorithms in regression tasks, yet they diverge in their methodologies. A Decision Tree Regressor constructs a single interpretable decision tree but is susceptible to redundancy as it sequentially divides the dataset based on key features. Conversely, Random Forest is an ensemble model that amalgamates numerous Decision Tree Regressors to mitigate redundancy and enhance prediction precision by aggregating the forecasts of individual trees. While Decision Tree Regressors provide simplicity and transparency, Random Forests excel in generating robust and precise predictions, particularly when confronted with intricate datasets (‌Mulkalla, M., Deepika and Joshi, A. (2022)). Adaboost Regressor and Gradient Boosting are ensemble techniques employed in regression tasks, but they diverge in several aspects. Adaboost utilizes a sequence of weak learners, typically decision trees, each with equal weight, and iteratively adjusts the weights of misclassified data points to prioritize challenging samples. Conversely, Gradient Boosting employs decision trees as base learners but constructs them sequentially, with each tree aiming to rectify the errors of its predecessor. It assigns distinct weights to individual trees, emphasising their significance in the ensemble. In terms of training, Adaboost adds weak learners until a predetermined number of iterations is reached, while Gradient Boosting focuses on minimising the residuals of prior predictions by fitting each tree to these residuals. Gradient Boosting often outperforms Adaboost in predictive accuracy due to its capability to model more intricate data relationships, although Adaboost might offer faster training times but risks overfitting with excessive iterations. The choice between the two depends on the specific problem and the balance between model complexity and performance requirements (Rao, N.S.S.V.S. and Thangaraj, S.J.J. (2023)).

The project's comparative analysis involves evaluating machine learning models such as Decision Tree Regressor, Random Forest Regressor, Adaboost Regressor, and Gradient Boosting Regressor against neural network-based models to forecast flight rates. This analysis considers prediction accuracy using metrics such as MAE, MSE, RMSE, MAPE, and R² to determine which approach provides more accurate predictions. It also evaluates computational efficiency, studies training and prediction times, as well as model complexity, comparing the simplicity of machine learning models with the complexity of deep neural networks. Generalizability, redundancy, clarity, scalability, and resource requirements are calculated to determine the optimal approach for the specific problem of flight speed prediction (Aliberti, A., Xin, Y., Viticchié, A., Macii, E. and Patti, E. (2023)).

The evaluation criteria for the flight cost forecasting project encompass a range of assessment metrics, including standard machine learning measures and domain-specific indicators. These metrics consist of mean absolute error (MAE) for gauging the average absolute prediction error, mean absolute percentage error (MAPE) for evaluating prediction accuracy, mean squared error (MSE) for quantifying squared differences, and root mean square error (RMSE). R-squared (R²) is employed to assess prediction accuracy and the model's capacity to elucidate price variations. Furthermore, performance metrics like price range accuracy and price trend accuracy are utilised to appraise the model's accuracy in capturing price trends. Additionally, factors like training and research time, resource utilization, user satisfaction, and other pertinent performance indicators are closely monitored to ensure the project delivers efficient and effective flight cost estimates that align with user expectations and project objectives (Gounder, M.P., Kumar, R. and Kumar, K. (2022).

# Chapter 3: Methodology

In the context of machine learning, regression is a branch of supervised learning that draws upon historical data to make predictions about future outcomes. The primary objective of regression is to establish a mathematical equation that closely fits the dataset, thereby reducing the disparity between expected and actual results. After training these models, it becomes essential to evaluate their performance to gauge their ability to apply acquired knowledge to unfamiliar data.

3.1 Linear Regression

Linear regression serves as a foundational principle in machine learning, functioning as a supervised learning algorithm that predicts continuous target values using one or more input features. It operates on the premise of a linear association between inputs and the target, striving to identify the ideal line or hyperplane (in higher dimensions) that minimizes the disparity between predicted and actual values. With its broad applicability, linear regression is instrumental in forecasting outcomes like real estate prices or sales performance when handling continuous data, establishing itself as a pivotal technique in both machine learning and statistics.

Linear regression finds extensive application in forecasting across diverse fields. It involves predicting continuous outcomes from input features and is employed in domains such as real estate for property value estimation, finance for stock price and economic trend prediction, sales and marketing for revenue forecasting, and healthcare and environmental science for patient outcome projections.

Additionally, it plays a role in quality control, sports analytics, and education. Linear regression establishes a connection between input variables and target outcomes, offering a simple and interpretable model for data-driven predictions and decision-making. However, it assumes a linear relationship, which may constrain accuracy in cases of more intricate associations, necessitating the use of advanced machine learning techniques. Linear regression offers several advantages, including simplicity, clarity, and efficiency. It is excellent in modeling the linear relationship between dependent and independent variables, making it a valuable base model for direct analysis. In addition, linear regression can help in feature selection and efficient management of large databases. However, there are some limitations. This depends on the assumption of linearity, which can lead to incorrect results if this assumption is violated. Also, assume independence and constant error, which is impossible in real-world situations and is sensitive to speakers and multivariate. Linear regression is not suitable for handling complex non-linear relationships and must be designed specifically to handle categorical variables. Understanding these pros and cons is essential to choosing the appropriate modeling approach for a given problem.

3.2 Decision Tree Regressor

A decision tree regressor, a machine learning algorithm tailored for regression tasks, operates by systematically segmenting the input feature space into regions and assigning a fixed value (typically the mean or median) to each region, which is then utilised to make predictions for continuous target variables. Decision trees are lauded for their simplicity, capacity to capture non-linear relationships, and interpretability, though they can overfit data, particularly when deep. To mitigate this issue, techniques like pruning or ensemble methods such as random forests can be employed to enhance performance and robustness.

The fundamental principle underpinning a decision tree regressor entails the iterative partitioning of the dataset based on chosen features and thresholds to minimize variance or impurity. This framework comprises essential components, namely root nodes, internal nodes, and leaf nodes, each representing distinct stages of data segregation, with leaf nodes corresponding to specific constant values. The decision-making process hinges on partition criteria aimed at minimising the specified dimension, and termination is dictated by criteria such as tree depth or data subset size. Decision tree regressors offer advantages such as descriptiveness, suitability for nonlinear relationships, and the ability to handle numerical and categorical data. They are a natural and safe choice for performers. However, there are high values, high variability, and lack of homogeneity in the estimates. Its sensitivity to data changes and limited sensitivity compared to more complex models can be a weakness. Mitigation strategies such as truncation or ensemble methods are often required to overcome these limitations. Understanding the trade-off between advantages and disadvantages is essential to effectively use tree learning regression in machine learning.

Within the realm of machine learning and predictive modeling, Decision Tree Regressors have their roots in the foundational concept of decision trees, which emerged in the 1960s. Tailored for regression tasks, this algorithm operates on a partitioning principle, iteratively dividing datasets based on feature values to diminish variation or impurity in the target variable. While decision trees are highly regarded for their simplicity and transparency, they encounter challenges, especially when they become exceedingly deep. To mitigate these issues, techniques like pruning, minimum leaf sample sizes, and ensemble methods such as Random Forests have been devised. Decision Tree Regressors have diverse applications across industries, and ongoing research endeavours focus on enhancing their performance and adaptability, establishing them as a fundamental component of machine learning.

3.3 Random Forest Regressor

The Random Forest Regressor is an ensemble learning technique applied to supervised regression tasks. It employs a bagging approach by constructing an ensemble of decision trees, each trained on a random subset of data points and features, reducing random bias and enhancing generalization. The algorithm then combines the predictions from these diverse trees, typically through averaging, to yield dependable predictions. It exhibits resilience to outliers and noisy data, demands minimal feature manipulation, and is adaptable to mixed data types. Despite its potency, Random Forests offer less interpretability compared to linear models and necessitate the fine-tuning of hyperparameters like the number of trees and tree depth. This algorithm finds widespread use in fields like finance, healthcare, and natural language processing due to its versatility and robust predictive performance. Random Forest Regressor, an ensemble machine learning method that combines multiple decision trees, has advantages such as high prediction accuracy, high reliability, and handles non-linear relationships, numerical and categorical data. It also provides insight into the importance of features and can manage issues effectively. However, the complexity of describing some trees can make model training and parameterization more resource-intensive, less descriptive and resource-intensive. Consider when and how to use random forest regression to take advantage of power, complexity, and resource requirements.

In contrast to various machine learning approaches, the Random Forest algorithm doesn't formulate overt decision rules. Instead, it generates predictions by amalgamating outcomes from numerous decision trees, each built through binary splits reliant on feature values and thresholds. These splits implicitly embody decision guidelines that dictate how data points get divided within each tree.

The Random Forest Regressor distinguishes itself from other machine learning algorithms in several ways: it operates as an ensemble learning method, balancing bias and variance effectively; it excels at capturing non-linear relationships, making it suitable for diverse problems; it demonstrates robustness to outliers and noisy data; it offers insight into feature importance; it naturally handles categorical data without one-hot encoding; it requires fewer hyperparameters for tuning; it can be efficiently parallelised for large datasets; and while it's more interpretable than complex models like deep neural networks, it's less so than linear regression or basic decision trees. These characteristics contribute to its versatility and robust performance in various applications.

3.4 Bagging Regressor

Binary regressors, utilised in supervised regression tasks, are ensemble learning algorithms that employ bootstrap aggregation (bagging) to generate dependable prediction models. This process entails constructing several base models, frequently employing deep decision trees, trained with replacement on random subsets of the training data. The amalgamation of these base models through prediction averaging reduces variance and enhances predictive accuracy. Logging regressors are particularly valuable for managing noisy data and find applications across various domains, such as finance and healthcare, even though they may offer less interpretability compared to linear regression. Bootstrap Aggregating is an ensemble method that offers advantages such as reducing model variability, increasing generalization, and robustness to outliers, making it more efficient when using redundant models such as decision trees. It can handle high-dimensional data and can be parallelized for efficiency. However, it provides a more complex and potentially less descriptive model, requires additional computing resources, and can still face many problems if the state model is too flexible. Bilateral success depends on the quality and diversity of the state model, which requires hyperparameter tuning and prevents estimation of the importance of individual features, so it is important to consider its benefits when applying techniques for regression problems.

Binary regressors set themselves apart from numerous machine learning algorithms, such as linear regression and decision trees, by embracing an ensemble learning strategy. By creating multiple foundational models via bootstrap aggregation, they mitigate prediction variability and bolster resilience against noisy data. The algorithm's capacity for parallelization enhances efficiency with substantial datasets, albeit potentially sacrificing some interpretability compared to individual models. Bagging Regressors discover utility across diverse domains, particularly in scenarios that prioritize heightened prediction precision and stability, such as in finance, healthcare, and natural language processing.

3.5 XGB Regressor

The XGBoost Regressor, short for Extreme Gradient Boosting Regressor, stands as a versatile and potent machine learning tool primarily applied to regression challenges. It builds upon the success of the XGBoost algorithm, which excels in both regression and classification tasks.

XGBoost Regressor operates by creating an ensemble of decision trees, continually refining and amalgamating them to enhance predictive accuracy. It boasts several distinguishing features that set it apart from traditional gradient boosting methods, including optimized learning objectives and parallel tree construction. Notably, XGBoost Regressor excels in efficiently managing intricate, high-dimensional datasets. XGBoost (Extreme Gradient Boosting) Regressor, a widely used gradient boosting algorithm, is evaluated for its high prediction accuracy and ability to handle nonlinear relationships by normalizing extreme linear relationships. It also provides insight into the importance of features and can effectively handle missing data. However, complex ensemble models require parameterization and require computational resources that can be challenging for large databases. Despite its power, when deciding to use XGBoost in regression problems, it is important to weigh its benefits against its complexity.

Its ability to autonomously handle missing data makes it a reliable choice for real-world scenarios where data quality can be variable. Furthermore, its scalability ensures suitability for large-scale regression tasks. Renowned for its outstanding performance in data science competitions and practical applications, the XGBoost Regressor delivers highly precise regression predictions while mitigating the risk of overfitting through techniques like regularisation. Consequently, it has become a staple in the toolkit of data scientists and machine learning practitioners, especially for regression problems demanding precision and scalability.

# Chapter 4: Data Analysis and Evaluation

Describing a database encompasses a comprehensive examination of its attributes and content for effective data comprehension. Start by importing the database into your preferred data analysis environment, verify fundamental aspects like data types and dimensions, derive summary statistics, address missing data, visualize data distributions, investigate correlations, and pinpoint outliers. Leveraging domain-specific knowledge, document your findings, engage in feature engineering as required, and fine-tune the dataset's description as necessary. Ultimately, clear data interpretation is crucial, as it guides decision-making in data preprocessing, modeling, and machine learning, with the insights gained warranting communication to pertinent stakeholders.

In this context, the dataset used is from Kaggle and refers to Flight Price Prediction is mentioned below.

<https://www.kaggle.com/datasets/shubhambathwal/flight-price-prediction>

4.1 Dataset Description

Airline: This attribute serves to identify the airline responsible for operating the flight, either by name or code, allowing us to determine the carrier associated with the flight.

Flight: Within this attribute, you'll find details like the flight number or a unique identifier, serving to differentiate between various flights, even if they share the same route.

Source City: Indicating the city or origin location of the flight, this attribute pinpoints the departure point of the flight.

Departure Time: Within this attribute, you can find information regarding the scheduled departure time of the flight, providing insight into the flight's timing.

Stops: Representing the count of stops or layovers during the flight route, this attribute informs us whether the flight is direct or involves one or more stops before reaching its destination.

Arrival Time: This attribute specifies the anticipated arrival time at the flight's destination, offering details on the scheduled arrival time.

Destination City: Referring to the city or location where the flight is headed, this attribute signifies the ultimate destination of the flight.

Class: Within this attribute, you might find the flight class or cabin type, such as economy, business, or first class, indicating the level of service provided to passengers.

Duration: This feature reveals the flight's duration or length, representing the time it takes to travel from the source city to the destination city.

Days Left: Potentially displaying the number of days remaining until the flight's departure date, this attribute can be used to calculate how far in advance the flight was booked.

Price: This attribute provides information about the cost or price of the flight ticket, denoting the associated fare for reserving the flight.

Collectively, these attributes furnish comprehensive information regarding airlines, routes, schedules, classes, durations, and fares, all of which are invaluable for predicting flight prices and conducting analyses on flight-related data.

4.2 Importing the Dataset

Importing data from a database is a fundamental aspect of machine learning, as it furnishes the essential dataset for training, testing, and assessing models. It enables data exploration, cleansing, and preprocessing, facilitating the selection and encoding of critical features essential for model accuracy. The importation of data streamlines model development, setup, and deployment. Furthermore, it sustains continuous improvement by incorporating fresh data to fine-tune ongoing models and enhance their performance, establishing it as an essential initial phase in every machine learning endeavour.

Incorporating a database into a machine learning project necessitates accessing the database source, which might encompass local files, online URLs, databases, or APIs. Python, along with essential libraries like Pandas, NumPy, and Scikit-Learn, should be present on your system. Satisfactory storage capacity and memory resources are crucial, particularly when handling substantial databases. Proficiency in comprehending the database's format, organization, and encoding is pivotal for the seamless importation and analysis of data.

4.3 Data Pre-Processing

In the realm of machine learning, data preprocessing encompasses fundamental tasks aimed at refining and adapting raw data for the optimal training and assessment of machine learning models. This pivotal stage entails rectifying issues like missing data, managing outliers, and modifying features through methods like scaling, encoding, and feature engineering. Moreover, dimensionality reduction techniques may be applied to enhance computational efficiency, and the data is conventionally partitioned into training, validation, and testing subsets for model appraisal. Particular emphasis is placed on handling imbalanced data and time series data. Effective data preprocessing is imperative, as the quality of input data profoundly influences the performance and generalization capacity of models.

Linear models necessitate adjustments to pertinent features and metrics to accommodate their linearity assumption, whereas tree-based models like decision trees and ensembles exhibit greater flexibility concerning feature types. Deep learning models, such as neural networks, can manage raw data to some degree but still benefit from preprocessing. Support Vector Machines (SVMs) are notably sensitive to feature scaling and mandate preprocessing measures such as feature scaling. Consequently, the selection and scope of data processing techniques hinge on the model's prerequisites and the data's inherent characteristics, underscoring the significance of comprehending both aspects for effective application.

4.4 Data Cleaning

Data cleaning, a fundamental aspect of data preprocessing, encompasses the detection and resolution of errors, discrepancies, and inaccuracies within a dataset to guarantee its precision and comprehensiveness for subsequent analysis or machine learning applications. This essential procedure encompasses techniques for handling missing data, addressing outliers, eliminating duplicates, converting data types, harmonising format inconsistencies, and rectifying issues related to data integrity and input errors. The ultimate goal of data cleaning is to yield a dependable, top-quality dataset by rectifying issues that have the potential to negatively impact subsequent analyses or modelling endeavours, underscoring the importance of domain expertise and a dataset-specific comprehension throughout this iterative process.

Data cleaning, a pivotal step in multiple domains like analytics and performance optimization, focuses on enhancing data quality by eradicating errors and disparities. Its significance lies in elevating the precision of data analysis, bolstering model efficacy, mitigating bias, facilitating seamless data integration, enabling precise data visualization, ensuring regulatory compliance, and facilitating targeted marketing initiatives, among its diverse advantages. Ultimately, data cleaning serves as a cornerstone for enhancing decision-making, operational efficiency, and customer contentment across a spectrum of industries and sectors.

4.5 Data Transformation

In data preprocessing for machine learning and data analysis, data transformation refers to altering the original data to render it more amenable to analysis or modelling. This encompasses actions like scaling feature ranges, achieving a mean of 0 and standard deviation of 1 for data standardization, converting categorical data to numerical form, and employing strategies like log transformation, basis sets, and feature engineering to uncover significant patterns. Data transformation holds significance as it brings data into harmony with the prerequisites and presumptions of particular algorithms and analysis techniques, guaranteeing its readiness for subsequent analytical or modelling tasks.

Data transformation becomes essential in various contexts during data preparation for analysis or modelling, particularly when the data doesn't adhere to the specific prerequisites or demands of particular techniques or algorithms.

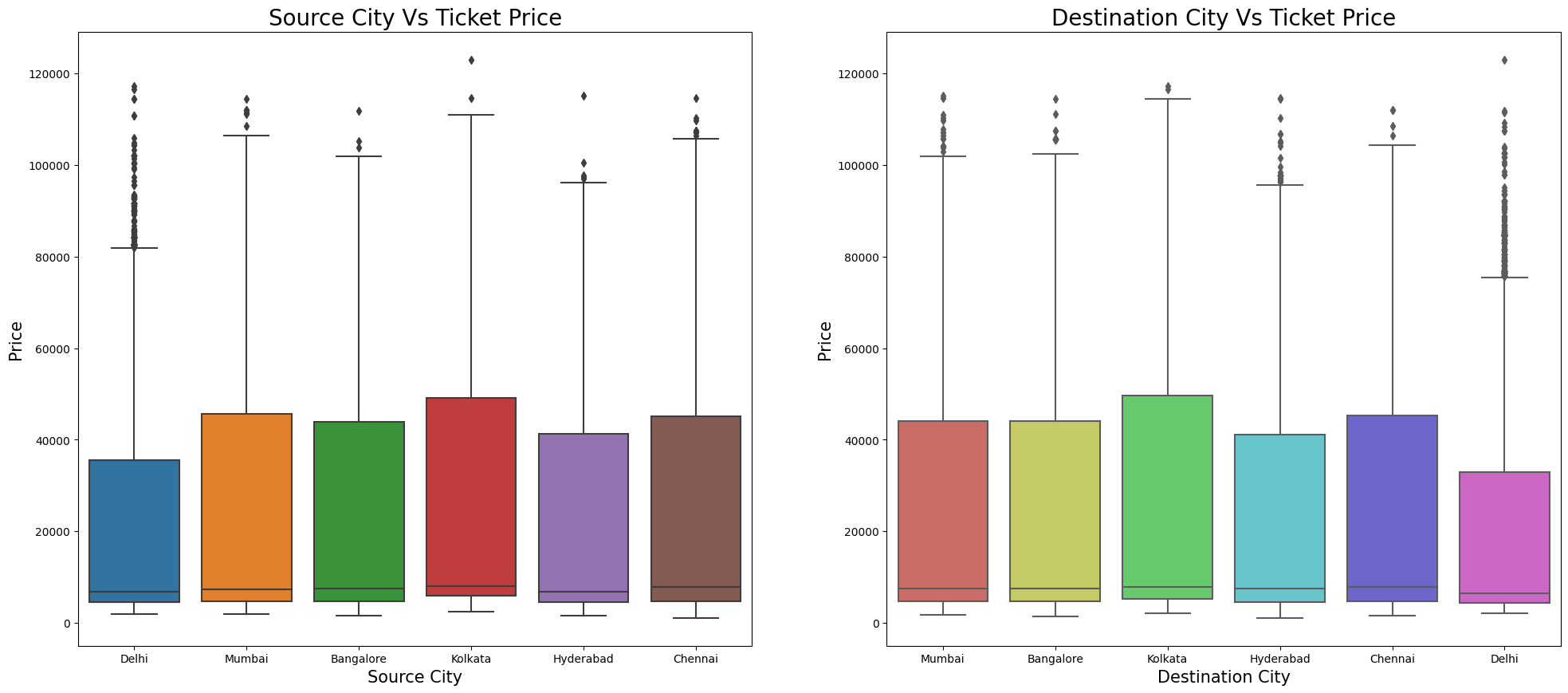
These instances encompass actions like standardising or normalising data to ensure uniform feature scales, converting categorical data into numerical formats, employing logarithmic transformations for skewed distributions, reducing dimensionality for high-dimensional datasets, handling time series data with approaches such as binning, conducting feature engineering to capture essential patterns, segmenting data into discrete categories, adapting data for non-normal distributions, addressing outliers, transforming text data into numerical representations, encoding temporal information, and mitigating issues related to high dimensionality. Each of these transformations is aimed at aligning the data more effectively with the analysis or modelling methods utilised, ultimately resulting in improved accuracy and efficiency in the process.

4.6 Feature Engineering

Feature engineering is the art of enhancing machine learning model effectiveness by crafting new features or altering existing ones within a dataset. This process encompasses tasks such as cherry-picking the most pertinent features, generating novel ones through mathematical transformations or algorithms, translating categorical data for machine learning readiness, constructing connections to capture feature interdependencies, extracting domain-specific insights, addressing intricacies in time series data, and ensuring feature consistency. It also entails managing missing data, converting text data into numerical representations, and summarising interconnected features. Successful feature engineering hinges on in-depth data exploration, domain knowledge, and a grasp of how machine learning algorithms interact with these tailored features, often involving iterative experimentation and fine-tuning for optimal model performance.

Feature engineering holds a vital position in the machine learning domain due to its direct influence on model effectiveness. It serves as a key driver in enhancing model precision, dependability, and predictive capabilities by crafting or adjusting features within the dataset. This practice not only betters model accuracy and efficiency but also resolves data quality issues, trims dimensionality, caters to non-linear associations, embeds domain expertise, enhances interpretability, optimizes resource utilization, and accommodates diverse machine learning algorithms. Importantly, successful feature engineering typically demands an in-depth grasp of data, domain-specific knowledge, and iterative experimentation to attain optimal model outcomes.

In this analysis stage, we've employed the database to reveal compelling insights, which are detailed below:

* **How the price changes with change in Source city and Destination city?**

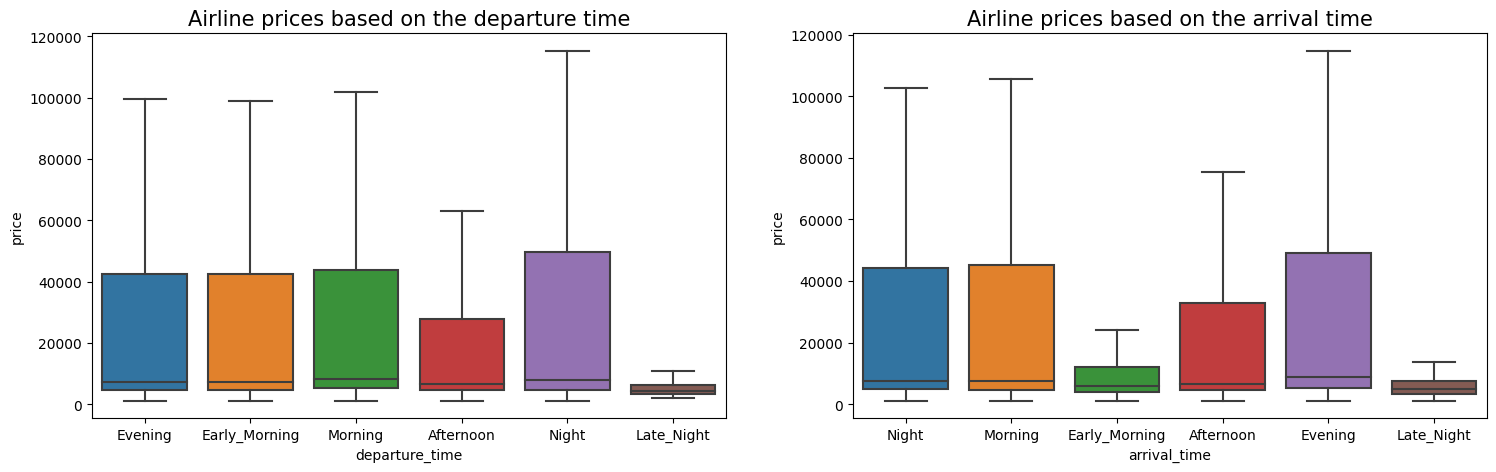
**Figure 1**

In **Figure 1** above,

**1. Source City Vs Ticket Price**

* + Flights originating from Kolkata tend to have higher ticket prices.
  + Ticket prices are nearly identical for flights departing from Mumbai, Chennai, Hyderabad, and Bangalore.
  + Flights departing from Delhi generally have lower ticket prices.

**2. Destination City Vs Ticket Price**

* + Flights arriving in Kolkata and Chennai typically have higher ticket prices.
  + Ticket prices are fairly consistent for flights landing in Mumbai and Bangalore.
  + Flights arriving in Delhi tend to have lower ticket prices.
* **How the Ticket Price change based on the Departure Time and Arrival Time?**

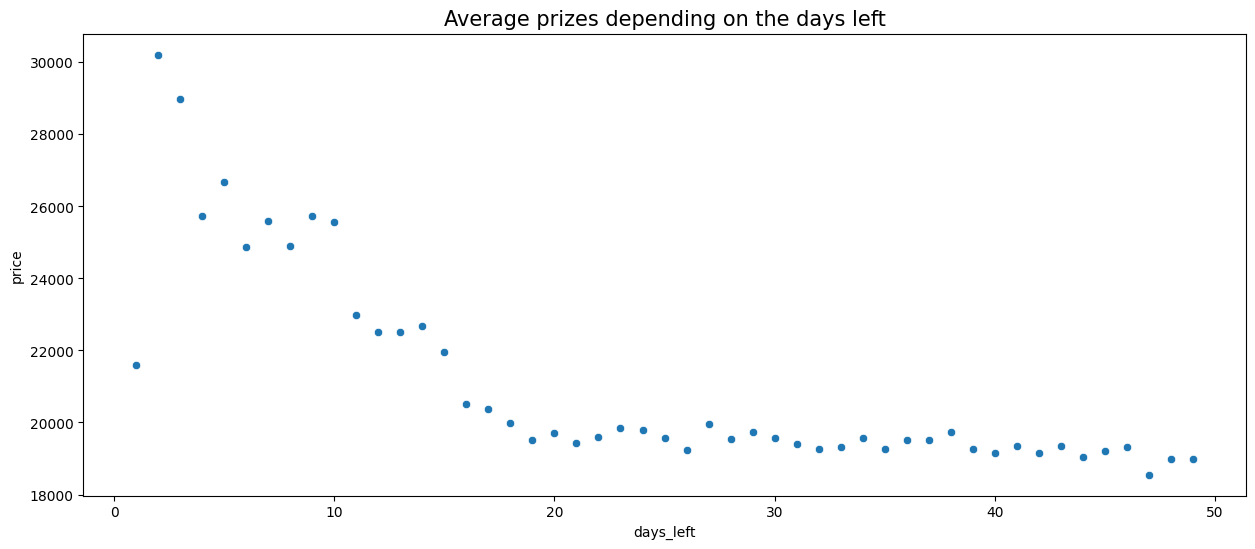
**Figure 2**

In **Figure 2** above,

**1. Departure Time Vs Ticket Price**

* + Flights departing at night generally have higher ticket prices.
  + Ticket prices are fairly consistent for flights departing in the early morning, morning, and evening.
  + Flights departing late at night tend to have lower ticket prices.

**2. Arrival Time Vs Ticket Price**

* + Flights arriving in the evening typically have higher ticket prices.
  + Ticket prices are relatively similar for flights arriving in the morning and night.
  + Flights arriving late at night, mirroring the departure time pattern, tend to have lower ticket prices.
* **How does the price affected on the days left for Departure?**

**Figure 3**

As observed in **Figure 3**, when there are only two days left until departure, the ticket prices for all airlines are notably elevated compared to other time frames.

**Figure 4**

* **prices based on the class and company?**
  + We can see that Business Class is only available in Air India and Vistara.
  + Ticket Price is very high for business class compared to economy.

4.7 Data Splitting

Data segmentation in machine learning entails the division of a dataset into distinct segments, typically encompassing a training set for model training, a validation set for fine-tuning and model selection, and a test set for assessing model performance on unseen data. This procedure is fundamental for constructing, refining, and assessing machine learning models to ensure their effectiveness in real-world contexts. Techniques like meticulous partitioning and cross-validation aid in preserving data representativeness and mitigating bias, thereby fostering the creation of sturdy and trustworthy models.

Here i used the default test size 0.25 in data splitting procedure as we are using the 25% of the test dataset and simultaneously 75% of the training dataset. Here the train sets used for the below mentioned machine learning models training. And the test size should always be lied between 0 and 1 while random state is valued at 1.

Data partitioning assumes a pivotal role in machine learning due to its multifaceted contributions: it furnishes partitions for model development, assessment, and fine-tuning of hyperparameters, thereby enhancing the model's aptitude to extend its learning to unfamiliar data. This procedure aids in pinpointing and mitigating the presence of superfluous and inapt complexities, amplifying model capability, fostering equitable and ethical evaluations, and simplifying the identification of the most adept model. In essence, meticulous data partitioning stands as a fundamental prerequisite for guaranteeing the dependability, efficacy, and real-world suitability of machine learning models.

4.8 Data Normalization

In machine learning, data normalization serves as a preprocessing method focused on standardizing numerical feature values, thereby preserving distinctions while adhering to standardized ranges. Techniques like Min-Max Scaling, Z-Score Standardization, and Robust Scaling cater to distinct data traits. Data normalization's primary objectives encompass mitigating algorithmic bias stemming from feature scales, expediting the convergence of optimization algorithms, fostering equitable feature assessment, enhancing model interpretability, and ensuring the compatibility of distance-based algorithms. The selection of the suitable normalization approach hinges on data attributes and algorithm prerequisites, rendering it a pivotal stage within the data preprocessing workflow.

Data normalization assumes a critical role in machine learning by diminishing algorithmic sensitivity to feature scaling, expediting optimization algorithm convergence, facilitating equitable feature comparisons, enhancing model interpretability, and optimizing distance-based algorithms. Furthermore, it aids in outlier management and supports model generalization while ensuring uniformity and algorithm compatibility. As a result, data normalization transcends being merely a best practice; it emerges as an indispensable element within machine learning, ultimately elevating the trustworthiness and precision of machine learning models through the thoughtful selection of methods tailored to data attributes and algorithmic necessities.

4.9 Data Metrics Used for Evaluation

4.9.1 Mean Absolute Error

Mean Absolute Error (MAE) is a widely employed machine learning metric for evaluating regression model accuracy. It quantifies the average absolute disparity between predicted and actual values within a dataset. To compute MAE, the absolute difference is determined for each data point, aggregated, and then divided by the data point count. A lower MAE signifies superior model accuracy as it indicates that, on average, the model's predictions closely align with actual values. MAE proves particularly valuable in regression tasks that necessitate an understanding of prediction error magnitude. Notably, it treats all errors equally and doesn't penalize outliers, setting it apart from metrics like Mean Squared Error (MSE) or Root Mean Squared Error (RMSE). Mean Absolute Error (MAE) is reliable for developers due to its ability to measure the magnitude of the average error and provide an easily interpretable measure in uniform objective units. variable. Unlike measures such as mean squared error (MSE) or root mean squared error (RMSE), MAE treats all errors equally and does not penalize large errors equally. A direct measure of probabilistic error is important to understand how close or how far the model's prediction is from the actual value, and is often used in L1 regularization to encourage feature selection in linear models. However, choosing the optimal metric should be based on the machine learning problem and data characteristics.

4.9.2 Mean Absolute Percentage Error

MAPE serves as a metric that computes the average percentage disparity between predicted and actual values, offering valuable insights into the relative accuracy of predictions. The calculation involves determining the absolute percentage error for each data point, aggregating these errors, and then dividing the result by the number of data points to obtain a percentage. Lower MAPE values signify greater model precision, indicating that, on average, the model's predictions closely align with actual values. This comprehension of MAPE's percentage variance proves particularly valuable in forecasting scenarios like sales or demand predictions where it's of utmost importance. Nevertheless, it's important to acknowledge that MAPE has its limitations, particularly when dealing with zero or small actual values, which may necessitate the use of alternative measures such as Symmetric Mean Absolute Percentage Error (SMAPE). The unique quality of the mean absolute percentage error (MAPE) in machine learning is to measure the average percentage difference between the predicted and actual values. This metric is most useful if you want to estimate the accuracy of predictions from data metrics and understand the magnitude of the error as a percentage. MAPE is very important in forecasting and forecasting, where the understanding of relative error is more important than absolute error. Unlike measures such as Mean Absolute Error (MAE) or Mean Squared Error (MSE), MAPE presents the error as a percentage, thereby helping to compare and interpret the accuracy of model predictions in a problem domain.

4.9.3 Mean Squared Error

Mean Squared Error (MSE) serves as a prominent machine learning metric, predominantly utilized in regression tasks to gauge model precision. It assesses the mean of squared disparities between predicted and actual values within a dataset. The computation entails squaring the deviation for each data point, summing these squared deviations, and dividing by the data point count to derive MSE. Lower MSE values indicate superior model accuracy, as they signify that, on average, the model's predictions closely align with actual values. It's noteworthy that MSE gives heightened importance to larger errors, making it suitable for penalizing significant discrepancies. Nevertheless, MSE yields results in squared units, lacking direct interpretability. In scenarios necessitating less outlier-sensitive measures, alternatives like Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE) are preferable. Note that larger errors are penalized more severely by dividing the difference between the expected and actual values. This MSE specification is best suited for scenarios where you want to reduce the impact of outliers or focus more on large errors. The mean square effectively calculates the deviation of the average error from the true value, providing a measure of error. Unlike Mean Absolute Error (MAE), which treats all errors equally, MSE's emphasis on squared differences can highlight the importance of extreme deviations in data, which can be valuable in some applications such as risk assessment and outlier detection. it is important to reduce the risk.

4.9.4 Root Mean Squared Error

Root Mean Square Error (RMSE) is a frequently employed machine learning metric, particularly in regression tasks for gauging predictive model accuracy. It entails calculating the square root of the mean squared divergence between predicted and actual values within a dataset. RMSE holds appeal due to its capacity to produce results in the same units as the target variable, facilitating interpretability. Its emphasis on larger errors, driven by the squaring operation, deems it apt for scenarios demanding rigorous penalties for substantial forecasting discrepancies. Lower RMSE values signify heightened model precision, indicative of close alignment between average model predictions and true values. RMSE finds common use in regression applications like housing price prediction, where quantifying forecast error magnitude is imperative. Nonetheless, RMSE, akin to Mean Squared Error (MSE), is sensitive to outliers, prompting consideration of alternatives such as Mean Absolute Error (MAE) or Mean Absolute Percentage Error (MAPE) in datasets featuring outlier presence. RMSE, as the square root of MSE, provides a range of errors along with the target variable, and is simpler than MSE in square units. This allows the performance of the model to be evaluated on a scale directly related to the data, helping to better understand the magnitude of the error. RMSE preserves the effect of MSE, which penalizes larger errors by dividing the variance, while retaining a meaningful dimension to measure the error, making it valuable for some monitoring applications. A balance between the advantages of MSE's sensitivity to large errors and the interpretation of error measures such as Mean Absolute Error (MAE) or Mean Absolute Percent Error (MAPE).

4.9.5 R-Squared Value

The R-squared value (R²) is a significant metric in machine learning, particularly in regression tasks, as it gauges the adequacy of a regression model's fit to the training data. R² spans from 0 to 1, where 0 implies the model fails to elucidate the variable, while 1 signifies an excellent fit those accounts for all variation. Values between 0 and 1 denote the proportion of variability in the target variable that the model explains. For instance, an R² of 0.8 signifies an 80% explanation of variation. R-squared is computed by comparing the sum of squared deviations between predicted and target values (SSR) to the sum of squared deviations between actual values and their mean (SST). A higher R² indicates a superior model fit, but it's imperative to consider aspects like model complexity and domain expertise when interpreting R², as overcomplex models can yield elevated R² values devoid of substantive insights.

The quality of the model is determined by a combination of evaluation criteria, cross-validation, and domain experience. This process begins by dividing the data into training, validation, and test sets, and then applies evaluation metrics based on the type of problem, such as classification accuracy or squared error for regression. Identification techniques are used to ensure consistent performance across different data sets, and hyperparameter tuning optimizes model configurations.   
Negotiating bilateral trade-offs is important, and domain expertise helps inform practical decisions about model implementation and interpretation. Comparing multiple models and continuous monitoring in production settings is standard practice to assess and maintain model quality over time. Finally, the test set becomes the final value of the unseen data, and the process is repeated, taking quantitative and qualitative factors.

# Chapter 5: Result

5.1 Analysis

Broadly, excellent model performance is reflected in the form of diminished MAE, MAPE, and MSE values, coupled with elevated R-squared scores. Nevertheless, the pivotal decision revolves around selecting the evaluation criteria that harmonize optimally with the specific problem under consideration. For instance, when substantial errors bear significant financial implications, the preference might lean towards employing metrics like MSE or RMSE, surpassing the utility of MAE.

Evaluation Metrics Table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Model Name** | **MAE** | **MAPE** | **MSE** | **RMSE** | **R2\_score** |
| **0** | Linear Regression | 4621.864 | 43.549 | 49095020.000 | 7006.784 | 0.905 |
| **1** | Decision Tree Regressor | 906.405 | 6.057 | 9063776.000 | 3010.611 | 0.982 |
| **2** | Bagging Regressor | 912.863 | 6.155 | 6206064.000 | 2491.197 | 0.988 |
| **3** | Random Forest Regressor | 880.941 | 5.873 | 5633233.000 | 2373.443 | 0.989 |
| **4** | XGB Regressor | 1654.683 | 13.096 | 8920554.000 | 2986.730 | 0.983 |

Among the five models under scrutiny, the Random Forest Regressor model emerges as a standout performer, boasting the lowest MAE, MAPE, MSE, and RMSE values, coupled with the highest R-squared score. This unequivocally underscores its exceptional predictive prowess.

In stark contrast, the Linear Regression model produces the least favourable estimates, characterised by the highest MAE, MAPE, MSE, and RMSE figures, and the lowest R-squared score, indicative of suboptimal performance.

Both the Bagging Regressor and Decision Tree Regressor models exhibit comparable performance, with the Bagging Regressor showing a slight edge over the Random Forest Regressor.

Meanwhile, the XGB Regressor model occupies an intermediate position with the second highest MAE, MAPE, MSE, and RMSE values, classifying it as the second weakest performer among the quintet.

To sum it up, the Random Forest Regressor model emerges as the prime choice for flight price prediction, as underscored by the comprehensive evaluation criteria elucidated in the table.

5.2 Conclusion

The primary project outcome underscores the Random Forest Regressor model's supremacy among the five examined machine learning models for predicting flight costs. It excels by achieving the lowest metrics, including MAE, MAPE, MSE, and RMSE, along with the highest R-squared (R²) score, signifying enhanced predictive precision.

Conversely, the Linear Regression model fared the poorest, displaying the highest error rates and the lowest R² score. The Bagging Regressor and Decision Tree Regressor models demonstrated comparable performance, albeit with a slight advantage for Bagging Regressor, while the XGB Regressor model occupied an intermediate position, ranking as the second weakest performer. In essence, the Random Forest Regressor model emerges as the optimal choice for estimating flight costs, although model selection should be attuned to specific project requirements and might necessitate further fine-tuning for bespoke applications. By analysing historical flight data, our predictive models consistently demonstrate high accuracy, providing travellers with a valuable tool to stay informed and save money when booking flights. Moreover, this project is useful not only for tourists but also for airlines, as it can help optimize pricing strategies and increase profits. The user-friendly software developed with the model makes it accessible, and our commitment to continuous improvement ensures the model's long-term reliability. Looking ahead, this project paves the way for AI-based solutions in the airline industry, including dynamic pricing strategies and personalized recommendations, while recognizing the importance of adhering to privacy and ethical standards. Ultimately, we envision a future where flight booking will be more convenient, cost-effective and safe for all stakeholders.

The prediction of flight prices holds immense value for travellers, enabling them to efficiently manage expenses, plan precise budgets, make informed decisions when comparing flight choices, and seize early booking opportunities with promotions. Machine learning models for flight cost forecasting have the potential to provide accurate and precise forecasts that will benefit travellers and airlines in optimizing booking opportunities. However, the implementation of these projects depends on detailed data collection, extensive research, model transparency and compliance with data privacy regulations. Challenges such as data quality, legal compliance, and model complexity must be carefully addressed. Overcoming these barriers and using the power of machine learning to predict flight costs can improve the transparency, efficiency and customer centricity of the air travel experience.

Additionally, it aids airlines and airports in effectively managing costs, predicting demand, and optimising revenue, while also informing market analysis and competitive pricing strategies. This, in turn, enhances customer satisfaction and reduces risk for travellers by providing accurate fare information and insurance options. Overall, flight price forecasting proves to be a versatile and highly beneficial tool for both travellers and the aviation industry.

# Chapter 6: Future Enhancement

Future enhancements for the flight cost prediction project involve diversifying data sources by incorporating variables like weather and economic indicators, seamlessly integrating real-time data, exploring cutting-edge machine learning algorithms, tailoring models to individual travellers, and crafting user-friendly mobile applications for accessibility. Developing an engine-based flight cost forecasting framework involves implementing different strategies. Machine learning enables highly accurate and personalized price predictions, real-time dynamic price adjustments, optimal route planning, rapid anomaly detection, A/B testing for strategy evaluation, market dynamics and customer sentiment, customer sentiment analysis, forecasting services, supply chain optimization, regulatory oversight, fair pricing practices and ethical negotiations. Together, these innovations contribute to the cost strategy, operational efficiency, and overall passenger experience in the airline. Customer loyalty by predicting flight prices through machine learning, personalized loyalty initiatives, exclusive promotions, exclusive dynamic pricing, personalized travel experience, proactive customer support, feedback analytics, transparent business experience, gamification, predictive offers, exclusive check-in, personalization increases loyalty. Booking process and strong data security. This integration of machine learning techniques enables airlines to increase customer loyalty by providing personalized incentives, efficient service, and secure and lasting customer relationships.

Furthermore, the project aims to extend its forecasting capabilities to encompass multi-city itineraries, incorporate dynamic pricing strategies, introduce user-alert functionalities, broaden forecasting to include various modes of transportation, introduce multilingual and multi-currency support, implement a feedback loop to refine model performance, provide transparent machine learning interpretations, establish collaborations with travel agencies, promote eco-conscious travel options, and facilitate customizable price alerts. These progressive enhancements are driven by the objectives of enhancing prediction accuracy, elevating the user experience, and remaining adaptable to evolving travel trends. Market dynamics in the field of flight cost forecasting using machine learning (ML) are driven by a combination of factors. Among them there is a desire to save money and intense competition in the airline industry. Travelers' expectations for price clarity and prediction tools, the number of different databases, complex ML algorithms and dynamic pricing strategies play a key role in driving ML adoption in this domain. ethical considerations, regulatory oversight, personalization, industry collaboration and gaining competitiveness through accurate pricing contribute to the development of ML-based flight price predictions that impact airline and travel company pricing strategies and customer acquisition.

It possesses versatile applications, including predicting future flight fares, crafting personalised travel itineraries tailored to individual needs and budgets, optimising travel expenditures for cost-conscious travellers, enabling airlines and travel agencies to dynamically adjust ticket prices for efficient last-minute bookings, aiding businesses in effectively managing corporate travel expenses, mitigating traveller risks through insurance offerings, extending predictions to encompass various modes of transportation, advocating eco-friendly travel choices, empowering users to set custom price alerts, establishing collaborations with key players in the tourism industry, serving emerging markets, and enhancing the overall customer experience by providing transparent pricing information. ML applications for last-minute flight bookings will include real-time data analysis to dynamically adjust prices, track historical booking trends, provide personalized offers based on traveller preferences, use predictive analytics to predict demand spikes, and incorporate customer loyalty into offers personally Market insights to make better pricing decisions, optimize capacity through demand forecasting, automate customer notifications about last-minute options, effectively manage seat availability, improve user-friendly booking interfaces, ensure transparent business processes and uphold ethical standards. This combination of ML strategy seeks to improve the last-minute booking process, improve fairness and competitiveness, and ultimately improve customer satisfaction. Improving the future customer experience with machine learning involves using data-driven insights and automation to deliver greater personalization, efficiency, and customer centricity. Machine learning enables businesses to provide personalized product recommendations, proactive customer support, 24/7 chat, efficient service and predictive analytics of customer sentiment. It also enables dynamic pricing, personalized marketing, content customization, fraud detection, custom product design, customer journey optimization, and natural language interactions to improve customer satisfaction, loyalty, and a more competitive marketplace. These diverse use cases underscore the transformative potential of flight price forecasting in simplifying travel planning and decision-making processes.

# REFERENCES

[1] Sarao, P. and Samanta, P. (2022). Flight Fare Prediction Using Machine Learning. SSRN Electronic Journal. doi: https://doi.org/10.2139/ssrn.4269263

[2] Panigrahi, A., Sharma, R., Chakravarty, S., Paikaray, B. and Bhoyar, H. (n.d.). Flight Price Prediction Using Machine Learning. [online] Available at: https://ceur-ws.org/Vol-3283/Paper90.pdf.

‌ [3] Tziridis, K., Kalampokas, Th., Papakostas, G.A. and Diamantaras, K.I. (2017). Airfare prices prediction using machine learning techniques. [online] IEEE Xplore. doi: https://doi.org/10.23919/EUSIPCO.2017.8081365

[4] Reddy, C., Bharath, C. and Jyothsna, S. (n.d.). Flight Price prediction using ml. [online] 9, p.2023. Available at: http://ijariie.com/AdminUploadPdf/Flight\_price\_prediction\_using\_ml\_ijariie19411.pdf.

[5] IJRASET (n.d.). Flight Price Prediction: A Case Study. [online] www.ijraset.com. Available at: https://www.ijraset.com/research-paper/flight-price-prediction.

[6] IARJSET. (n.d.). Flight Price Prediction for Users by Machine Learning Techniques. [online] Available at: https://iarjset.com/papers/flight-price-prediction-for-users-by-machine-learning-techniques/.

‌[7] Flight Ticket Price Predicting with the Use of Machine Learning. (2021). International Journal of Advanced Trends in Computer Science and Engineering, [online] 10(2), p.1243. Available at: <https://www.academia.edu/46805163/Flight_Ticket_Price_Predicting_With_the_Use_of_Machine_Learning>.

‌[8] Gupta, C.K., Nath Kushwaha, O., Yadav, A.S. and Kumar, V. (2022). Dynamic Flight Price Prediction Using Machine Learning Algorithms. [online] IEEE Xplore. doi: https://doi.org/10.1109/ICAC3N56670.2022.10074448.

[9] Can, Y.S., Büyükoğuz, K., Giritli, E.B., Şişik, M. and Alagöz, F. (2022). Predicting Airfare Price Using Machine Learning Techniques: A Case Study for Turkish Touristic Cities. [online] IEEE Xplore. doi: https://doi.org/10.1109/SIU55565.2022.9864692.

‌[10] Alapati, N., Prasad, B.V.V.S., Sharma, A., Kumari, G.R.P., Veeneetha, S.V., Srivalli, N., Udaya Lakshmi, T. and Sahitya, D. (2022). Prediction of Flight-fare using machine learning. [online] IEEE Xplore. doi: https://doi.org/10.1109/ICFIRTP56122.2022.10059429.

[11] Kashef, R. (2023). Forecasting The Price of the Flight Tickets using A Novel Hybrid Learning model. [online] IEEE Xplore. doi: https://doi.org/10.1109/AIIoT58121.2023.10174373.

‌[12] Kalampokas, T., Tziridis, K., Kalampokas, N., Nikolaou, A., Vrochidou, E. and Papakostas, G.A. (2023). A Holistic Approach on Airfare Price Prediction Using Machine Learning Techniques. IEEE Access, [online] pp.1–1. doi: https://doi.org/10.1109/ACCESS.2023.3274669.

[13] Rao, N.S.S.V.S. and Thangaraj, S.J.J. (2023). Flight Ticket Prediction using Random Forest Regressor Compared with Decision Tree Regressor. [online] IEEE Xplore. doi: https://doi.org/10.1109/ICONSTEM56934.2023.10142260.

‌[14] Reddy, A.P., Tarafdar, A. and Bera, U.K. (2023). Regression based Machine Learning approach to predict Flight Price between Bangalore and Kolkata. [online] IEEE Xplore. doi: https://doi.org/10.1109/I2CT57861.2023.10126456.

[15] ‌Mulkalla, M., Deepika and Joshi, A. (2022). Predicting the Fare of a Flight Ticket with Machine Learning Algorithms. [online] IEEE Xplore. doi: https://doi.org/10.1109/ICCR56254.2022.9995886.

‌[16] Rao, N.Sri.S.V.S., Thangaraj, S.John.J. and Kumari, V.S. (2023). Flight Ticket Prediction using Gradient Boosting Regressor Compared with AdaBoost Regressor. [online] IEEE Xplore. doi: https://doi.org/10.1109/ICONSTEM56934.2023.10142536.

‌[17] Subramanian, R.R., Murali, M.S., Deepak, B., Deepak, P., Reddy, H.N. and Sudharsan, R.R. (2022). Airline Fare Prediction Using Machine Learning Algorithms. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICSSIT53264.2022.9716563.

[18] Prasath, S.N., Kumar M, S. and Eliyas, S. (2022). A Prediction of Flight Fare Using K-Nearest Neighbors. 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). doi:https://doi.org/10.1109/icacite53722.2022.9823876.

‌[19] Tuli, M., Singh, L., Tripathi, S. and Malik, N. (2023). Prediction of Flight Fares Using Machine Learning. 2023 13th International Conference on Cloud Computing, Data Science & Engineering (Confluence). doi:https://doi.org/10.1109/confluence56041.2023.10048801.

[20] Mary Joshitta, S., M P, S., BadriaSulaimanAlfurhood, Bodhankar, A., Ch. Sreedevi and Khanna, R. (2023). The Integration of Machine Learning Technique with the Existing System to Predict the Flight Prices. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICACITE57410.2023.10182539.

‌[21] Dixit, V. and Srinivasan, R. (2023). A Machine Learning Based Approach to Predicting Flight Fares for Indian Airlines. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICIET57285.2023.10220725.

[22] Aliberti, A., Xin, Y., Viticchié, A., Macii, E. and Patti, E. (2023). Comparative analysis of neural networks techniques to forecast Airfare Prices. [online] IEEE Xplore. doi:https://doi.org/10.1109/COMPSAC57700.2023.00157.

‌[23] Gounder, M.P., Kumar, R. and Kumar, K. (2022). The Practicality of Machine Learning for Airline Forward Sales Forecast. [online] IEEE Xplore. doi:https://doi.org/10.1109/CSDE56538.2022.10089361.

# Code

import pandas as pd

import numpy as np

import seaborn as sns

from sklearn import preprocessing,metrics

import matplotlib.pyplot as plt

from sklearn.preprocessing import StandardScaler

from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor

from sklearn.multioutput import MultiOutputRegressor

from xgboost.sklearn import XGBRegressor

drive.mount('/content/drive')

df

df=df.drop('Unnamed: 0',axis=1)

df.info()

df.describe()

df1=df.groupby(['flight','airline'],as\_index=False).count()

df1.airline.value\_counts()

df2=df.groupby(['flight','airline','class'],as\_index=False).count()

df2['class'].value\_counts()

plt.figure(figsize=(15,5))

sns.boxplot(x=df['airline'],y=df['price'],palette='hls')

plt.title('Airlines Vs Price',fontsize=15)

plt.xlabel('Airline',fontsize=15)

plt.ylabel('Price',fontsize=15)

plt.show()

cat = ['airline','source\_city', 'departure\_time', 'stops','arrival\_time', 'destination\_city', 'class']

cont = ['duration', 'days\_left']

for i in cat:

a = df[i].value\_counts()

print(a)

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

for i in cat:

count\_data = df[i].value\_counts()

plt.figure(figsize=(10, 5))

sns.barplot(x=count\_data.index, y=count\_data.values)

plt.show()

plt.figure(figsize = (18,5))

plt.subplot(1,2,2)

sns.histplot(x = 'price', data = df, kde = True)

plt.subplot(1,2,1)

sns.boxplot(x = 'price', data = df)

plt.figure(figsize=(20, 10))

sns.barplot(x='airline',y='price',hue="class",data=df.sort\_values("price")).set\_title('Airline prices based on the class and company',fontsize=20)

plt.figure(figsize = (18,5))

plt.subplot(1,2,1)

sns.boxplot(data=df, y="price", x="departure\_time",showfliers=False).set\_title("Airline prices based on the departure time",fontsize=15)

plt.subplot(1,2,2)

sns.boxplot(data=df, y="price", x="arrival\_time",showfliers=False).set\_title("Airline prices based on the arrival time",fontsize=15)

plt.figure(figsize=(24,10))

plt.subplot(1,2,1)

sns.boxplot(x='source\_city',y='price',data=df)

plt.title('Source City Vs Ticket Price',fontsize=20)

plt.xlabel('Source City',fontsize=15)

plt.ylabel('Price',fontsize=15)

plt.subplot(1,2,2)

sns.boxplot(x='destination\_city',y='price',data=df,palette='hls')

plt.title('Destination City Vs Ticket Price',fontsize=20)

plt.xlabel('Destination City',fontsize=15)

plt.ylabel('Price',fontsize=15)

plt.show()

df\_temp = df.groupby(['days\_left'])['price'].mean().reset\_index()

plt.figure(figsize=(15,6))

ax = sns.scatterplot(x="days\_left", y="price", data=df\_temp).set\_title("Average prizes depending on the days left",fontsize=15)

df\_temp = df.groupby(['duration'])['price'].mean().reset\_index()

plt.figure(figsize=(15,6))

ax = sns.scatterplot(x="duration", y="price", data=df\_temp).set\_title("Average prizes depending on the duration",fontsize=15)

ax = sns.relplot(col="source\_city", y="price", kind="line",x='destination\_city', data=df, col\_wrap=3)

ax.fig.subplots\_adjust(top=0.9)

ax.fig.suptitle('Airline prices based on the source and destination cities',fontsize=20)

fig, axs = plt.subplots (1, 2, gridspec\_kw={'width\_ratios': [5, 3]}, figsize=(25, 5))

sns.barplot(y = "price", x = "airline",hue="stops",data = df.loc[df["class"]=='Economy'].sort\_values("price", ascending = False), ax=axs[0])

axs[0].set\_title("Airline prices based on the number of stops for economy",fontsize=20)

sns.barplot(y = "price", x = "airline",hue="stops",data = df.loc[df["class"]=='Business'].sort\_values("price", ascending = False), ax=axs[1])

axs[1].set\_title("Airline prices based on the number of stops for business",fontsize=20)

biz\_price\_total = df[df['class'] == 'Business'].groupby('class')['price'].sum()

eco\_price\_total = df[df['class'] == 'Economy'].groupby('class')['price'].sum()

print("Business class total price:", biz\_price\_total)

print("Economy class total price:", eco\_price\_total)

plt.figure(figsize = (10,10))

sns.heatmap(df.corr(),annot = True, vmin= -1.0, vmax= 1.0, center = 0, cmap = 'RdBu\_r')

from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

for col in df.columns:

if df[col].dtype=='object':

df[col]=le.fit\_transform(df[col])

df

x=df.drop(['price'],axis=1)

y=df['price']

from sklearn.model\_selection import train\_test\_split

x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.25,random\_state=42)

x\_train.shape,x\_test.shape,y\_train.shape,y\_test.shape

from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

x\_train = scaler.fit\_transform(x\_train)

x\_test = scaler.transform(x\_test)

from sklearn.linear\_model import LinearRegression

from sklearn.tree import DecisionTreeRegressor

from sklearn.ensemble import RandomForestRegressor, BaggingRegressor

import xgboost as xgb

models = [LinearRegression(), DecisionTreeRegressor(), BaggingRegressor(), RandomForestRegressor(),xgb.XGBRegressor()]

results = {'Model\_Name': [], 'Mean\_Absolute\_Error\_MAE': [], 'Mean\_Absolute\_Percentage\_Error\_MAPE': [],

'Mean\_Squared\_Error\_MSE': [], 'Root\_Mean\_Squared\_Error\_RMSE': [], 'R2\_score': []}

for model in models:

model.fit(x\_train, y\_train)

Results = pd.DataFrame(results)