



**Faculty of Engineering & Technology
Electrical & Computer Engineering Department**

ENCS5141-Intelligent Systems Laboratory

**Case Study #1 Data Cleaning and Feature Engineering for the
Titanic Dataset**

Instructor:

Dr. Yazan AbuFarha

Prepared By:

Abd Khuffash-1200970

Section:

2

Date:

30-3-2024

Abstract/Objectives

This study delves into the exploration and analysis of the Titanic dataset, aiming to predict survival outcomes using machine learning techniques. Initially, libraries such as Seaborn, NumPy, Pandas, and Matplotlib are imported to facilitate data handling and visualization. The dataset is loaded and explored to understand its structure and content. Data preprocessing involves handling missing values and outliers, employing techniques like Z-score method for outlier detection. Further, correlation analysis is conducted to understand the relationships between features and survival. Machine learning models, including Decision Trees, are employed to predict survival, with Grid Search optimizing model parameters. The study concludes with an evaluation of model performance and insights into feature importance. This comprehensive analysis offers valuable insights into the factors influencing survival on the Titanic.

Table of Contents

Contents

Abstract/Objectives.....	i
Table of Contents	ii
Table of Figures.....	iii
Introduction.....	2
Procedure and Discussion	4
Conclusion	11

Table of Figures

Figure 1. Data Processing	2
Figure 2. Count of Survivors	4
Figure 3. Correlation Heatmap	5
Figure 4. Scatter Plot for outliers in columns	6
Figure 5. Null values after the handling	6
Figure 6. Correlation analysis results	7
Figure 7. Decision Tree	8

Introduction

In the realm of data science and machine learning, thorough exploration and preprocessing of data lay the groundwork for building robust models. The Titanic dataset, a classic in the field, provides a rich source for such exploration. In this theoretical exposition, we delve into the process of importing, analyzing, processing, and modeling this dataset.

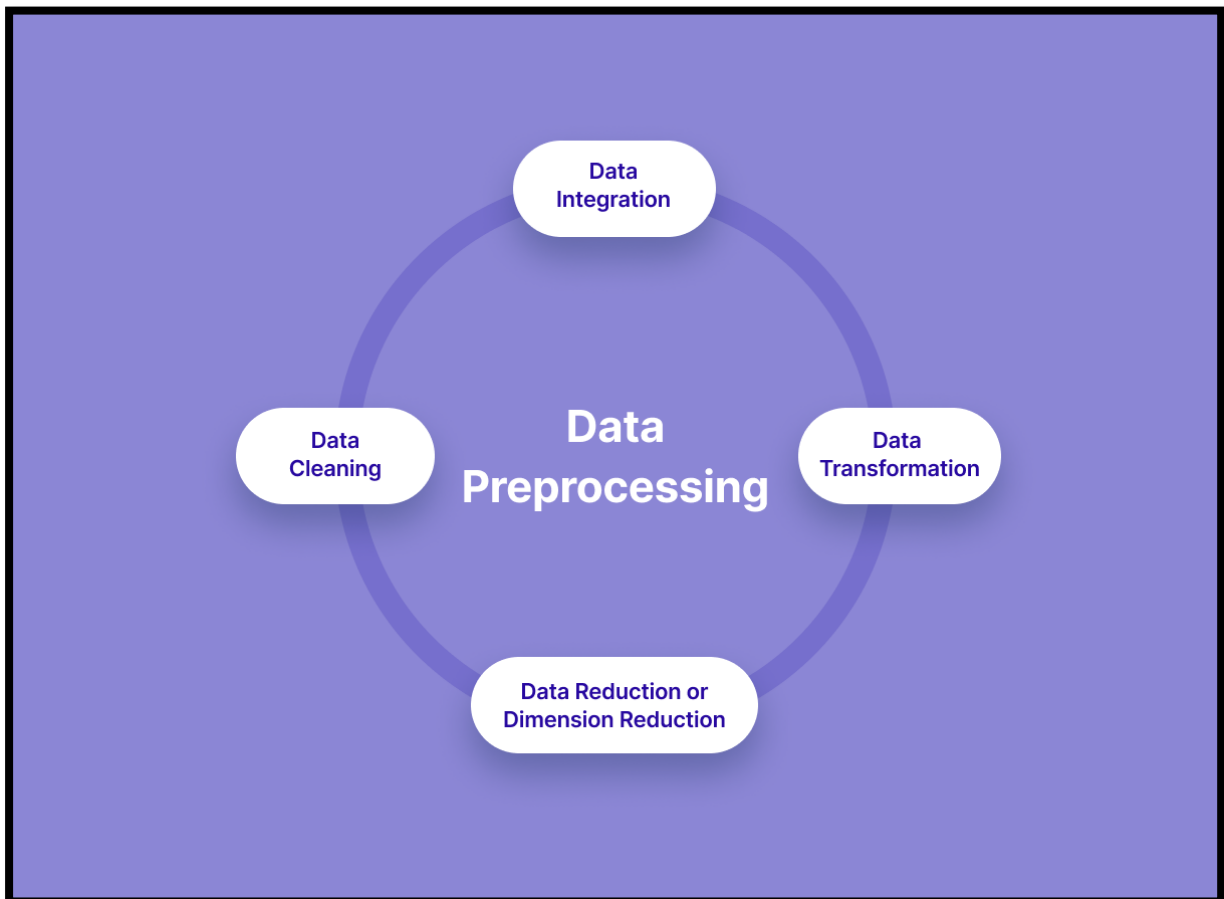


Figure 1. Data Processing

Data Cleaning and Preprocessing is a crucial step in any data analysis project. Missing values need to be handled appropriately, and outliers may need to be addressed depending on their impact on the analysis. Features may need to be transformed or encoded to be suitable for machine learning algorithms.

Feature Engineering, Understanding the relevance of each feature to the target variable is essential. This can be done through correlation analysis or domain knowledge. Feature engineering involves creating new features or transforming existing ones to improve model performance.

Model Training and Evaluation, in this case, a Decision Tree classifier and Random forest classifier are used as the machine learning model. Decision trees are intuitive to interpret and can handle both numerical and categorical data. Model evaluation is done using metrics like

accuracy, precision, recall, and F1-score. Grid Search is employed to find the best hyperparameters for the model, optimizing its performance.

Visualizations like box plots, scatter plots, count plots, and heatmaps are used throughout the analysis to understand the distribution of data, identify outliers, explore relationships between variables, and visualize model performance.

Data analysis and machine learning are often iterative processes. After building a model, it's important to evaluate its performance, iterate on feature engineering, try different algorithms, and fine-tune hyperparameters to improve results.

Procedure and Discussion

The Study starts by importing the needed libraries, imported libraries include Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn, and others for data manipulation, visualization, and machine learning. The Titanic dataset was loaded using the `sns.load_dataset('titanic')` function from the Seaborn library.

Then started exploring the dataset, initial exploration included displaying the first few rows, summarizing data types and structure, obtaining descriptive statistics, identifying missing values, and visualizing data distributions and relationships.

Multiple figures were displayed, for instance the count of survivors:

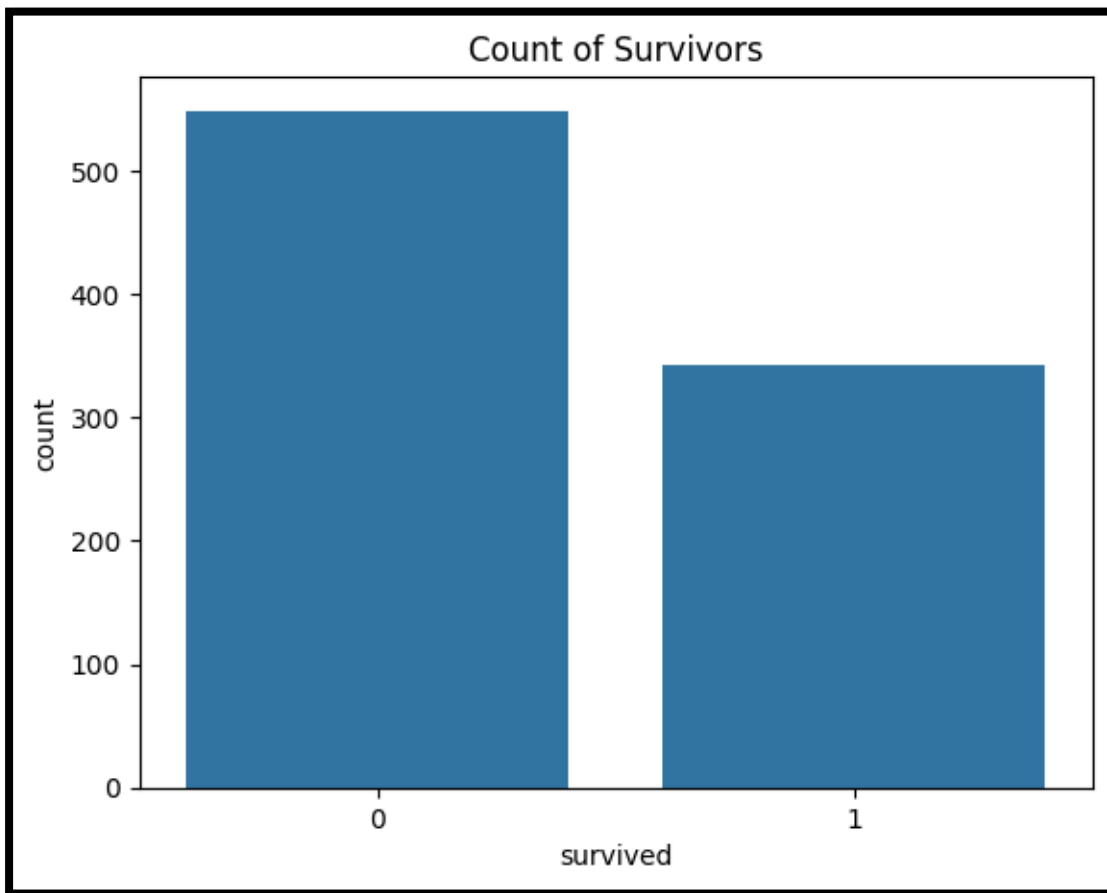


Figure 2. Count of Survivors

A Correlation heatmap for the dataset:

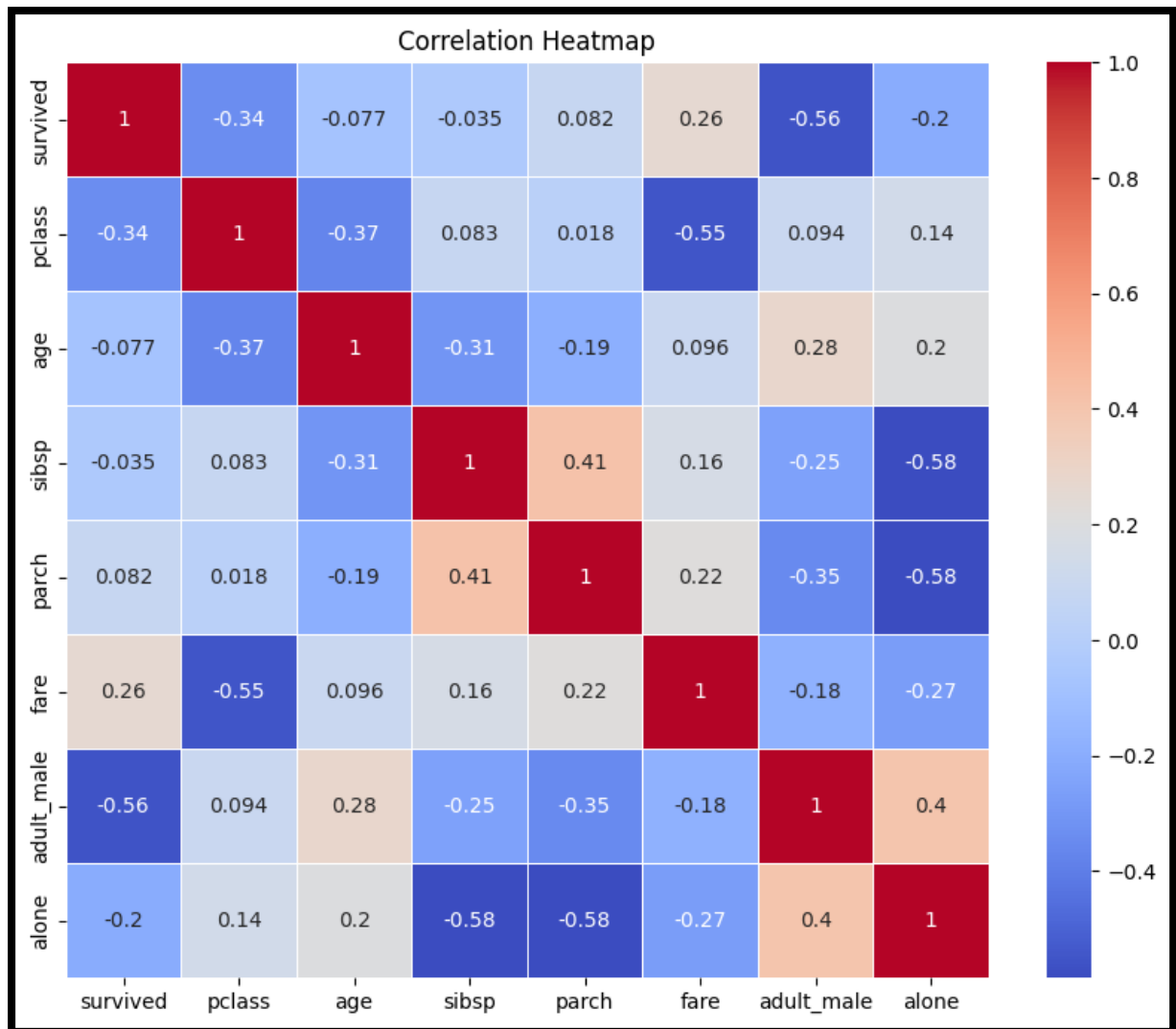


Figure 3. Correlation Heatmap

For processing and cleaning the data, Outliers were detected and handled using the Z-score method, Missing values in features such as 'age', 'embarked', 'deck', and 'embark_town' were addressed through imputation or removal, Categorical columns were encoded using Label Encoding, Correlation analysis was conducted to understand relationships between features and the target variable 'survived'.

Here is a scatterplot for detecting anomalies:

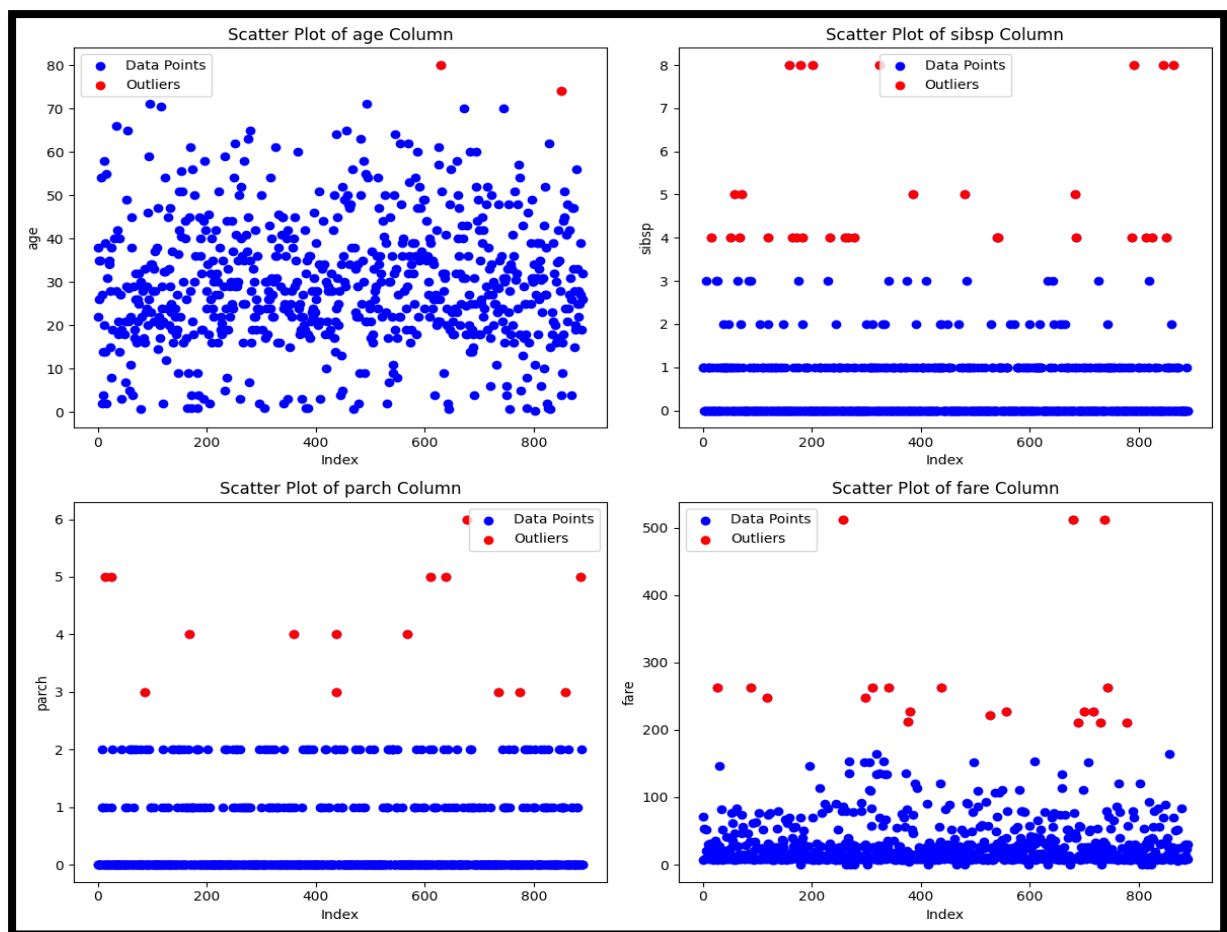


Figure 4. Scatter Plot for outliers in columns

After cleaning the data this is what the dataset looked like:

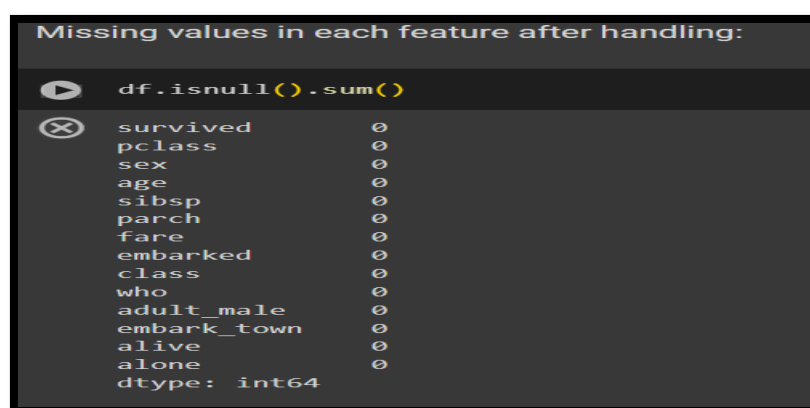
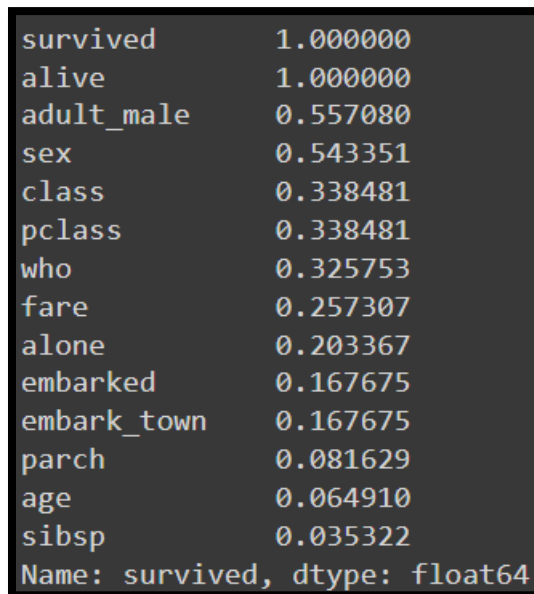


Figure 5. Null values after the handling

Finding the relevance of each feature to the target 'Survived', through a correlation analysis:



survived	1.000000
alive	1.000000
adult_male	0.557080
sex	0.543351
class	0.338481
pclass	0.338481
who	0.325753
fare	0.257307
alone	0.203367
embarked	0.167675
embark_town	0.167675
parch	0.081629
age	0.064910
sibsp	0.035322
Name: survived, dtype: float64	

Figure 6. Correlation analysis results

Based on correlation coefficients, several factors are associated with survival rates aboard the Titanic. Strong positive correlations were observed between survival and being alive, being female, or not being an adult male. Moderate positive correlations were found between survival and passenger class or certain passenger designations. Weak positive correlations were noted between survival and fare paid, or traveling alone. Weak negative correlations were observed between survival and factors such as port of embarkation, number of parents/children aboard, age, and number of siblings/spouses aboard. Overall, being female, belonging to a higher passenger class, paying a higher fare, and traveling alone were associated with higher survival chances, while age and familial relations had weaker associations with survival.

After preparing and cleaning the dataset, it was ready for utilizing it with a machine learning algorithm, first the dataset was splitted using `train_test_split` from `skit` library (80% train, 20% test), `Min-Max` scaller was used, also the dimensionality was reduced using Principal component analysis.

Two models were trained, Decision Tree and Random Forest Classifiers, each model was trained with default parameters and then using a `GridsearchCV`, parameters were tuned to find the best for each model, and here are the results:

Decision tree:

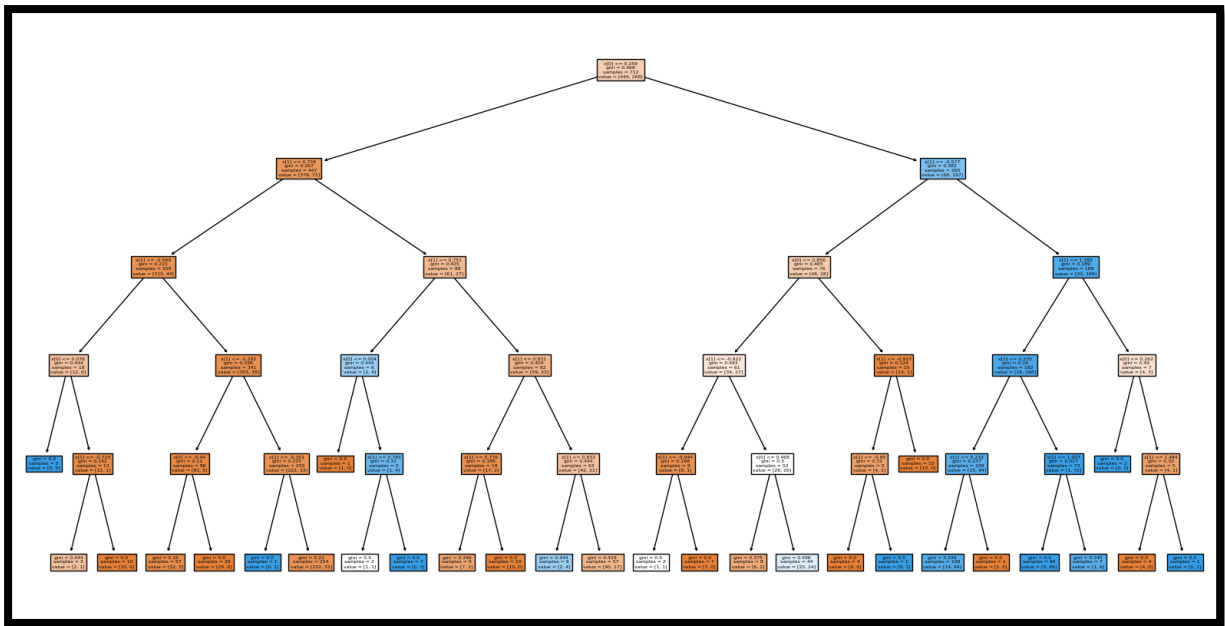


Figure 7. Decision Tree

Decision Tree classifier was initially trained with default parameters, and then a grid search was performed to optimize the model's parameters.

1. Default Decision Tree Model:

* Accuracy: 0.7988

- * **Confusion Matrix:**

[[85 20]]

[16 58]]

* The default model had an accuracy of around 79.88%. Precision, recall, and F1-score for class 0 (survived) are relatively higher compared to class 1 (not survived), indicating that the model performs slightly better at predicting class 0.

2. Grid Search Optimized Model:

```
* Best Parameters: {'criterion': 'gini', 'max_depth': 5, 'min_samples_leaf': 1,
'min_samples_split': 2}
```

* Best Score: 0.8062

* Accuracy: 0.8212

- * Confusion Matrix:

[[92 13]

[19 55]]

- * The grid search optimized model achieved an accuracy of around 82.12%, which is slightly better than the default model.

3. Comparison

- * The optimized model outperforms the default model, indicating that parameter tuning through grid search improved the model's performance.

- * Both models seem to have better performance in predicting class 0 (survived) than class 1 (not survived), which is evident from higher precision, recall, and F1-score for class 0 in both cases.

- * The optimized model shows improvement particularly in terms of reducing false positives (predicted survived but actually not survived), as evident from the decrease in the confusion matrix's (0,1) entry.

Overall, the grid search optimization helped in fine-tuning the parameters of the Decision Tree classifier, resulting in a more accurate model for predicting survival outcomes.

For the Random Forest:

Random Forest classifier was initially trained with default parameters, and then a grid search was performed to optimize the model's parameters.

1. Default Random Forest Model:

- * Accuracy: 0.7933

- * Confusion Matrix:

[[84 21]

[16 58]]

- * The default model had an accuracy of around 79.33%. Precision, recall, and F1-score for class 0 (survived) are relatively higher compared to class 1 (not survived), indicating that the model performs slightly better at predicting class 0.

2. Grid Search Optimized Model:

- * Best Parameters: {'max_depth': None, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 200}

- * Accuracy: 0.8045

- * Confusion Matrix:

[[93 12]

[23 51]]

- * The grid search optimized model achieved an accuracy of around 80.45%, which is slightly better than the default model.

3. Comparison

- * Similar to the Decision Tree, the optimized Random Forest model outperforms the default model, indicating that parameter tuning through grid search improved the model's performance.

- * Both models seem to have better performance in predicting class 0 (survived) than class 1 (not survived), which is evident from higher precision, recall, and F1-score for class 0 in both cases.

- * The optimized model shows improvement particularly in terms of reducing false negatives (predicted not survived but actually survived), as evident from the increase in the confusion matrix's (1,0) entry.

In summary, similar to the Decision Tree classifier, the Random Forest classifier benefited from parameter tuning through grid search, resulting in a more accurate model for predicting survival outcomes.

Conclusion

In summary, this exposition navigated the data exploration, preprocessing, modeling, and evaluation process using the Titanic dataset. Through meticulous steps, we refined the dataset for machine learning algorithms. The implementation of Decision Tree and Random Forest classifiers, along with parameter optimization, enhanced predictive accuracy. Both classifiers exhibited higher precision in predicting survival outcomes, emphasizing the iterative nature of data analysis and machine learning.