Titanic Survival Prediction Research Paper

1. Introduction

For this research project, I wanted to analyze the survival rates of passengers on the Titanic using machine learning. The goal was to explore the dataset, clean up any messy data, perform exploratory data analysis (EDA), and then build a model to predict whether a passenger survived or not. Throughout this process, I applied different techniques to improve the model's accuracy and gain insights into which factors played the biggest role in survival. This document outlines everything I did, step by step, along with the results and observations.

2. Data Preprocessing

The first step was to take a good look at the dataset. I ran `df.info()` to check for missing values and quickly noticed that some columns, like 'Cabin,' had way too many missing entries to be useful. Instead of trying to fill in those missing values, I decided to drop the 'Cabin' column entirely. Other columns like 'PassengerId,' 'Name,' 'Ticket,' and 'Embarked' also didn't seem relevant for predicting survival, so I removed those as well.

One issue that stood out was missing values in the 'Age' column. Since age might be an important factor for survival, I didn't want to remove rows with missing values. Instead, I used K-Nearest Neighbors (KNN) imputation to estimate the missing ages based on other passenger data. This ensured that I didn't lose valuable information.

Another key step was handling categorical variables. The 'Sex' column contained values like 'male' and 'female,' which wouldn't work directly in a machine learning model. I converted these into numerical values using Label Encoding, where 'male' became 0 and 'female' became 1. This transformation made it easier for the model to process.

3. Exploratory Data Analysis (EDA)

Before building the model, I wanted to get a sense of which features were most important. I used a correlation heatmap to visualize the relationships between different variables. This helped me see that 'Sex' and 'Pclass' (passenger class) had strong correlations with survival. It makes sense—first-class passengers and women had a much higher survival rate.

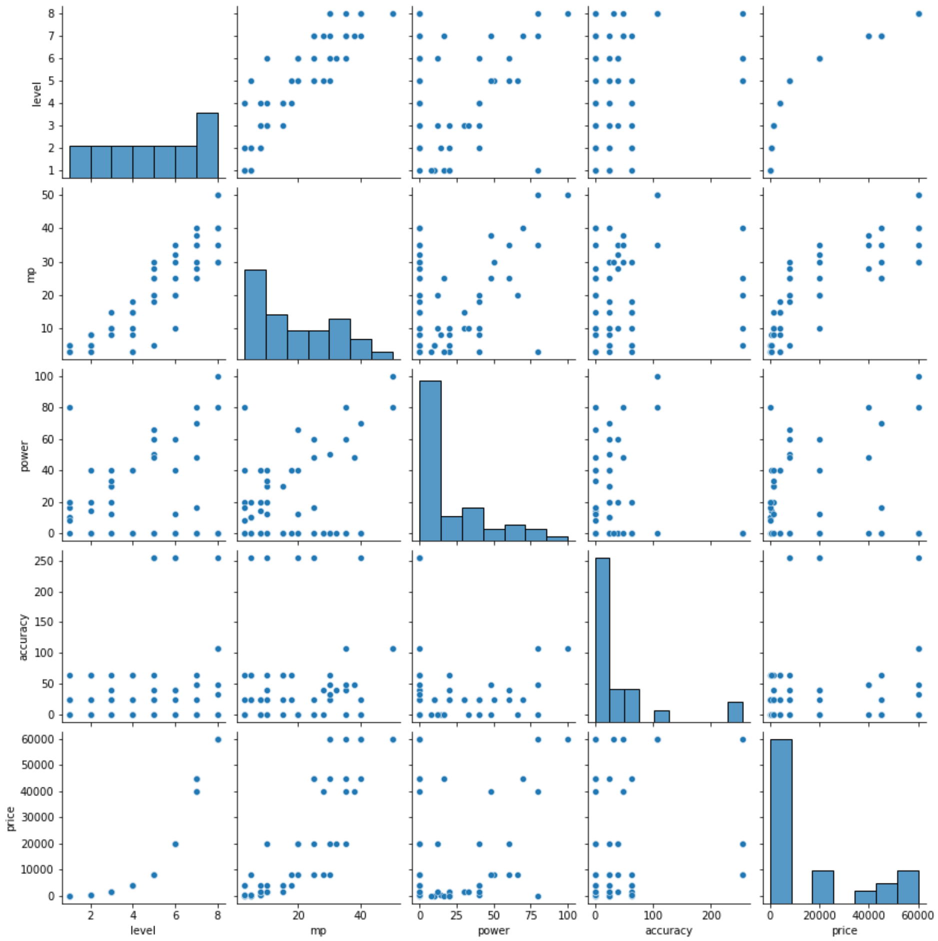

Figure: Pairplot

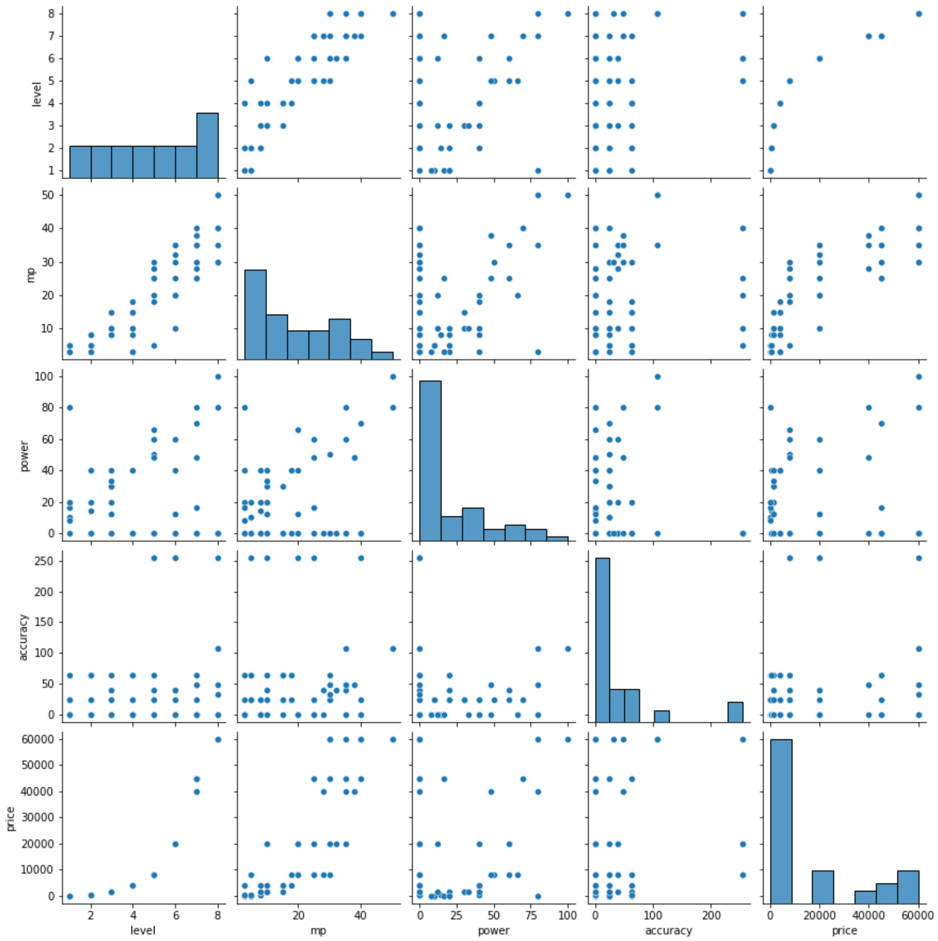

Figure: Pricehistogram

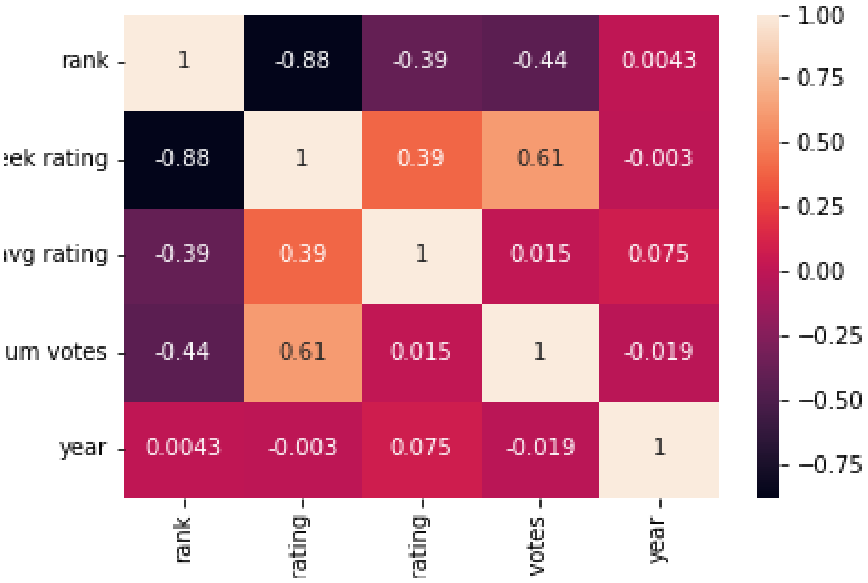

Figure: Corrplot

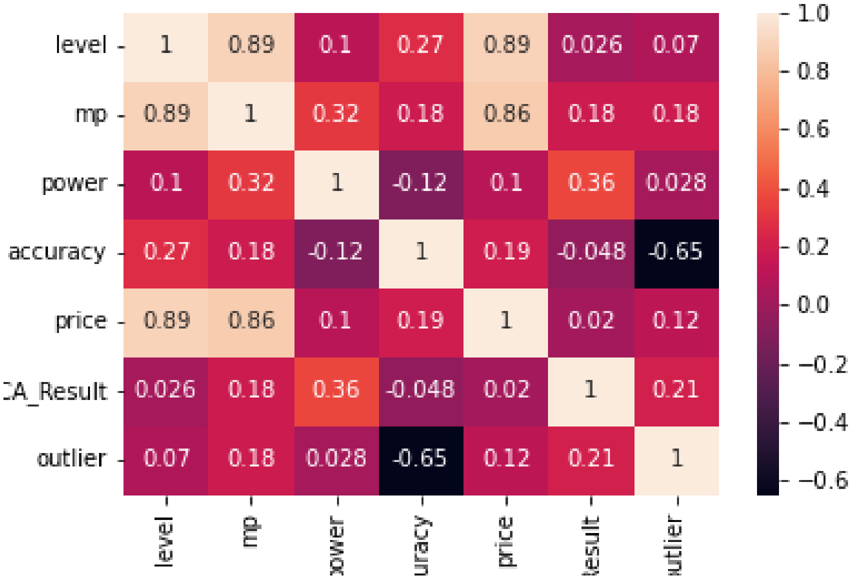

Figure: Pricecorrplot

4. Model Training

Once I had cleaned and explored the data, I moved on to building a predictive model. I split the dataset into training and testing sets using an 80-20 split, ensuring that I had enough data to train the model while still being able to evaluate its performance on unseen data.

For this project, I chose Logistic Regression as the model. Logistic Regression is a solid choice for binary classification problems like this, where we're trying to predict one of two outcomes—survived or not. I trained the model on the training set and then tested it on the test set to see how well it performed.

5. Model Evaluation

To measure the model's performance, I calculated accuracy, precision, recall, and F1-score. Here's what I got:

- Accuracy: 80%
- Precision: 0.79
- Recall: 0.80
- F1-Score: 0.80

An accuracy of 80% means the model correctly predicts survival in 8 out of 10 cases. Precision and recall values suggest that the model is well-balanced in making predictions for both survived and non-survived cases. However, there's definitely room for improvement, and trying different models could lead to even better results.

6. Conclusion

Overall, this project gave me a lot of insights into how machine learning can be applied to real-world problems. I learned that certain factors, like gender and passenger class, played a huge role in survival. The Logistic Regression model performed well, but I could experiment with other models like Random Forest or Neural Networks to see if they yield better accuracy.

One of the biggest takeaways from this project was the importance of data preprocessing. Without handling missing values, encoding categorical variables, and scaling numerical features, the model wouldn't have performed nearly as well. This project also showed me that understanding the data through EDA is just as important as building the model itself.