Titanic Dataset Analysis Report

1. Introduction

The Titanic tragedy is still one of the best-researched maritime disasters in history, and the dataset here provides significant insights on what predicts survival among passengers. The project entailed an analysis of the Titanic dataset to extract survival patterns based on passenger data as well as ticket data. The main aims were to clean the data, manage missing values efficiently, and conduct exploratory data analysis (EDA) to discern influential variables in survival rates.

2. Tools Utilized

- **Python**: Primary programming language utilized for data analysis.
- **Pandas**: Utilized for data cleaning and manipulation.
- **NumPy**: Utilized for numerical computation and missing data handling.
- **Matplotlib & Seaborn**: Utilized for plotting distributions and data visualization.
- **Jupyter Notebook**: Utilized for interactive coding and step-by-step analysis.

3. Steps in Creating the Project

Data Cleaning

- Recognized missing values in columns like *Age*, *Cabin*, and *Embarked*.
- Imputed missing *Age* values with the median age to preserve distribution integrity.
- Removed the *Cabin* column due to high levels of missing data.
- Imputed missing *Embarked* values with the mode (most frequent embarkation point).
- Transformed categorical variables such as *Sex* and *Embarked* into suitable formats for analysis.

Charts used:

- **Bar Plots / Count Plots**: To show counts of categorical variables such as survival counts by gender, passenger class (Pclass), and embarkation port. These plots helped reveal differences in survival rates across groups.

- **Histograms with KDE (Kernel Density Estimate)**: To visualize the age distribution of passengers, comparing survivors and non-survivors.
- **Pie Charts**: To depict proportions of passengers by embarkation port.
- **Box Plots**: To explore the distribution of fares paid across different passenger classes and survival status.
- **Scatter Plots**: To examine relationships between continuous variables, such as Age vs. Fare.
- **Heatmaps**: To visualize correlation matrices among numerical features, identifying potential relationships.

These visualizations, created using libraries like Matplotlib and Seaborn, were instrumental in uncovering patterns such as higher survival rates among females, first-class passengers, and children, as well as fare-related trends.

Exploratory Data Analysis (EDA)

- Computed overall survival rate (~38%).
- Examined survival rates by gender, and found females had much higher survival probabilities than males.
- Analysed passenger class (Pclass), reporting greater survival in first- and second-class passengers.
- Researched age profile of survivors and non-survivors:
    - Survivors were on average younger than non-survivors.
- Researched fare paid, reporting passengers who paid more were more likely to survive.

4. Conclusion

The Titanic dataset analysis, following extensive data cleaning and exploratory analysis, identified a number of key factors determining survival. The most important predictors were gender and passenger class, with the highest survival rates for females and first-class passengers. Age was also an important factor, with children having significantly higher survival probabilities, consistent with evacuation priorities at the time of the disaster. The project illustrates the capacity of data cleansing and EDA to reveal rich insights from old datasets, leading to a potential platform for deeper statistical exploration or further predictive modelling.

jupyter Titanic_analysis Last Checkpoint: 5 days ago

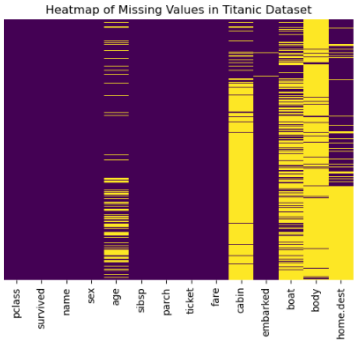
File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python [conda env:base] *

[3]:
```
import seaborn as sns
sns.heatmap(titanic.isnull(), yticklabels=False, cbar=False, cmap='viridis')
plt.title('Heatmap of Missing Values in Titanic Dataset')
plt.show()
```



[4]:
```
sns.set_style('darkgrid')
palette = ['#ffb482', '#e5b5f7']
sns.countplot(x='survived' , hue='sex', data= titanic, palette= palette)
plt.show()
```

jupyter Titanic_analysis Last Checkpoint: 5 days ago

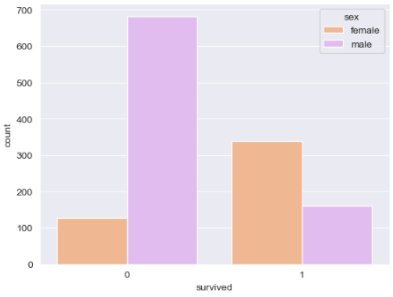
File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python [conda env:base] *

[4]:
```
sns.set_style('darkgrid')
palette = ['#ffb482', '#e5b5f7']
sns.countplot(x='survived' , hue='sex', data= titanic, palette= palette)
plt.show()
```



[5]:
```
print("\nDetailed Info:")
print(titanic.info())

print("\nSummary Statistics:")
print(titanic.describe())

print("\nColumn Names and Data Types:")
print(titanic.dtypes)
```

jupyter Titanic_analysis Last Checkpoint: 5 days ago

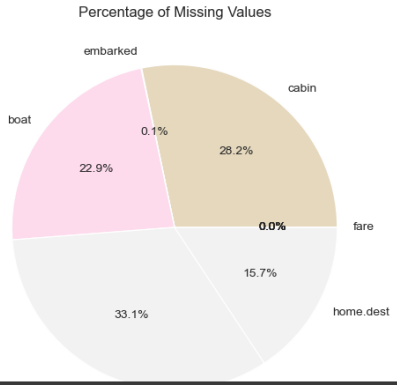
File Edit View Run Kernel Settings Help

Trusted

Code

JupyterLab Python [conda env:base] *

```
missing_percent = titanic.isnull().sum() / len(titanic) * 100
missing_percent.plot(kind='pie', autopct='%1.1f%%', figsize=(6, 6), colormap='Pastel1')
plt.title("Percentage of Missing Values")
plt.ylabel("")
plt.show()
```



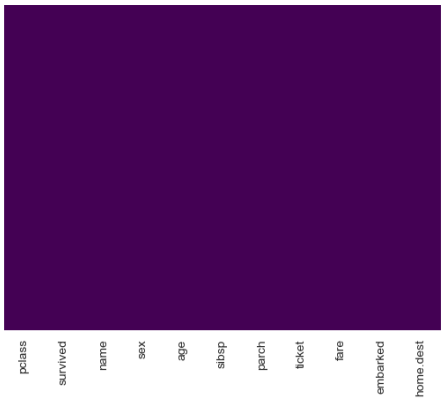
jupyter Titanic_analysis Last Checkpoint: 5 days ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base]

[18]:
```
titanic.dropna(inplace=True)
```

[19]:
```
sns.heatmap(titanic.isnull(), yticklabels=False, cbar=False, cmap='viridis')
plt.show()
```

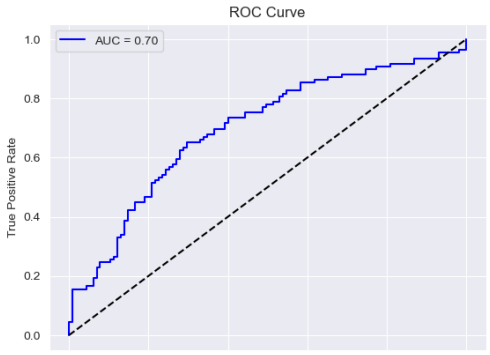


jupyter Titanic_analysis Last Checkpoint: 5 days ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base]

```
plt.plot([0, 1], [0, 1], 'k--')  # Diagonal line
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curve')
plt.legend()
plt.show()
```



jupyter Titanic_analysis Last Checkpoint: 5 days ago

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python [conda env:base]

[40]:
```
data = titanic
sns.regplot(x='age', y='fare', data=data, line_kws={"color": "red", "alpha": 0.7, "lw": 2})
plt.xlabel("Age")
plt.ylabel("Fare")
plt.title("Regression Chart: Age vs Fare")
plt.show()
```