Spacetime Anomaly Passenger Recovery: Predicting Lost Passengers of the Spaceship Titanic

1. Project Overview

This project leverages data from the Spaceship Titanic dataset on Kaggle to solve a binary classification problem. The dataset contains simulated records of passengers aboard the Spaceship Titanic, including demographics, travel details, and onboard interactions. The goal is to predict which passengers were transported to an alternate dimension due to a spacetime anomaly. This project utilizes **supervised learning** techniques such as Logistic Regression, Random Forest, and Gradient Boosting to classify passengers based on the available data.

2. Dataset Description

The dataset consists of passenger records from the Spaceship Titanic. Each row represents an individual passenger, and the columns provide various attributes about the passengers, their travel details, and other related information. The data is structured in two main files:

1. **Train.csv** (8693 rows, 14 columns):
    - Contains the labeled training data with features and the target variable `Transported`.
    - The `Transported` column is a binary indicator (True/False) of whether a passenger was transported to the alternate dimension.
2. **Test.csv** (4277 rows, 13 columns):
    - Contains the unlabeled test data without the `Transported` column.
    - Used for model evaluation and prediction.

Features in the Dataset

1. **PassengerId**: A unique identifier for each passenger (e.g., "0001_01" where "0001" is a group identifier and "01" is an individual within the group).
2. **HomePlanet**: The planet the passenger departed from (e.g., Earth, Europa, Mars).
3. **CryoSleep**: A boolean indicating whether the passenger opted for cryogenic sleep during the journey.
4. **Cabin**: The cabin number, including deck and side of the ship (e.g., "B/45/P").
5. **Destination**: The final destination of the passenger (e.g., TRAPPIST-1e, 55 Cancri e, PSO J318.5-22).
6. **Age**: The passenger's age in years.
7. **VIP**: A boolean indicating whether the passenger paid for special VIP services.
8. **RoomService**: The amount billed for room service.
9. **FoodCourt**: The amount billed for food court purchases.
10. **ShoppingMall**: The amount billed for shopping mall purchases.
11. **Spa**: The amount billed for spa services.
12. **VRDeck**: The amount billed for virtual reality deck services.

13. **Name**: The name of the passenger (e.g., "Doe, John").
14. **Transported**: The target variable indicating whether the passenger was transported to the alternate dimension (True/False).

3. Data Cleaning

3.1. Overview

The dataset comprises passenger information for a fictional Titanic spaceship. The primary objective of the data cleaning process was to prepare the dataset for analysis by handling missing values, standardizing formats, and ensuring consistency across columns.

Check for missing values:

Column	Missing Values	Percentage
CryoSleep	217	2.496261
ShoppingMall	208	2.39273
VIP	203	2.335212
HomePlanet	201	2.312205
Name	200	2.300702
Cabin	199	2.289198
VRDeck	188	2.16266
FoodCourt	183	2.105142
Spa	183	2.105142
Destination	182	2.093639
RoomService	181	2.082135
Age	179	2.059128
PassengerId	0	0
Transported	0	0

We have 3 type of columns:

- Numeric columns: Age, RoomService, FoodCourt, ShoppingMall, Spa, VRDeck
- Binary columns: CryoSleep, VIP, Transported
- Category columns: PassengerId, HomePlanet, Cabin, Destination, Name

Columns that have missing values: HomePlanet, CryoSleep, Cabin, Destination, Age, VIP, RoomService, FoodCourt, ShoppingMall, Spa, VRDeck, Name

Number of uniques for each category column:

Column	Number of unique values
HomePlanet	3
PassengerId	8693
Cabin	6560
Destination	3

3.2. Remove unnecessary columns

Columns PassengerId and Name have too many unique values. So I decided to drop them

3.3. Handling Missing Values

- Several columns contained missing values, including Cabin, HomePlanet, CryoSleep, Destination, Age, VIP, RoomService, FoodCourt, ShoppingMall, Spa, VRDeck

  Techniques Applied:
- Missing categorical and binary columns (e.g., HomePlanet, CryoSleep, Destination, VIP): Imputed with the most frequent value (mode).
- Numerical columns (e.g., Age, RoomService, FoodCourt, ShoppingMall, Spa, VRDeck): Set their values to 0

3.4. Splitting and Standardizing the category columns

- Split Cabin into 3 columns Deck, Number and Side using delimiter '/'
- Apply One-Hot Encoding to these columns HomePlanet, Destination, Deck and Side

3.5. Convert binary columns to numeric columns

Columns affected: VIP, Transported, CryoSleep

3.6. Outlier Detection and Removal

- Focus on Numerical Data
- Using Z-Score to remove the data points that have z-score > 3 or < -3

3.7. Normalization

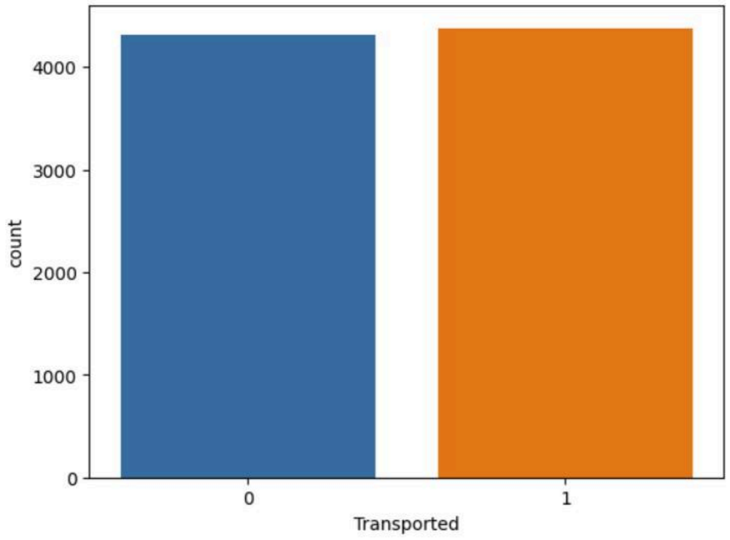
Scale all the features to range [0, 1] using MinMaxScaler from sklearn

Discussion

After completing the data cleaning process, we converted all features to numeric values, removed unnecessary columns, and handled missing data. The dataset is now ready for the classification step.

4. Exploratory Data Analysis

Check target variable balance

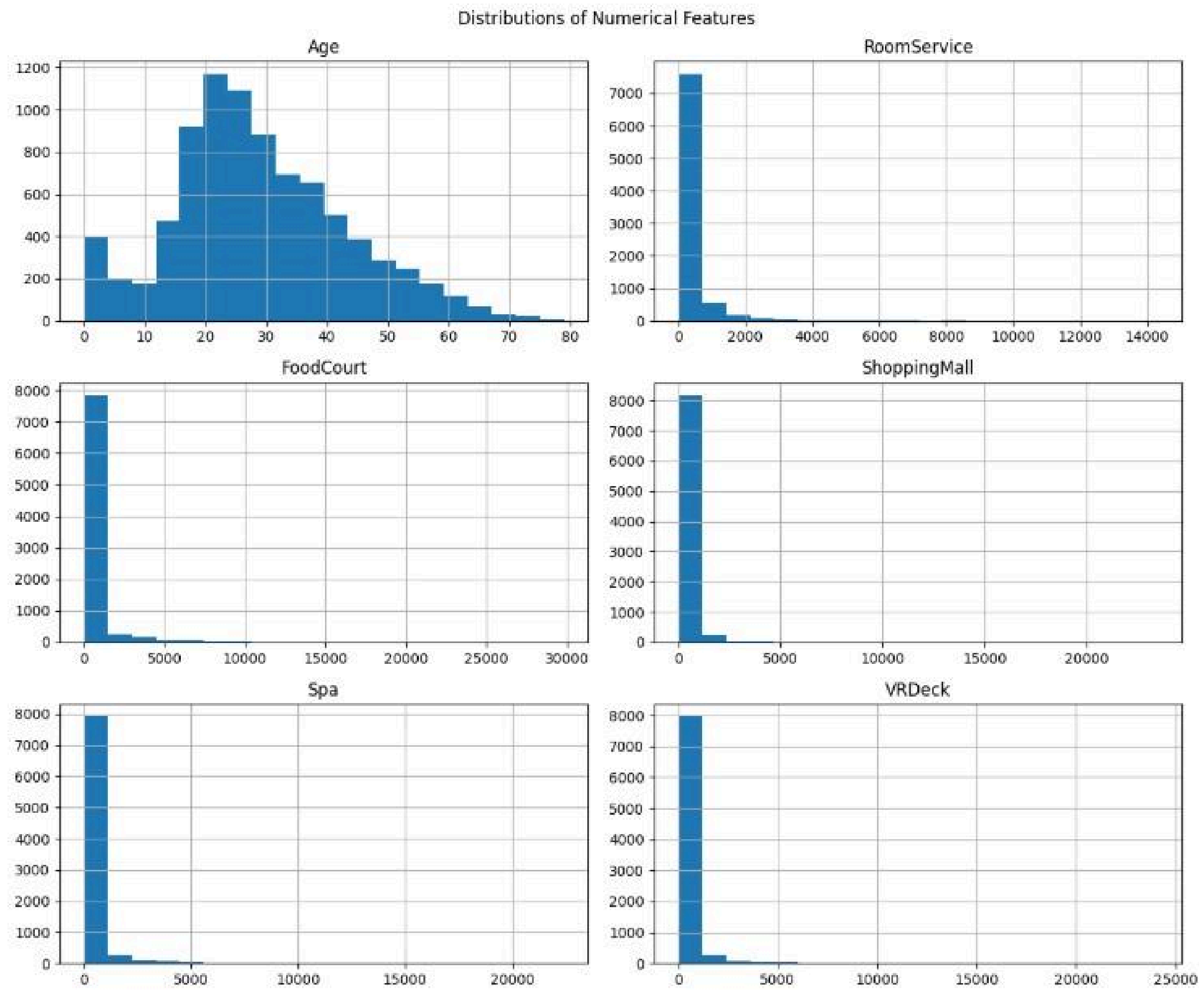


The target variable, Transported, is evenly distributed, with nearly 50% of the passengers transported and 50% not transported. This balance ensures that models won't need extensive handling for class imbalance.

Balanced classes make accuracy, precision, recall, and F1-score equally important as evaluation metrics.

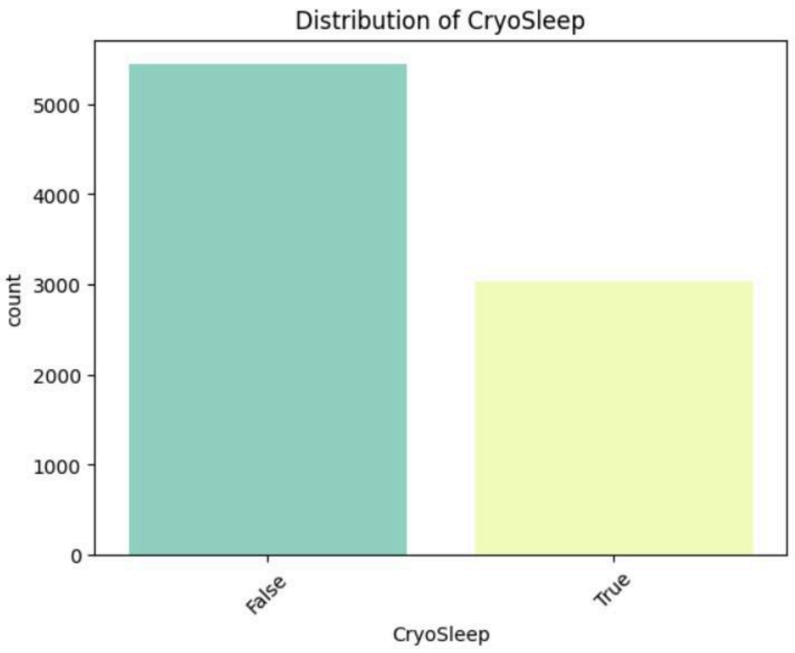
The dataset is well-suited for models like Logistic Regression and Decision Trees without additional sampling techniques.

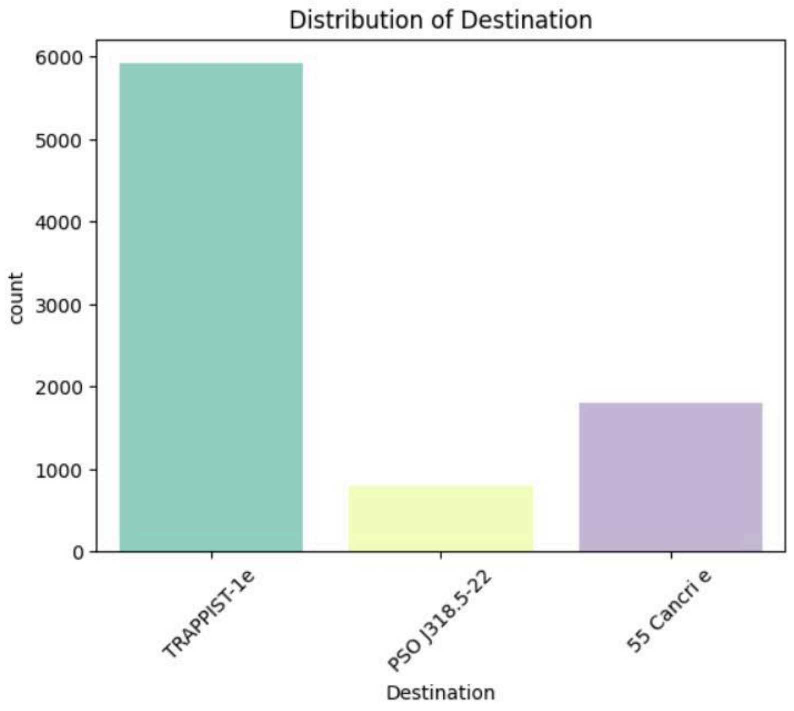
Visualize Data Distributions of Numerical Features

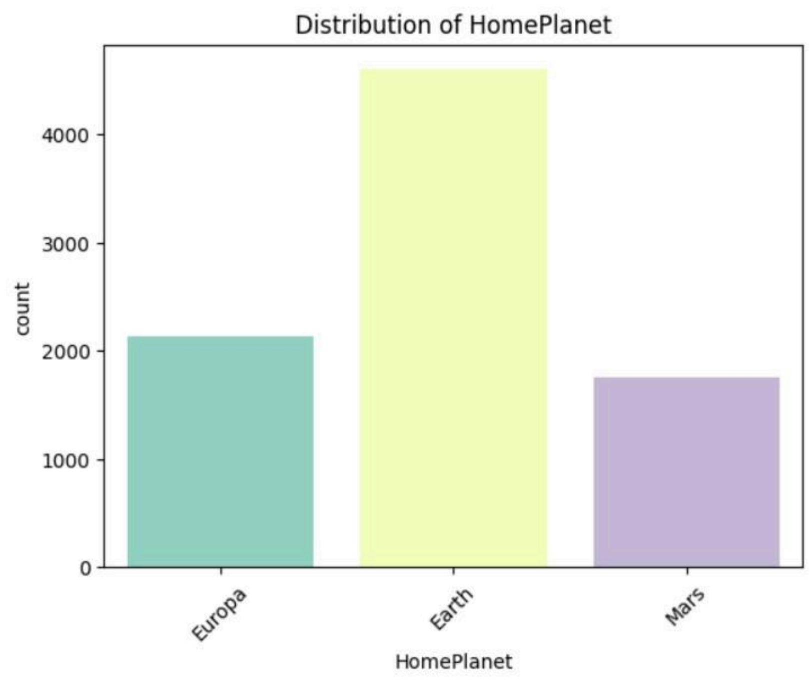


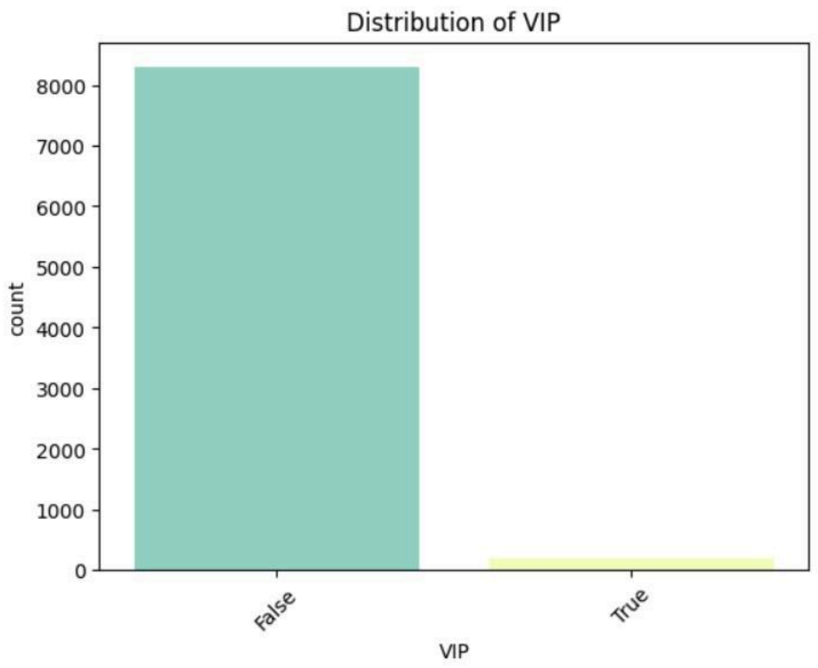
- **Age**: Distribution is slightly right-skewed, with a significant proportion of younger passengers.
- **Spending Features**: Many passengers spent little or nothing on services (`RoomService`, `FoodCourt`, `ShoppingMall`, `Spa`, `VRDeck`), with a few high spenders causing a long tail.
- The skewed spending distributions suggest the need for transformations (e.g., log or square root) to normalize the data for models that assume normality.

Visualize Data Distributions of Categorical Features


Distribution of CryoSleep
count
5000
4000
3000
2000
1000
0
False
True
CryoSleep


Distribution of Destination
count
6000
5000
4000
3000
2000
1000
0
TRAPPIST-1e
PSO J318.5-22
55 Cancri e
Destination





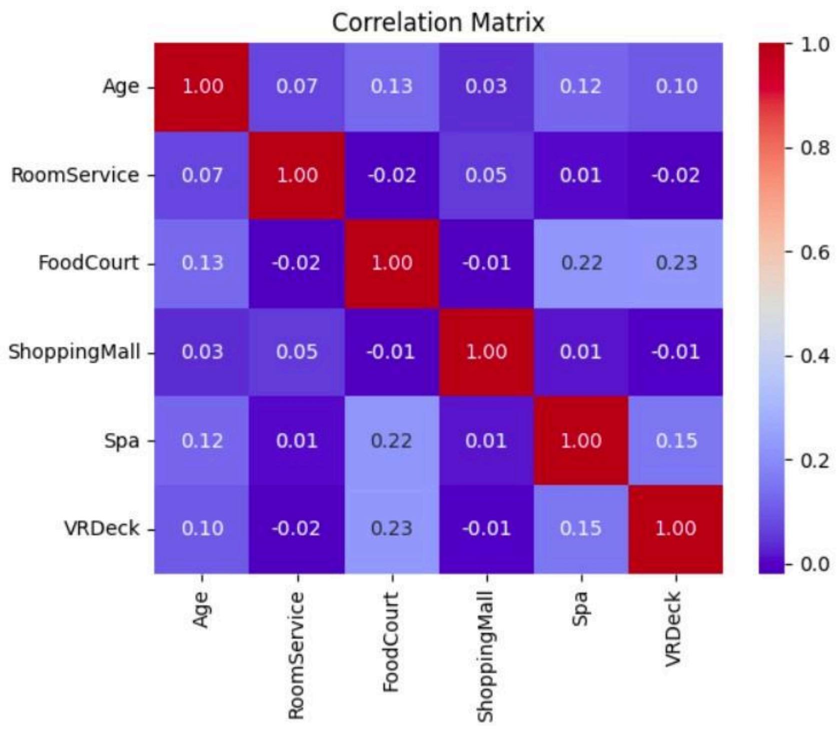
HomePlanet: Most passengers are from `Earth`, followed by `Europa` and `Mars`.

Destination: The majority are traveling to `TRAPPIST-1e`.

CryoSleep: Many passengers are not in cryogenic sleep, with some missing data.

VIP: Only a small fraction of passengers are VIP.

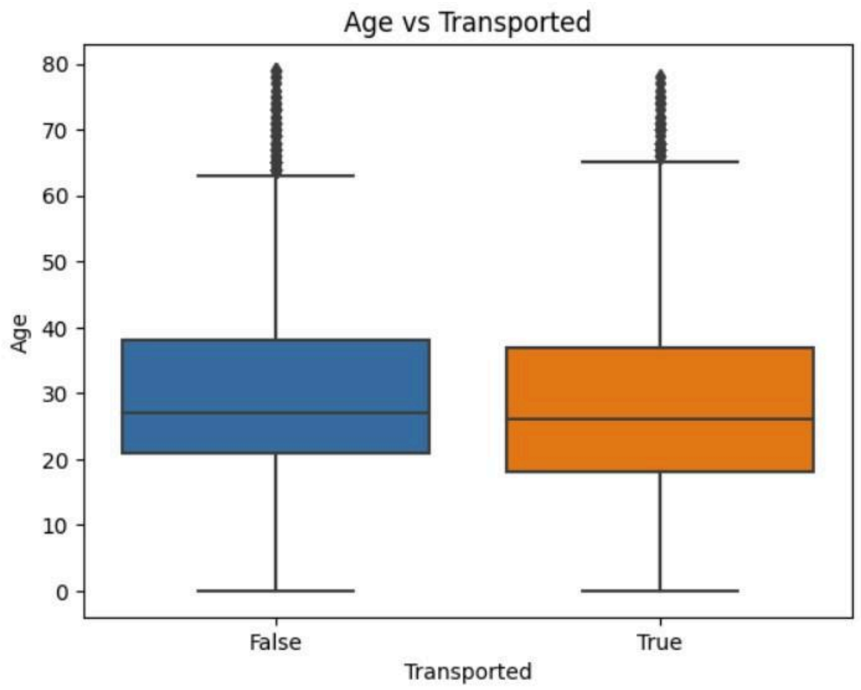
Correlation Matrix of numerical columns

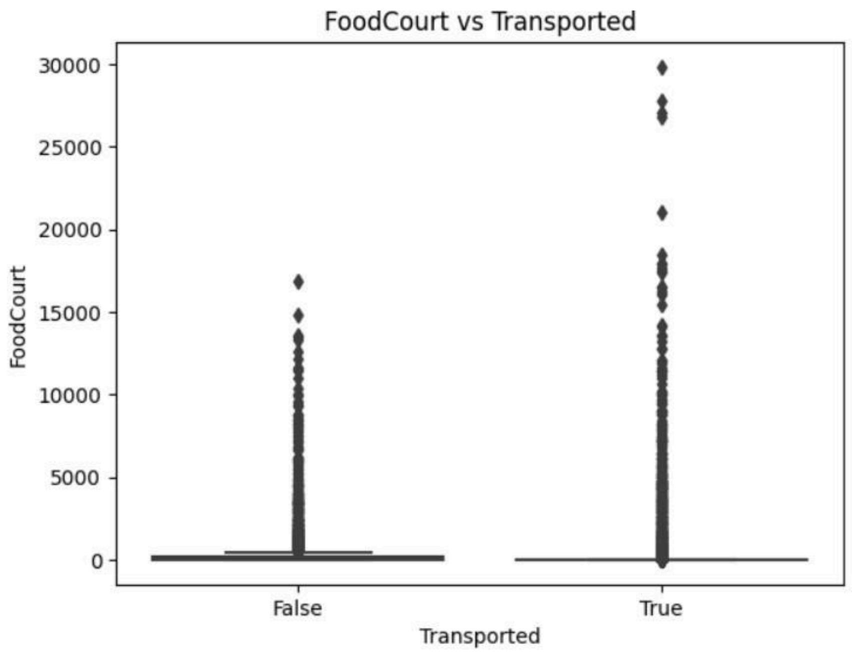


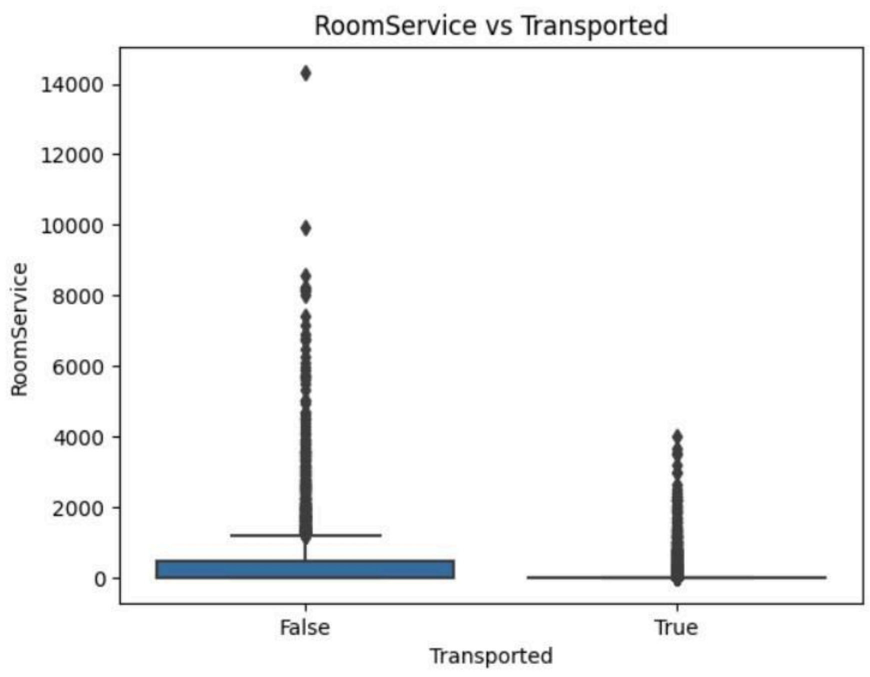
	Age	RoomService	FoodCourt	ShoppingMall	Spa	VRDeck
Age	1.00	0.07	0.13	0.03	0.12	0.10
RoomService	0.07	1.00	-0.02	0.05	0.01	-0.02
FoodCourt	0.13	-0.02	1.00	-0.01	0.22	0.23
ShoppingMall	0.03	0.05	-0.01	1.00	0.01	-0.01
Spa	0.12	0.01	0.22	0.01	1.00	0.15
VRDeck	0.10	-0.02	0.23	-0.01	0.15	1.00

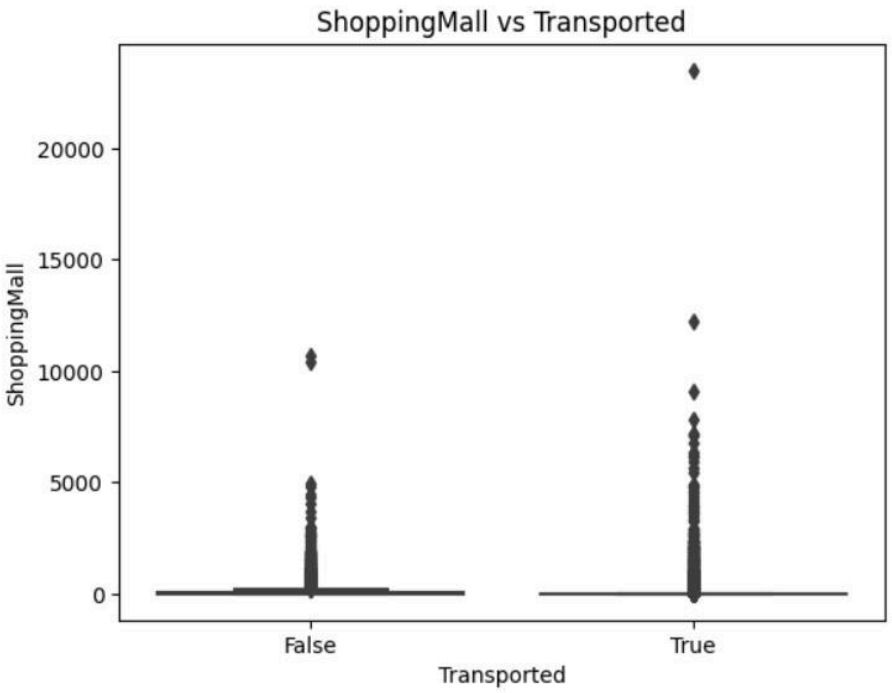
There are weak correlations between these features. So, we should keep all of them

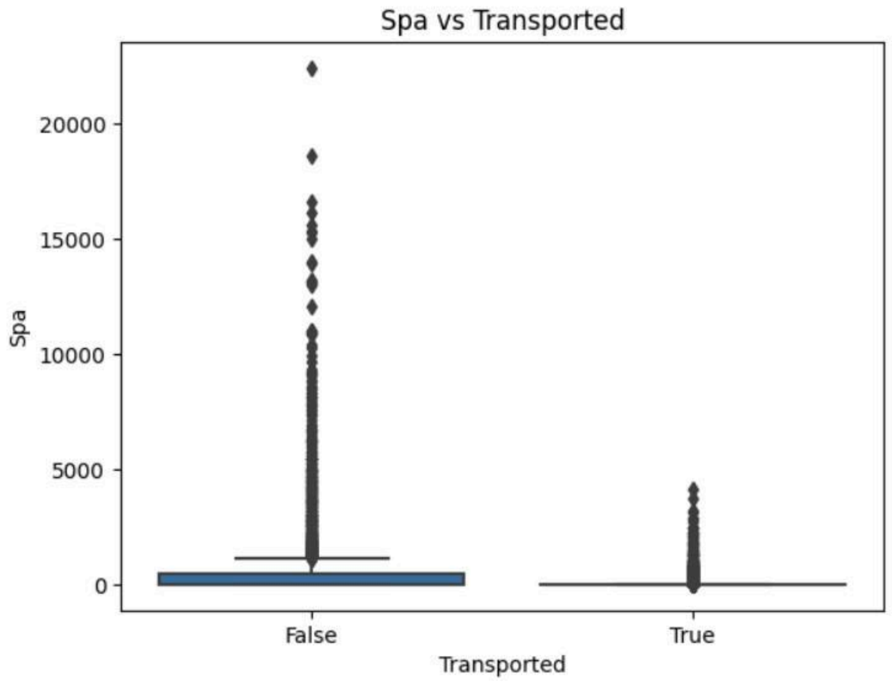
Relationship between numerical features with Target Variable

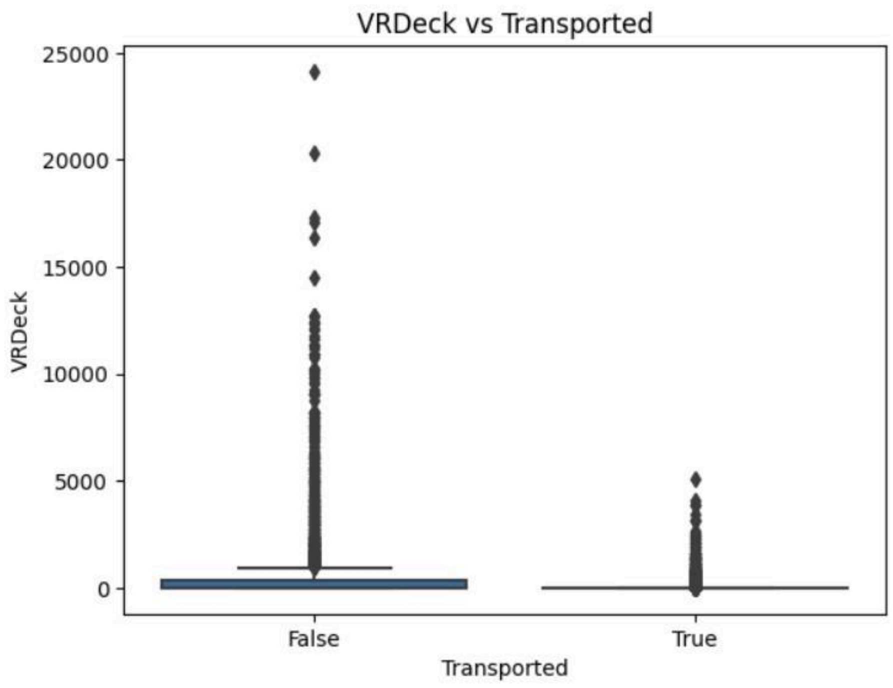

Age vs Transported
Age
80
70
60
50
40
30
20
10
0
False
True
Transported


FoodCourt vs Transported
FoodCourt
30000
25000
20000
15000
10000
5000
0
False
True
Transported


RoomService vs Transported
RoomService
Transported
False
True
0
2000
4000
6000
8000
10000
12000
14000

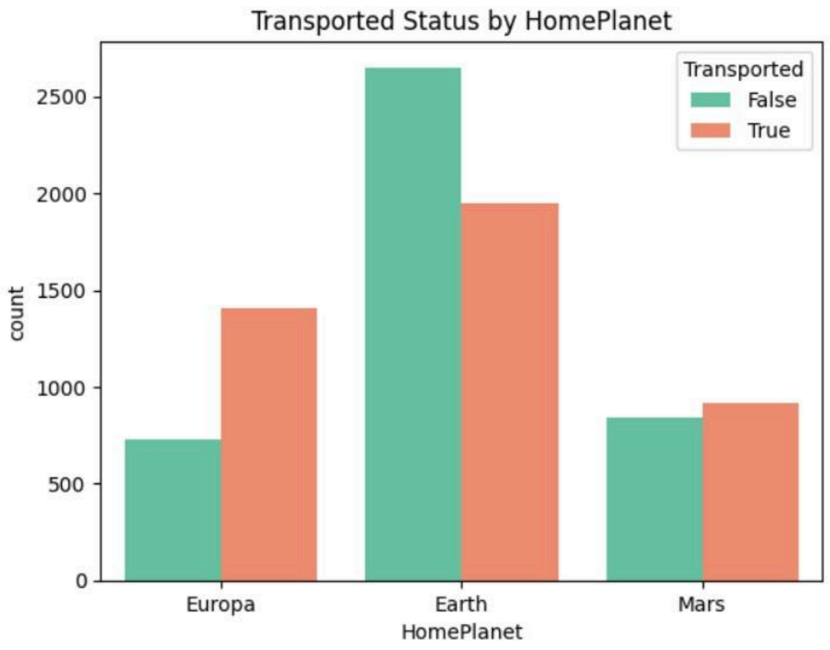

ShoppingMall vs Transported
ShoppingMall
20000
15000
10000
5000
0
False
True
Transported

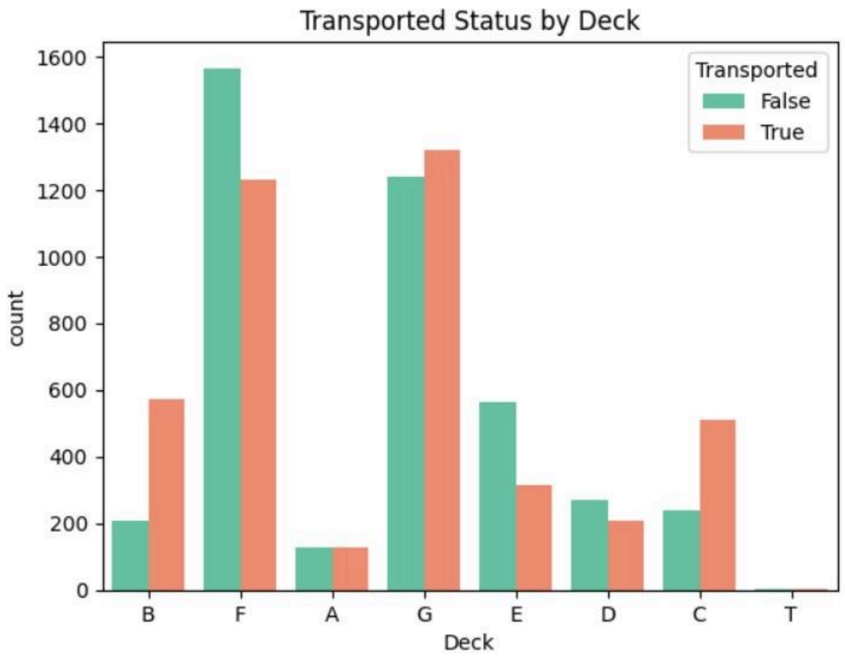

Spa vs Transported
Spa
20000
15000
10000
5000
0
False
True
Transported



- Passengers who spent high (> 5000) on services: `RoomService`, `Spa`, `VRDeck` were **less** likely to be transported.
- Passengers who spent high on services: `FoodCourt > 20000`, `ShoppingMall > 14000` were **more** likely to be transported.
- Age shows no significant difference between transported and non-transported passengers.

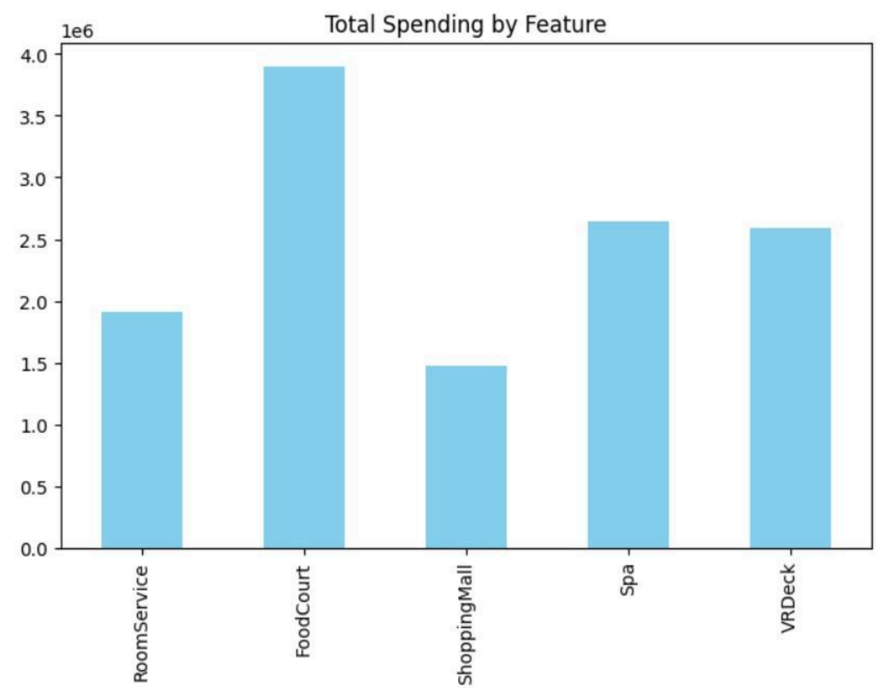
Relationship between categorical features with Target Variable



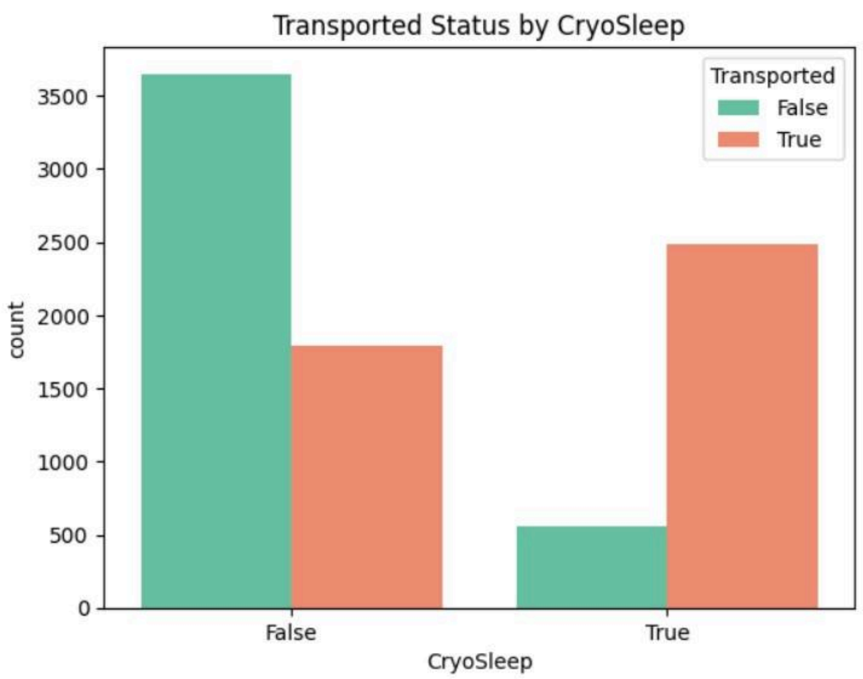


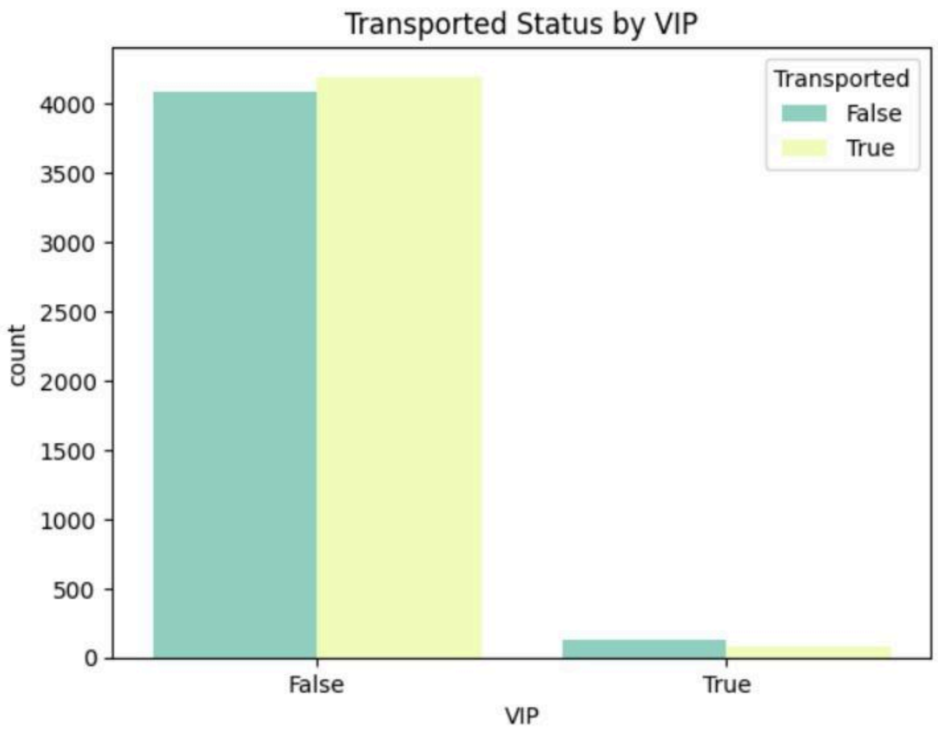
There are no obvious relationships between these variables vs Transported

Analyze features like RoomService, FoodCourt, ShoppingMall, Spa, and VRDeck to identify trends in passenger spending.



Analyze CryoSleep and VIP





Passengers in cryogenic sleep are more likely to be transported.
VIP passengers do not show a clear trend with respect to Transported.
Discussion:

- Cryogenic sleep has a strong predictive relationship, likely linked to reduced interactions during the journey.
- VIP status might have minimal direct impact but could interact with other features like spending or cabin type.

Conclusion on EDA:

- The features in the dataset show minimal correlation with each other, suggesting that multicollinearity is not a concern. Therefore, all features have been retained for modeling. Additionally, the weak correlation between features and the target variable indicates that relationships may be non-linear. As such, classifiers that can capture non-linear patterns, such as Decision Trees, AdaBoost, Random Forest, and Support Vector Machines (SVM) with non-linear kernels, are suitable choices. Logistic Regression may also be used as a baseline for comparison, despite its linear nature.

5. Modelling

5.1. Choice of Models

The selection of Logistic Regression, Decision Tree, AdaBoost, Random Forest, and SVM is well-suited for this binary classification task. These models address both linear and non-linear relationships, making the approach robust. I also tried the experiment with XGBoost Classifier. The inclusion of ensemble methods like AdaBoost, Random Forest and XGBoost highlights an effort to improve performance.

5.2. Techniques to Reduce Overfitting and Handle Data Imbalance

- **Overfitting**:
    - Regularization is applied in Logistic Regression (`C` parameter).
    - Tree-based models use depth limits (`max_depth`) and minimum sample constraints (`min_samples_split`, `min_samples_leaf`).
    - SVM uses regularization (`C`) and kernel parameters.
      These strategies effectively mitigate overfitting.
- **Data Imbalance**:
  The target variable is evenly distributed, so imbalance handling is unnecessary.

5.3. Model Performance Comparison

Model	Train Accuracy	Validation Accuracy	Test Accuracy
Logistic Regression	79.52%	78.26%	79.17%
Decision Tree	77.29%	76.19%	76.88%
AdaBoost	80.99%	80.16%	79.85%
Random Forest	81.00%	79.47%	80.08%
SVM	80.14%	78.95%	80.27%

XGBoost	81.19%	79.87%	80.01%

5.4. Discussion

- **Best Model**: SVM slightly outperforms others on the test set, with Random Forest being a close second.
- **Trade-offs**: Random Forest provides more interpretability than SVM and might be preferred if explainability is crucial.
- **Next Steps**: Further improve models by feature engineering, especially for spending-related variables, and exploring advanced techniques like boosting variants or neural networks if computational resources allow.