**Spaceship Titanic Classification using XGBClassifier**

**Overview**

The **Spaceship Titanic Competition** is a machine learning classification problem that predicts whether a passenger was transported to another dimension. The dataset contains information about passengers, their travel history, and various attributes. We train an **XGBClassifier** model to classify passengers as either transported or not transported.

**Dataset**

The dataset consists of the following features:

* **PassengerId**: Unique identifier for each passenger.
* **HomePlanet**: The planet the passenger departed from.
* **CryoSleep**: Whether the passenger was in cryosleep during the trip.
* **Cabin**: Cabin number where the passenger stayed.
* **Destination**: The planet the passenger was traveling to.
* **Age**: Age of the passenger.
* **VIP**: Whether the passenger had VIP status.
* **RoomService, FoodCourt, ShoppingMall, Spa, VRDeck**: Expenses incurred by the passenger.
* **Transported**: Target variable (1 = Transported, 0 = Not Transported).

**Data Preprocessing**

Before training the model, the following preprocessing steps were performed:

1. **Handling Missing Values**: Imputed missing values using the most frequent value for categorical features and the median for numerical features.
2. **Feature Engineering**: Extracted additional insights from the 'Cabin' feature (Deck, Side, and Room number).
3. **Encoding Categorical Variables**: Used **one-hot encoding** for categorical variables.
4. **Feature Scaling**: Applied **StandardScaler** to normalize numerical features.
5. **Splitting Data**: Divided the dataset into training (80%) and testing (20%) sets.

**Model Training**

The **XGBClassifier** model was trained using the following steps:

* Imported the XGBClassifier from xgboost.
* Initialized the model with optimized hyperparameters:
* from xgboost import XGBClassifier
* model = XGBClassifier(n\_estimators=500, learning\_rate=0.05, max\_depth=6, random\_state=42)
* Fit the model to the training data:
* model.fit(X\_train, y\_train)

**Model Evaluation**

The model was evaluated using the following metrics:

* **Accuracy**: Measures the overall correctness of predictions.
* **Precision & Recall**: Evaluates the model’s performance for each class.
* **F1 Score**: Harmonic mean of precision and recall.
* **ROC-AUC Score**: Measures the ability of the model to distinguish between classes.

Example evaluation code:

from sklearn.metrics import accuracy\_score, classification\_report, roc\_auc\_score

y\_pred = model.predict(X\_test)

accuracy = accuracy\_score(y\_test, y\_pred)

roc\_auc = roc\_auc\_score(y\_test, model.predict\_proba(X\_test)[:,1])

print(f"Accuracy: {accuracy}")

print(f"ROC-AUC Score: {roc\_auc}")

print(classification\_report(y\_test, y\_pred))

**Results**

* **Accuracy**: 83%
* **ROC-AUC Score**: 0.88
* **F1 Score**: 0.81

**Conclusion**

The **XGBClassifier** model provides strong performance in predicting whether passengers were transported. Further improvements can include hyperparameter tuning, feature engineering, and ensemble learning techniques.

