Insights Report – Model Training & Hyperparameter Tuning

1. Dataset Overview

The Titanic dataset was used for this task. It contains demographic and travel details of passengers, with the goal of predicting survival (0 = No, 1 = Yes). After loading the dataset, irrelevant columns such as PassengerId, Name, Ticket, and Cabin were dropped to focus on meaningful features.

2. Data Cleaning and Preprocessing

Missing values in numeric columns (e.g., Age, Fare) were filled with their median values. Categorical features (like Sex and Embarked) were filled with the mode. All categorical variables were converted to numeric format using one-hot encoding, producing binary columns suitable for machine learning models.

3. Train-Test Split

The data was split into training and testing sets using stratified sampling to maintain the balance of survived vs. non-survived classes. 80% of data was allocated for training and 20% for testing.

4. Random Forest Classifier

A Random Forest model was trained using RandomizedSearchCV for hyperparameter tuning. Parameters tuned included the number of trees, tree depth, and minimum samples per split or leaf. The optimal parameters were chosen based on the ROC-AUC score using 5-fold cross-validation. The best model was saved to disk as 'model.pkl' for future deployment.

5. Support Vector Machine Classifier

A Support Vector Machine model was developed using a pipeline that included StandardScaler for feature scaling followed by the SVC classifier. GridSearchCV was used to search combinations of C, kernel type, and gamma. The optimal SVM model achieved strong ROC-AUC scores and was ready for comparison with other models.

6. Feature Engineering Notes

Features included numeric variables such as Age, Fare, SibSp, and Parch, as well as encoded categorical variables like Sex_male, Embarked_Q, and Embarked_S. These features provided the model with diverse information for survival prediction.

7. Conclusion

The models were successfully trained and optimized. Random Forest and SVM were both fine-tuned for hyperparameters, demonstrating solid predictive performance. The Random Forest model was exported as a .pkl file for deployment in an API or further analysis.