Model Development Phase Template

Date	11 July 2024
Team ID	SWTID1720369851
Project Title	Ecommerce Shipping Prediction
Maximum Marks	6 Marks

Model Selection Report

In the forthcoming Model Selection Report, various models will be outlined, detailing their descriptions, hyperparameters, and performance metrics, including Accuracy or F1 Score. This comprehensive report will provide insights into the chosen models and their effectiveness.

Model Selection Report:

Model	Description	Hyperparameters	Performance Metric (e.g., Accuracy, F1 Score)
Logistic Regression	It's a type of regression analysis that estimates the probability that a given input point belongs to a certain class.	penalty = ['l2','l1','elasticnet'] C = [0.0001, 0.001, 0.002] solver = ['newton-cg', 'lbfgs', 'liblinear', 'sag', 'saga'] hyperparameters = dict(penalty=penalty, C=C , solver=solver)	Confusion Matrix : [[531 377] [437 855]] Accuracy_Score: 63.0 % F1 Score: 67.75 Precision Score: 69.399 Recall Score : 66.176 AUC Score : 62.328

Decision Tree Classifier	It uses a tree-like model of decisions and their possible consequences, including outcomes, resource costs, and utility.	max_depth = [int(x) for x in np.linspace(1, 110, num = 30)] min_samples_split = [2, 5, 10, 100] min_samples_leaf = [1, 2, 4, 10, 20, 50] max_features = ['auto', 'sqrt'] criterion = ['gini', 'entropy'] splitter = ['best', 'random'] hyperparameters = dict(max_depth=max_depth, min_samples_split=min_samples_split, min_samples_leaf=min_samples_leaf, max_features=max_features, criterion=criterion, splitter=splitter)	Confusion Matrix : [[508 400] [392 900]] Accuracy_Score: 64.0 % F1 Score: 69.444 Precision Score: 69.231 Recall Score : 69.659 AUC Score : 62.803

Random Forest	Random Forest is an ensemble learning method that constructs multiple decision trees during training and merges their outputs to improve classification accuracy and control overfitting.	**NA**	Confusion Matrix : [[611 297] [446 846]] Accuracy_Score: 66.227 % F1 Score: 69.487 Precision Score: 74.016 Recall Score : 65.48 AUC Score : 66.385
SVM	Support Vector Machine (SVM) is a powerful supervised learning algorithm used for classification and regression tasks.	**NA**	Confusion Matrix : [[566 342] [402 890]] Accuracy_Score: 66.182 % F1 Score: 70.523 Precision Score: 72.24 Recall Score : 68.885 AUC Score : 65.61
XGBoost	XGBoost (Extreme Gradient Boosting) is a highly efficient and scalable machine learning algorithm for classification and regression tasks.	**NA**	Confusion Matrix : [[567 341] [442 850]] Accuracy_Score: 64.409 % F1 Score: 68.466 Precision Score: 71.369 Recall Score : 65.789 AUC Score : 64.117

KNN	K-Nearest Neighbors (KNN) classifies a data point based on the majority class of its k-nearest neighbors in the feature space.	n_neighbors = list(range(2,30)) p=[1,2] algorithm = ['auto', 'ball_tree', 'kd_tree', 'brute'] hyperparameters = dict(n_neighbors=n_neighbors, p=p, algorithm=algorithm)	Confusion Matrix : [[556 352] [409 883]] Accuracy_Score: 65.409 % F1 Score: 69.885 Precision Score: 71.498 Recall Score : 68.344 AUC Score : 64.789