REPORT

Himanshu, MDS202327

Task 1

Steps followed:

1. Dropped the CustomerID column
2. Encoded the three qualitative features i.e. Gender, PromotionResponse, EmailOptIn using OneHotEncoder()
3. Splitted the encoded data into train and test data (70-30)
4. Fitted the AdaBoostClassifier() using GridSearchCV with 5-fold cross-validation with scoring set as recall having hyper-parameters as
    a. estimator - A weak classifier
        i. DecisionTreeClassifier(max_depth=1)
        ii. RandomForestClassifier(max_depth=1)
        iii. GaussianNB()
    b. learning_rate - 0.5, 1, 1.5, 2, 2.5
5. Fitted the RandomForestClassifier() using GridSearchCV with 5-fold cross-validation with scoring set as recall having hyper-parameters as
    a. n_estimators – 50, 100, 200, 500
    b. criterion – gini, entropy, log_loss
    c. max_features – sqrt, log2
6. Computed the Confusion Matrix, Overall Accuracy, Precision and Recall

Results:

Model	Parameters	Overall Accuracy	Precision	Recall
Ada Boost	learning_rate: 2.5	0.527	0.530	0.981
Random Forest	criterion: entropy max_features: sqrt n_estimators: 500	0.500	0.528	0.581

Conclusion:

AdaBoost fitted with weak estimator as DecisionTreeClassifier(max_depth=1) gave the best results across the categories with high recall which is important when calculating customer churn.

Task 2

Sub Task A: Gender

Steps followed:

1. Dropped the InvoiceID column
2. Encoded the three qualitative features i.e. CustomerType, ProductType, PaymentType and Branch using OneHotEncoder()
3. Splitted the encoded data into train and test data (70-30)
4. Fitted the DecisionTreeClassifier() using GridSearchCV with 5-fold cross-validation on both type of encoded data having hyper-parameters as
   a. criterion – gini, entropy, log_loss
   b. max_depth – 4, 6, 8, 10, 15, 20
5. Fitted the RandomForestClassifier() using GridSearchCV with 5-fold cross-validation on both type of encoded data having hyper-parameters as
   a. n_estimators – 50, 100, 200, 300, 400, 500
   b. criterion – gini, entropy, log_loss
   c. max_features – sqrt, log2, None
6. Computed the Confusion Matrix, Overall Accuracy, Precision and Recall

Results:

Model	Parameter	Overall Accuracy	Precision	Recall
Decision Tree	criterion: entropy max_depth: 6 max_features: sqrt	0.560	0.592	0.404
Random Forest	criterion: gini max_features: sqrt n_estimators: 300	0.503	0.507	0.450

Conclusion:

Decision Tree fitted on the data gave the better results in overall accuracy and precision and comparable result in the recall category. It is chosen since it is more interpretable and less complex of a model then the random forest.

Sub Task B: Rating

Steps followed:

1. Dropped the InvoiceID column
2. Encoded the three qualitative features i.e. CustomerType, ProductType, PaymentType and Branch using OneHotEncoder()
3. Splitted the encoded data into train and test data (70-30)
4. Fitted the DecisionTreeRegressor() using GridSearchCV with 5-fold cross-validation having hyper-parameters as
    a. criterion – absolute_error
    b. max_depth – 1,2,3,4,5,6,7,8,9,10
    c. max_features – sqrt, log2
5. Fitted the LinearRegression()
6. Fitted the Ridge(), Lasso(), ElasticNet() using GridSearchCV with 5-fold cross-validation having hyper-parameters as
    a. alpha – 0.5, 1, 1.5, 2, 2.5, 3
    b. l1_ratio – 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
7. Computed the Mean Absolute Error

Results:

Model	Parameter	MAE
Decision Tree	max_depth: 2 max_features: sqrt	1.561
Linear Regression	default	1.538
Ridge Regression	alpha: 3	1.538
Lasso Regression	alpha: 1	1.535
Elastic Net Regression	alpha: 1 l1_Ratio: 0.7	1.535

Conclusion:

Lasso Regression fitted on the data gave the best result. It is chosen since it is more interpretable since it imposes l1 penalty pushing most of the coefficients to zero. Also, the decision tree classifier was returning the tree with least depth out of the hyper-parameters passes, which doesn’t seem reliable. Absolute error was chosen as it felt more natural to compare the ratings.