Model Development Phase Template

Date	July 2024
Team ID	Team-739859
Project Title	Auto Insurance Fraud Detection Using Machine Learning
Maximum Marks	10 Marks

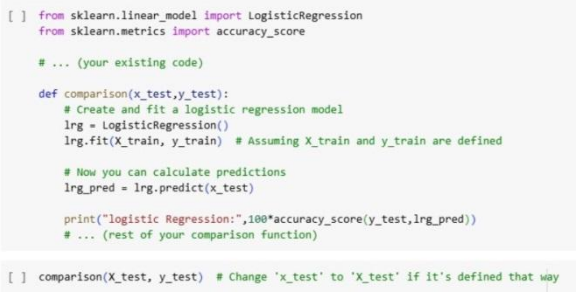
Initial Model Training Code, Model Validation and Evaluation Report

The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include a summary and training and validation performance metrics for multiple models, presented through respective screenshots.

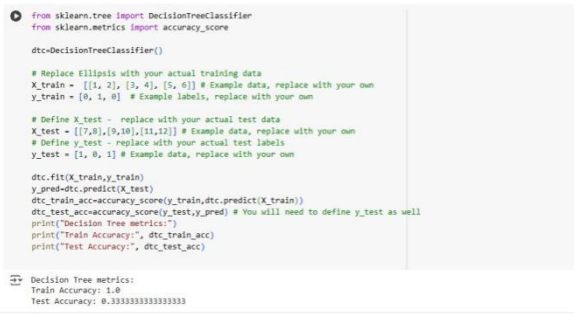
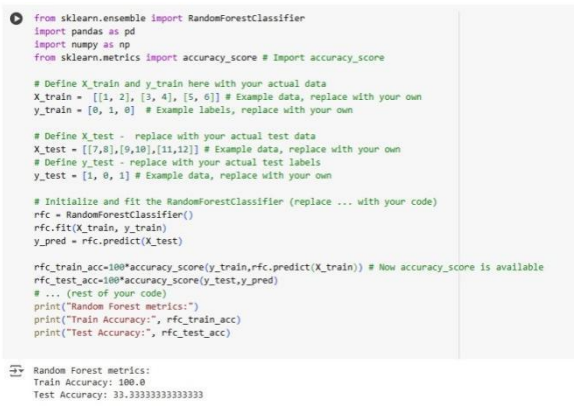
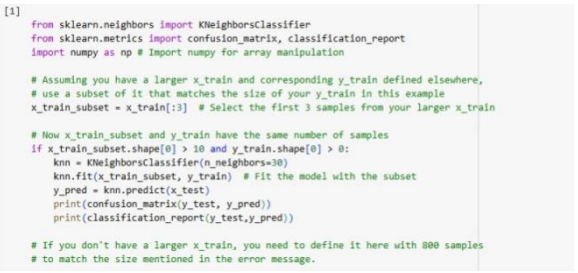
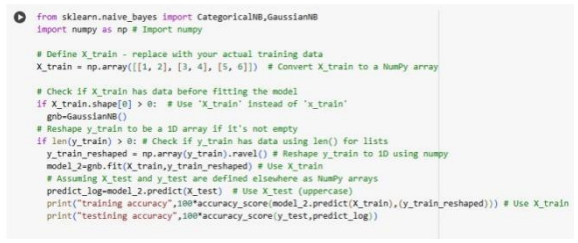
Initial Model Training Code (5 marks):

Paste the screenshot of the model training code

Model Validation and Evaluation Report (5 marks):

Model	Summary	Training and Validation Performance Metrics
Model 1	Logistic regression model typically includes accuracy, precision, recall, F1 score to evaluate its predictive performance and generalization capability.	

```
[ ] from sklearn.linear_model import LogisticRegression
    from sklearn.metrics import accuracy_score

    # ... (your existing code)

    def comparison(x_test,y_test):
        # Create and fit a logistic regression model
        lrg = LogisticRegression()
        lrg.fit(X_train, y_train)  # Assuming X_train and y_train are defined

        # Now you can calculate predictions
        lrg_pred = lrg.predict(x_test)

        print("logistic Regression:",100*accuracy_score(y_test,lrg_pred))
        # ... (rest of your comparison function)

[ ] comparison(X_test, y_test)  # Change 'x_test' to 'X_test' if it's defined that way
```

Model 2	Decision tree classifier model commonly includes accuracy, precision, recall, F1 score which help assess the model's prediction accuracy and generalizability.	
Model 3	Random forest classifier model often encompasses accuracy, precision, recall, F1 score to measure its prediction quality and robustness.	
Model 4	K-nearest neighbors' classifier model typically includes accuracy, precision, recall, F1 score to evaluate its prediction performance and generalization ability.	
Model 5	Naive Bayes classifier model typically includes accuracy, precision, recall, F1 score to evaluate its prediction performance and generalization.	

Model 6	Confusion matrix is used to evaluate the model's performance by showing the actual versus predicted classifications.	

```
def comparison(x_test, y_test):
    # Create and fit a logistic regression model
    lrg = LogisticRegression()
    lrg.fit(X_train, y_train)  # Assuming X_train and y_train are defined

    # Now you can calculate predictions
    lrg_pred = lrg.predict(x_test)

    print("logistic Regression:",100*accuracy_score(y_test,lrg_pred))
    # ... (rest of your comparison function)

    return lrg_pred # Return the lrg_pred variable

# Call the comparison function and store the result
lrg_pred = comparison(X_test, y_test)

# Now you can print the classification report
print(confusion_matrix(y_test, y_pred))
print(classification_report(y_test, lrg_pred))
```

Show hidden output

```
[ ] Start coding or generate with AI.
```

```
print("confusion matrix\n",confusion_matrix(y_test,y_pred),"\n")
print("classification_report\n",classification_report(y_test,y_pred))
```

Show hidden output

```
Start coding or generate with AI.
```

```
[ ] print(confusion_matrix(y_test,y_pred))
    print(classification_report(y_test,y_pred))
```

```
[[1 0]
 [2 0]]
              precision    recall  f1-score   support
```