Model Development Phase Template

Date	21 June 2024
Team ID	740142
Project Title	Life Style Change Due To Covid Prediction
Maximum Marks	4 Marks

Initial Model Training Code, Model Validation and Evaluation Report

The initial Random Forest model shows promising results for predicting lifestyle changes due to COVID-19 based on demographic and behavioral attributes. Further refinement of the model, including hyperparameter tuning and feature engineering, may enhance its predictive performance.

Initial Model Training Code:

```
from sklearn.ensemble import RandomForestClassifier
# Initialize the Random Forest Classifier
model2 = RandomForestClassifier()

# Fit the model
model2.fit(X_train, y_train)

# Make predictions on the test set
y_pred = model2.predict(X_test)

# Model Accuracy
accuracy = accuracy_score(y_test, y_pred)

# Evaluate the model
print("Accuracy: ", accuracy * 100)
print("\nClassification Report: \n", classification_report(y_test, y_pred))
```

```
from sklearn.tree import DecisionTreeClassifier

# Initialize the Decision Tree Classifier
model3 = DecisionTreeClassifier(random_state=42)

# Fit the model
model3.fit(X_train, y_train)

# Make predictions on the test set
y_pred = model3.predict(X_test)

# Model Accuracy
accuracy = accuracy_score(y_test, y_pred)

# Evaluate the model
print("Accuracy: ", accuracy * 100)
print("\nClassification Report: \n", classification_report(y_test, y_pred))
```

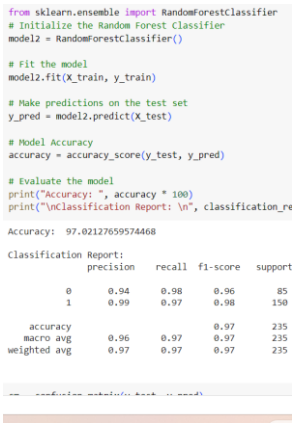
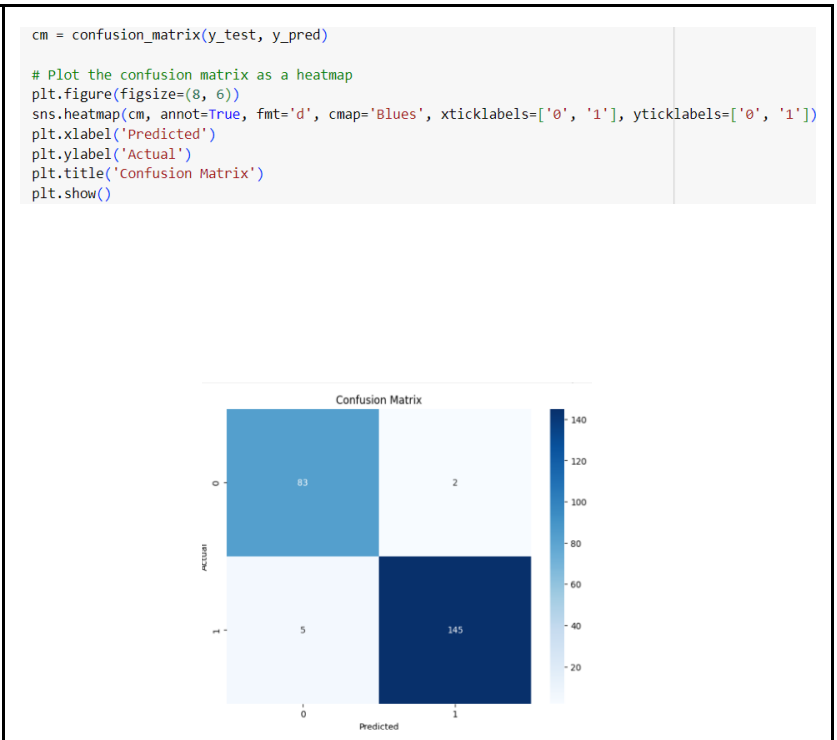
```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report
```

```
model1=LogisticRegression()
```

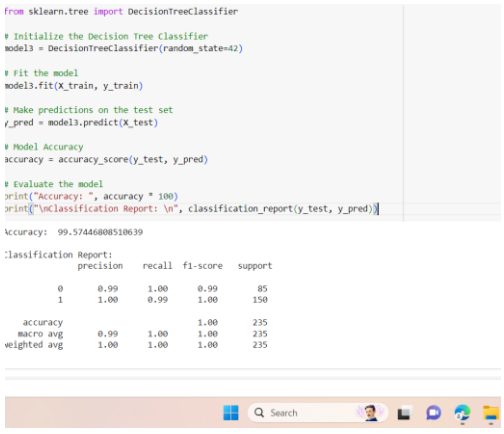
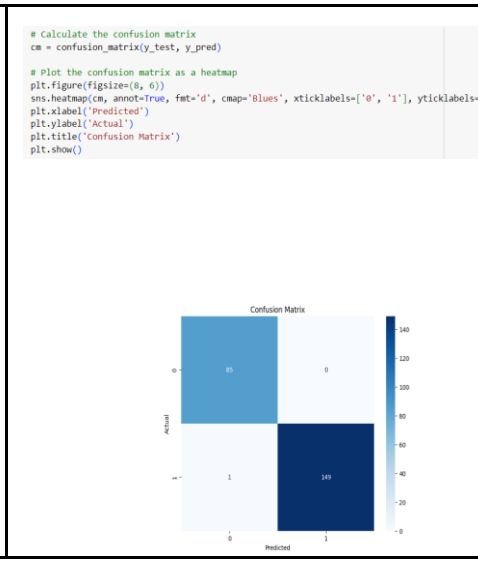
```
model1.fit(X_train,y_train)
```

```
▾ LogisticRegression
LogisticRegression()
```

Model	Classification Report	F1 Score	Confusion Matrix

Random Forest		97%	

Model Validation and Evaluation Report:

Decision Tree		99%	

Logistic Regression	`print('Accuracy:',accuracy*100)`
	`print('\nClassification Report:',classification_report(y_test,y_pred))`
	
	`Accuracy: 82.97872340425532`
	
	`Classification Report:`
	`precision recall f1-score support`
	
	`0 0.82 0.68 0.74 85`
	`1 0.84 0.91 0.87 150`
	
	`accuracy 0.83 235`
	`macro avg 0.83 0.80 0.81 235`
	`weighted avg 0.83 0.83 0.83 235`
		82%	`cm = confusion_matrix(y_test, y_pred)`
			
			`# Plot the confusion matrix as a heatmap`
			`plt.figure(figsize=(8, 6))`
			`sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['0', '1'], yticklabels=['0', '1'])`
			`plt.xlabel('Predicted')`
			`plt.ylabel('Actual')`
			`plt.title('Confusion Matrix')`
			`plt.show()`