

Model Development Phase Template

Date	15 July 2024
Team ID	739885
Project Title	Golden Harvest: A predictive model for apple quality assurance
Maximum Marks	4 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 classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

Initial Model Training Code:

```
model_=DecisionTreeClassifier()
model_.fit(x_train,y_train)
dt_pred=model_.predict(x_test)
acc_score=accuracy_score(y_test,dt_pred)
from sklearn.metrics import accuracy_score
acc_score=accuracy_score(y_test,dt_pred)
print("acc_score of decision tree model %.2f" % acc_score)
```

```
acc_score of decision tree model 0.80
```

```
model=RandomForestClassifier(n_estimators=100)
model.fit(x_train,y_train)
forest=model.predict(x_test)
accuracy=accuracy_score(y_test,forest)
print("acc_score of randomForest model %.2f"%accuracy)
```

```
acc_score of randomForest model 0.91
0.915
```

```
model.score(x_test,y_test)
```

```
0.9
```

```
model1=xgb.XGBClassifier().fit(x_train,y_train)
y_pred=model1.predict(x_test)
model1.score(x_test,y_test)
```

```
0.915
```

```
print("acc_score of model %.2f"%accuracy_score(y_test,forest))
```

```
acc_score of model 0.90
```

```
reg_model=LogisticRegression()
reg_model.fit(x_train,y_train)
reg_pred=reg_model.predict(x_test)
reg_acc_score=accuracy_score(y_test,reg_pred)
print("acc_score of logistic regression model %.2f"%accuracy_score(y_test,reg_pred))
```

```
acc_score of logistic regression model 0.75
```

Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix																														
Decision tree	<pre>from sklearn.metrics import classification_report print(classification_report(dt_pred,y_test))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.82</td><td>0.81</td><td>0.81</td><td>406</td></tr><tr><td>1</td><td>0.80</td><td>0.81</td><td>0.81</td><td>394</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.81</td><td>800</td></tr><tr><td>macro avg</td><td>0.81</td><td>0.81</td><td>0.81</td><td>800</td></tr><tr><td>weighted avg</td><td>0.81</td><td>0.81</td><td>0.81</td><td>800</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.82	0.81	0.81	406	1	0.80	0.81	0.81	394	accuracy			0.81	800	macro avg	0.81	0.81	0.81	800	weighted avg	0.81	0.81	0.81	800	80%	-
	precision	recall	f1-score	support																													
0	0.82	0.81	0.81	406																													
1	0.80	0.81	0.81	394																													
accuracy			0.81	800																													
macro avg	0.81	0.81	0.81	800																													
weighted avg	0.81	0.81	0.81	800																													
Random forest	<pre>from sklearn.metrics import classification_report print(classification_report(forest,y_test))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.91</td><td>0.90</td><td>0.90</td><td>405</td></tr><tr><td>1</td><td>0.89</td><td>0.90</td><td>0.90</td><td>395</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.90</td><td>800</td></tr><tr><td>macro avg</td><td>0.90</td><td>0.90</td><td>0.90</td><td>800</td></tr><tr><td>weighted avg</td><td>0.90</td><td>0.90</td><td>0.90</td><td>800</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.91	0.90	0.90	405	1	0.89	0.90	0.90	395	accuracy			0.90	800	macro avg	0.90	0.90	0.90	800	weighted avg	0.90	0.90	0.90	800	91%	-
	precision	recall	f1-score	support																													
0	0.91	0.90	0.90	405																													
1	0.89	0.90	0.90	395																													
accuracy			0.90	800																													
macro avg	0.90	0.90	0.90	800																													
weighted avg	0.90	0.90	0.90	800																													
XGB	<pre>print(classification_report(y_pred,y_test))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.90</td><td>0.91</td><td>0.90</td><td>400</td></tr><tr><td>1</td><td>0.90</td><td>0.90</td><td>0.90</td><td>400</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.90</td><td>800</td></tr><tr><td>macro avg</td><td>0.90</td><td>0.90</td><td>0.90</td><td>800</td></tr><tr><td>weighted avg</td><td>0.90</td><td>0.90</td><td>0.90</td><td>800</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.90	0.91	0.90	400	1	0.90	0.90	0.90	400	accuracy			0.90	800	macro avg	0.90	0.90	0.90	800	weighted avg	0.90	0.90	0.90	800	90%	-
	precision	recall	f1-score	support																													
0	0.90	0.91	0.90	400																													
1	0.90	0.90	0.90	400																													
accuracy			0.90	800																													
macro avg	0.90	0.90	0.90	800																													
weighted avg	0.90	0.90	0.90	800																													
Logistic Regression	<pre>print(classification_report(reg_pred,y_test))</pre> <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>0.75</td><td>0.76</td><td>0.75</td><td>400</td></tr><tr><td>1</td><td>0.75</td><td>0.75</td><td>0.75</td><td>400</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.75</td><td>800</td></tr><tr><td>macro avg</td><td>0.75</td><td>0.75</td><td>0.75</td><td>800</td></tr><tr><td>weighted avg</td><td>0.75</td><td>0.75</td><td>0.75</td><td>800</td></tr></tbody></table>		precision	recall	f1-score	support	0	0.75	0.76	0.75	400	1	0.75	0.75	0.75	400	accuracy			0.75	800	macro avg	0.75	0.75	0.75	800	weighted avg	0.75	0.75	0.75	800	75%	-
	precision	recall	f1-score	support																													
0	0.75	0.76	0.75	400																													
1	0.75	0.75	0.75	400																													
accuracy			0.75	800																													
macro avg	0.75	0.75	0.75	800																													
weighted avg	0.75	0.75	0.75	800																													