Model Development Phase Template

Date	13 july 2024
Team ID	739805
Project Title	Prediction and Analysis of Liver Patient Data Using Machine Learning
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:

Logistic Regression

```
# LogisticRegression
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(x_train, y_train)
y_pred_lr = lr.predict(x_test)
y_pred_lr
```

KNeighborsClassifier

```
#KNeighborsClassifier
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(x_train, y_train)
ypred_knn = knn.predict(x_test)
```

SVC

```
#SVC()
from sklearn.svm import SVC
svm = SVC()
svm.fit(x_train, y_train)
y_pred_svm = svm.predict(x_test)
```

RandomForestClassifier

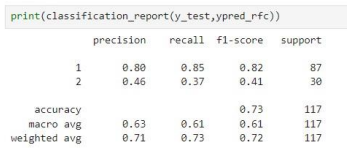
```
from sklearn.ensemble import RandomForestClassifier

rfc = RandomForestClassifier()
rfc.fit(x_train, y_train)
ypred_rfc = rfc.predict(x_test)
```

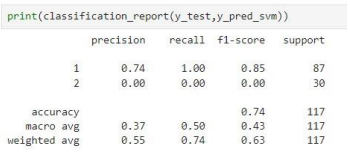
Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix

Logistic Regression	`print(classification_report(y_test,y_pred))`
	precision recall f1-score support
	1 0.75 0.91 0.83 128
	2 0.45 0.19 0.27 47
	accuracy 0.72 175
	macro avg 0.60 0.55 0.55 175
	weighted avg 0.67 0.72 0.68 175
		`lr_acc = accuracy_score(y_pred, y_test)`
		`lr_acc`
		0.72
			`conmat=confusion_matrix(y_test,y_pred)`
			`print(conmat)`
			[[117 11]
			[38 9]]

K neighbors Classifier	`print(classification_report(y_test,ypred_knn))`
	precision recall f1-score support
	1 0.81 0.80 0.80 109
	2 0.42 0.43 0.43 37
	accuracy 0.71 146
	macro avg 0.61 0.62 0.61 146
	weighted avg 0.71 0.71 0.71 146
		`knn_acc = accuracy_score(ypred_knn, y_test)`
		`print(knn_acc)`
		0.7054794520547946
			`confusion_matrix(y_test,ypred_knn)`
			array([[87, 22],
			[21, 16]], dtype=int64)

Random Forest Classifier	`print(classification_report(y_test,ypred_rfc))`
	 
	precision recall f1-score support
	1 0.80 0.85 0.82 87
	2 0.46 0.37 0.41 30
	accuracy 0.73 117
	macro avg 0.63 0.61 0.61 117
	weighted avg 0.71 0.73 0.72 117
		`rfc_acc = accuracy_score(ypred_rfc, y_test)`
		`print(rfc_acc)`
		0.7264957264957265
			`confusion_matrix(y_test,ypred_rfc)`
			array([[74, 13],
			[19, 11]], dtype=int64)

SVC	`print(classification_report(y_test,y_pred_svm))`  precision recall f1-score support 1 0.74 1.00 0.85 87 2 0.00 0.00 0.00 30 accuracy 0.74 117 macro avg 0.37 0.50 0.43 117 weighted avg 0.55 0.74 0.63 117	`accuracy_score(y_pred_svm, y_test)` 0.7435897435897436	`confusion_matrix(y_test,y_pred_svm)` array([[87, 0], [30, 0]], dtype=int64)