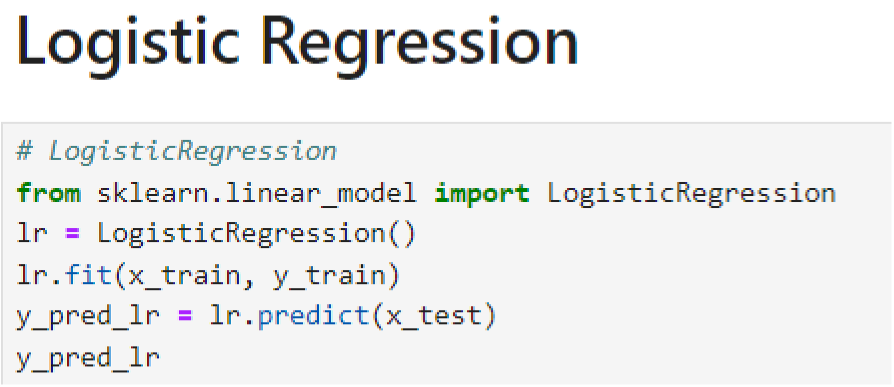
# Model Development Phase Template

| | |
|---|---|
| Date | 10 July 2024 |
| Team ID | SWTID1720013031 |
| 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

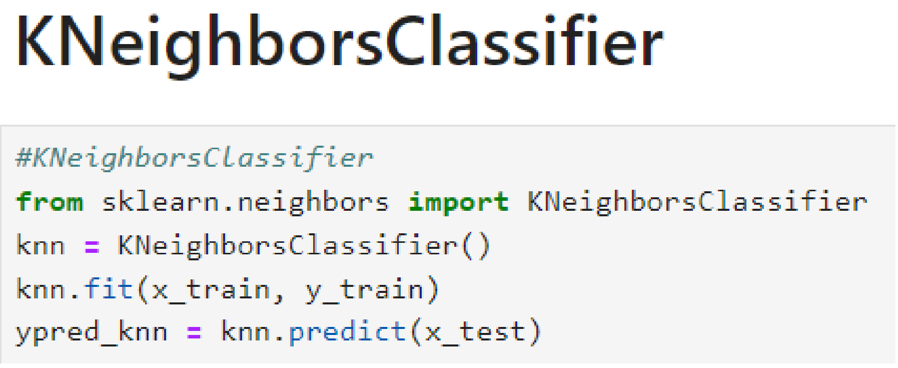
The initial model training code will be showcased in the future through a screenshot. The model validation and evaluation report will include classification reports, accuracy, and confusion matrices for multiple models, presented through respective screenshots.

## Initial Model Training Code:



# Logistic Regression

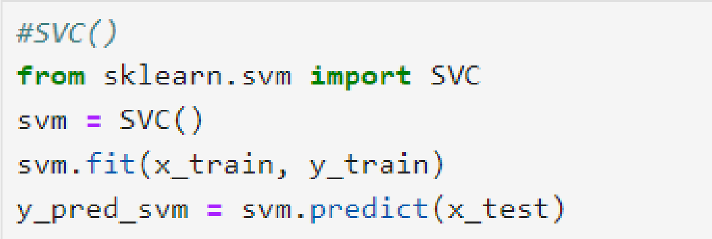
```
# LogisticRegression
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(x_train, y_train)
y_pred_lr = lr.predict(x_test)
y_pred_lr
```



# KNeighborsClassifier

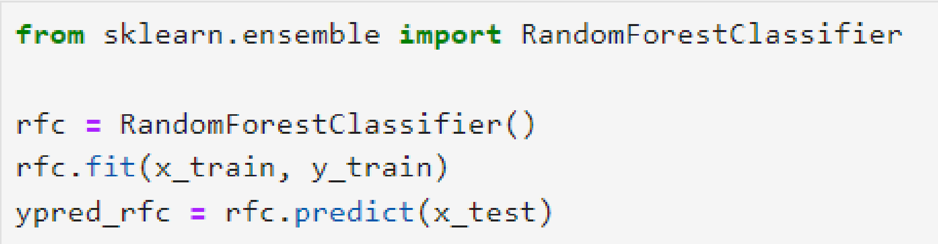
```
#KNeighborsClassifier
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier()
knn.fit(x_train, y_train)
ypred_knn = knn.predict(x_test)
```

# SVC



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

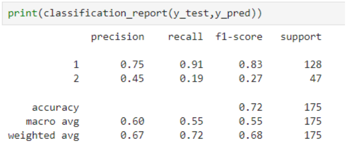
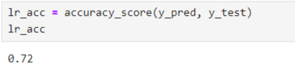
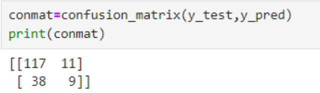
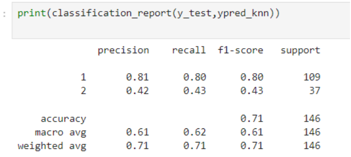
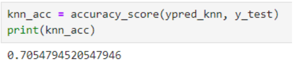
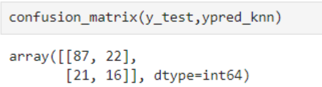
# RandomForestClassifier

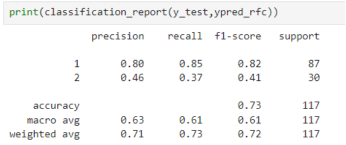
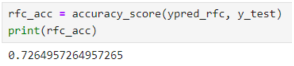
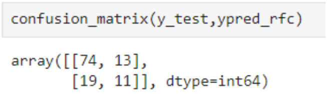
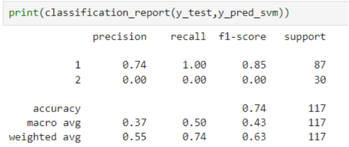
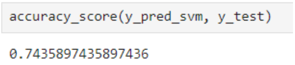
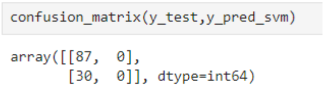


```
from sklearn.ensemble import RandomForestClassifier

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<br>Regression | <br>print(classification_report(y_test,y_pred))<br>precision recall f1-score support<br>1 0.75 0.91 0.83 128<br>2 0.45 0.19 0.27 47<br>accuracy 0.72 175<br>macro avg 0.60 0.55 0.55 175<br>weighted avg 0.67 0.72 0.68 175 | <br>lr_acc = accuracy_score(y_pred, y_test)<br>lr_acc<br>0.72 | <br>conmat=confusion_matrix(y_test,y_pred)<br>print(conmat)<br>[[117 11]<br>[ 38 9]] |
| K neighbors<br>Classifier | <br>print(classification_report(y_test,ypred_knn))<br>precision recall f1-score support<br>1 0.81 0.80 0.80 109<br>2 0.42 0.43 0.43 37<br>accuracy 0.71 146<br>macro avg 0.61 0.62 0.61 146<br>weighted avg 0.71 0.71 0.71 146 | <br>knn_acc = accuracy_score(ypred_knn, y_test)<br>print(knn_acc)<br>0.7054794520547946 | <br>confusion_matrix(y_test,ypred_knn)<br>array([[87, 22],<br>[21, 16]], dtype=int64) |

| | | | |
|---|---|---|---|
| Random Forest<br>Classifier | <br>print(classification_report(y_test,ypred_rfc))<br>precision recall f1-score support<br>1 0.80 0.85 0.82 87<br>2 0.46 0.37 0.41 30<br>accuracy 0.73 117<br>macro avg 0.63 0.61 0.61 117<br>weighted avg 0.71 0.73 0.72 117 | <br>rfc_acc = accuracy_score(ypred_rfc, y_test)<br>print(rfc_acc)<br>0.7264957264957265 | <br>confusion_matrix(y_test,ypred_rfc)<br>array([[74, 13],<br>[19, 11]], dtype=int64) |
| SVC | <br>print(classification_report(y_test,y_pred_svm))<br>precision recall f1-score support<br>1 0.74 1.00 0.85 87<br>2 0.00 0.00 0.00 30<br>accuracy 0.74 117<br>macro avg 0.37 0.50 0.43 117<br>weighted avg 0.55 0.74 0.63 117 | <br>accuracy_score(y_pred_svm, y_test)<br>0.7435897435897436 | <br>confusion_matrix(y_test,y_pred_svm)<br>array([[87, 0],<br>[30, 0]], dtype=int64) |