Model Development Phase

Date	10 July 2024
Team ID	SWTID1721205662
Project Title	Early Prediction of Chronic Kidney Disease 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:

1) **Splitting the data into test and train split.**

```
from sklearn.model_selection import train_test_split

x_train,x_test,y_train,y_test=train_test_split(x_scaled,y,test_size=0.3, random_state=0)

x_train.shape

x_test.shape
```

2)Decision tree classifier

```
from sklearn.tree import DecisionTreeClassifier
df=DecisionTreeClassifier(criterion='entropy',random_state=0)


df.fit(x_train,y_train)


pred_dt=df.predict(x_test)


pred_dt
```

3)Evaluation metrics

```
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report


accuracy=accuracy_score(y_test,pred)
conmat=confusion_matrix(y_test,pred)


print(accuracy)
print(conmat)
```

4) Logisitic regression

```
from sklearn.linear_model import LogisticRegression


lr=LogisticRegression()


lr.fit(x_train,y_train)


pred1=lr.predict(x_train)


pred=lr.predict(x_test)
```

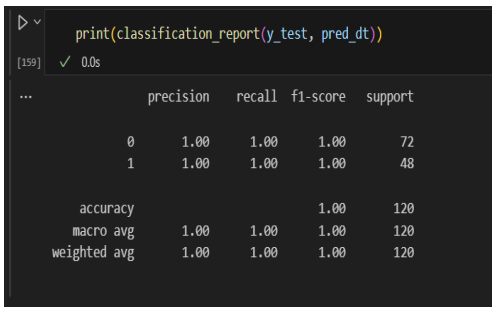
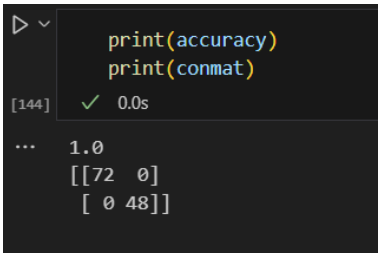
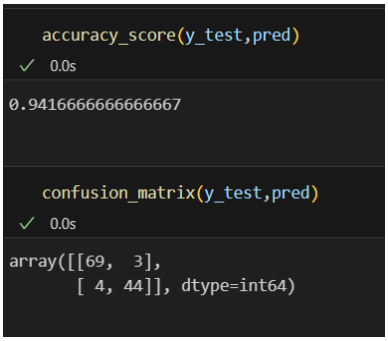
5) Evaluation metrics

```
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix

accuracy_score(y_train,pred1)

accuracy_score(y_test,pred)

confusion_matrix(y_test,pred)

print(classification_report(y_test, pred))
```

6) KNN Classifier

```
x_train1,x_test1,y_train1,y_test1=train_test_split(x,y,test_size=0.2, random_state=0)

from sklearn.neighbors import KNeighborsClassifier
knn=KNeighborsClassifier()

knn.fit(x_train1,y_train1)

pred_knn=knn.predict(x_test1)

pred_knn
```

Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix
Decision tree classifier		1.00	
Logistic Regression		0.941	
KNN Classification		0.68