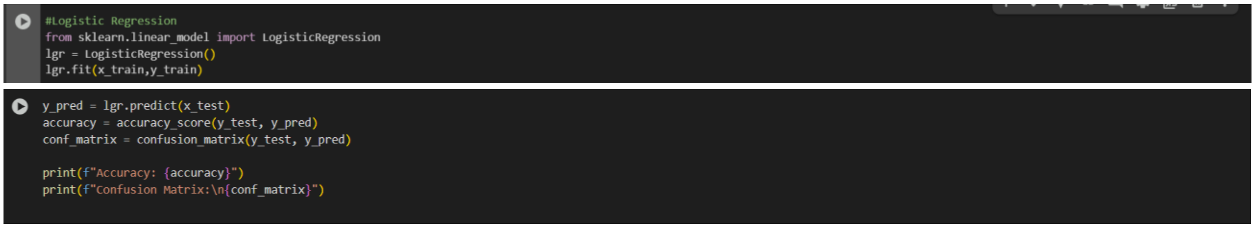
# Model Development Phase

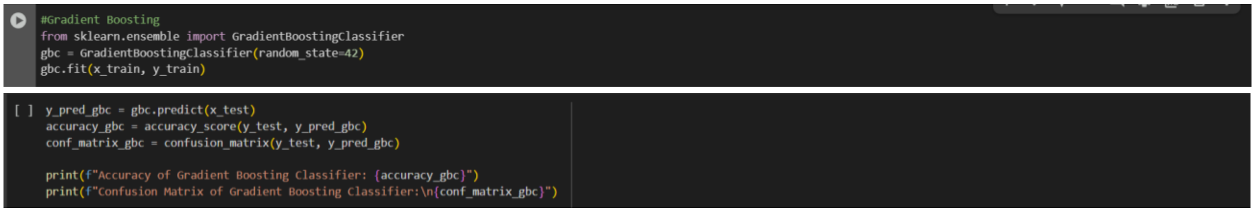
|  |  |
|---|---|
| Date | 18 June 2025 |
| Team ID | SWTID1749620488 |
| Project Title | Early Prediction for Chronic Kidney Disease Detection: A Progressive Approach to Health Management |
| Maximum Marks | 4 Marks |

## Initial Model Training Code:



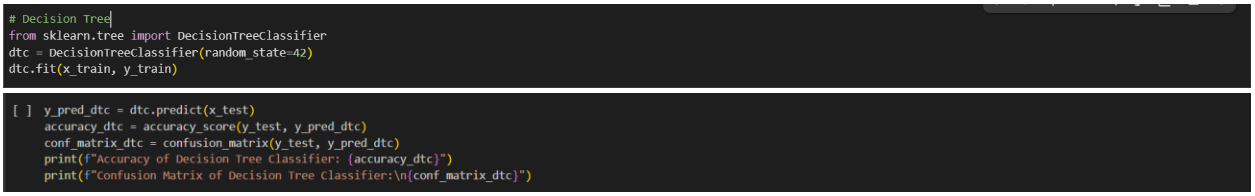
```
#Logistic Regression
from sklearn.linear_model import LogisticRegression
lgr = LogisticRegression()
lgr.fit(x_train,y_train)
```

```
y_pred = lgr.predict(x_test)
accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)

print(f"Accuracy: {accuracy}")
print(f"Confusion Matrix:\n{conf_matrix}")
```



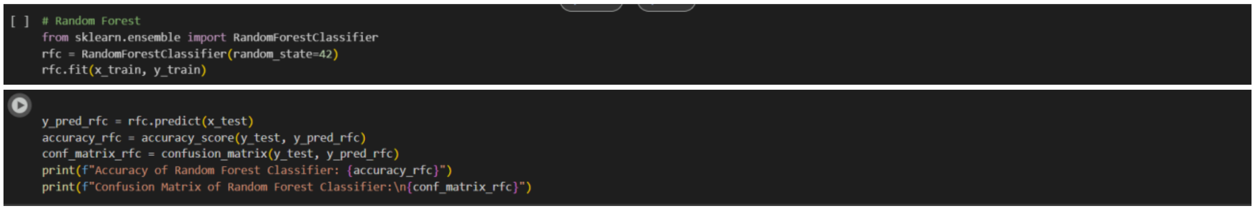
```
#Gradient Boosting
from sklearn.ensemble import GradientBoostingClassifier
gbc = GradientBoostingClassifier(random_state=42)
gbc.fit(x_train, y_train)
```

```
[ ] y_pred_gbc = gbc.predict(x_test)
    accuracy_gbc = accuracy_score(y_test, y_pred_gbc)
    conf_matrix_gbc = confusion_matrix(y_test, y_pred_gbc)

    print(f"Accuracy of Gradient Boosting Classifier: {accuracy_gbc}")
    print(f"Confusion Matrix of Gradient Boosting Classifier:\n{conf_matrix_gbc}")
```



```
# Decision Tree
from sklearn.tree import DecisionTreeClassifier
dtc = DecisionTreeClassifier(random_state=42)
dtc.fit(x_train, y_train)
```

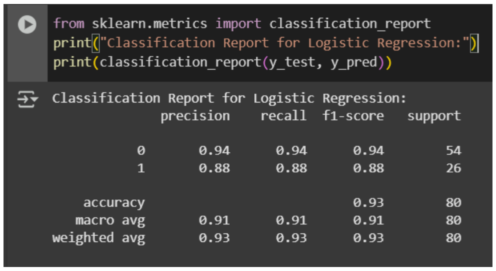
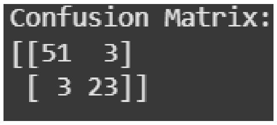
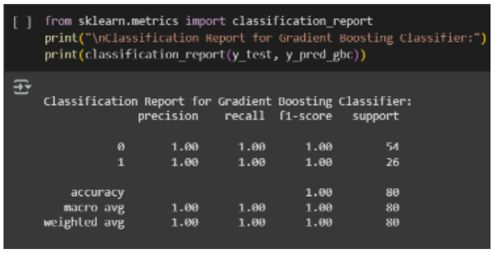
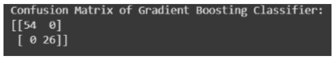
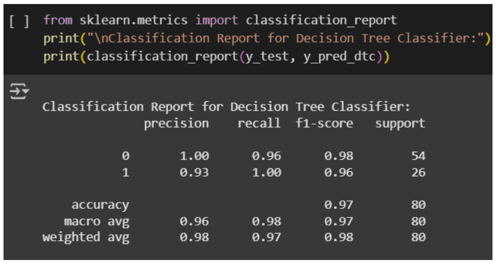
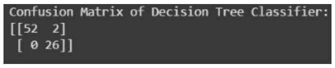
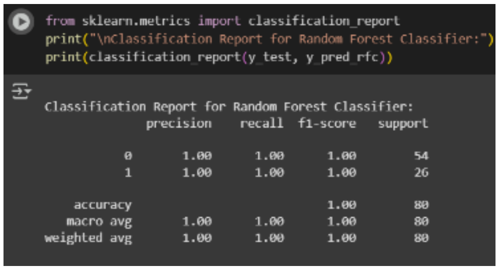
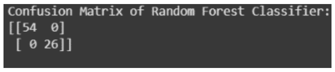
```
[ ] y_pred_dtc = dtc.predict(x_test)
    accuracy_dtc = accuracy_score(y_test, y_pred_dtc)
    conf_matrix_dtc = confusion_matrix(y_test, y_pred_dtc)
    print(f"Accuracy of Decision Tree Classifier: {accuracy_dtc}")
    print(f"Confusion Matrix of Decision Tree Classifier:\n{conf_matrix_dtc}")
```



```
[ ] # Random Forest
    from sklearn.ensemble import RandomForestClassifier
    rfc = RandomForestClassifier(random_state=42)
    rfc.fit(x_train, y_train)
```

```
y_pred_rfc = rfc.predict(x_test)
accuracy_rfc = accuracy_score(y_test, y_pred_rfc)
conf_matrix_rfc = confusion_matrix(y_test, y_pred_rfc)
print(f"Accuracy of Random Forest Classifier: {accuracy_rfc}")
print(f"Confusion Matrix of Random Forest Classifier:\n{conf_matrix_rfc}")
```

## Model Validation and Evaluation Report:

| Model | Classification Report | Accuracy | Confusion Matrix |
|---|---|---|---|
| Logistic Regression | <br>from sklearn.metrics import classification_report<br>print("Classification Report for Logistic Regression:")<br>print(classification_report(y_test, y_pred))<br>Classification Report for Logistic Regression:<br>precision recall f1-score support<br>0 0.94 0.94 0.94 54<br>1 0.88 0.88 0.88 26<br>accuracy 0.93 80<br>macro avg 0.91 0.91 0.91 80<br>weighted avg 0.93 0.93 0.93 80 | 92.5% | <br>Confusion Matrix:<br>[[51  3]<br> [ 3 23]] |
| Gradient Boosting | <br>[ ] from sklearn.metrics import classification_report<br>print("\nClassification Report for Gradient Boosting Classifier:")<br>print(classification_report(y_test, y_pred_gbc))<br>Classification Report for Gradient Boosting Classifier:<br>precision recall f1-score support<br>0 1.00 1.00 1.00 54<br>1 1.00 1.00 1.00 26<br>accuracy 1.00 80<br>macro avg 1.00 1.00 1.00 80<br>weighted avg 1.00 1.00 1.00 80 | 100% | <br>Confusion Matrix of Gradient Boosting Classifier:<br>[[54  0]<br> [ 0 26]] |
| Decision Tree | <br>[ ] from sklearn.metrics import classification_report<br>print("\nClassification Report for Decision Tree Classifier:")<br>print(classification_report(y_test, y_pred_dtc))<br>Classification Report for Decision Tree Classifier:<br>precision recall f1-score support<br>0 1.00 0.96 0.98 54<br>1 0.93 1.00 0.96 26<br>accuracy 0.97 80<br>macro avg 0.96 0.98 0.97 80<br>weighted avg 0.98 0.97 0.98 80 | 97.5% | <br>Confusion Matrix of Decision Tree Classifier:<br>[[52  2]<br> [ 0 26]] |
| Random Forest | <br>from sklearn.metrics import classification_report<br>print("\nClassification Report for Random Forest Classifier:")<br>print(classification_report(y_test, y_pred_rfc))<br>Classification Report for Random Forest Classifier:<br>precision recall f1-score support<br>0 1.00 1.00 1.00 54<br>1 1.00 1.00 1.00 26<br>accuracy 1.00 80<br>macro avg 1.00 1.00 1.00 80<br>weighted avg 1.00 1.00 1.00 80 | 100% | <br>Confusion Matrix of Random Forest Classifier:<br>[[54  0]<br> [ 0 26]] |