Project Development Phase
Model Performance Test

Date	9th NOvember 2022
Team ID	Team-592607
Project Name	Diabetes Prediction Using Machine Learning
Maximum Marks	10 Marks

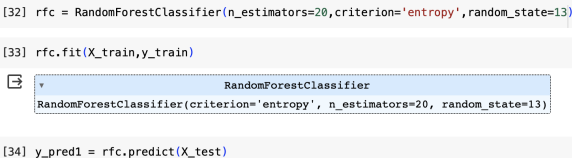**Team Member:** AS. Vikram Aditya

Team Member: PVS. Uday Kiran

Team Member: K. Rami Reddy

Team Leader: L. Mohith

Model Performance Testing:

Project team shall fill the following information in the model performance testing template.

S.No.	Parameter	Values	Screenshot

1.	Metrics	**Regression Model:** MAE, MSE , RMSE , R2 score
		
		Classification Model: Confusion Matrix, Accuray Score & Classification Report
			[32] rfc = RandomForestClassifier(n_estimators=20,criterion='entropy',random_state=13)
			
			[33] rfc.fit(X_train,y_train)
			
			RandomForestClassifier
			RandomForestClassifier(criterion='entropy', n_estimators=20, random_state=13)
			
			[34] y_pred1 = rfc.predict(X_test)
			
			Regression Model:
			
			``python
			from sklearn.svm import SVC
			from sklearn.datasets import make_classification
			from sklearn.pipeline import Pipeline
			from sklearn.metrics import mean_absolute_error
			from sklearn.metrics import mean_squared_error
			from sklearn.metrics import r2_score
			``
			
			[53] MAE = mean_absolute_error(y_test,y_pred1)
			MAE.round(2)
			
			0.12
			
			[54] MSE = mean_squared_error(y_test,y_pred1)
			MSE.round(2)
			
			0.23
			
			[55] RMSE = MSE**(1/2)
			RMSE.round(2)
			
			0.48
			
			[56] R2 = r2_score(y_test,y_pred1)
			R2
			
			0.654524878682704
			
			Classification Model:

			``python
pd.crosstab(y_test,y_pred1)
```

<br>

| col_0 | 0.0 | 1.0 | 2.0 |
|---|---|---|---|
| Diabetes_012 | | | |
| 0.0 | 23983 | 571 | 3897 |
| 1.0 | 0 | 28421 | 17 |
| 2.0 | 759 | 510 | 27367 |

```python
[41] as1 = accuracy_score(y_test,y_pred1)
```

```python
[42] as1
```
```
0.9327214264834843
```

```python
[48] classificationreport1 = classification_report(y_test,y_pred1)
    print(classificationreport1)
```
```
              precision    recall  f1-score   support

         0.0       0.97      0.84      0.90     28451
         1.0       0.96      1.00      0.98     28438
         2.0       0.87      0.96      0.91     28636

    accuracy                           0.93     85525
   macro avg       0.94      0.93      0.93     85525
weighted avg       0.94      0.93      0.93     85525
``` |
| 2. | Tune the Model | **Hyperparameter Tuning, Validation Method** | **Hyperparameter Tuning:**

```python
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import GridSearchCV
pipe = make_pipeline((RandomForestClassifier()))
grid_param = [
    {"randomforestclassifier": [RandomForestClassifier()],
     "randomforestclassifier__n_estimators":[10,20,50],
     "randomforestclassifier__random_state":[6,7,13]}]
gridsearch = GridSearchCV(pipe,grid_param,cv=5,verbose=0,n_jobs=-1)
```

```python
[65] best_model =gridsearch.fit(X_train,y_train)
```

```python
best_model.score(X_test,y_test)
```
```
0.9347208418591055
```

Validation Method: KFold Method

```python
[68] from sklearn.model_selection import KFold
    kfold = KFold(5,shuffle=True)
    for train,test in kfold.split(y):
        print("Train:",train,"Test:",test)

Train: [     0      2      3 ... 570162 570163 570164] Test: [     1      4      5 ... 570130 570148 570161]
Train: [     0      1      2 ... 570161 570162 570163] Test: [     3     11     15 ... 570154 570156 570164]
Train: [     0      1      2 ... 570160 570161 570164] Test: [     8     14     18 ... 570158 570162 570163]
Train: [     1      3      4 ... 570162 570163 570164] Test: [     0      2     10 ... 570146 570150 570157]
Train: [     0      1      2 ... 570162 570163 570164] Test: [     6      7      9 ... 570153 570159 570160]
``` |