Model Development Phase Template

Date	24 April 2024
Team ID	Team-738169
Project Title	Rainfall Prediction Using Machine Learning
Maximum Marks	4 Marks

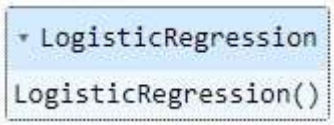
Initial Model Training Code, Model Validation and Evaluation Report

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.Logistic Regression.

```
logreg = LogisticRegression()
logreg.fit(X_train_res, y_train_res)
```




```
y_pred2 = logreg.predict(X_test)
print(confusion_matrix(y_test,y_pred2))
print(accuracy_score(y_test,y_pred2))
print(classification_report(y_test,y_pred2))
```

2.Decision Tree Classifier.

```
model_dt =  DecisionTreeClassifier(criterion='gini', random_state = 100, max_depth = 6, min_samples_leaf = 8)
```

```
model_dt.fit(X_train_res, y_train_res)
```

```
DecisionTreeClassifier(max_depth=6, min_samples_leaf=8, random_state=100)
```

```
y_pred = model_dt.predict(X_test)
print(confusion_matrix(y_test,y_pred))
print(accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))
```

3.Random Forest Classifier.

```
rf=RandomForestClassifier()
rf.fit(X_train_res,y_train_res)
```

```
RandomForestClassifier()
```

```
y_pred1 = rf.predict(X_test)
print(confusion_matrix(y_test,y_pred1))
print(accuracy_score(y_test,y_pred1))
print(classification_report(y_test,y_pred1))
```

4. KNeighbors Classifier.

```
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(X_train_res, y_train_res)
```

```
KNeighborsClassifier(n_neighbors=3)
```

```
y_pred4 = knn.predict(X_test)
print(confusion_matrix(y_test,y_pred4))
print(accuracy_score(y_test,y_pred4))
print(classification_report(y_test,y_pred4))
```

5.SVC.

```
svc = SVC()
svc.fit(X_train_res, y_train_res)
```

```
▾ SVC
SVC()
```

```
y_pred5 = svc.predict(X_test)
print(confusion_matrix(y_test,y_pred5))
print(accuracy_score(y_test,y_pred5))
print(classification_report(y_test,y_pred5))
```

6. Xgboost Classifier.

```
xgb = XGBClassifier()
xgb.fit(X_train_res, y_train_res)
```

```
▾                                XGBClassifier
XGBClassifier(base_score=None, booster=None, callbacks=None,
              colsample_bylevel=None, colsample_bynode=None,
              colsample_bytree=None, device=None, early_stopping_rounds=None,
              enable_categorical=False, eval_metric=None, feature_types=None,
              gamma=None, grow_policy=None, importance_type=None,
              interaction_constraints=None, learning_rate=None, max_bin=None,
              max_cat_threshold=None, max_cat_to_onehot=None,
              max_delta_step=None, max_depth=None, max_leaves=None,
              min_child_weight=None, missing=nan, monotone_constraints=None,
              multi_strategy=None, n_estimators=None, n_jobs=None,
              num_parallel_tree=None, random_state=None, ...)
```

```
y_pred = xgb.predict(X_test)
print(confusion_matrix(y_test,y_pred))
print(accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))
```

Model Validation and Evaluation Report:

Model	Classification Report	Accuracy	Confusion Matrix
Logistic Regression	precision recall f1-score support 0 0.92 0.77 0.84 22717 1 0.48 0.76 0.59 6375 accuracy 0.77 29092 macro avg 0.70 0.77 0.71 29092 weighted avg 0.82 0.77 0.78 29092	77%	[[17456 5261] [1508 4867]]
Decision Tree Classifier	precision recall f1-score support 0 0.90 0.75 0.82 22717 1 0.45 0.72 0.55 6375 accuracy 0.75 29092 macro avg 0.68 0.73 0.69 29092 weighted avg 0.80 0.75 0.76 29092	75%	[[17113 5604] [1807 4568]]
Random Forest Classifier	precision recall f1-score support 0 0.90 0.91 0.90 22717 1 0.66 0.62 0.64 6375 accuracy 0.85 29092 macro avg 0.78 0.77 0.77 29092 weighted avg 0.84 0.85 0.85 29092	85%	[[20694 2023] [2410 3965]]

KNeighbors Classifier	precision recall f1-score support 0 0.91 0.77 0.83 22717 1 0.46 0.72 0.56 6375 accuracy 0.76 29092 macro avg 0.68 0.74 0.70 29092 weighted avg 0.81 0.76 0.77 29092	76%	[[17410 5307] [1811 4564]]
SVC	precision recall f1-score support 0 0.92 0.78 0.85 22717 1 0.49 0.75 0.60 6375 accuracy 0.78 29092 macro avg 0.71 0.77 0.72 29092 weighted avg 0.83 0.78 0.79 29092	78%	[[17823 4894] [1595 4780]]
Xgboost Classifier	precision recall f1-score support 0 0.89 0.94 0.91 22717 1 0.72 0.57 0.63 6375 accuracy 0.86 29092 macro avg 0.80 0.75 0.77 29092 weighted avg 0.85 0.86 0.85 29092	86%	[[21321 1396] [2768 3607]]