

Project Development Phase  
Model Performance Test

Date	17 November 2022
Team ID	PNT2022TMID37820
Project Name	Web Phishing Detection
Maximum Marks	10 Marks

Model Performance Testing:

S.No.	Parameter	Values	Screenshot
1.	Metrics	Regression Model: <b>LOGISTICREGRESSION</b> MAE - 0.16 MSE - 0.33 RMSE - 0.57 R2 score - 0.66	Attached Below
2.	Tune the Model	Hyperparameter Tuning: 1. Best parameter - {'C': 0.1, 'penalty': 'l2'} 2. Accuracy - 0.93  Validation Method : <b>Grid Search Cross Validation</b>	Attached Below

1. METRICS:  
REGRESSION MODEL – **LOGISTICREGRESSION**

Working with LogisticRegression

▶

```
# Linear regression model
from sklearn.linear_model import LogisticRegression

# instantiate the model
log = LogisticRegression()

# fit the model
log.fit(x_train,y_train)
```

[8] ✓ 0.2s

...

▼ LogisticRegression

LogisticRegression()

```
pred=log.predict(x_test)
```

[9] ✓ 0.4s

```
pred
```

[10] ✓ 0.3s

```
array([-1, -1, 1, ..., -1, -1, 1], dtype=int64)
```

EVALUATION METRICS:

- ✓ R2 Score
- ✓ Mean Square Error (MSE)
- ✓ RMSE (Root Mean Square Error)
- ✓ Mean Absolute Error (MAE)

Evaluation Metrics

```

from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
MSE = mean_squared_error(pred, y_test)

```

[11]
✓
0.4s

```

mean_absolute_error(pred, y_test)

```

[12]
✓
0.4s

...
0.16644052464947987

```

MSE

```

[13]
✓
0.6s

...
0.33288104929895973

```

RMSE=np.sqrt(MSE)
print(RMSE)

```

[14]
✓
0.4s

...
0.5769584467697476

```

R2_score=r2_score(pred, y_test)
print(R2_score)

```

[15]
✓
0.3s

...
0.6627729239543096

2.Tune the Model:

Grid Search Cross Validation Method used for Hyperparameter Tuning

Grid search cross validation

```

from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LogisticRegression

```

[102]
✓
0.3s

```

grid={"C":np.logspace(-3,3,7), "penalty":["l1","l2"]}# l1 lasso l2 ridge
log_cv=GridSearchCV(log,grid,cv=10)
log_cv.fit(x_train,y_train)

print("Tuned hpyerparameter :(best parameter) ",log_cv.best_params_)
print("accuracy :",log_cv.best_score_)

```

[103]
✓
3.8s

...
Tuned hpyerparameter :(best parameter) {'C': 0.1, 'penalty': 'l2'}
accuracy : 0.9313666692231