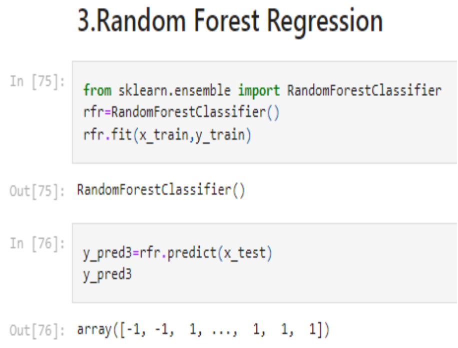
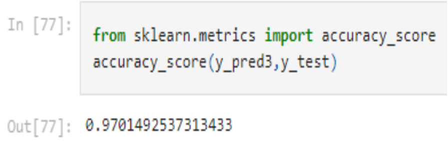
Project Development Phase
Model Performance Test

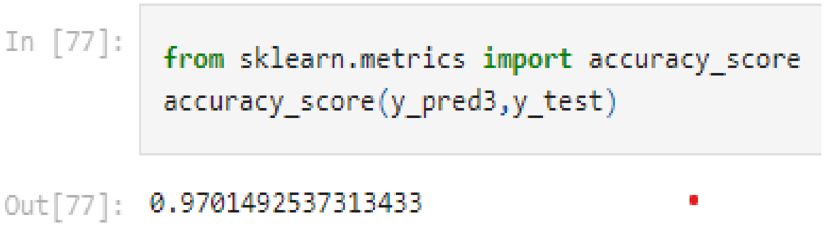
Date	18 November 2022
Team ID	PNT2022TMID33764
Project Name	Project – Web Phishing Detection
Maximum Marks	10 Marks

Model Performance Testing:

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

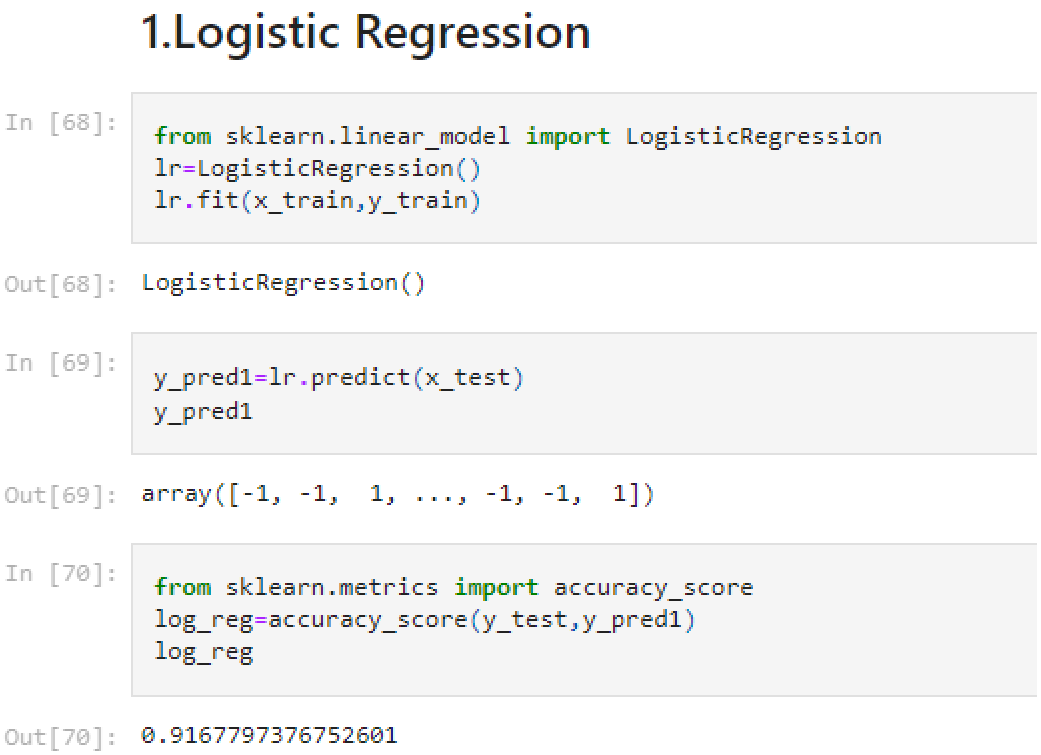
S.No.	Parameter	Values	Screenshot
1.	Metrics	**Classification Model:** RandomForest Classifier. **Evaluation Metrics:** ConfusionMatrix - Accuracy Score- & Classification Report.	 3.Random Forest Regression In [75]: from sklearn.ensemble import RandomForestClassifier rfr=RandomForestClassifier() rfr.fit(x_train,y_train) Out[75]: RandomForestClassifier() In [76]: y_pred3=rfr.predict(x_test) y_pred3 Out[76]: array([-1, -1, 1, ..., 1, 1, 1])  In [77]: from sklearn.metrics import accuracy_score accuracy_score(y_pred3,y_test) Out[77]: 0.9701492537313433

EVALUATION METRICS:



```
In [77]: from sklearn.metrics import accuracy_score
         accuracy_score(y_pred3,y_test)

Out[77]: 0.9701492537313433
```

COMPARING PERFORMANCE OF DIFFERENT ML MODELS:



1.Logistic Regression

```
In [68]: from sklearn.linear_model import LogisticRegression
         lr=LogisticRegression()
         lr.fit(x_train,y_train)

Out[68]: LogisticRegression()

In [69]: y_pred1=lr.predict(x_test)
         y_pred1

Out[69]: array([-1, -1,  1, ..., -1, -1,  1])

In [70]: from sklearn.metrics import accuracy_score
         log_reg=accuracy_score(y_test,y_pred1)
         log_reg

Out[70]: 0.9167797376752601
```

2.Decision Tree

In [71]:

```
from sklearn.tree import DecisionTreeClassifier
dtr=DecisionTreeClassifier()
dtr.fit(x_train,y_train)
```

Out[71]:

```
DecisionTreeClassifier()
```

In [73]:

```
y_pred2=dtr.predict(x_test)
y_pred2
```

Out[73]:

```
array([-1, -1,  1, ...,  1,  1,  1])
```

In [74]:

```
from sklearn.metrics import accuracy_score
accuracy_score(y_pred2,y_test)
```

Out[74]:

```
0.9647218453188603
```

3.Random Forest Regression

In [75]:

```
from sklearn.ensemble import RandomForestClassifier
rfr=RandomForestClassifier()
rfr.fit(x_train,y_train)
```

Out[75]:

```
RandomForestClassifier()
```

In [76]:

```
y_pred3=rfr.predict(x_test)
y_pred3
```

Out[76]:

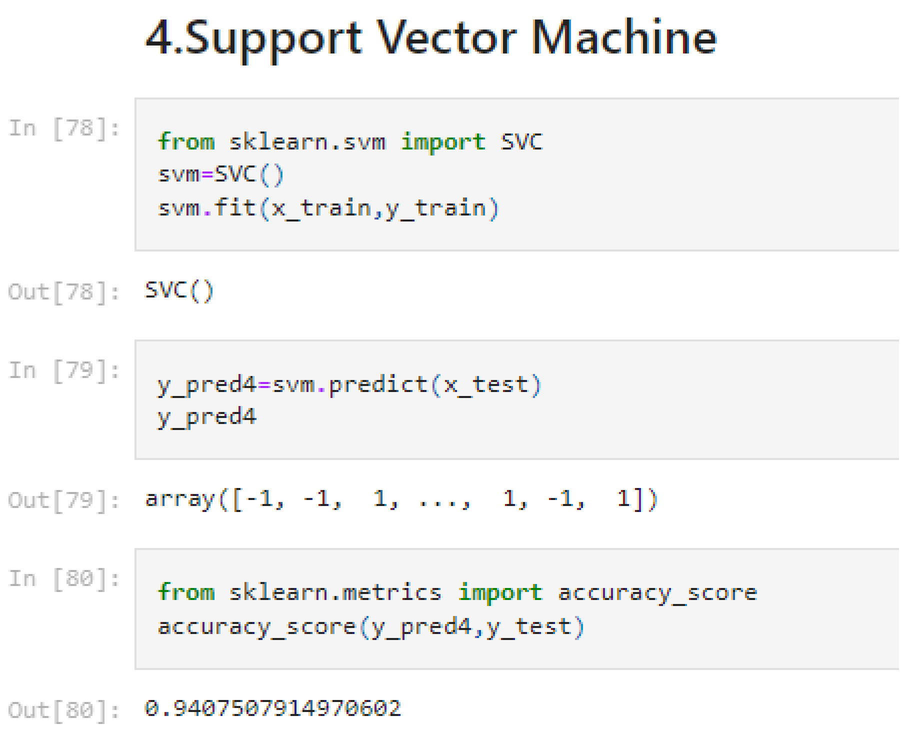
```
array([-1, -1,  1, ...,  1,  1,  1])
```

In [77]:

```
from sklearn.metrics import accuracy_score
accuracy_score(y_pred3,y_test)
```

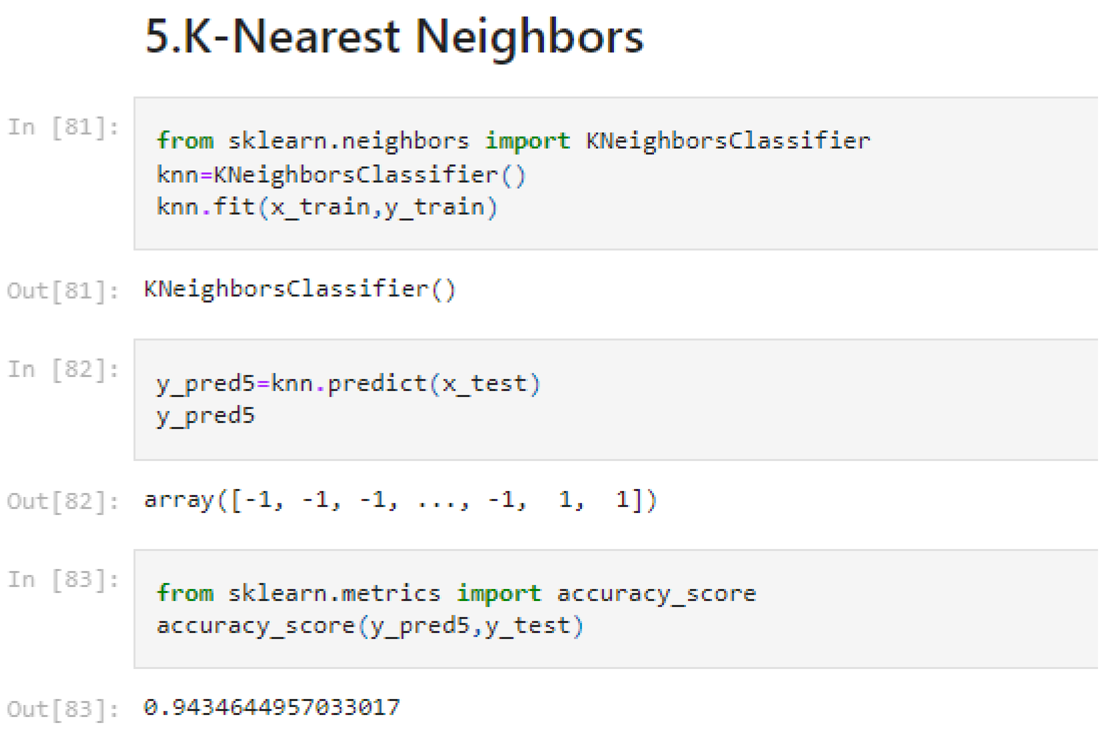
Out[77]:

```
0.9701492537313433
```



4.Support Vector Machine

```
In [78]: from sklearn.svm import SVC
         svm=SVC()
         svm.fit(x_train,y_train)

Out[78]: SVC()

In [79]: y_pred4=svm.predict(x_test)
         y_pred4

Out[79]: array([-1, -1,  1, ...,  1, -1,  1])

In [80]: from sklearn.metrics import accuracy_score
         accuracy_score(y_pred4,y_test)

Out[80]: 0.9407507914970602
```



5.K-Nearest Neighbors

```
In [81]: from sklearn.neighbors import KNeighborsClassifier
         knn=KNeighborsClassifier()
         knn.fit(x_train,y_train)

Out[81]: KNeighborsClassifier()

In [82]: y_pred5=knn.predict(x_test)
         y_pred5

Out[82]: array([-1, -1, -1, ..., -1,  1,  1])

In [83]: from sklearn.metrics import accuracy_score
         accuracy_score(y_pred5,y_test)

Out[83]: 0.9434644957033017
```

From the above results of Machine Learning algorithms, It clears that Random Forest Regression Machine Learning Algorithm has the greatest accuracy than other Machine Learning algorithms.