

Project Development Phase Model Performance Test

Date	21 November 2023
Team ID	PNT2022TMID592399
Project Name	Project - Disease Prediction using Machine Learning
Maximum Marks	10 Marks

Model Performance Testing:

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

Regression Model:

MAE - , MSE - , RMSE - , R2 score -

```
In [70]: from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
```

```
In [71]: mae = mean_absolute_error(y_test_encoded, y_pred3_encoded )
print(f'Mean Absolute Error (MAE): {mae}')

# Mean Squared Error (MSE)
mse = mean_squared_error(y_test_encoded, y_pred3_encoded )
print(f'Mean Squared Error (MSE): {mse}')

# Root Mean Squared Error (RMSE)
rmse = np.sqrt(mse)
print(f'Root Mean Squared Error (RMSE): {rmse}')

# R-squared (R2 score)
r2 = r2_score(y_test_encoded, y_pred3_encoded )
print(f'R-squared (R2 score): {r2}')
```

```
Mean Absolute Error (MAE): 0.023809523809523808
Mean Squared Error (MSE): 0.023809523809523808
Root Mean Squared Error (RMSE): 0.1543033499620919
R-squared (R2 score): 0.999826521550567
```

Classification Model:

Confusion Matrix - , Accuray Score- & Classification Report -

```
In [63]: from sklearn.metrics import accuracy_score, confusion_matrix, classification_report, roc_auc_score, roc_curve
print(accuracy_score(y_pred3, y_test))
```

0.9761904761904762

0.9761904761904762

```
In [64]: print(classification_report(y_test, y_pred3))
```

	precision	recall	f1-score	support
(vertigo) Paroymsal	1.00	1.00	1.00	1
Positional Vertigo	1.00	1.00	1.00	1
AIDS	1.00	1.00	1.00	1
Acne	1.00	1.00	1.00	1
Alcoholic hepatitis	1.00	1.00	1.00	1
Allergy	1.00	1.00	1.00	1
Arthritis	1.00	1.00	1.00	1
Bronchial Asthma	1.00	1.00	1.00	1
Cervical spondylosis	1.00	1.00	1.00	1
Chicken pox	1.00	1.00	1.00	1
Chronic cholestasis	1.00	1.00	1.00	1
Common Cold	1.00	1.00	1.00	1
Dengue	1.00	1.00	1.00	1
Diabetes	1.00	1.00	1.00	1
Dimorphic hemmorhoids(piles)	1.00	1.00	1.00	1
Drug Reaction	0.50	1.00	0.67	1
Fungal infection	1.00	0.50	0.67	2
GERD	1.00	1.00	1.00	1
Gastroenteritis	1.00	1.00	1.00	1
Heart attack	1.00	1.00	1.00	1
Hepatitis B	1.00	1.00	1.00	1
Hepatitis C	1.00	1.00	1.00	1
Hepatitis D	1.00	1.00	1.00	1
Hepatitis E	1.00	1.00	1.00	1
Hypertension	1.00	1.00	1.00	1
Hyperthyroidism	1.00	1.00	1.00	1
Hypoglycemia	1.00	1.00	1.00	1
Hypothyroidism	1.00	1.00	1.00	1
Impetigo	1.00	1.00	1.00	1
Jaundice	1.00	1.00	1.00	1
Malaria	1.00	1.00	1.00	1
Migraine	1.00	1.00	1.00	1
Osteoarthritis	1.00	1.00	1.00	1
Paralysis (brain hemorrhage)	1.00	1.00	1.00	1
Peptic ulcer disease	1.00	1.00	1.00	1
Pneumonia	1.00	1.00	1.00	1
Psoriasis	1.00	1.00	1.00	1
Tuberculosis	1.00	1.00	1.00	1
Typhoid	1.00	1.00	1.00	1
Urinary tract infection	1.00	1.00	1.00	1
Varicose veins	1.00	1.00	1.00	1
hepatitis A	1.00	1.00	1.00	1
accuracy			0.98	42
macro avg	0.99	0.99	0.98	42
weighted avg	0.99	0.98	0.98	42

```
In [65]: cm = confusion_matrix(y_test,y_pred3)
print(cm)
```

```
[[1 0 0 ... 0 0 0]
 [0 1 0 ... 0 0 0]
 [0 0 1 ... 0 0 0]
 ...
 [0 0 0 ... 1 0 0]
 [0 0 0 ... 0 1 0]
 [0 0 0 ... 0 0 1]]
```

```
In [66]: pd.crosstab(y_test,y_pred3)
```

Tune the Model

Hyperparameter Tuning -
Validation Method -

Performance Testing & Hyperparameter Tuning

In [43]: `x_train.shape`

Out[43]: (3936, 89)

In [44]: `a=rfc.feature_importances_
print(a)
print(len(a))`

```
[0.00731905 0.00618061 0.00886982 0.00719549 0.01095323 0.00963366  
0.00911179 0.00731008 0.01071234 0.00657018 0.00900867 0.01050089  
0.00292587 0.00524764 0.01237832 0.01180145 0.0115499 0.00871327  
0.00849029 0.0098106 0.00972926 0.00549846 0.00494291 0.00935291  
0.00587691 0.0057526 0.01224661 0.00787538 0.01016185 0.00918115  
0.00873608 0.01610834 0.00998105 0.00783738 0.00329483 0.00953713  
0.01890555 0.01429013 0.00749357 0.01256548 0.01233597 0.01264167  
0.0141116 0.01347112 0.00800466 0.01544509 0.01728203 0.01474086  
0.00815067 0.01316282 0.00797272 0.01403469 0.01183176 0.01402696  
0.01005894 0.01406593 0.01367007 0.00650629 0.01208545 0.01142946  
0.00764562 0.01170572 0.02565916 0.00524023 0.01250634 0.0134911  
0.01460353 0.02251584 0.00781893 0.00908393 0.01752008 0.00673618  
0.00699536 0.02724331 0.01254273 0.01204232 0.00791377 0.02060109  
0.00779964 0.02123292 0.02067059 0.00776417 0.0221095 0.02561995  
0.00902168 0.00726267 0.01043231 0.00929853 0.00824331]
```

89

In [45]: `col=x.columns
feat_imp={}
for i,j in zip(a,col):
 feat_imp[j]=i`

In [46]: `feat_imp`

Out[46]: {'itching': 0.007319048042835398,
'skin_rash': 0.006180613815992404,
'nodal_skin_eruptions': 0.00886981820978196,
'continuous_sneezing': 0.007195491736036008,
'shivering': 0.010953230794687727,
'chills': 0.009633658382390069,
'joint_pain': 0.009111789899665584,
'stomach_pain': 0.007310084487467259,
'acidity': 0.010712342888172164,
'ulcers_on_tongue': 0.0065701754821160035,
'muscle_wasting': 0.00900867283535562,
'vomiting': 0.010500890689574969,
'burning_micturition': 0.0029258679030467077,
'spotting_urination': 0.005247644478514162,
'fatigue': 0.012378316418070523,
'weight_loss': 0.011801445076106249,
'restlessness': 0.011549895436480091,
'lethargy': 0.00871327138248388,
'patches_in_throat': 0.008490294649098808,
'cough': 0.00981060420180069,
'high_fever': 0.009729255416372548,
'sunken_eyes': 0.00549845618016258,
'breathlessness': 0.004942913624074378,

```
In [47]: print(x_val.shape)
print(x_train.shape)
print(x_test.shape)
```

```
(984, 89)
(3936, 89)
(42, 89)
```

```
In [48]: import pickle
rfc1_results=[]
```

```
In [49]: knn1_results=[]
```

```
In [ ]:
```

```
In [50]: for main in [0.020,0.018,0.016,0.014,0.012,0.01,0.008]:
to_drop=[]
for i,j in zip(feats.keys(),feats.values()):
if j<main:
to_drop.append(i)

x_new=x.drop(to_drop,axis=1)
y_new=y
x1_train,x1_val,y1_train,y1_val=train_test_split(x_new,y_new,test_size=0.2)
x1_test = x_test.drop(to_drop,axis=1)
y1_test=y_test
rfc_new=RandomForestClassifier()
rfc_new.fit(x_train,y_train)
temp1=model_evaluation(rfc_new,x1_train.shape[1])
rfc_results.append(temp1)
knn_new=KNeighborsClassifier()
knn_new.fit(x_train, y_train)
temp2=model_evaluation(knn_new,x1_train.shape[1])
knn_res.append(temp2)
```

```
Number of Features: 8
The Training Accuracy of the algorithm is 1.0
The Validation Accuracy of the algorithm is 1.0
The Testing Accuracy of the algorithm is 0.9761904761904762
Number of Features: 8
```

```
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. 'skew', 'kurtosis'), the default behavior of 'mode' typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of 'keepdims' will become False, the 'axis' over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set 'keepdims' to True or False to avoid this warning.
```

```
mode, _ = stats.mode(y[neigh_ind, k], axis=1)
```

```
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. 'skew', 'kurtosis'), the default behavior of 'mode' typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of 'keepdims' will become False, the 'axis' over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set 'keepdims' to True or False to avoid this warning.
```

```
mode, _ = stats.mode(y[neigh_ind, k], axis=1)
```

```
/opt/anaconda3/lib/python3.9/site-packages/sklearn/neighbors/_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. 'skew', 'kurtosis'), the default behavior of 'mode' typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of 'keepdims' will become False
```