```python
import numpy
import matplotlib.pyplot as plt
import pandas
import math
from keras.models import Sequential
from keras.layers import Dense
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error


from google.colab import files

uploaded = files.upload()
```

Escolher arquivos dataset_teste_selecao.csv

- **dataset_teste_selecao.csv**(application/vnd.ms-excel) - 220838 bytes, last modified: 11/01/2022 - 100% done

Saving dataset teste selecao csv to dataset teste selecao (13) csv

```python
df = pd.read_csv('dataset_teste_selecao.csv', encoding = "ISO-8859-1")
```

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças

	paciente_id	coriza	tosse_seca_ou_produtiva	calafrios	feb

```
df.describe()
```

	tempo_vmi_total	tempo_uti_total	tempo_internacao	tempo_referencia_internac
count	229.000000	385.000000	573.000000	535.0000
mean	8.524017	6.722078	7.104712	3.7289
std	7.288274	5.998104	5.738125	3.0148
min	0.000000	0.000000	0.000000	0.0000
25%	3.000000	2.000000	3.000000	2.0000
50%	7.000000	5.000000	5.000000	3.0000
75%	11.000000	9.000000	9.000000	5.0000
max	40.000000	34.000000	37.000000	18.0000

```
df = df.replace(["Sim", "Não"], [1, 0])
df = df.replace(["Masculino", "Feminino"], [1, 0])
df = df.replace(["Caso suspeito", "Caso confirmado"], [1, 0])
df = df.replace(["obito", "alta"], [1, 0])
```

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças

| | paciente_id | coriza | tosse_seca_ou_produtiva | calafrios | feb |

```
columns = ['paciente_id', 'nomemacro','bairro','idh_label','cor_autorreferida','estado_civ
df.drop(columns, inplace=True, axis=1)
```

	paciente_id	coriza		calafrios	
2	99c5e07b4d5de9d18c350cdf64c5aa3d	0		0	0

```
df.head()
```

	coriza	tosse_seca_ou_produtiva	calafrios	febre	dispneia	fadiga	anorexia	mi
0	0	0	0	0	0	0	0	
1	0	1	0	1	1	0	0	
2	0	0	0	1	1	0	0	
3	0	0	0	0	1	0	0	

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças

```
df=  df.fillna(0)
```

```
df.head()
```

	coriza	tosse_seca_ou_produtiva	calafrios	febre	dispneia	fadiga	anorexia	mi
0	0	0	0	0	0	0	0	
1	0	1	0	1	1	0	0	
2	0	0	0	1	1	0	0	
3	0	0	0	0	1	0	0	
4	0	0	0	0	1	0	0	

```
df = df.astype('int')
```

```
df.info()
```

```
0    coriza                              576 non-null    int64
1    tosse_seca_ou_produtiva             576 non-null    int64
2    calafrios                           576 non-null    int64
3    febre                               576 non-null    int64
4    dispneia                            576 non-null    int64
5    fadiga                              576 non-null    int64
6    anorexia                            576 non-null    int64
7    mialgia                             576 non-null    int64
8    astenia                             576 non-null    int64
9    dor_de_garganta                     576 non-null    int64
10   congestao_nasal                     576 non-null    int64
11   cefaleia                            576 non-null    int64
12   diarreia                            576 non-null    int64
13   nausea                              576 non-null    int64
14   vomitos                             576 non-null    int64
15   anosmia                             576 non-null    int64
16   ageusia                             576 non-null    int64
17   tabagismo                           576 non-null    int64
18   drogas                              576 non-null    int64
19   etilismo                            576 non-null    int64
20   diabetes                            576 non-null    int64
```

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças

```
23   doenca_cardiaca                     576 non-null    int64
24   doenca_pulmonar                     576 non-null    int64
25   doenca_reumatologica                576 non-null    int64
26   doenca_autoimune                    576 non-null    int64
27   doenca_renal_cronica                576 non-null    int64
28   doenca_hepatica_cronica             576 non-null    int64
29   doenca_neurologica                  576 non-null    int64
30   paciente_chegou_com_suporte_respiratorio  576 non-null  int64
31   vmi                                 576 non-null    int64
32   uti                                 576 non-null    int64
33   necessidade_transfusional           576 non-null    int64
34   fenomenos_tromboembolicos_tvp       576 non-null    int64
35   fenomenos_tromboembolicos_tep       576 non-null    int64
36   sepse                               576 non-null    int64
37   disfuncao_hepatica                  576 non-null    int64
38   choque_septico                      576 non-null    int64
39   disfuncao_multipla_de_orgaos        576 non-null    int64
40   choque_hipovolemico                 576 non-null    int64
41   complicacao_neurologica             576 non-null    int64
42   hemorragia                          576 non-null    int64
43   hemodialise                         576 non-null    int64
44   tempo_vmi_total                     576 non-null    int64
45   tempo_uti_total                     576 non-null    int64
46   tempo_internacao                    576 non-null    int64
47   tempo_referencia_internacao         576 non-null    int64
48   sexo                                576 non-null    int64
```

```
49  idade                                    576 non-null    int64
50  tipo_caso_à_admissão                     576 non-null    int64
51  tempo_sintomas_internacao                576 non-null    int64
52  desfecho                                 576 non-null    int64
53  obito_menos24horas                       576 non-null    int64
54  obito_vm                                 576 non-null    int64
55  obito_uti                                576 non-null    int64
dtypes: int64(56)
memory usage: 252.1 KB
```

df.columns

```
Index(['coriza', 'tosse_seca_ou_produtiva', 'calafrios', 'febre', 'dispneia',
       'fadiga', 'anorexia', 'mialgia', 'astenia', 'dor_de_garganta',
       'congestao_nasal', 'cefaleia', 'diarreia', 'nausea', 'vomitos',
       'anosmia', 'ageusia', 'tabagismo', 'drogas', 'etilismo', 'diabetes',
       'obesidade', 'hipertensao_arterial', 'doenca_cardiaca',
       'doenca_pulmonar', 'doenca_reumatologica', 'doenca_autoimune',
       'doenca_renal_cronica', 'doenca_hepatica_cronica', 'doenca_neurologica',
       'paciente_chegou_com_suporte_respiratorio', 'vmi', 'uti',
       'necessidade_transfusional', 'fenomenos_tromboembolicos_tvp',
       'fenomenos_tromboembolicos_tep', 'sepse', 'disfuncao_hepatica',
       'choque_septico', 'disfuncao_multipla_de_orgaos', 'choque_hipovolemico',
       'complicacao_neurologica', 'hemorragia', 'hemodialise',
       'tempo_vmi_total', 'tempo_uti_total', 'tempo_internacao',
       'tempo_referencia_internacao', 'sexo', 'idade', 'tipo_caso_à_admissão',
       'tempo_sintomas_internacao', 'desfecho', 'obito_menos24horas',
       'obito_vm', 'obito_uti'],
      dtype='object')
```

> Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
> Mostrar diferenças

	coriza	tosse_seca_ou_produtiva	calafrios	febre	dispneia	fadiga	anorexia	mi
0	0	0	0	0	0	0	0	0
1	0	1	0	1	1	0	0	
2	0	0	0	1	1	0	0	
3	0	0	0	0	1	0	0	
4	0	0	0	0	1	0	0	

X = df.drop(['desfecho'],axis=1)

y = df.desfecho

y.head()

```
0    1
```

```
1    0
2    0
3    1
4    1
Name: desfecho, dtype: int64
```

```python
X_train, X_test, y_train, y_test = train_test_split(X,y,random_state=12,test_size=0.3)
```

```python
from imblearn.over_sampling import SMOTE
```

```python
sm = SMOTE(random_state=12)
X_train_bal, y_train_bal = sm.fit_resample(X_train,y_train)
```

```python
from sklearn.neural_network import MLPClassifier
```

```python
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(n_estimators=100,random_state=12,verbose=1,n_jobs=-1).fit(X
```

```
[Parallel(n_jobs=-1)]: Using backend ThreadingBackend with 2 concurrent workers.
[Parallel(n_jobs=-1)]: Done  46 tasks       | elapsed:    0.1s
[Parallel(n_jobs=-1)]: Done 100 out of 100 | elapsed:    0.2s finished
```

```python
neurons = [50,100,50]
```

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças

```python
mlp = MLPClassifier(hidden_layer_sizes=neurons,activation='logistic',random_state=12,verbo
                    early_stopping=True,validation_fraction=0.2,max_iter=100)
mlp.fit(X_train_bal,y_train_bal)
```

```
Validation score: 0.962617
Iteration 38, loss = 0.22383143
Validation score: 0.971963
Iteration 39, loss = 0.21557945
Validation score: 0.971963
Iteration 40, loss = 0.21048426
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
  "X does not have valid feature names, but"
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
  "X does not have valid feature names, but"
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
  "X does not have valid feature names, but"
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
  "X does not have valid feature names, but"
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
  "X does not have valid feature names, but"
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
  "X does not have valid feature names, but"
/usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
```

```
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    Validation score: 0.962617
    Iteration 41, loss = 0.21068731
    Validation score: 0.962617
    Iteration 42, loss = 0.20208031
    Validation score: 0.971963
    Iteration 43, loss = 0.19864009
    Validation score: 0.971963
    Iteration 44, loss = 0.19762363
    Validation score: 0.971963
    Iteration 45, loss = 0.19452592
    Validation score: 0.971963
    Validation score did not improve more than tol=0.000100 for 10 consecutive epochs.
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
    /usr/local/lib/python3.7/dist-packages/sklearn/base.py:451: UserWarning: X does no
      "X does not have valid feature names, but"
```

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças

```
      "X does not have valid feature names, but"
    MLPClassifier(activation='logistic', early_stopping=True,
                  hidden_layer_sizes=[50, 100, 50], max_iter=100, random_state=12,
                  validation_fraction=0.2, verbose=True)
```

```python
pred = mlp.predict(X_test)


from sklearn.metrics import mean_squared_error, confusion_matrix, accuracy_score, precisio
from sklearn.metrics import recall_score, cohen_kappa_score, balanced_accuracy_score


S = recall_score(y_test,pred,pos_label=1)
E = recall_score(y_test,pred,pos_label=0)

beta = 0.7
wtdAcc = (beta*E) + ((1-beta) * S)
wtdAcc
```

```
    0.923971278652306
```

```python
gmean = pow((S*E),0.5)
```

gmean

        0.9255643836787986

```
print('Mean squared error: ' + str(mean_squared_error(y_test,pred)))

print('\nConfusion matrix')
print(confusion_matrix(y_test,pred))

print('\nAccuracy: ' + str(accuracy_score(y_test, pred)))

print('Weighted accuracy: ' + str(wtdAcc))

print('Precision: ' + str(precision_score(y_test,pred)))

print('Recall: ' + str(S))

print('Especificidade: ' + str(E))

#print('F1-Score: ' + str(f1_score(y_test, predRF)))
print('G-Mean: ' + str(gmean))

print('\nROC AUC Score: ' + str(roc_auc_score(y_test,pred)))

print('Cohen-Kappa Score: ' + str(cohen_kappa_score(y_test,pred)))
```

        Mean squared error: 0.07514450867052024

        Confusion matrix

    Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
    Mostrar diferenças

        Accuracy: 0.9248554913294798
        Weighted accuracy: 0.923971278652306
        Precision: 0.8918918918918919
        Recall: 0.9295774647887324
        Especificidade: 0.9215686274509803
        G-Mean: 0.9255643836787986

        ROC AUC Score: 0.9255730461198565
        Cohen-Kappa Score: 0.8457158537421966

✓ 0s conclusão: 18:10 ● ✕

Falha no salvamento automático. Este arquivo foi atualizado remotamente ou em outra guia.
Mostrar diferenças