

```
#####
#####
#   Enunciado do Exemplo:
#
#   Suponha que você está usando um algoritmo para um aprendizado de máquina que precisa prever se pacientes
#   de uma clínica estão infectados ou não por um vírus. Após treinar seu algoritmo com dados de treino, você
#   escolhe 10 valores de teste e monta a tabela a seguir:
#
#       Predição   |   Real
#   -----
#       Tem Vírus   |   Tem Vírus
#       Não tem Vírus | Não tem Vírus
#       Tem Vírus   |   Tem Vírus
#       Tem Vírus   |   Tem Vírus
#       Não tem Vírus | Tem Vírus
#       Tem Vírus   |   Não tem Vírus
#       Tem Vírus   |   Tem Vírus
#       Tem Vírus   |   Não tem Vírus
#       Tem Vírus   |   Tem Vírus
#       Tem Vírus   |   Tem Vírus
#
#   Exemplo de como construir uma matriz de confusão e como calcular os parâmetros a seguir:
#
#   - Acurácia;
#   - Sensibilidade ou Recall;
#   - Especificidade;
#   - Precisão;
#   - F1-Score (Média Harmônica entre Precisão e Recall)
#
#####
#####

# Importando as bibliotecas básicas
import pandas as pd
import seaborn as sn
import matplotlib.pyplot as plt

# Importando as bibliotecas para o cálculo dos parâmetros relacionados à matriz de confusão
from sklearn import metrics

# Tabela ou dataset com os resultados de teste
dados_de_teste = {
```

```

'Resultado_Predito_Virus': ['Tem Vírus', 'Não tem Vírus', 'Tem Vírus',
'Tem Vírus', 'Não tem Vírus', 'Tem Vírus', 'Tem Vírus', 'Tem Vírus', 'T
em Vírus', 'Tem Vírus'],
'Resultado_Real_Virus'    : ['Tem Vírus', 'Não tem Vírus', 'Tem Vírus',
'Tem Vírus', 'Tem Vírus', 'Não tem Vírus', 'Tem Vírus', 'Não tem Vírus'
, 'Tem Vírus', 'Tem Vírus']
}

df = pd.DataFrame(data = dados_de_teste, columns = ['Resultado_Predito_
Virus', 'Resultado_Real_Virus'])
print(df)

# Mapeando os classificadores do problema
# Tem Vírus = 1
# Não tem Vírus = 0
df['Resultado_Predito_Virus'] = df['Resultado_Predito_Virus'].map({'Tem
Vírus': 1, 'Não tem Vírus': 0})
df['Resultado_Real_Virus'] = df['Resultado_Real_Virus'].map({'Tem Vírus
': 1, 'Não tem Vírus': 0})

# Criando a matriz de confusão para os dados de teste
matriz_confusao = pd.crosstab(df['Resultado_Predito_Virus'], df['Result
ado_Real_Virus'], rownames = ['Resultado_Predito_Virus'], colnames=['Re
sultado_Real_Virus'], margins = False)
print(matriz_confusao)

# Transformando a matriz de confusão para podermos visualizar e interpr
etar a mesma como um mapa de calor
sn.heatmap(matriz_confusao, annot = True)

#Mostrando a matriz de confusão com a bibliotecas matplotlib
plt.show()

#####
#####
#
# Calculando e apresentando os parâmetros: Recall, Especificidade, Acur
ácia, Precisão e F1-Score
#
#####
#####

matriz_confusao = metrics.confusion_matrix(df['Resultado_Predito_Virus'
], df['Resultado_Real_Virus'])

#####
#####

```

```

#
# Para o calculo correto dos parâmetros foi preciso inverter a posição
# de cada lista de dados do dataset, de modo que:
#
# df['Resultado_Predito_Virus'], df['Resultado_Real_Virus'] --
> df['Resultado_Real_Virus'], df['Resultado_Predito_Virus']
#
#####
#####
Sensibilidade_Recall = metrics.recall_score(df['Resultado_Real_Virus'],
df['Resultado_Predito_Virus'])
print("\n Sensibilidade (Recall): ", Sensibilidade_Recall)

Especificidade = metrics.recall_score(df['Resultado_Real_Virus'], df['R
esultado_Predito_Virus'], pos_label = 0)
print("\n Especificidade: ", Especificidade)

Acuracia = metrics.accuracy_score(df['Resultado_Real_Virus'], df['Resul
tado_Predito_Virus'])
print("\n Acurácia: ", Acuracia)

Precisao = metrics.precision_score(df['Resultado_Real_Virus'], df['Resu
ltado_Predito_Virus'])
print("\n Precisão: ", Precisao)

F1_Score = metrics.f1_score(df['Resultado_Real_Virus'], df['Resultado_P
redito_Virus'])
print("\n F1-Score: ", F1_Score)

```