Questao_5

February 28, 2021

0.1 Questão 5

```python
In [35]: import pandas as pd
         import numpy as np
```

```python
In [36]: np.random.seed(42)
```

```python
In [37]: path = "teste_smarkio_lbs.xls"
         df2 = pd.read_excel(path, 'NLP')
```

```python
In [38]: df2.head()
```

```
Out[38]:                                               letra  artista
         0  Jay-z Uh-uh-uh You ready b? Let's go get 'em. ...  Beyoncé
         1  Your challengers are a young group from Housto...  Beyoncé
         2  Dum-da-de-da Do, do, do, do, do, do (Coming do...  Beyoncé
         3  If I ain't got nothing I got you If I ain't go...  Beyoncé
         4  Six inch heels She walked in the club like nob...  Beyoncé
```

```python
In [39]: # Codificando as artistas: Rihanna = 0 e Beyonce = 1
         df2['artista'] = np.where(df2['artista']=='Rihanna',0,1)
```

```python
In [40]: df2.head()
```

```
Out[40]:                                               letra  artista
         0  Jay-z Uh-uh-uh You ready b? Let's go get 'em. ...        1
         1  Your challengers are a young group from Housto...        1
         2  Dum-da-de-da Do, do, do, do, do, do (Coming do...        1
         3  If I ain't got nothing I got you If I ain't go...        1
         4  Six inch heels She walked in the club like nob...        1
```

```python
In [41]: from sklearn.model_selection import train_test_split
         X_treino,X_teste,y_treino,y_teste = train_test_split(df2['letra'],\
                                                              df2['artista'],\
                                                              test_size=0.3)
```

```python
In [42]: y_treino.head()
```

```
Out[42]: 94     1
         289    0
         291    0
         23     1
         365    0
         Name: artista, dtype: int32
```

```python
In [43]: # Obtenção da matriz tf-idf da amostra de treino para input no modelo
         from sklearn.feature_extraction.text import CountVectorizer
         from sklearn.feature_extraction.text import TfidfTransformer

         vetor= CountVectorizer(stop_words='english',strip_accents='unicode')
         X_treino_tf = vetor.fit_transform(X_treino)

         transf = TfidfTransformer()
         X_treino_tfidf = transf.fit_transform(X_treino_tf)
```

```python
In [44]: from sklearn.naive_bayes import MultinomialNB
         modelo = MultinomialNB().fit(X_treino_tfidf,y_treino)
```

```python
In [45]: # Preparação da amostra de validação para predição
         X_teste_tf = vetor.transform(X_teste)
         X_teste_tfidf = transf.transform(X_teste_tf)
```

```python
In [46]: predicao=modelo.predict(X_teste_tfidf)
```

```python
In [47]: # 0 - Rihanna/ 1 - Beyonce
         from sklearn import metrics

         metrics.confusion_matrix(y_teste,predicao)
```

```
Out[47]: array([[30, 45],
                [ 5, 76]], dtype=int64)
```

```python
In [48]: from sklearn.metrics import accuracy_score,precision_score,recall_score,f1_score
         acuracia = accuracy_score(y_teste,predicao)
         precisao = precision_score(y_teste,predicao)
         recall = recall_score(y_teste,predicao)
         f1 = f1_score(y_teste,predicao)
```

```python
In [49]: print('Métricas: \n - Acuracia: {},\n - Precisao: {},\n - Recall: {},\n - F1: {}'.
               format(acuracia,precisao,recall,f1))
```

```
Métricas:
 - Acuracia: 0.6794871794871795,
 - Precisao: 0.628099173553719,
 - Recall: 0.9382716049382716,
 - F1: 0.7524752475247526
```