

Reconhecimento do Gênero de Pessoas a Partir dos Sinais de suas Vozes

Kallil M. Caparroz¹, Rodrigo C. Anater²

¹Universidade Tecnológica Federal do Paraná - Câmpus Pato Branco

kallil@alunos.utfpr.edu.br, rodrigoanater@alunos.utfpr.edu.br

Abstract. *During the past two decades, the Machine Learning technique has become one of the main studies of information technology. With the increasing amount of information to be analyzed, there is a good reason to develop more and more computational algorithms to analyze these data. The objective of this work is to use different algorithms in the same dataset and compare which one is the best to be used.*

Resumo. *Durante as duas últimas décadas, a técnica de Aprendizado de Máquinas tornou-se um dos principais estudos da tecnologia de informação. Com o aumento da quantidade de informações a serem analisadas, há uma boa razão para desenvolver mais e mais algoritmos computacionais a fim de analisar esses dados. O objetivo deste trabalho é usar diferentes algoritmos no mesmo conjunto de dados e comparar qual deles é o melhor a ser usado.*

1. Introdução

O termo *Machine Learning* se refere à detecção, por parte do computador, em reconhecer padrões em uma base de dados. Nas últimas décadas, tornou-se uma ferramenta fundamental para qualquer tarefa que requer a extração de informações de grandes conjuntos de dados.

Estamos cercados por tecnologia baseada em *Machine Learning*: Mecanismos de pesquisa aprendem como nos disponibilizar os melhores resultados, softwares anti-spam aprendem como filtrar nossos e-mails, assim como transações de cartões de créditos são mantidas em segurança por um software que aprende a detectar fraudes. [Shalev-Shwartz and Ben-David 2014] Para tal, vários algoritmos conseguem fazer com que um computador possa automaticamente detectar os padrões de uma base de dados.

Nesse trabalho será utilizada uma base de dados criada para identificar uma voz como masculina ou feminina baseada nas propriedades acústicas da voz e da fala. A base de dados consiste em 3 168 amostras de voz gravadas, coletadas de falantes masculinos e femininos, tendo 22 características, todas envolvendo a frequência sonora da fala, sendo algumas delas:

- Frequência média (kHz)
- Desvio padrão da frequência
- Mediana da frequência (kHz)
- Centróide da frequência

Para o reconhecimento dos padrões dessa base de dados serão utilizados duas técnicas: Regressão Linear e Máquinas de Vetores de Suporte.

2. Regressão Linear

O principal objetivo por trás da regressão linear é de estimar um valor $y \in \mathbb{R}$ dado uma característica x . Por exemplo, pode-se estimar o valor de um estoque no dia seguinte de acordo com as vendas dos dias anteriores, os batimentos cardíacos de um atleta de acordo com a distância que percorreu, entre outras aplicações [Smola and Vishwanathan 2008].

A Figura 1 mostra um bom exemplo de um método de regressão linear. Os pontos na figura são amostras obtidas das características x da base de dados, o objetivo é criar uma função $f(x)$ que se aproxime da melhor forma dos valores observados.

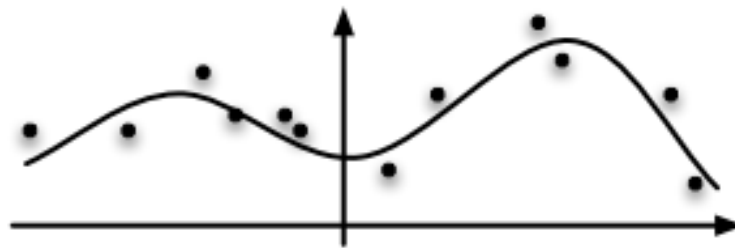


Figura 1. Exemplo de regressão linear

A função $f(x)$ utilizada pode ser representada pela forma:

$$f(x) = b_0 + b_1 * x_1 + b_1 * x_1 + b_2 * x_2 + \dots + b_n * x_n$$

onde $x_{1,2,..n}$ correspondem às características da base de dados utilizada, b_0 corresponde à constante e $b_{1,2,..n}$ são os coeficientes. Os dois últimos são termos que o método de regressão linear irá ajustar de forma a obter uma função que se aproxime dos valores dados pelo conjunto de treinamento.

Abaixo o código em linguagem python do processo de aprendizado da base de dados:

```
1 # Importing the libraries
2 import numpy as np
3 import pandas as pd
4
5 # Lista de Parametros
6 RATIO_TESTE_TREINO = 1000/3168
7 rs = np.random.randint(0,100)
8
9
10 #Importando a base de dados
11 dataset = pd.read_csv('voice.csv')
12 X = dataset.iloc[:, :-1].values
13 y = dataset.iloc[:, 20].values
14
```

```

15 # Transformando as labels em numeros
16 from sklearn.preprocessing import LabelEncoder, OneHotEncoder
17 labelencoder_y = LabelEncoder()
18 y = labelencoder_y.fit_transform(y)
19 onehotencoder = OneHotEncoder(categorical_features = [0])
20 y = onehotencoder.fit_transform(y).toarray()
21 y = np.transpose(y)
22
23 # Dividindo a base de dados em treinamento e teste
24 from sklearn.cross_validation import train_test_split
25 X_train, X_test, y_train, y_test =
    train_test_split(X,y,test_size=RATIO_TESTE_TREINO, stratify=y)
26
27 # Metodo de regressao linear simples
28 from sklearn.linear_model import LinearRegression
29 regressor = LinearRegression()
30 regressor.fit(X_train, y_train)
31
32 #Previendo os resultados da base de testes
33 y_pred = regressor.predict(X_test)
34 y_pred = np.round(y_pred)
35
36 #Calcular a precisao dos resultados
37 from sklearn.metrics import accuracy_score
38 print('Accuracy simple linear regression: %.2f\n' %
    accuracy_score(y_test, y_pred))

```

3. SVM

Referências

- Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms*. Cambridge university press.
- Smola, A. and Vishwanathan, S. (2008). Introduction to machine learning. *Cambridge University, UK*, 32:34.