

Sistemas Inteligentes - T951

## Primeiro trabalho: Modelos Lineares.

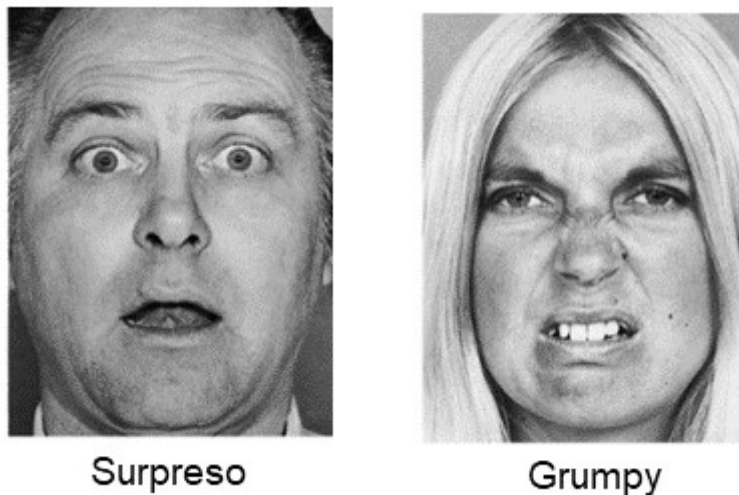
*Professor: Prof. Msc. Paulo Cirillo Souza Barbosa*

### O que fazer?

Será fornecido um conjunto de dados referente aos sinais de eletromiografia, captados nos músculos faciais: Corrugador do Supercílio (Sensor 1); Zigomático Maior (Sensor 2).

No presente conjunto de dados, tem-se 20 000 observações para os dois sensores, em classes totalmente balanceadas, ou seja, 10 000 são referentes ao gesto facial Surpreso, e as demais 10 000 amostras são referentes ao gesto Grumpy. Tais gestos podem ser visualizados na Figura 1.

Figura 1: Gestos (classes) presentes no conjunto de dados



O arquivo fornecido via AVA, trata-se de um .json contendo 10 rodadas de aquisições. Para cada rodada, há o indicativo (rótulo) referente aos gestos Surpreso e Grumpy. Acessando cada um destes, pode-se verificar dois vetores com os dados captados referentes aos sensores 1 e 2.

Pede-se inicialmente que faça a identificação de  $P$  (número de preditores),  $N$  (Quantidade de amostras) e  $C$  (Quantidade de classes). Em seguida, realizar o acesso ao arquivo .json e os organize conforme as matrizes que se seguem:

$$\mathbf{X} \in \mathbb{R}^{N \times P} \quad \mathbf{Y} \in \mathbb{R}^{N \times C}$$

De modo a auxiliá-los, disponibiliza-se o algoritmo em **Python** que realiza essa organização.

**OBS:** Nem sempre o autor do conjunto de dados, realizará essa organização. Estou disponibilizando esta, exclusivamente e unicamente para facilitar o primeiro trabalho.

```
1 import json
2 import numpy as np
3
4 f = open('EMG.json') #Coloque o arquivo na mesma pasta deste algoritmo
5 data = json.load(f)
6 c = 2
7 p = 2
8
9 X = np.empty((0,p))
```

```

10 Y = np.empty((0,c))
11
12 for j in data:
13     it = 0
14     data1= data[j]
15     for i in data1:
16
17         #ORGANIZANDO X
18         aux1= np.array(data1[i][0])
19         aux1.shape=(len(aux1),1)
20         aux2= np.array(data1[i][1])
21         aux2.shape=(len(aux2),1)
22         X = np.concatenate((X,np.concatenate((aux1,aux2),axis=1)),axis=0)
23
24         #ORGANIZANDO Y
25         y = np.zeros((len(aux2),c))
26         y[:,it] = 1
27         Y = np.concatenate((Y,y),axis=0)
28         it+=1

```

## 1) Análise inicial dos dados.

Como o presente conjunto de dados possui apenas dois preditores, pede-se que construa o gráfico de espalhamento (scatter plot).

## 1) Testando os Algoritmos Implementados

Pede-se que implemente os classificadores descritos nas próximas seções, porém, para todos deve-se calcular medidas de desempenhos. Tais medidas são: **Acurácia**; **Sensibilidade**; **Especificidade**. Para uma melhor confiabilidade dos resultados, solicita-se que o treinamento e teste de cada classificador seja executado 100 vezes. Para isso, deve-se para cada rodada embaralhar os dados e dividi-los em 80% das amostras para treinamento e 20% para teste.

Ao final das 100 rodadas, para cada classificador deve-se ter 100 dados referentes a Acurácia, Sensibilidade e Especificidade. Com isto posto, deve-se calcular a média, mediana, desvio padrão, menor e maior valor, para cada uma das medidas de desempenho.

Com esta análise, é possível verificar qual modelo possui uma melhor capacidade de classificação. Pede-se então para cada classificador com os melhores hiperparâmetros definidos, que faça mais uma rodada de treinamento e teste e construa-se o gráfico da matriz de confusão.

## 1) OLS.

1. Faça a implementação do método dos mínimos quadrados ordinário.
2. Faça a implementação do método dos mínimos quadrados regularizado.
  - Faça o treinamento do modelo utilizando valores de  $\lambda = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.9, 1\}$
  - **Pergunta:** O que significa  $\lambda = 0$ ??
3. Em sala foi realizada uma discussão sobre a adição de um vetor coluna de **1s** no início da matriz de dados **X**.
  - O que implicaria adicionar este vetor coluna para o presente trabalho?
  - Qual a interpretação geométrica?
  - O resultado obtido é melhor ou pior?

## 1) PS.

1. Faça a implementação do algoritmo do Perceptron Simples.
2. Faça o treinamento do modelo com as taxas de aprendizagem  $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.9, 1\}$ .

3. Para todos os casos, inicialize todos os vetores de peso com  $\mathbf{0}$ .
4. Dada uma rede perceptron, com apenas um neurônio artificial. Seria possível resolver um problema com classes não linearmente separáveis? Por quê?
5. Se o vetor de pesos não fosse inicializado com um vetor nulo, e sim com valores pequenos e aleatórios, a convergência terminaria com a mesma quantidade de épocas? A fronteira de decisão final é melhor ou pior?
6. Se  $x_0$  fosse nulo, qual é o desafio para que o modelo consiga dividir duas classes que são linearmente separáveis?

## 1) ADALINE.

1. Faça a implementação do algoritmo do ADALINE.
2. Faça o treinamento do modelo com as taxas de aprendizagem  $\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.9, 1\}$ .
3. Para todos os casos, inicialize todos os vetores de peso com  $\mathbf{0}$ .
4. Se estes vetores  $\mathbf{w}$  fossem inicializados com valores diferentes de nulo, o hiperplano ótimo se difere?
5. Explique as diferenças principais entre Perceptron Simples e o Adaline.
6. Discorra sobre eventuais instabilidades no processo de treinamento, quando é escolhido valores elevados OU valores ínfimos para a taxa de aprendizagem.
7. Para seu conjunto de dados, trace a curva de aprendizagem para o treinamento do ADALINE.
8. É correto dizer que o ADALINE é uma rede neural com maior tendência de imunidade com relação ao Perceptron Simples, a eventuais ruídos que podem afetar o processo de divisão das classes? Explique o motivo!