

# **Relatório LAMIA 22**

## **Prática: Reconhecimento de Emoções com TensorFlow 2.0 e Python (III)**

**Kaique Medeiros Lima**

### **1 Introdução**

Nesse card, é apresentado o reconhecimento de emoções que utiliza inteligência artificial para interpretar estados emocionais por meio de expressões faciais, voz e gestos. Com aplicações em diversas áreas, seu desenvolvimento envolve redes neurais e diversas técnicas de manipulação.

### **2 Descrição da atividade**

#### **2.1 Introdução ao reconhecimento de emoções**

##### **2.1.1 Expressões faciais**

Fornecem informações sobre a resposta emocional e exercem um papel fundamental na interação humana e como forma de comunicação não verbal, podendo complementá-la ou até mesmo transmitir uma mensagem completa por si só.

Pode também contribuir para o efeito de mensagem falada do que a entonação ou a própria mensagem.

##### **2.1.2 Aplicações**

- Carros inteligentes
- Jogos
- Saúde
- Indústria
- Educação
- Marketing
- Cinema

## 2.2 Redes Neurais Artificiais

### 2.2.1 Neurônio Artificial

Nodos de entrada recebem as informações de entrada que é multiplicada pelos pesos de cada nodo e aplicados às funções de soma e função degrau.

### 2.2.2 Redes multicamada

### 2.2.3 Função soma e função de ativação

As redes multicamadas são usadas quando os operadores são não linearmente separáveis. Existe também a função sigmoide:

$$y = \frac{1}{1 + e^{-x}}$$

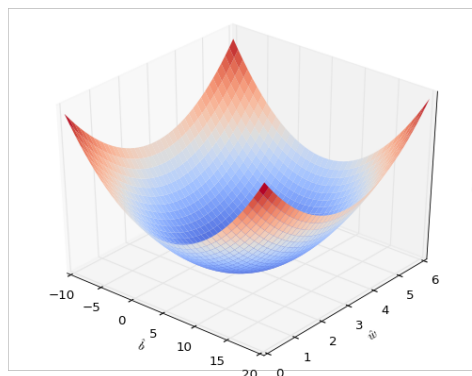
### 2.2.4 Cálculo do erro

O cálculo é feito pelo quão longe a resposta está de estar certa, subtraindo a resposta calculada da resposta correta temos o erro.

### 2.2.5 Descida do gradiente

Representa as combinações de pesos que resultam no erro. Os pesos iniciais resultam no erro da parte vermelha da imagem e a meta é descer o erro até a parte azul, o fundo significando um erro menor. A fórmula é:

$$d(y \cdot (1 - y))/dx, \text{ tal que } y = \frac{1}{1 + e^{-x}}$$



### 2.2.6 Cálculo do parâmetro delta

O parâmetro delta é o último passo prévio ao gradiente, é dividido em duas partes, a primeira se calcula na camada de saída e a segunda na camada densa escondida. Na camada de saída é feita com a multiplicação do erro pela derivada, já na segunda, entre o valor da derivada, peso e resultado da camada de saída.

### 2.2.7 Backpropagation

Consiste no reajuste de pesos da rede neural de acordo com os resultados obtidos durante o treinamento.

### 2.2.8 Bias, erro, descida do gradiente estocásticas e mais parâmetros

**Bias:** Neurônio adicional que se liga em todos os outros, mas com valores diferentes, mudando a saída mesmo que sejam com os mesmos valores.

**Erro:** Resposta correta - Resposta calculada.

**Stochastic Gradient Descent (SGD):** Previne mínimos locais.

**Learning rate:** Taxa de aprendizado.

**Batch size:** Tamanho do lote de uso.

**Epochs:** Quantidade de vezes que o programa vai ser rodado.

### 2.2.9 Funções de ativação

**Degrau:** Retorna 1 se a entrada for positiva e 0 caso contrário, usada em perceptrons.

**Sigmoide:** Mapeia valores para o intervalo (0,1), útil em probabilidades, mas sofre com o problema do gradiente desaparecendo.

**Tangente hiperbólica:** Similar à sigmoide, mas varia entre (-1,1), centrada na origem, reduzindo deslocamento médio.

**ReLU:** Retorna 0 para valores negativos e x para positivos, acelerando o aprendizado, mas pode sofrer com neurônios mortos.

**Linear:** Mantém a entrada inalterada, usada em regressão, mas não introduz não linearidade.

**Softmax:** Converte um vetor de valores em probabilidades normalizadas, usada em classificação multiclasse.

## 3 Conclusão

A utilização de inteligência artificial para o reconhecimento de emoções tem se mostrado uma ferramenta poderosa em diversas áreas, desde a indústria até a saúde. Redes neurais artificiais possibilitam avanços significativos nesse campo.