



UNIVERSIDADE FEDERAL DO PIAUÍ

CAMPUS MINISTRO PETRÔNIO PORTELLA

CURSO: CIÊNCIA DA COMPUTAÇÃO - 2020.2

MATERIA: INTELIGÊNCIA ARTIFICIAL

PROF.: VINICIUS PONTE MACHADO

RELATORIO

Francisco Jair de Oliveira Neres

Anderson Geraldo de Carvalho Baeta

Perceptron e Redes Neurais

Redes neurais são sistemas de computação com nós interconectados que funcionam como os neurônios do cérebro humano. Usando algoritmos, elas podem reconhecer padrões escondidos e correlações em dados brutos, agrupá-los e classificá-los, e – com o tempo – aprender e melhorar continuamente.

No trabalho foi dado um problema para classificar indivíduos, esses indivíduos foram divididos em duas classes, jogadores de futebol e jogadores de tênis. Para poder classificar os indivíduos para saber em qual classe cada um pertence foi implementando um *perceptron* na linguagem Python. Uma classe Chamada Perceptron foi criada para pode classificar os jogadores, essa classe recebe como parâmetro no seu construtor as entradas e as saídas, pesos das entradas, o valor do Bias e o *learning rate* (Taxa de aprendizagem), os dois últimos parâmetros não são obrigatórios, caso não seja escolhido valores para eles o algoritmo colocar o valor um para os dois.

Quando os valores necessários para realizar os treinamentos da classe Perceptron forem inseridos chamamos o método treino, nesse método e realizado todos os cálculos necessários para o treinamento do modelo do perceptron. No começo do treinamento temos a soma dos produtos dos pesos com as entradas, após esses cálculos pegamos o valor e jogamos na função de ativação, a função usada foi a step, caso o valor dos cálculos seja maior que zero ela retorna um, caso seja contrario ela retorna zero, após obter o valor da função de ativação, verificamos se o modelo acertou ou se ele errou, caso ele acerte a execução vai para o próximo passo, mas caso ele erre os valores dos pesos são ajustados e esse ajuste e feito com o valor do erro (valor da saída esperada menor valor errado), o valor do erro e usado para ajustar os pesos sinápticos, após o ajuste o algoritmo continua para testar o próximo caso, e faz esse mesmo processo até que o modelo aprenda todas as ocorrências.

Para a os pesos sinápticos $W_1 = 0.9$ e $W_2 = 0.7$ e taxa de aprendizagem 0.8, temos os valores dos pesos de cada época.

```
EPOCA: 0
PESO 1: 0.9
PESO 2: -0.30000000000000004
EPOCA: 1
PESO 1: 1.9
PESO 2: -0.30000000000000004
EPOCA: 2
PESO 1: 1.9
PESO 2: -0.30000000000000004
```