# **Universidade Positivo**

Curso: Ciência da Computação

Relatório Técnico

Título: Análise de Previsão de Votos em Filmes e Séries através de Rede Neural

Aluno: Anthonny Leonardo Amarante de Camargo, Gabriell de Souza Zappelini

Disciplina: Sistemas Inteligentes

Professor: Malgarete Rodrigues da Costa

Data: 23/11/2023

#### Sumário

Introdução

- Metodologia
- Desenvolvimento
- Resultados e Discussão
- DataSet Utilizado

## 1. Introdução

O presente relatório tem como objetivo explorar a utilização de redes neurais para previsão de votos em filmes e séries. O conjunto de dados utilizado contém informações sobre ano de lançamento, duração e pontuação no IMDB, buscando relacionar esses dados com a quantidade de votos recebidos.

## 2. Metodologia

### 2.1. Pré-processamento dos Dados

Os dados foram carregados a partir do arquivo 'imdb\_movies\_shows.csv' e passaram por um processo de limpeza, onde foram removidas as linhas com valores ausentes nos campos relevantes: 'release\_year', 'runtime', 'imdb\_score' e 'imdb\_votes'. Em seguida, normalizamos os dados para facilitar o treinamento da rede neural, garantindo média zero e desvio padrão um para cada atributo.

#### 2.2. Implementação da Rede Neural

A arquitetura da rede neural implementada consiste em uma camada de entrada com 3 atributos, uma camada escondida com 8 neurônios e uma camada de saída, mas alteramos algumas vezes para teste. Para a função de ativação, utilizamos a função sigmoide para as camadas intermediárias e tangente hiperbólica para a camada de saída.

#### 2.3. Treinamento e Teste da Rede Neural

O treinamento da rede foi realizado em 50 épocas de padrão, mas alterados durante os testes. Durante o treinamento, os pesos foram ajustados utilizando o algoritmo de retro propagação de erro.

#### 3. Desenvolvimento

O codigo utilizado para o desenvolvimento é o "main.py"

#### 4. Resultados e Discussão

Após o treinamento, foram obtidos os resultados e métricas de desempenho do modelo. A análise dos resultados inclui a observação do erro médio ao longo das épocas e a avaliação do erro nos dados de teste.

Durante a fase de testes, a função de ativação escolhida para análise foi a função SIGMOIDE, a qual descreve a relação entre a entrada e a saída de uma unidade em uma rede neural. A escolha dessa função de ativação é fundamental para compreender o comportamento e a eficácia da rede neural em questão.

#### Teste 1:

Porção de dados = 3 numEpocas = 50 eta = 0.005

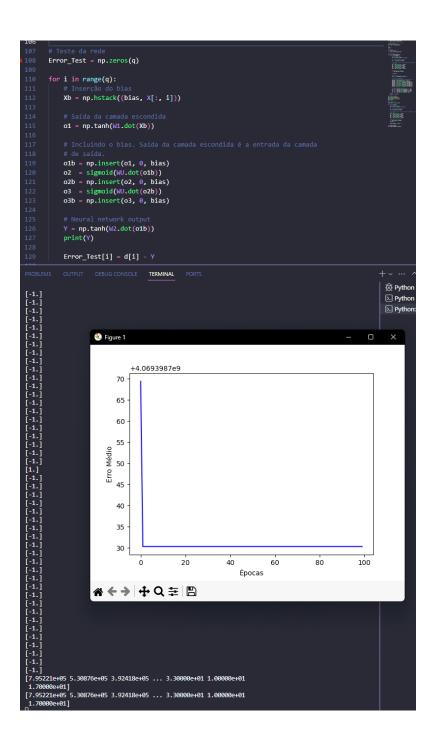
```
# Teste da rede
Error_Test = np.zeros(q)
          for i in range(q):
                # Saida da camada escondida
o1 = np.tanh(W1.dot(Xb))
                # de saída.
olb = np.insert(o1, 0, bias)
o2 = sigmoid(WU.dot(o1b))
o2b = np.insert(o2, 0, bias)
o3 = sigmoid(WU.dot(o2b))
o3b = np.insert(o3, 0, bias)
                # Neural network output
Y = np.tanh(W2.dot(o1b))
print(Y)
print(Error_Test)
print(np.round(Error_Test))
  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Figure 1
                   +4.06939873e9
            5
        Erro
            2
             0 1
                                                             20
Épocas
[-1.]
[7.95221e+05 5.30876e+05 3.92418e+05 ... 3.30000e+01 1.00000e+01
1.70000e+01]
[7.95221e+05 5.30876e+05 3.92418e+05 ... 3.30000e+01 1.00000e+01
1.70000e+01]
```

# Teste 2:

Porção de dados = 3

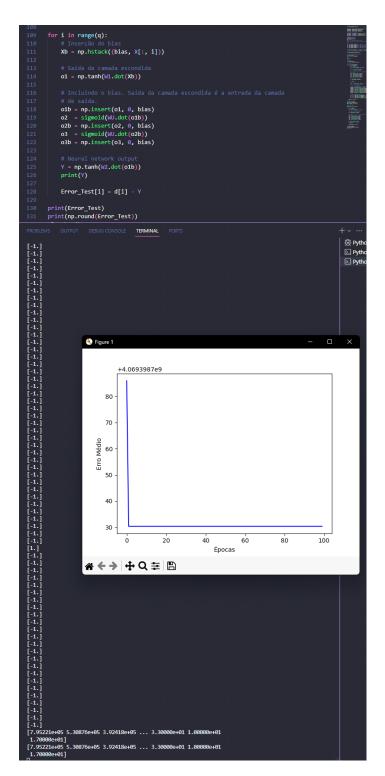
numEpocas = 100

eta = 0.005



## Teste 3:

```
Porção de dados = 3
numEpocas = <u>100</u>
eta = 0.8
```



# **Teste 4:**

Porção de dados = 3

numEpocas = 10

```
Error_Test = np.zeros(q)
              # Saída da camada escondio
o1 = np.tanh(W1.dot(Xb))
              # Incluindo o bias. Saída da camada escondida é a entrada da camada
# de saída.
              # de saída.
o1b = np.insert(o1, 0, bias)
o2 = sigmoid(WU.dot(o1b))
o2b = np.insert(o2, 0, bias)
o3 = sigmoid(WU.dot(o2b))
o3b = np.insert(o3, 0, bias)
              # Neural network output
Y = np.tanh(W2.dot(o1b))
print(Y)
         print(Error_Test)
print(np.round(Error_Test))
               OUTPUT DEBUG CONSOLE TERMINAL PORTS
               Figure 1
                                +4.06939873e9
                                                                          Épocas
               ☆ ♦ ♦ | 4 Q ≨ | 🖺
             +05 5.30876e+05 3.92418e+05 ... 3.30000e+01 1.00000e+01
[7.95221e+05 5.30876e+05 3.92418e+05 ... 3.3000e+01 1.00000e+01 1.70000e+01]
```

## 5. DataSet Utilizado

• https://www.kaggle.com/datasets/maso0dahmed/netflix-movies-and-shows/data