



CENTRO UNIVERSITÁRIO INTERNACIONAL UNINTER
ESCOLA SUPERIOR POLITÉCNICA
GRADUAÇÃO EM CIENCIAS DE DADOS
DISCIPLINA DE INTELIGÊNCIA ARTIFICIAL APLICADA

Atividades prática

Treinamento do Neurônio de Rosenblatt

1 - Objetivo

Este trabalho descreve o processo e resultados alcançados durante a construção de uma versão do neurônio de Rosenblatt, como parte da avaliação de aprendizado da disciplina “Inteligência artificial aplicada”.

2 - Detalhes

O treinamento do neurônio de Rosenblatt será realizado de forma a conseguir que um RU, em específico o de número 3303411, produza uma saída de valor 1 no neurônio, assim como todos valores maiores que o citado RU. De forma análoga todos os valores de RU inferiores a 3303411 devem produzir uma saída igual a -1.

Para isso será realizado um treinamento supervisionado para calibração dos pesos em um neurônio artificial, que será desenvolvido em planilha eletrônica, de forma que os pesos sejam atualizados até atingirem um certo nível de classificação sobre a massa de treinamento. A expectativa é que o nível de classificação, seja considerado satisfatório para o trabalho, quanto o neurônio for capaz de dividir os dados em 2 conjuntos tendo como ponto de divisão o RU alvo. Estes valores serão detalhados nas sessões seguintes.

Uma massa de teste, contendo RUs diferentes dos de treinamento será confrontada contra o neurônio treinado, e gerará uma avaliação quanto a capacidade de conhecimento do neurônio treinado quanto ao padrão existente nos dados.

2.1 - Planilha eletrônica

Como ferramenta de planilha eletrônica foi utilizado o google sheets que funciona em nuvem e pode ser acessado de forma online. A ferramenta possui funções compatíveis com o excel que foi a ferramenta inicialmente indicada, a título de sugestão, para o trabalho. Desenvolveu-se para este trabalho duas versões de planilha eletrônica tanto para o treinamento quanto para o teste sendo denominadas Neuronio-Rosenblatt-V1 (V1) e Neuronio-Rosenblatt-V2 (V2), com urls de acesso a seguir, respectivamente:

Uma versão exportada será disponibilizada com este trabalho e a versão online estará disponível para acesso e conferência durante todo o período da disciplina.

2.2 - Neuronio-Rosenblatt-V1

A versão Neuronio-Rosenblatt-V1 foi desenvolvida usando como características para alimentação do neurônio os dígitos formadores do número RU, dessa forma o RU 3303411 gera um vetor de entrada na seguinte forma : [3,3,0,3,4,1,1]. Esse neurônio por consequência tem 7 pesos associados além de um bias que por conta de pesquisas ficou sendo atualizado com a política usada para atualização dos pesos. O processo de atualização dos pesos foi desenvolvido através de uma Macro, que pegava o cálculo dos novos pesos calculados para cada entrada e retroalimentava os pesos atuais do neurônio. A figura1e figura2 mostra a aparência final da planilha Neuronio-Rosenblatt-V1, que possui duas abas sendo uma para treinamento e outra para Testes.

Na aba de treinamento foram inseridas 250 entradas escolhidas aleatoriamente tendo como parâmetro de escolha aleatória o número do RU específico de acordo com a fórmula:
 $f(x) = RU + \text{AleatorioEntre}(-1000, 1000).$

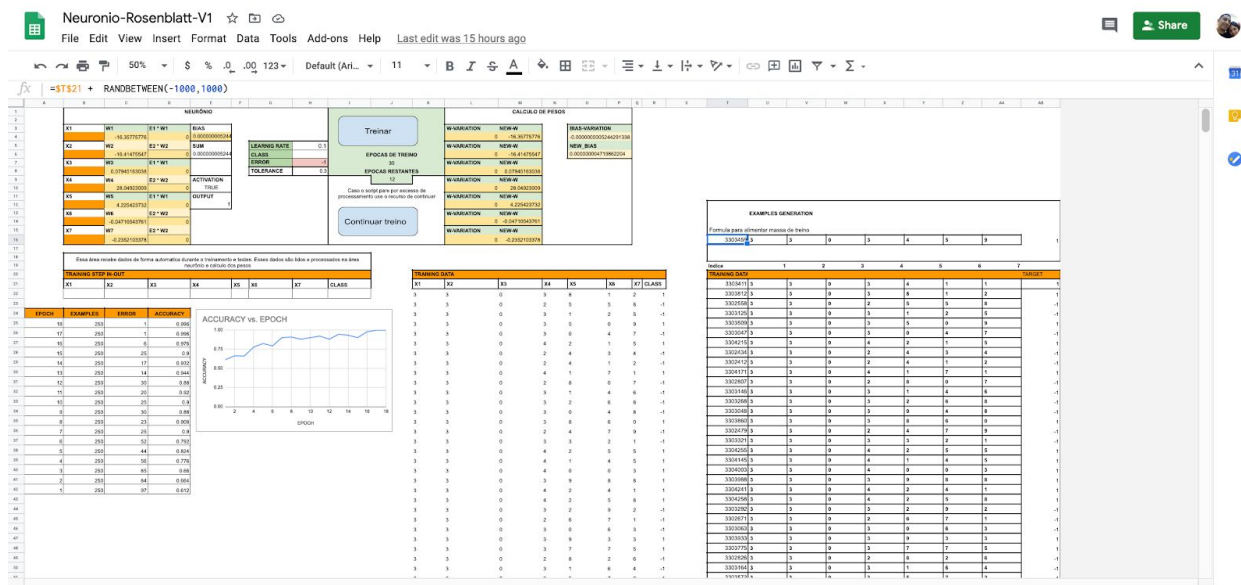


Figura1 - planilha Neuronio-Rosenblatt-V1 aba de treino

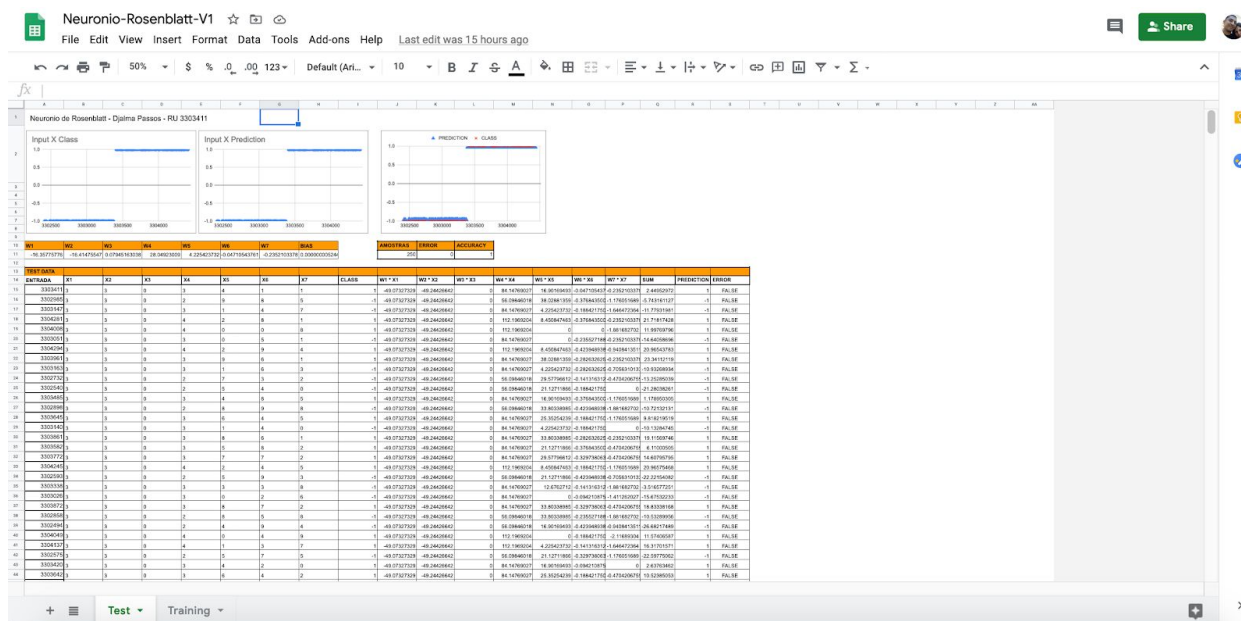


Figura2 - planilha Neuronio-Rosenblatt-V1 aba de teste

A região exibida na figura3 mostra em detalhes as entradas do neurônio (x1 até x7), pesos (W), além de calculos de pesos veses entradas, ativação, saída, taxa de aprendizado, saída esperada (class), erro.

Na área referente aos calculos dos pesos figuram a variação de cada peso(w-variation), novo peso calculado (new-w), variação do bias e novo bias. Esse processo é atualizado através de uma macro que inicia no botão “treinar” e por conta de tempo maximo de execução no google

sheets pode ser executada em novas épocas no botão “continuar treino”. O processo da macro pode ser conferido através do menu tools -> macros -> manage macros ou tools -> script editor.

| NEURÔNIO | | | | | CALCULO DE PESOS | | | | |
|----------|----|---------|------------|---------------|--|-------------|----------------|-----------------------|----------------------|
| X1 | W1 | E1 * W1 | BIAS | 0.00000005244 | <div>Treinar</div> <div>EPOCAS DE TREINO 30</div> <div>EPOCAS RESTANTES 12</div> <div>Caso o script pare por excesso de processamento use o recurso de continuar</div> <div>Continuar treino</div> | W-VARIATION | NEW-W | BIAS-VARIATION | |
| X2 | W2 | E2 * W2 | SUM | 0.00000005244 | | 0 | -16.35775778 | -0.000000005244291338 | |
| X3 | W3 | E1 * W1 | | | | W-VARIATION | NEW-W | NEW_BIAS | 0.000000004719862204 |
| X4 | W4 | E2 * W2 | ACTIVATION | TRUE | | 0 | -16.41475547 | | |
| X5 | W5 | E1 * W1 | OUTPUT | 1 | | W-VARIATION | NEW-W | | |
| X6 | W6 | E2 * W2 | | | | 0 | 0.07945163038 | | |
| X7 | W7 | E2 * W2 | | | | W-VARIATION | NEW-W | | |
| | | | | | | 0 | 28.04923009 | | |
| | | | | | | W-VARIATION | NEW-W | | |
| | | | | | | 0 | 4.225423732 | | |
| | | | | | | W-VARIATION | NEW-W | | |
| | | | | | | 0 | -0.04710543761 | | |
| | | | | | | W-VARIATION | NEW-W | | |
| | | | | | | 0 | -0.2352103378 | | |

Figura3 - Neuronio-Rosenblatt-V1 detalhe neurônio

Cada epoca de treino usou como entrada 250 amostras que foram escolhidas inicialmente através de um processo aleatório com intervalo de -1000 até +1000 ao redor do RU alvo.

O treino da rede se perpetuou até que a taxa de acerto parasse de evoluir, e isso aconteceu na casa dos 99,6% de acurácia na epoca de número 18. A figura4 mostra a evolução da acurácia do neurônio durante o treinamento. Os pesos finais da rede na rodada final de treinamento são indicados abaixo na tabela1.

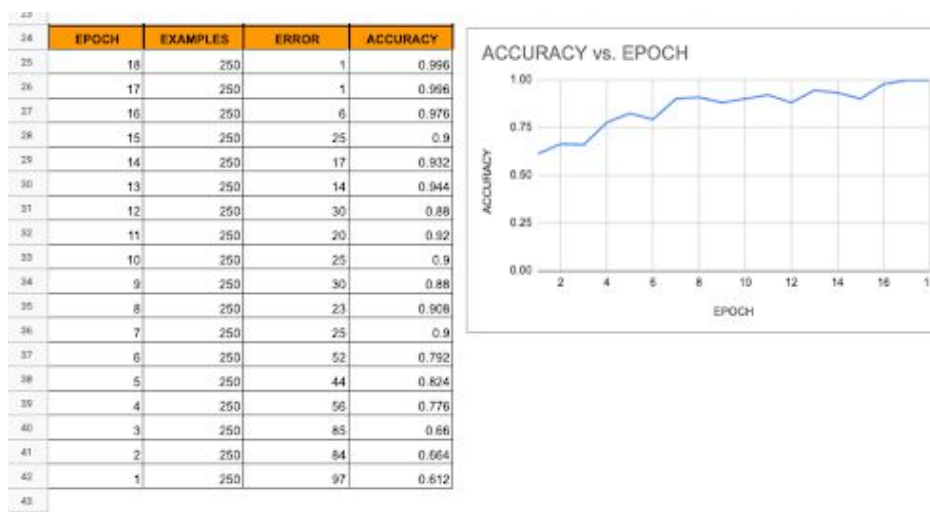


Figura4 - Neuronio-Rosenblatt-V1 evolução acurácia no treino.

| W1 | W2 | W3 | W4 | W5 | W6 | W7 | BIAS |
|------------------|------------------|-------------------|-----------------|-----------------|--------------------|-------------------|------------------------------|
| -16.357757 76 | -16.414755 47 | 0.07945163 038 | 28.0492300 9 | 4.22542373 2 | -0.0471054 3761 | -0.2352103 378 | 0.00000000 524429133 8 |

Tabela1 - Neuronio-Rosenblatt-V1 pesos após treino.

Após o treino foram gerados 250 novos dados seguindo a mesma política usada para o treinamento. Esses dados foram confrontados contra linhas da planilha de teste onde cada linha simula o neurônio treinado na aba de treino, e cada linha da planilha de testes é alimentada com os pesos finais da tabela1. A tabela2 representa uma linha da aba de treinamento onde são alimentados os pesos para avaliação de uma trada específica quanto a ser da classe do RU (prediction = 1) ou de outra classe (prediction = -1).

| TES T DAT A | | | | | | | | | | | | | | | | | | |
|----------------------|----|----|----|----|----|----|----|-----------|--------------------------|--------------------------|------------|-------------------------|-------------------------|--------------------------------|---------------------------|------------------------|------------------------|---------------|
| ENT RA DA | X1 | X2 | X3 | X4 | X5 | X6 | X7 | CLA SS | W1 * X1 | W2 * X2 | W3 * X3 | W4 * X4 | W5 * X5 | W6 * X6 | W7 * X7 | SU M | PR EDI CTI ON | ER RO R |
| 330 341 1 | 3 | 3 | 0 | 3 | 4 | 1 | 1 | 1 | -49. 073 273 29 | -49. 244 266 42 | | 84.1 476 902 7 | 16.9 016 949 3 | -0.0 471 054 376 1 | -0.2 352 103 378 | 2.4 495 297 2 | | FAL SE |

Tabela2 - Neuronio-Rosenblatt-V1 pesos após treino.

O resultado do teste foi sobre as 250 amostras foi de 250 predições corretas e 0 (zero) predições erradas. A tabela3 exibe o resultado do treinamento. A figura5 exibe os gráficos que resumem o comportamento do treinamento. Podemos ver que as entradas tiveram o comportamento de degrau no limiar do RU especificado. O mesmo comportamento se mostrou nas predições e podemos ver a sobreposição de ambos no gráfico prediction X class.

| AMOSTRAS | ERROR | ACCURACY |
|----------|-------|----------|
| 250 | 0 | 1 |

Tabela3 - Neuronio-Rosenblatt-V1 acurácia de teste.

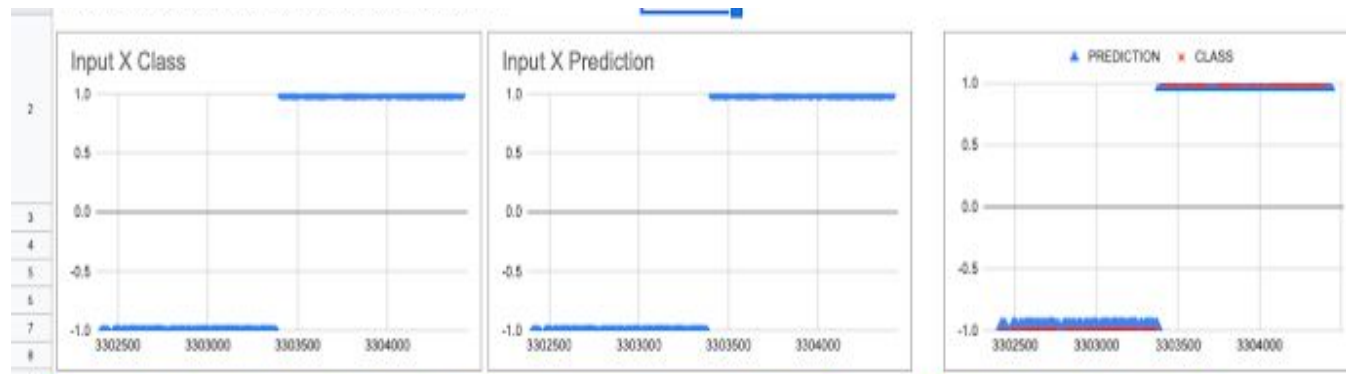


Figura5 - Neuronio-Rosenblatt-V1 gráficos de teste.

2.3 - Neuronio-Rosenblatt-V2

A versão Neuronio-Rosenblatt-V2 também usou treinamento supervisionado para treinamento dos pesos, porém a arquitetura dos neurônio teve alguma variações que abordaremos a seguir.

- A entrada para cada dentrito do neurônio foi o próprio valor do RU
- O Número de entradas necessárias para o treinamento foi 2.
- Diferentemente da V1 o bias desse neurônio ficou fixo com valor -1.

O processo de treinamento foi simplificado necessitando de menos programação e substituiu-se as macros por atualizações de formulas no excel. Da mesma forma que na V1 o treinamento durou até que o neurônio não sofresse mais atualizações de pesos por novas entradas.

Tanto o treinamento quanto o processo de testes receberam como entradas 1012 entradas de um intervalo de -2999999 até 2999999 em relação ao RU alvo. Sendo os dados de treino diferentes dos dados de teste. A quantidade de rodadas para o treinamento foi de aproximadamente 8.

Na tabela4 temos os valores pesos para o neurônio após todas as rodadas de treinamento. Na figura6 temos como resultado os gráficos que relacionam entrada, saída esperada e valores préditos. Na tabela5 temos um exemplo de dado de treinamento no

neurônio da V2. A acurácia de treino e de testes teve o mesmo valor final de acordo com a tabela6.

| W1 | W2 |
|----------------------|-------------------|
| 0.000000004518174196 | 0.000000298199497 |

Tabela4 - Neurônio-Rosenblatt-V2 pesos finais do neurônio com 2 entradas.

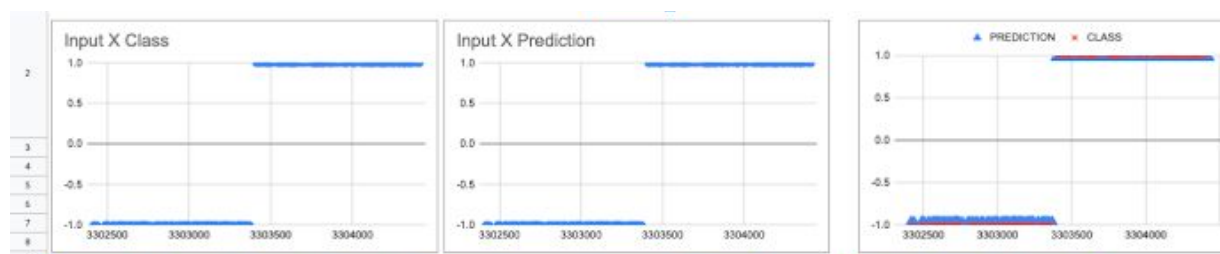


Figura6 - Neurônio-Rosenblatt-V2 gráficos de teste.

| ENTRA DA | CLA SS | W1 | W2 | W1 * X1 | W2 * X2 | SUM | PREDICT ION | ERROR |
|-------------|-----------|--------------------------|-----------------------|-------------------|-----------------|------------------|----------------|-------|
| 60233 73 | 1 | 0.000000004518 174196 | 0.0000002981 99497 | 0.027214 64846 | 1.79616 6799 | 0.823381 4471 | 1 | 0 |

Tabela5 - Neurônio-Rosenblatt-V2 exemplo de dados de testes aplicado ao neurônio v2.

| PREDICTIONS | ERROR | ACCURACY |
|-------------|-------|----------|
| 1012 | 1012 | 1 |

Tabela6 - Neurônio-Rosenblatt-V2 acurácia de treino e teste.

3 - Conclusão

Neste trabalho foram apresentadas 2 implementações de neurônio de rosenblatt sendo uma construída com 8 entradas que recebiam, individualmente, como características, os dígitos formadores do RU. Já a segunda versão foi construída com 2 entradas que recebiam como característica o próprio valor do RU.

Ambas as soluções chegaram a um valor de divisão dos dados com acurácia na casa das unidades, porém implementar a segunda versão pediu menor esforço de pré-processamento, menos processamento durante o treinamento. Que resultou na possibilidade de entrar com

uma quantidade maior de amostras e alcançar o fim do treinamento em menor número de épocas.

Dessa forma conclui-se esse trabalho com 2 implementações que resolveram o problema proposto sendo a V2 mais rápida de implementar e mais performática para a classificação do RU 3303411 em duas classes específicas.