



**UNIVERSIDADE FEDERAL DO PARÁ
INSTITUTO DE TECNOLOGIA - ITEC
FACULDADE DE ENGENHARIA ELÉTRICA E BIOMÉDICA
CURSO DE ENGENHARIA ELÉTRICA**

RODRIGO GOMES DUTRA 201607140080

**RELATÓRIO DE REDES NEURAIS APLICADAS EM CLASSIFICAÇÃO DE
CANCER DE MAMA**

**Belém
2019**

SUMÁRIO

1	INTRODUÇÃO	2
1.1	BANCO DE DADOS	2
2	DESENVOLVIMENTO	4
2.1	RESULTADOS DA MELHOR TOPOLOGIA	4
3	CONCLUSÃO	6
	REFERÊNCIAS	7

1 INTRODUÇÃO

Este relatório é relativo à atividade proposta em sala de aula na disciplina de Inteligência computacional, essa atividade consiste em aplicar uma rede neural para a resolução de um problema de classificação. A rede neural será aplicada em um banco de dados de mamografias, para classificar um caso positivo de câncer ou não.

O banco de dados consiste em dados de mamografia retirado de um repositório do site UCI, o qual será detalhado mais adiante nesse documento.

O software que será utilizado para os fins de gerar a estrutura da rede neural será o Python3(ROSSUM, 1995), com auxílio das bibliotecas Tensorflow(ABADI et al., 2016) e Scikit(PEDREGOSA et al., 2011), as quais foram utilizadas respectivamente para gerar a estrutura da rede neural e manipular a base de dados.

1.1 BANCO DE DADOS

O banco de dados utilizado foi retirado do *UCImachinelearningrepository*(ASUNCION; NEWMAN, 2007), os dados pertencentes ao banco foram retirados de mamografias digitais, contendo casos de câncer malignos e benignos nessas regiões. Os dados consiste em uma tabela com dados de, Bi-RADS, Idade, Forma, Margem, Densidade, Severidade.

As entradas do banco de dados representam respectivamente:

- BI-RADS é um acrônimo para *Breast Imaging – Reporting and Data System*, representado uma avaliação ordinal de 1 a 5.
- Idade do paciente em anos (inteiro)
- Forma da massa, podendo ser redonda = 1, oval = 2, lobular = 3, irregular = 4. (nominal)
- Margem: margem de massa: circunscrita = 1, microlobulada = 2, obscurecido = 3, mal definido = 4, espiculado = 5 (nominal)
- Densidade: densidade de massa: alta = 1, iso = 2, baixa = 3, contendo gordura = 4 (ordinal)
- Severidade: benigna = 0 ou maligna = 1 (binominal, resultado)

Dessa forma, o banco de dados tem 5 entradas, representado as variáveis de Bi-RADS, Idade, Forma, Margem, Densidade. A saída do problema de classificação é a severidade do tumor, dessa forma podendo ser aplicada para a predição de do tipo de tumor do paciente.

A divisão da base de dados em dados de treino, validação e teste foi realizada com a biblioteca Scikit(PEDREGOSA et al., 2011), a qual randomiza a seleção de índice para a divisão

do banco de dados em dada proporção previamente programada. A proporção utilizada foi de 60% para treino 20% para validação e 20% para teste.

Dessa forma resultando na distribuição mostrada na Tabela 1 :

Tabela 1 – Distribuição de dados

Lote de dados	Porcentagem	Classe 1	Classe 2	Total
Treino	60%	301	283	584
Validação	20%	67	59	126
Teste	20%	62	62	124
Total	100%	430	404	834

A distribuição de dados foi realizada de forma randômica com o pacote Scikit(PEDREGOSA et al., 2011), além disso os dados em sua totalidade foram normalizados por coluna, dessa forma todo o banco de dados tem entradas e saídas variando de 0 a 1.

2 DESENVOLVIMENTO

Para realizar a análise e classificação do banco de dados de vinhos utilizando a rede neural foi criado um código utilizando o Python3 (ROSSUM, 1995).

Foi utilizado o algoritmo de Adam para o treinamento da rede neural, com uma taxa de aprendizagem de, 0.001 e decaimento por época de 0.0005. Utilizando-se dos dados de treinamento definidos na Tabela 1. Para a função de erro, depois de realizado-se a análise da melhor função baseado-se no desempenho, a acurácia do modelo.

A fim de estudar o comportamento da rede neural conforme o número de neurônios na camada escondida, foi necessário mais que uma tentativa para cada quantidade de neurônios, para assim tentar tirar influência do caráter randômico no erro inerente em relação a saída da rede neural e a saída real do banco de dados para as mesmas entradas.

Nesse sentido foram realizadas 2 repetições de criação da rede neural variando-se o número de neurônios, utilizando-se dos valores: 1;5;10;15;30 de neurônios, de forma que em cada repetição era salvo os erros de teste relativos a atual iteração. Dessa forma, foi possível gerar uma tabela com os valores de acurácia coletados.

Tabela 2 – Acurácia vs número de neurônios

Nº de neurônios	repetição 1	repetição 2	Média	Acurácia repetição 1	Acurácia repetição 2	Acurácia média
5	102	100	101.0	80.95238095238095	79.36507936507937	80.158730
10	100	104	102.0	79.36507936507937	82.53968253968253	80.9523809
15	103	103	103.0	81.74603174603175	81.74603174603175	81.7460317
30	103	105	104.0	81.74603174603175	83.33333333333333	82.5396825

2.1 RESULTADOS DA MELHOR TOPOLOGIA

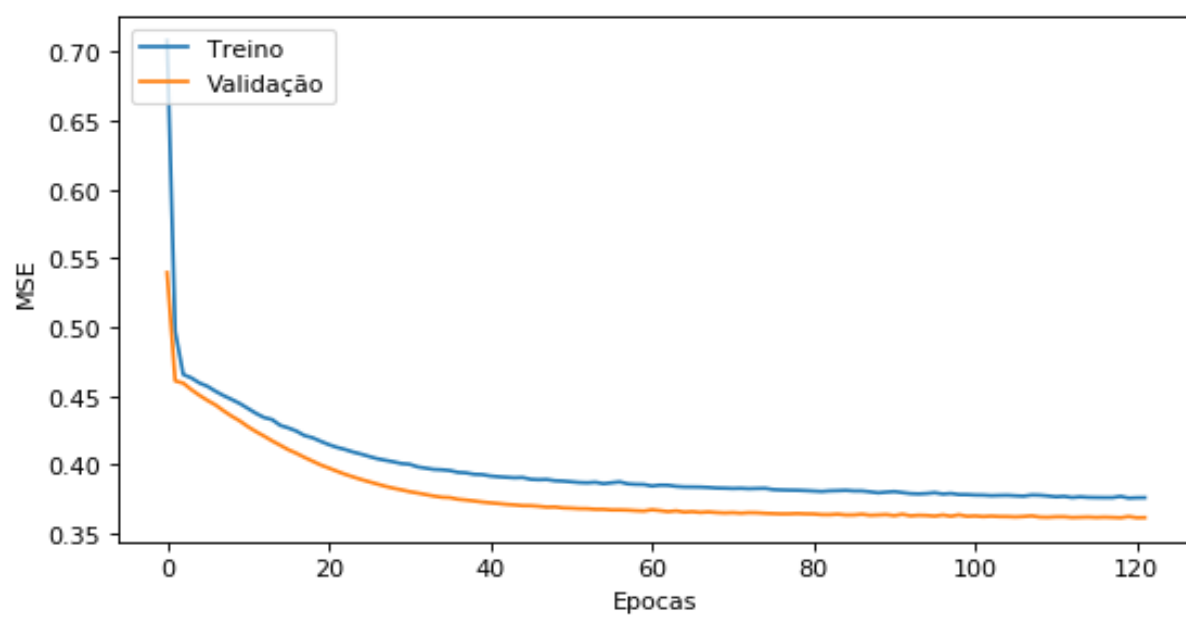
A melhor topologia seguindo os resultados da tabela 1 foi a topologia com 30 neurônios na camada escondida, o comportamento dessa topologia pode ser melhor observado através de seus resultados de treinamento e teste, visualizando-se a evolução do gráfico do erro e a matriz confusão da rede ,respectivamente.

A partir da matriz confusão é possível ver o quanto a rede neural acertou e errou para cada uma das 2 categorias de câncer, os severos e os não severos(malignos e benignos).

Tabela 3 – Matriz confusão

		Valores preditos	
		Severo	Não severo
Valores reais	Severo	55	7
	Não severo	12	50

Figura 1 – Gráfico do erro



3 CONCLUSÃO

A partir dos resultados das rede neural no problema de classificação proposto é possível concluir que avaliar o número de neurônios de uma rede neural é bastante importante para ampliar a acurácia da rede, de forma que foi possível aumentar a performance de acurácia média em quase 3%.

De posse dos resultados da melhor topologia pode-se concluir também que esta apresentou uma boa precisão, mas que ainda se distancia do ideal no qual seria próximo dos 99%, porém ainda assim uma topologia como as apresentadas nesse documento, poderiam ser aplicadas como um auxilio para detectar casos de câncer severos e assim ampliar a possibilidade de recuperação do paciente o quanto antes possível.

REFERÊNCIAS

ABADI, M. et al. Tensorflow: A system for large-scale machine learning. In: **12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)**. [S.l.: s.n.], 2016. p. 265–283.

ASUNCION, A.; NEWMAN, D. **UCI machine learning repository**. 2007.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in python. **Journal of machine learning research**, v. 12, n. Oct, p. 2825–2830, 2011.

ROSSUM, G. Python reference manual. CWI (Centre for Mathematics and Computer Science), 1995.