



RELATÓRIO DO TRABALHO PRÁTICO

Conhecimento e Raciocínio

CARLOS SANTOS - a2003035578@isec.pt

HUGO FERREIRA – a2020128305@isec.pt

2021-2022

RELATÓRIO DO TRABALHO PRÁTICO CONHECIMENTO E RACIOCÍNIO



ALUNO N.º 2003035578

Carlos Santos

ALUNO N.º 2020128305

Hugo Ferreira

ENTIDADE

Instituto Superior de Engenharia de Coimbra

Curso

Licenciatura em Engenharia Informática

PROFESSORES

Viriato António Pereira Marinho Marques

Anabela Borges Simões

COIMBRA – JUNHO – 2022



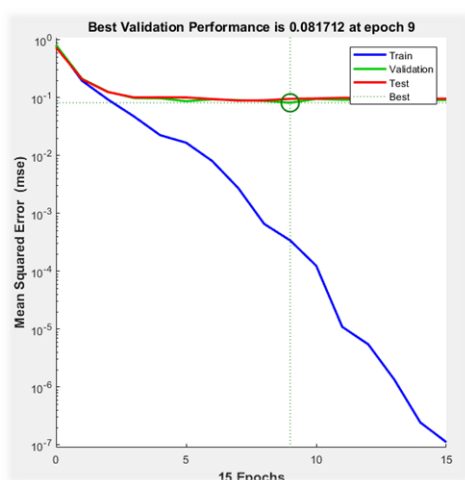
ÍNDICE

1 - INTRODUÇÃO	4
2 – Decisões tomadas	5
2.1 – Pré processamento das imagens:	5
2.2 – Targets	5
2.3 – Backup das imagens	5
2.4 – Dados guardados	5
3 – Testes fase b)	6
3.1 – Dimensão e camadas escondidas	6
4 – Testes fase c)	7
4.1 - 1ªFase:	7
4.2 - 2ªFase:	7
4.3 - 3ªFase:	7
5 – Aplicação	8
6 – Conclusões	11
7 – Bibliografia	12

1 - INTRODUÇÃO

Neste tema pretende-se que os estudantes aprofundem os seus conhecimentos sobre redes neuronais. O objetivo consiste na implementação e teste de diferentes arquiteturas de redes neuronais “feedforward” para classificar corretamente 6 formas geométricas (círculo, papagaio, paralelograma, quadrado, trapézio e triângulo). Pretendemos atingir a melhor precisão possível na identificação destas formas.

De forma a fazermos o estudo adequado para análise a cada lançamento com parâmetros diferentes geramos diversos outputs como estes:

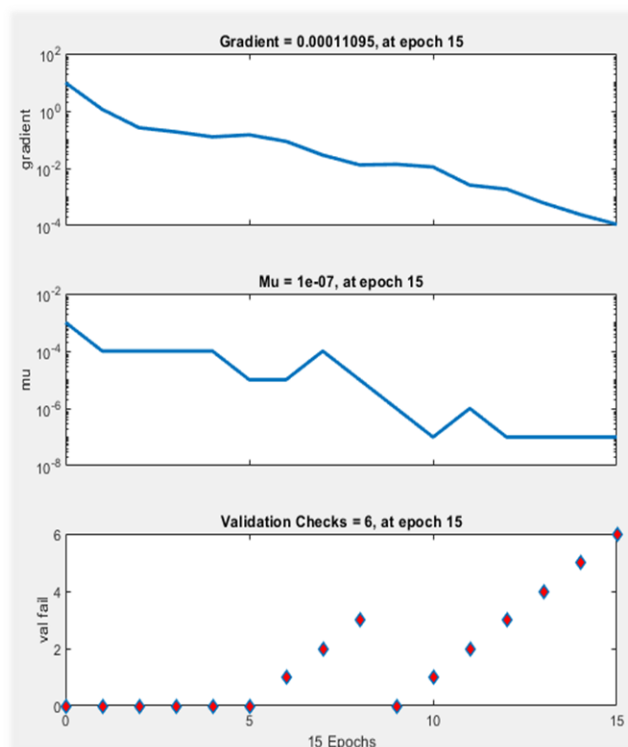
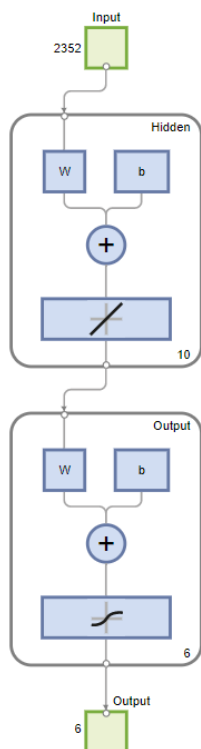


Confusion matrix

	1	2	3	4	5	6	
1	45 15.0%	2 0.7%	0 0.0%	0 0.0%	3 1.0%	1 0.3%	88.2% 11.8%
2	0 0.0%	45 15.0%	0 0.0%	0 0.0%	0 0.0%	3 1.0%	93.8% 6.2%
3	0 0.0%	0 0.0%	43 14.3%	0 0.0%	1 0.3%	1 0.3%	95.6% 4.4%
4	3 1.0%	0 0.0%	1 0.3%	50 16.7%	0 0.0%	2 0.7%	89.3% 10.7%
5	2 0.7%	1 0.3%	2 0.7%	0 0.0%	46 15.3%	0 0.0%	90.2% 9.8%
6	0 0.0%	2 0.7%	4 1.3%	0 0.0%	0 0.0%	43 14.3%	87.8% 12.2%
	90.0% 10.0%	90.0% 10.0%	86.0% 14.0%	100% 0.0%	92.0% 8.0%	86.0% 14.0%	90.7% 9.3%

Output Class

Target Class



2 – Decisões tomadas

2.1 – Pré processamento das imagens:

Para questões de otimização da aplicação optamos por redimensionar as imagens para uma resolução de 28x28. Passamos ainda as imagens para binário e fomos preenchendo a matriz dos inputs à medida que liamos as pastas pedidas.

2.2 – Targets

Para os targets decidimos fazer uma matriz com vetores com os números 1,2,3,4,5,6 que servem para identificação da figura geométrica em que o 1 é o círculo e assim adiante.

2.3 – Backup das imagens

À medida que íamos lendo as imagens, criámos um backup das mesmas em binário para garantir a segurança dos dados lidos (as mesmas encontram-se na pasta “Out/Binary”).

2.4 – Dados guardados

Em vez de apenas guardarmos a matriz de confusão para os testes feitos, decidimos guardar em pastas separadas para cada teste ainda o gráfico da “performance”, “trainstate”, as redes e a matriz de confusão.

3 – Testes fase b)

Durante a realização dos testes para a fase b retirámos várias conclusões as quais vamos falar em seguida.

3.1 – Dimensão e camadas escondidas

Como se pode ver no Excel, ao testar várias arquiteturas com diferente número de neurónios e camadas escondidas chegámos à conclusão de que os mesmos parâmetros alteram muito pouco a performance da rede.

3.2 – Função de treino

As funções de treino têm todas uma performance muito similar, no entanto observamos que a função “trainlm” e a “trainbfg” têm um tempo de execução muito longo, podendo mesmo chegar a várias horas.

No entanto, a única rede com 100% de performance foi a que usamos a função de treino “trainlm”.

Observámos que existem algoritmos muito bons e muitos maus, como não testámos todos, nos que usámos apenas se verificou uma diminuição significativa da performance na função de treino “traincgp”.

4 – Testes fase c)

4.1 - 1ªFase:

Sem treinar novamente a rede ao identificar as imagens da pasta “test”, a melhor rede neuronal da fase b) tem uma precisão de 73.3%.

Sem treinar novamente a rede ao identificar as imagens da pasta “train”, a melhor rede neuronal da fase b) tem uma precisão de 91.0%.

Sem treinar novamente a rede ao identificar as imagens da pasta “start”, a melhor rede neuronal da fase b) tem uma precisão de 70.0%.

4.2 - 2ªFase:

A melhor rede neuronal da fase b), treinada com a pasta “test”, tem uma precisão de 50.0% ao identificar as imagens da pasta “start”.

A melhor rede neuronal da fase b), treinada com a pasta “test”, tem uma precisão de 79.3% ao identificar as imagens da pasta “train”.

A melhor rede neuronal da fase b), treinada com a pasta “test”, tem uma precisão de 86.7% ao identificar as imagens da pasta test.

4.3 - 3ªFase:

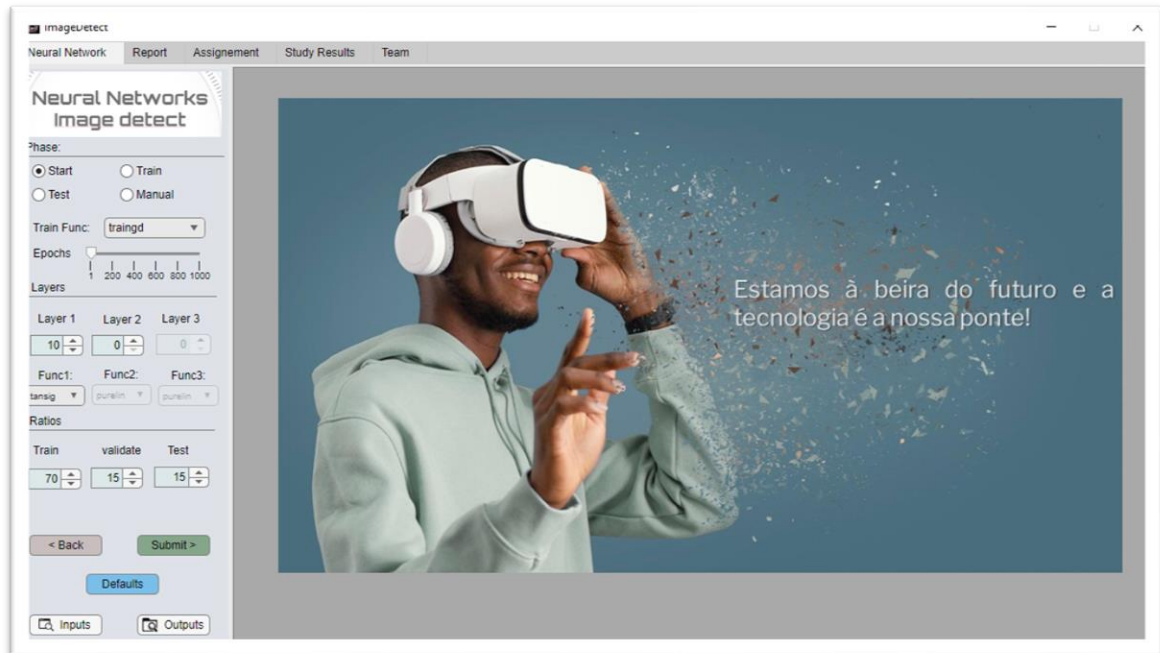
A melhor rede neuronal da fase b), treinadas com todas as pastas, tem uma precisão de 50.0% ao identificar as imagens da pasta “start”.

A melhor rede neuronal da fase b), treinadas com todas as pastas, tem uma precisão de 86.6% ao identificar as imagens da pasta “test”.

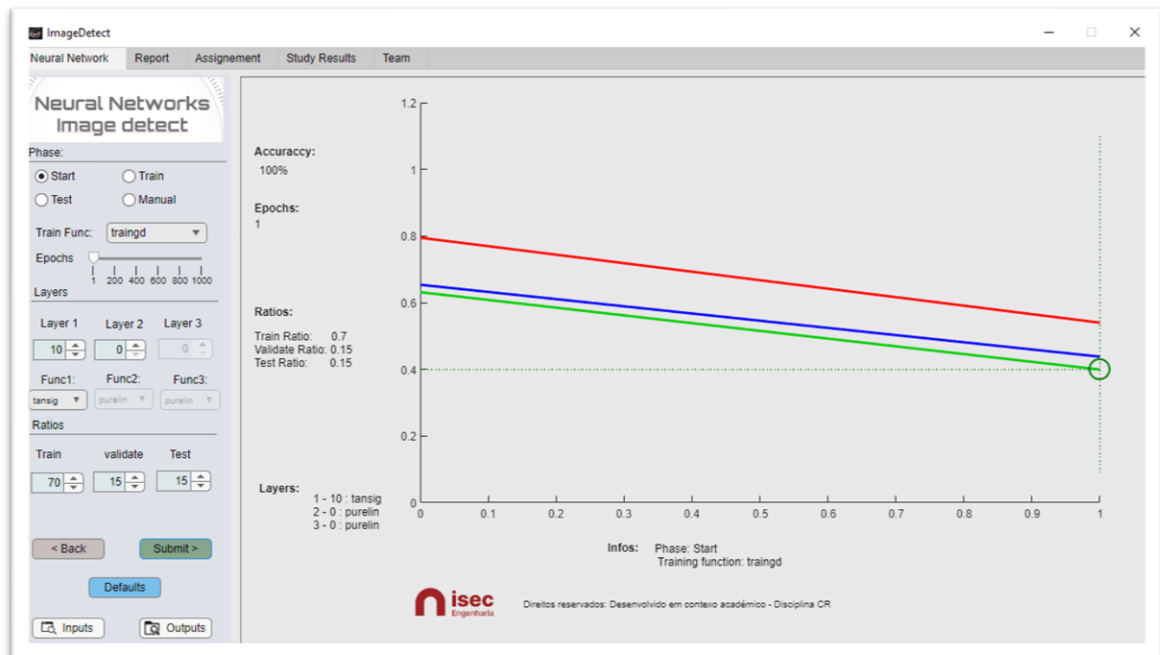
A melhor rede neuronal da fase b), treinadas com todas as pastas, tem uma precisão de 79.3% ao identificar as imagens da pasta “train”.

5 – Aplicação

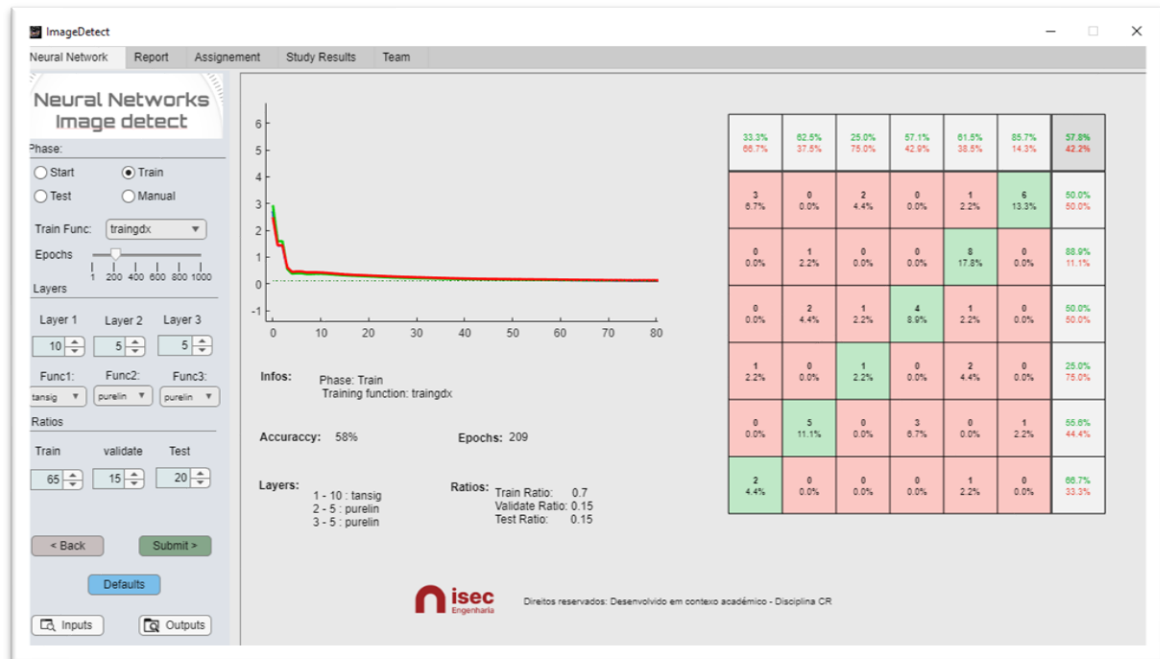
Início:



Lançamento “Phase Start”:



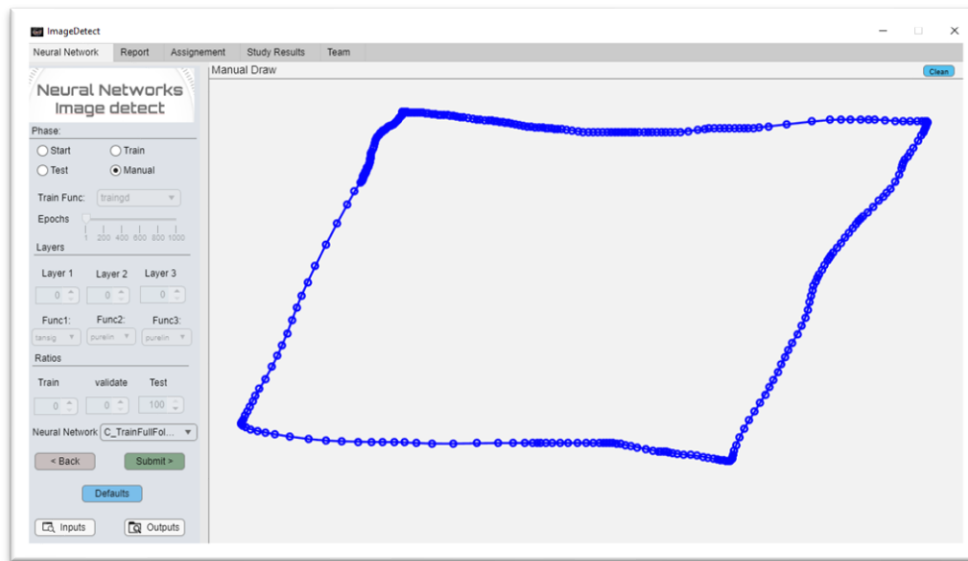
Lançamento “Phase Train”:



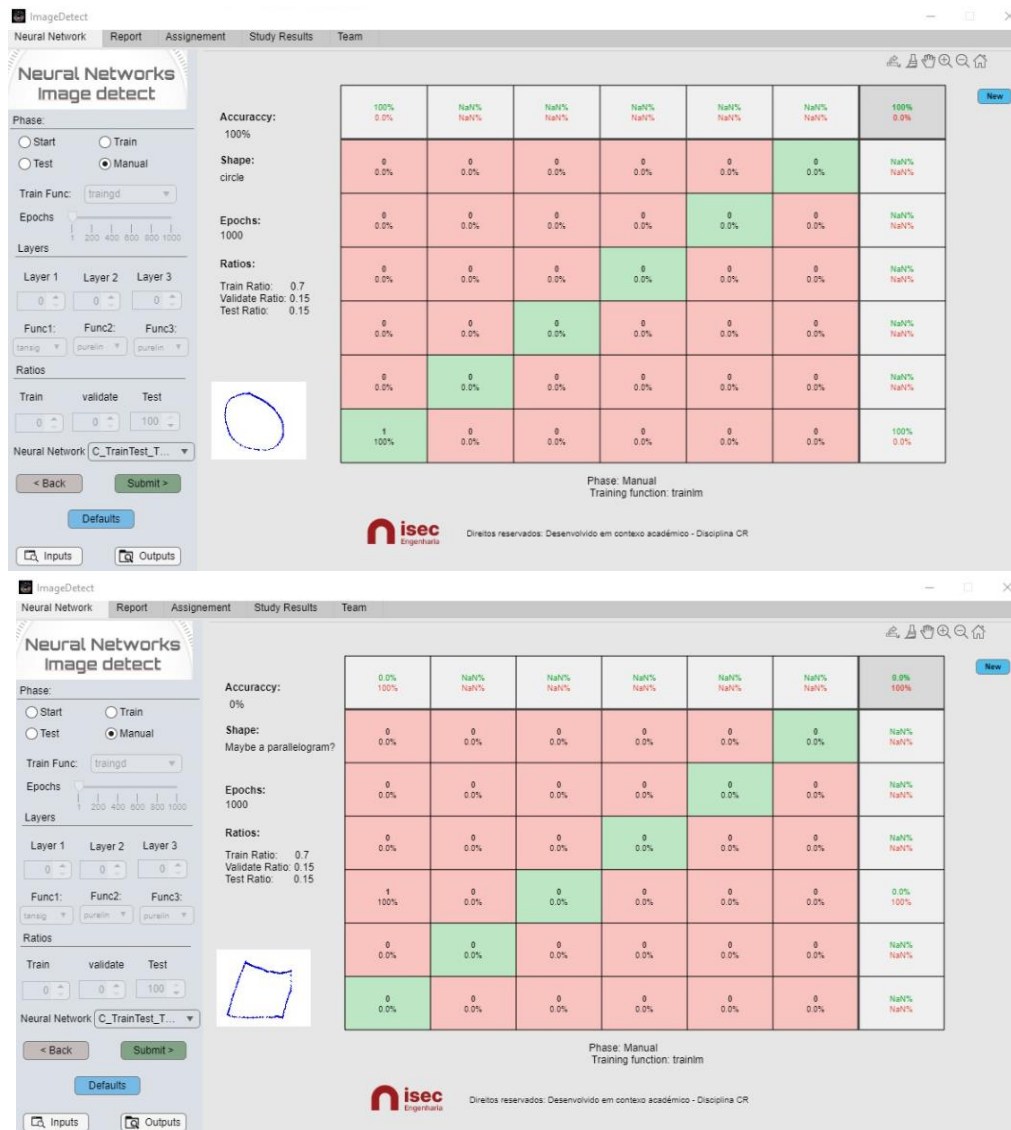
Lançamento “Phase Test”:



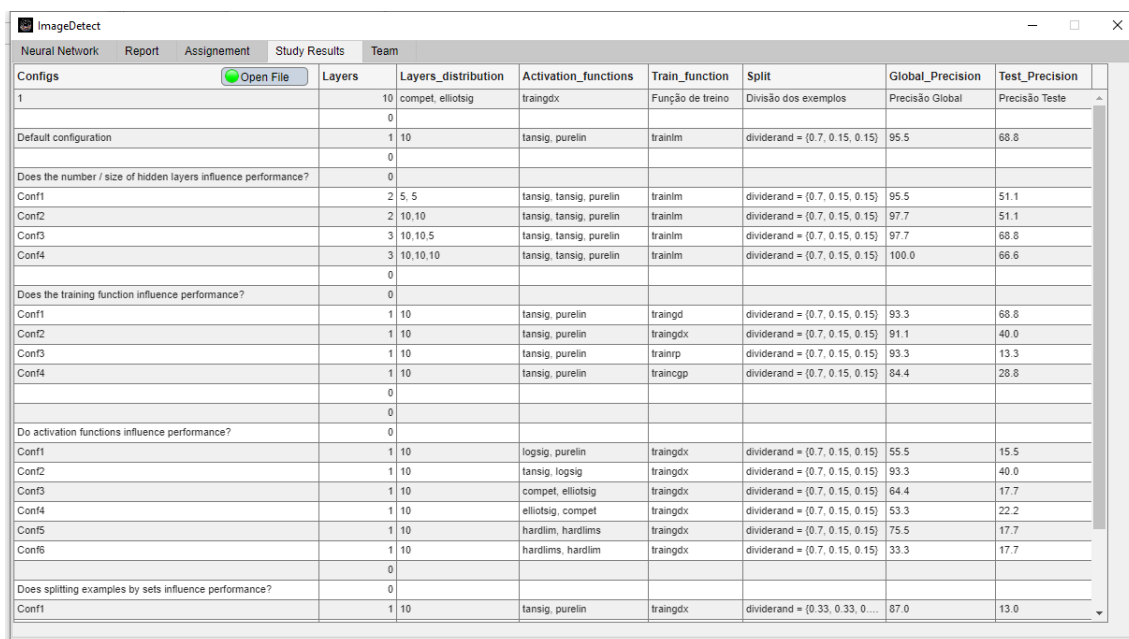
Desenho para “Phase Manual”:



Lançamento “Phase Manual”:



Resultados do estudo:



Configs	Layers	Layers_distribution	Activation_functions	Train_function	Split	Global_Precision	Test_Precision
1	10	compet, elliotstg	traingdx	Função de treino	Divisão dos exemplos	Precisão Global	Precisão Teste
Default configuration	1	10	tansig, purelin	trainlm	dividerand = (0.7, 0.15, 0.15)	95.5	68.8
Does the number / size of hidden layers influence performance?	0						
Conf1	2	5, 5	tansig, tansig, purelin	trainlm	dividerand = (0.7, 0.15, 0.15)	95.5	51.1
Conf2	2	10, 10	tansig, tansig, purelin	trainlm	dividerand = (0.7, 0.15, 0.15)	97.7	51.1
Conf3	3	10, 10, 5	tansig, tansig, purelin	trainlm	dividerand = (0.7, 0.15, 0.15)	97.7	68.8
Conf4	3	10, 10, 10	tansig, tansig, purelin	trainlm	dividerand = (0.7, 0.15, 0.15)	100.0	66.6
Does the training function influence performance?	0						
Conf1	1	10	tansig, purelin	traingd	dividerand = (0.7, 0.15, 0.15)	93.3	68.8
Conf2	1	10	tansig, purelin	traingdx	dividerand = (0.7, 0.15, 0.15)	91.1	40.0
Conf3	1	10	tansig, purelin	trainrp	dividerand = (0.7, 0.15, 0.15)	93.3	13.3
Conf4	1	10	tansig, purelin	traingcp	dividerand = (0.7, 0.15, 0.15)	84.4	28.8
Do activation functions influence performance?	0						
Conf1	1	10	logsig, purelin	traingdx	dividerand = (0.7, 0.15, 0.15)	55.5	15.5
Conf2	1	10	tansig, logsig	traingdx	dividerand = (0.7, 0.15, 0.15)	93.3	40.0
Conf3	1	10	compet, elliotstg	traingdx	dividerand = (0.7, 0.15, 0.15)	64.4	17.7
Conf4	1	10	elliotstg, compet	traingdx	dividerand = (0.7, 0.15, 0.15)	53.3	22.2
Conf5	1	10	hardlim, hardlims	traingdx	dividerand = (0.7, 0.15, 0.15)	75.5	17.7
Conf6	1	10	hardlims, hardlim	traingdx	dividerand = (0.7, 0.15, 0.15)	33.3	17.7
Does splitting examples by sets influence performance?	0						
Conf1	1	10	tansig, purelin	traingdx	dividerand = (0.33, 0.33, 0.33)	87.0	13.0

6 – Conclusões

Com este trabalho ficamos a conhecer melhor as redes neuronais “feedforwardnet”. Fizemos uma aplicação que cumpre todos os requisitos pedidos.

Através da nossa investigação percebemos a importância destas redes neuronais. As redes neuronais estão a transformar a forma como as pessoas e empresas interagem com os sistemas resolvendo problemas e tomam decisões melhores fazendo melhores previsões. Elas podem aprender e modelar relações entre entradas e saídas de dados que são não-lineares e complexos; realizar generalizações e inferências; revelar relacionamentos e padrões.

Como resultado, as redes neurais podem melhorar processos de decisão em diversas áreas, como por exemplo:

- Deteção de fraude;
- Otimização de logística para redes de transporte;
- Marketing direcionado;
- Previsões nos mercados financeiros de ações de mercado, moeda, opções, futuros, falência e classificação de títulos;

7 – Bibliografia

Como fontes utilizamos principalmente a Wikipédia, o moodle e os documentos da teórica:

- Moodle/Documentos Teórica

[Curso: 60023545 Conhecimento e Raciocínio LEI 2122 \(isec.pt\)](#)

- Investopedia - Neural Network

<https://www.investopedia.com/terms/n/neuralnetwork.asp#:~:text=A%20neural%20network%20is%20a,organic%20or%20artificial%20in%20nature.>

- SAS - Insights sobre Análise de Dados

https://www.sas.com/pt_br/insights/analytics/neural-networks.html

- Publisher Logo Review of Scientific Instruments - Neural networks and their applications

<https://aip.scitation.org/doi/abs/10.1063/1.1144830>

- cnn_wp.pdf (eet-china.com)

http://site.eet-china.com/webinar/pdf/Cadence_0425_webinar_WP.pdf

- Introduction to multi-layer feed-forward neural networks - ScienceDirect

<https://www.sciencedirect.com/science/article/abs/pii/S0169743997000610>

- msc-thesis.pdf (toronto.edu)

<http://www.cs.toronto.edu/~gkoch/files/msc-thesis.pdf>