Uma imagem com texto, ClipArt

Descrição gerada automaticamente

Licenciatura em Engenharia Informática

Conhecimento e Raciocínio

Trabalho Prático

*Neural Networks*

Prof.º Viriato António Pereira Marinho Marques

Prof.ª Anabela Borges Simões

Pedro Gonçalo dos Reis Correia

2018020558 – Turma P4

Índice

[Introdução 3](#_Toc106500668)

[Tratamento de imagens 4](#_Toc106500669)

[Estudo do treino de uma rede neuronal 5](#_Toc106500670)

[Estudo do treino de uma rede neuronal e o valor da precisão de teste 6](#_Toc106500671)

[Estudo das precisões de teste com diferentes conjuntos de exemplos 7](#_Toc106500672)

[Teste da pasta test sem re-treinar a rede 7](#_Toc106500673)

[Treino da rede com a pasta test e teste de cada pasta individualmente 8](#_Toc106500674)

[Simulação com formas geradas pelo utilizador 10](#_Toc106500675)

[Aplicação Gráfica 11](#_Toc106500676)

# Introdução

Este trabalho prático foi realizado no âmbito da Unidade Curricular de Conhecimento e Raciocínio da Licenciatura em Engenharia Informática do Instituto Superior de Engenharia de Coimbra, tem como objetivo o desenvolvimento de redes neuronais, de modo a poder efetuar um estudo estatístico de diferentes arquiteturas de rede. Este projeto foi desenvolvido em MATLAB e encontra-se dividido nas diversas alíneas do enunciado. Foi também desenvolvida uma aplicação gráfica, recorrendo à ferramenta GUIDE do MATLAB.

# Tratamento de imagens

Dado que o objetivo do projeto consiste em treinar redes neuronais para a identificação de diferentes formas geométricas, é necessário começar por transformar estas imagens em matrizes binárias que possam ser interpretadas pelo MATLAB.

Para este efeito, foi criada uma função *read\_images*. Nesta função está definido um mapa que liga os valores binários de *target output* para cada forma, com o seu nome, por exemplo, *circle* tem um *target output* de 100000.

São então percorridas todas as imagens de cada pasta do conjunto indicado, onde cada imagem é lida para uma variável. De modo a conseguir uma melhor eficiência e maior velocidade de treino, estas imagens são redimensionadas para imagens de 25x25. Depois deste redimensionamento, as imagens passam por um tratamento de cor para ficarem a preto e branco. Assim, garantimos que a matriz de inputs gerada de seguida é apenas composta por bits 0 e 1.

Esta matriz gerada é transformada num vetor, de modo a gerar uma coluna na matriz de inputs. A esta matriz de inputs é correspondente o seu target associado.

# Estudo do treino de uma rede neuronal

Para a realização desta tarefa foram utilizados todos os exemplos da pasta *start* para treinar a rede neuronal. Utilizando diferentes tipos de configuração, verificamos que como são utilizados todos os exemplos no treino, a precisão do treino é sempre muito perto de 100%.

Mesmo assim, esta configuração de rede pode não ser a melhor, visto que os dados não são em quantidade suficiente para controlar a qualidade da saída.

# Estudo do treino de uma rede neuronal e o valor da precisão de teste

No ficheiro Excel em anexo, estão detalhadas as diferentes configurações de redes neuronais usadas para estudar o impacto de cada variável no treino de uma rede neuronal.

Assim, verificamos que a melhor rede neuronal tem a seguinte arquitetura:

* Número de *hidden layers*: 1
* Número de neurónios: 10
* Função de treino: trainbr
* Funções de ativação: tansig, purelin
* Divisão dos exemplos: 0.7, 0.15, 0.15

Esta rede obteve uma precisão global de 83.7% e de teste 77.8%.

Será esta a rede a utilizar no restante projeto.

# Estudo das precisões de teste com diferentes conjuntos de exemplos

## Teste da pasta test sem re-treinar a rede

Uma imagem com mesa

Descrição gerada automaticamente

Figura 1 - Matriz de confusão da pasta test

Pela matriz de confusão apresentada acima podemos verificar que a precisão de teste da rede neuronal foi de 65%. Esta percentagem é aceitável, pois os exemplos dados para este teste, eram totalmente novos para a rede.

## Treino da rede com a pasta test e teste de cada pasta individualmente

Uma imagem com mesa

Descrição gerada automaticamente

Figura 2 - Matriz de confusão da pasta start

Depois de treinada a rede com os exemplos da pasta test, a simulação dos testes para a pasta start, teve uma precisão de 73.3%.

Uma imagem com mesa

Descrição gerada automaticamente

Figura 3 - Matriz de confusão da pasta train

Já para a pasta train, a percentagem de precisão foi de 83.7%.

Uma imagem com mesa

Descrição gerada automaticamente

A pasta de test, obteve o mesmo resultado que anteriormente, 65%.

Figura 4 - Matriz de confusão da pasta test

# Simulação com formas geradas pelo utilizador

Foram desenhadas 12 figuras, 2 de cada forma geométrica aprendida pela rede neuronal.

Foi efetuado um estudo da eficácia da rede na identificação das formas corretas, a qual não correu como esperado.

Como podemos verificar na tabela abaixo, apenas 2 formas foram corretamente identificadas, sendo ambas o círculo.

Mesmo assim, visto que as imagens desenhadas não têm o mesmo tamanho das imagens de treino, pode-se explicar algum deste falhanço.

Uma imagem com mesa

Descrição gerada automaticamente

Figura - Resultado dos testes com desenhos

# Aplicação Gráfica

De modo a generalizar o processo descrito anteriormente e utilizando algumas das funções desenvolvidas anteriormente, foi desenvolvida uma aplicação gráfica, utilizando a ferramenta GUIDE do MATLAB.

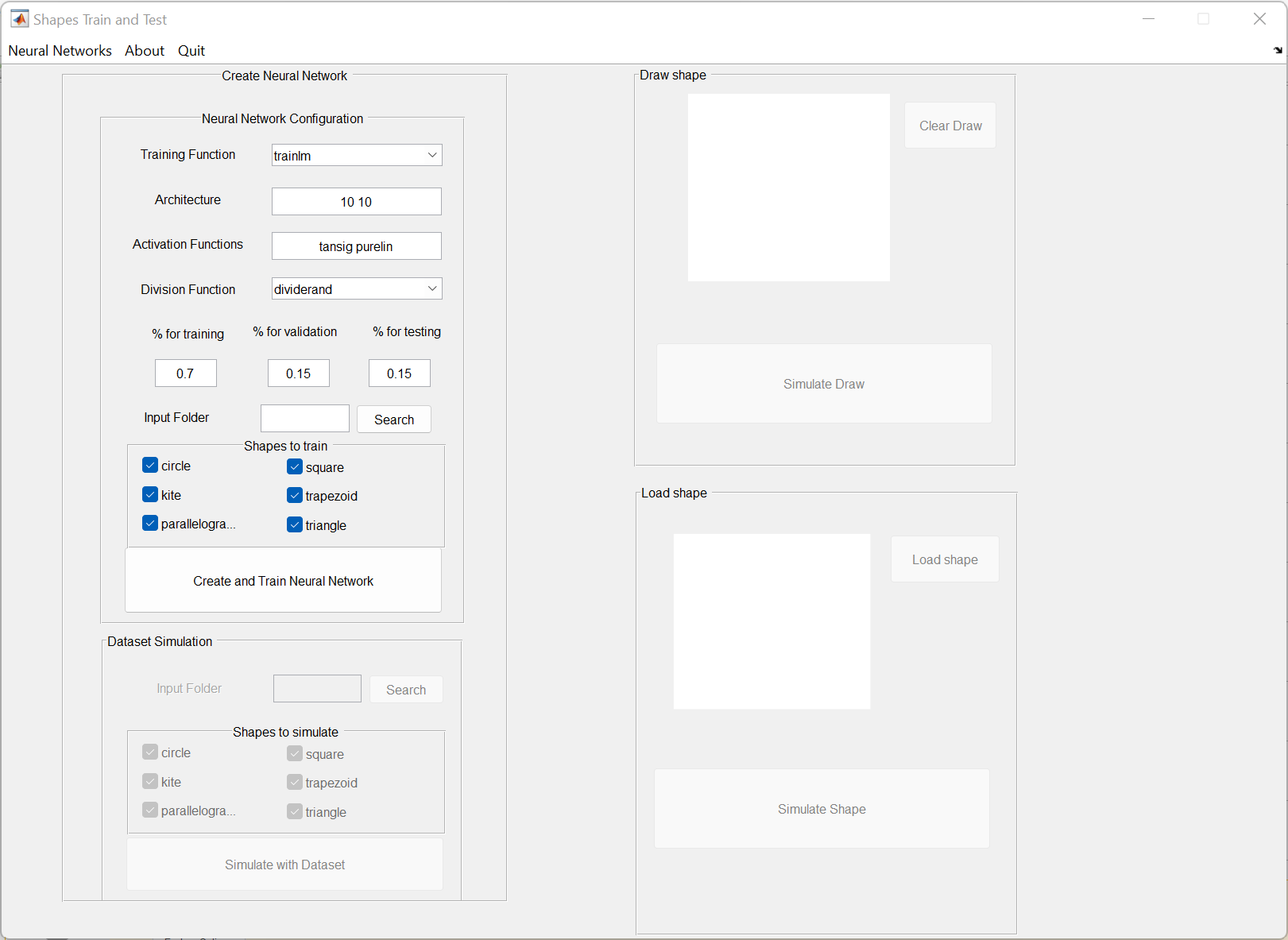


Figura 6 - Janela inicial da aplicação

Esta aplicação permite selecionar diferentes funções de treino previamente definidas, assim como definir o número de neurónios e camadas escondidas. As funções de ativação devem ser definidas pelo utilizador, as percentagens de divisão também. A função de divisão dos inputs é também selecionada de uma lista previamente carregada.

O utilizador tem a opção de selecionar quais as formas que pretende treinar. Este deve também definir a pasta onde se encontram as imagens de exemplo.

Existem outros três modos que são desbloqueados depois do utilizador carregar uma rede neuronal através do menu, ou depois de efetuar o treino de uma rede: simulação com um Dataset, onde o utilizador pode configurar quais formas pretende simular; simulação com um desenho, onde o utilizador pode desenhar uma forma; simulação com um ficheiro, na qual o utilizador escolhe qual a imagem que pretende simular.