

Faculdade de Engenharia da Universidade do Porto

Redes Neurais para Classificação de Expressões Faciais

Relatório Final



Inteligência Artificial

3º ano do Mestrado Integrado em Engenharia Informática e Computação

GRUPO E5_1

Lázaro Costa – up201405342 – up201405342@fe.up.pt

Miguel Lira – up201405324 – up201405324@fe.up.pt

Miriam Gonçalves – up201403441 – up201403441@fe.up.pt

21 de Maio de 2017

1. Objetivo

Este trabalho tem como objetivo a criação de uma Rede Neuronal Artificial para a classificação de expressões faciais. É pressuposta a utilização do algoritmo Retro Propagação do Gradiente ou *BackPropagation* para criar uma solução que proceda ao treino da rede, tendo por base um conjunto de dados disponibilizado. O conjunto de dados é constituído por 101 atributos que representam uma coordenada de um ponto da face humana, por um “alvo” /*target* que representa o sucesso ou insucesso na classificação dessa expressão e por um *timestamp*. A solução encontrada permite o treino da rede neuronal, o seu teste e retorna a classificação da expressão facial obtida.

2. Descrição

2.1. Especificação

O desenvolvimento deste trabalho dividiu-se em 3 partes sequenciais: uma primeira em que foi necessário o processamento dos dados fornecidos/ *dataset*, de seguida, procedeu-se ao treino da rede neuronal e, por fim, permitir a utilização da rede neuronal para a classificação de expressões faciais com base em entradas a serem fornecidas pelo utilizador. A implementação da rede neuronal foi feita recorrendo-se a uma *framework*, *Neuroph*, que permitiu a simplificação do desenvolvimento da arquitetura de uma rede neuronal comum através de uma biblioteca e uma ferramenta que ajudou à sua criação, treino e aplicação.

2.1.1. Descrição e análise do *dataset*

O *dataset* fornecido para o treino da rede neuronal é composto por 36 ficheiros, 9 ficheiros contêm o nome da expressão facial relativa ao indivíduo A em estudo e outros 9 relativas às expressões faciais do indivíduo B e os restantes 18 ficheiros com o resultado ou especificação para cada indivíduo, os dados da expressão facial são compostos por 1 *timestamp* e 100 coordenadas faciais tridimensionais (x,y,z). O x e o y são dados em pixéis e a terceira coordenada é dada em milímetros. Deste conjunto de atributos fazem parte coordenadas dos olhos, sobrancelhas, boca, contorno da face, nariz, a ponta do nariz, as linhas acima das sobrancelhas e as íris.

Os dados disponibilizados têm valores distintos e não contínuos, tornando-se necessário proceder à normalização destes nos limites de [0,1], evitando-se a possibilidade de levar a rede a tomar decisões erradas. Esta normalização de dados é feita recorrendo-se à classe *MinMaxNormalizer* fornecido pela *framework Neuroph*.

2.1.2. Estrutura da rede

Uma rede neuronal é uma rede de circuitos que simula a estrutura neuronal de organismos inteligentes e que adquire conhecimento através da experiência, tal como o cérebro humano. Este tipo de redes é constituído por 3 elementos: elemento de processamento, estrutura das ligações e a lei da aprendizagem que para este projeto será o algoritmo *backpropagation*. A informação armazenada pela rede é partilhada por todas as suas unidades de processamento(conexionismo).

Foi necessário a criação de 9 redes neuronais, cada uma respetiva a uma expressão facial diferente.

Depois de sabida o número de entradas da rede estudou-se a memória ocupada pela rede: $301 \cdot \text{nr}^\circ \text{ de neurónios} + 1 \cdot \text{nr}^\circ \text{ de neurónios}$, em que 301 corresponde ao número de entradas e 1 output de saída.

Sendo usado como algoritmo de aprendizagem o *backpropagation*, a rede será de múltiplas camadas, a camada de entrada, intermédia e de saída. É de notar que na camada intermédia é preciso ter 3 a 5 vezes mais ligações do que variáveis. A informação circula no sentido *input* -> *output*. Após ser calculado o output é necessário proceder ao cálculo do erro, e consequentemente à atualização dos valores da rede no caso do valor do gradiente do erro em relação aos respetivos valores das arestas for elevado, esta atualização é feita no sentido *output*->*input*.

Uma vez que o *dataset* contém 101 atributos que são coordenadas (x,y,z) e um *timestamp*, então as 9 redes de reconhecimento de expressões faciais terão 301 entradas e 9 saídas, que correspondem ao número de expressões faciais diferentes estudadas na base de dados utilizada. No final será avaliado o melhor output dado pelos nós de saídas das redes, ou seja, o valor mais baixo de erro e será escolhido esse como sendo a expressão facial.

2.1.3. Normalização de dados

No *dataset* são utilizadas as seguintes características sobre a face humana:

1. timestamp – identificação da imagem de cada frame
2. left eye – olho esquerdo (numérico: de 0 a 7)
3. right eye – olho direito (numérico: de 8 a 15)
4. left eyebrow – sobrancelha esquerda (numérico: de 16 a 25)
5. right eyebrow – sobrancelha direita (numérico: de 26 a 35)
6. nose – nariz (numérico: de 36 a 47)
7. mouth – boca (numérico: de 48 a 67)
8. face contour – contorno do rosto (numérico: de 68 a 86)
9. left iris – iris esquerda (numérico: 87)
10. right iris – iris direita (numérico: 88)
11. nose tip – ponta do nariz (numérico: 89)
12. line above left eyebrow – linha acima da sobrancelha esquerda (numérico: de 90 a 94)
13. line above right eyebrow – linha acima da sobrancelha direita (numérico: de 95 a 99)

2.1.4. Lei de Aprendizagem

Dado que se está a desenvolver uma rede neuronal de múltiplas camadas usar-se-á um tipo de aprendizagem supervisionada recorrendo-se ao algoritmo de *backpropagation*, tal como se referiu anteriormente no ponto 2.1.2.. Este método tenta minimizar a função de custo quadrático.

3. Desenvolvimento

As 9 redes neurais para a classificação de expressões faciais foram implementadas em *Java* recorrendo-se à *framework Neuroph*. Todo o código foi desenvolvido no IDE IntelliJ em Windows 10. A utilização do *Neuroph* permitiu que o desenvolvimento do projeto fosse rápido e fácil.

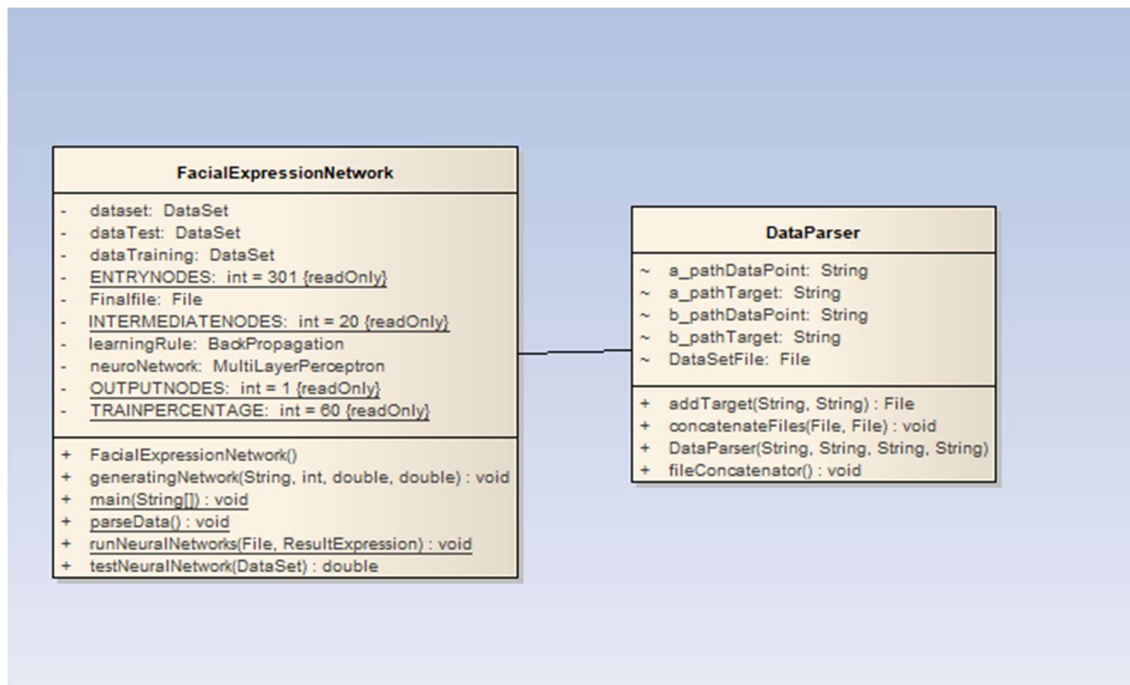


Figura 1: Diagrama de Classes

O resultado final em código permite ao utilizador introduzir um ficheiro que queira testar com dados compatíveis. E a partir deste é possível saber qual a expressão que se encontra mais presente nesses dados. No final, o programa retornará uma resposta com a expressão facial encontrada.

4. Experiências

Foi necessário a realização de diversas experiências com a finalidade de verificar os valores que permitiam a melhor configuração das 9 redes neurais, nomeadamente, o número de nós intermédios, a taxa de aprendizagem e a percentagem do *dataset* utilizada para treino da rede.

Para cada experiência, foram feitos 10 testes para cada valores nas 9 redes diferentes.

4.1. Taxa de Aprendizagem

Nesta experiência, variou-se o valor da taxa de aprendizagem entre 0.05, 0.10, 0.15 e 0.20. No final verificou-se, o erro médio, o erro de treino, o erro de teste e o tempo de execução médio de cada rede neuronal.

As redes neurais foram testadas com os seguintes parâmetros:

Número máximo de iterações: 1000

Erro máximo: 0,01

Número de nós intermédios: 20

Percentagem do *dataset* para treino: 60%

Tendo em conta esta configuração das 9 redes neurais, no final, calculou-se a média das 9 redes diferentes obtendo-se os seguintes resultados:

| Taxa de Aprendizagem | Erro de treino | Erro de teste | Erro médio | Tempo médio de execução |
|----------------------|----------------|---------------|-------------|-------------------------|
| 0.05 | 0,027662556 | 0,148686111 | 0,100282 | 29,00042222 |
| 0.10 | 0,034119789 | 0,257851 | 0,110856556 | 30,60735556 |
| 0.15 | 0,037793378 | 0,080171756 | 0,106579593 | 91,40881111 |
| 0.20 | 0,040636667 | 0,157426556 | 0,114596 | 46,983 |

Tabela 2: Experiência da taxa de aprendizagem

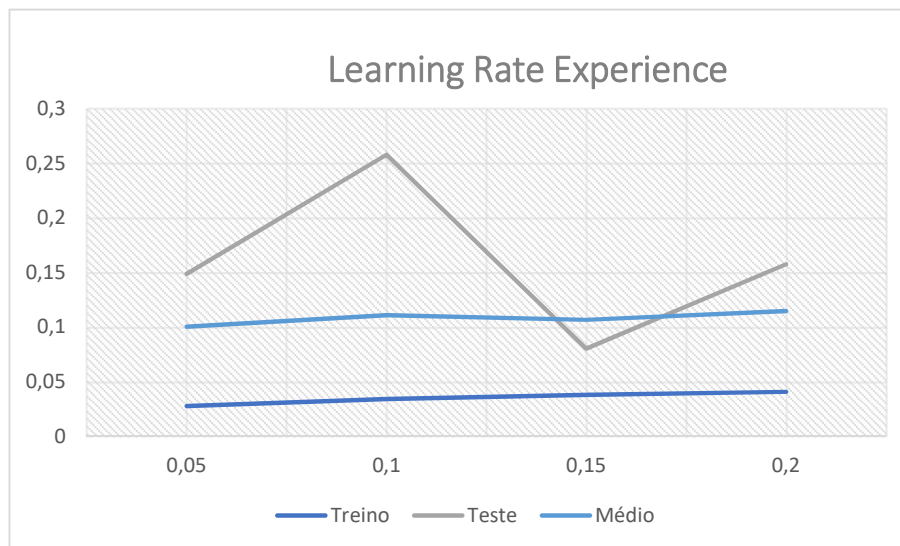


Gráfico 3: Experiência da taxa de aprendizagem

Depois de se analisar os valores anteriores verifica-se que para um taxa de aprendizagem de 0.15 os erros são em geral mais baixos, portanto as 9 redes neurais foram configuradas com este valor e optou-se por fazer o resto das experiências com o parâmetro de aprendizagem com esse valor.

4.2. Número de Nós Intermédios

Nesta experiência, incrementou-se o valor do número de nós intermédios entre 10 e 40 em intervalos de 10. No final verificou-se, o erro médio, o erro de treino, o erro de teste e o tempo de execução médio de cada rede neuronal.

As redes neuronais foram testadas com os seguintes parâmetros:

Número máximo de iterações: 1000

Erro máximo: 0,01

Taxa de aprendizagem: 0.15

Percentagem do *dataset* para treino: 60%

Tendo em conta esta configuração das 9 redes neuronais, no final, calculou-se a média das 9 redes diferentes obtendo-se os seguintes resultados:

| Nrº de nós na camada intermédia | Erro de treino | Erro de teste | Erro médio | Tempo médio de execução |
|---------------------------------|----------------|---------------|-------------|-------------------------|
| 10 | 0,040792774 | 0,081667035 | 0,113222464 | 22,43233333 |
| 20 | 0,039661966 | 0,08007057 | 0,11127056 | 32,62664444 |
| 30 | 0,041689803 | 0,083447313 | 0,113253416 | 45,8971679 |
| 40 | 0,035711734 | 0,075761187 | 0,104711104 | 56,33026667 |

Tabela 2: Experiência do número de nós na cama intermédia

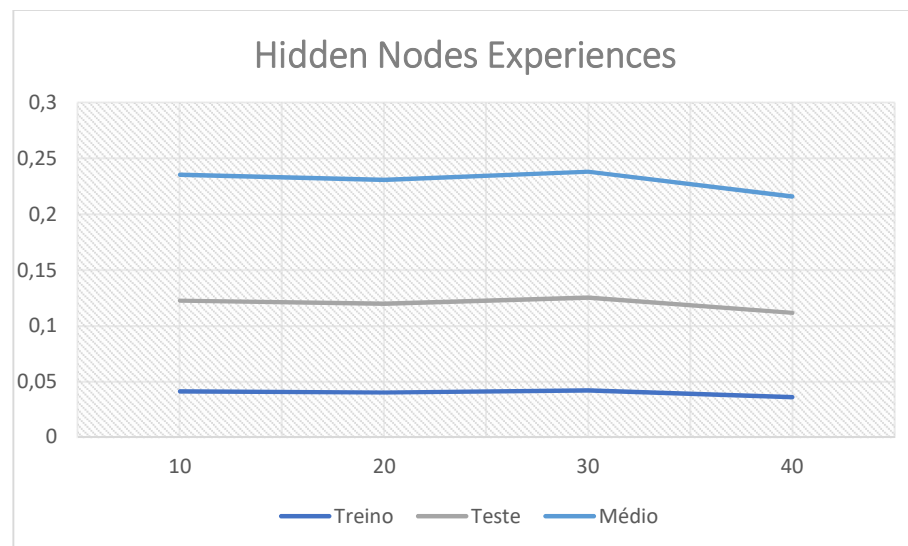


Gráfico 2: Experiência do número de nós da camada intermédia

Depois de se analisar os valores anteriores verificou-se que quando a rede é configurada com 40 nós na camada intermédia, esta obtém melhores resultados, portanto este valor foi o escolhido como valor final para as 9 redes neuronais diferentes e para as seguintes experiências.

4.3. Percentagem de *dataset* para Treino

Nesta experiência, os valores das percentagens de *dataset* para treino das rede neuronais tomaram os seguintes valores: 50%, 60%, 70% e 80%. No final verificou-se, o erro médio, o erro de treino, o erro de teste e o tempo de execução médio de cada rede neuronal.

As redes neuronais foram testadas com os seguintes parâmetros:

Número máximo de iterações: 1000

Erro máximo: 0,01

Taxa de aprendizagem: 0.15

Número de nós da camada intermédia: 40

Tendo em conta esta configuração das 9 redes neuronais, no final, calculou-se a média das 9 redes diferentes obtendo-se os seguintes resultados:

| Porcentagem para treino | Erro de treino | Erro de teste | Erro médio | Tempo médio de execução |
|-------------------------|----------------|---------------|-------------|-------------------------|
| 50 | 0,040874459 | 0,086025775 | 0,11584863 | 43,30676667 |
| 60 | 0,035711734 | 0,075761187 | 0,104711104 | 56,33026667 |
| 70 | 0,037889406 | 0,076322828 | 0,111980709 | 66,45208889 |
| 80 | 0,039120615 | 0,075014949 | 0,105228941 | 86,14361111 |

Tabela 3: Experiência da percentagem de dataset para treino

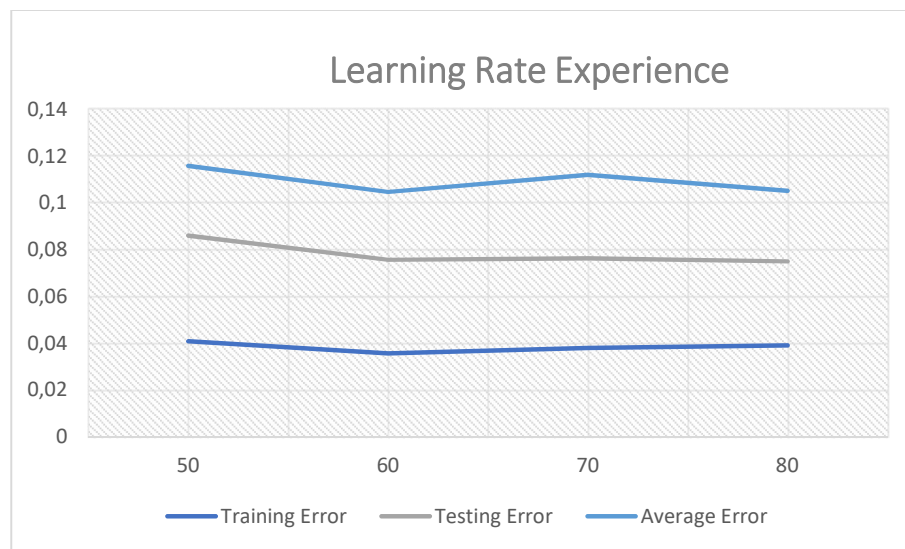


Gráfico 3: Experiência da percentagem de dataset para treino

Através dos resultados anteriores pode verificar-se que a rede apresenta melhores resultados quando 60% dos dados do *dataset* são utilizado para treino das redes neuronais.

5. Conclusões

Verifica-se que as redes neuronais podem desempenhar um papel importante na civilização para estudos estatísticos e tomadas de decisões. No entanto, também conclui-se que é uma área que ainda se encontra em desenvolvimento, pelo que algumas técnicas de implementação de redes neuronais podem vir a ser melhoradas.

É de notar que o contexto realista do desenvolvimento desta rede neuronal desperta um maior interesse por esta área e também por parte dos estudantes para esta matéria.

O desenvolvimento do código foi uma parte bastante rápida e simples, uma vez que a *framework* utilizada tinha todas as ferramentas presentes para tornar esta fase pouco complexa. Na parte de testes da rede para saber qual a melhor configuração de cada uma das redes foi mais demorada, uma vez que foi necessário testar 10 vezes para cada valor as 9 redes neuronais diferentes.

A participação equitativa de cada elemento de grupo tornou o desenvolvimento deste projeto mais rápido, todos se encontram bastante satisfeitos com o projeto final e acreditam que cumpre todos os requisitos pedidos.

6. Recursos

6.1. Bibliografia

- o <http://neuroph.sourceforge.net/>
- o <http://paginas.fe.up.pt/~eol/IA/IA0708/APONTAMENTOS/IA-NN.pdf>
- o <https://www.mq15.com/pt/articles/497>
- o FREITAS, F. A.; Peres, S.M.; Lima ; C. A. M. ; BARBOSA, F. V. . *Grammatical Facial Expressions Recognition with Machine Learning. In: 27th Florida Artificial Intelligence Research Society Conference (FLAIRS), 2014, Pensacola Beach. Proceedings of the 27th Florida Artificial Intelligence Research Society Conference (FLAIRS)*. Palo Alto: The AAAI Press, 2014. p. 180-185.

6.2. Software

- o *Neuroph*

6.3. Apêndice

Para o utilizador: ao correr o programa basta inserir o diretório do ficheiro a correr como argumento e o programa mostrará o resultado. Caso o *dataset* não se encontre presente, é necessário que o utilizador crie uma pasta com o nome *Experiences* com todos os ficheiros do *dataset* dentro desta e coloca-las ao mesmo nível da página source do programa.