Inteligência Artificial

Início Objectivos Programa Bibliografia Docentes Avaliação Horário Aulas Trabalhos

Aprendizagem com Redes Neuronais O tópico destas aulas é a contrução, treino e validação de uma rede neuronal, recorrendo à

que correspondem a parâmetros químicos, geológicos, mineralógicos e isotópicos:

utilização da ferramenta computacional NeurophStudio.

O Problema de Classificação Para avaliar o potencial de recursos minerais numa região no que respeita às matérias primas críticas (critical raw materials), uma equipa de especialistas de pesquisa mineral seleccionou um conjunto de atributos que melhor caracterizam diferentes mineralizações associadas a essas matérias primas e que suportem as suas campanhas de pesquisa. A equipa está interessada em estudar três tipos de mineralização: t1- 'elementos das terras raras', t2- 'estanho e volfrâmio', e t3- 'lítio'. Para o efeito, seleccionaram um conjunto de 178 amostras descritas por 13 atributos,

1. Lítio 2. Flúor 3. Elementos voláteis em turmalinas 4. Estanho 5. Distância ao granito 6. Cério 7. Extensão do halo de alteração hidrotermal 8. Nióbio 9. Neodímio 10. Minerais do grupo dos fosfatos

11. Conteúdo em W no rútilo 12. Delta 180 13. Temperatura de homogeneização em inclusões fluidas

respectivamente, por 1, 2, e 3. O ficheiro de dados está disponível aqui. Este problema é um exemplo típico de reconhecimento de padrões em que os atributos de

Cada uma das 178 amostras está classificada num dos três tipos de mineralização, codificado,

entrada estão relacionados com diferentes classes, e se pretende usar uma Rede Neuronal não só para classificar os dados disponíveis como também para aprender a classificar com exatidão amostras de matérias primas críticas recolhidos em futuras campanhas.

Procedimento Experimental

Para construir, treinar e avaliar o desempenho de uma rede neuronal consideram-se os seguintes passos:

1. Criar um projecto NeurophStudio; 2. Criar um conjunto de treino; 3. Normalizar os dados; 4. Criar uma rede neuronal e definir a parameterização de arquitectura e treino; 5. Treinar a rede e avaliar o desempenho da fase de treino 6. Testar a rede e avaliar o desempenho de generalização.

1. Criar um projecto NeurophStudio

Selecionar File -> New Project, escolher Neuroph Project e carregar no botão Next. Definir o nome do projecto e a sua localização no computador. Após terminar carregar em Finish. É criado um novo projecto o qual aparece na janela de projectos do lado esquerdo do NeurophStudio.

2. Criar um conjunto de treino Existem dois tipos de aprendizagem de redes neuronais, supervisionada e não supervisionada. Em aprendizagem supervisionada, os dados de treino estão classificados (outputs da rede) e a rede aprende a inferir a relação entre inputs e outputs. Neste tutorial só iremos trabalhar com aprendizagem supervisionada.

Para o conjunto de dados mineralizacoesOrg.txt há que decidir como recodificar o atributo classe em atributo(s) de output para a rede. Repare que a codificação original dos três tipos de classe é arbitrária. Proceder à recodificação, com um atributo binário para cada classe, e gravar o novo conjunto de dados. Para facilitar, a recodificação encontra-se aqui. Definir o parâmetro inputs igual ao número de atributos de entrada do conjunto de dados e definir o parâmetro outputs igual ao número de atributos de saída (3) de acordo com a recodificação adequada.

Para carregar o conjunto de dados de treino:

• Botão direito do rato selecionar New -> Data Set, atribuir um nome ao conjunto de • Definir os parâmetros: Type:supervised; Number of inputs:13; Number of • Carregar ChooseFile e selecionar o ficheiro do conjunto de dados (os valores dos

atributos estão separados por tabs); • Carregar Load e todos os dados serão carregados para a tabela; • Carregar em Finish um novo conjunto de treino aparece no projecto.

No final deverá inspecionar o conjunto de dados. 3. Normalizar os dados

Por forma a treinar a rede neuronal o conjunto de dados tem de ser normalizado. Os dados em questão são multivariados e os valores dos 13 atributos de entrada estão em diferentes escalas, com distintos intervalos de valores e diferentes variâncias. Existem diferentes métodos de normalização. Para o conjunto de dados deste tutorial vamos escolher a seguinte função de normalização:

 $x'_{if} = \frac{x_{if} - x_{min}(f)}{x_{max}(f) - x_{min}(f)}$

em que x_{if} é o valor do atributo f (f= 1, ..., 13) para a i-ésima entidade (i= 1, ..., 178), x'_{if} o correspondente valor normalizado, $x_{min(f)}$ e $x_{max(f)}$ correspondem, respectivamente, aos valores mínimo e máximo do atributo f.

Para normalizar os dados:

• Selecionar o conjunto de dados; • Com o botão direito do rato selecionar Normalize --> Max Min.

4. Definir uma Rede Neuronal

O Neuroph suporta as arquitecturas mais comuns de redes neuronais tais como Adaline, Perceptron, Multi Layer Perceptron, etc. No que se segue vamos usar a arquitectura Multi Layer Perceptron.

Definida a arquitectura da rede há que definir o número de camadas escondidas (hidden Layer) bem como o número de neurónios de cada camada escondida. A arquitectura de redes neuronais Multi Layer Perceptron com uma única camada escondida conjuntamente com a aplicação do algoritmo de backpropagation é suficientemente eficaz para classificar muitos problemas reais. Assim, vamos partir de uma arquitectura com uma única camada escondida. Existem algumas regras empíricas para determinar o número de neurónios a usar na camada

escondida, como por exemplo:

• O número de neurónios escondidos deve estar na gama entre a dimensão da camada de input e a dimensão da camada de output (r1); • O número de neurónios escondidos deve ser igual à dimensão da camada de input mais a da

camada de output, mais 1 (r2); • O número de neurónios escondidos deve ser menor do que 2 vezes a dimensão da camada de input (r3). Uma arquitectura com poucos neurónios na camada escondida resultará num cenário de

cenário de sobreajustamento (overfitting). A melhor arquitectura será aquela que estabelece o melhor cenário de compromisso de desempenho da rede neuronal. Em todo este exercício, vamos trabalhar com o algoritmo de Retropropagação do Erro com momento (Backpropagation with Momentum). Este algoritmo apresenta maior taxa de

subajustamento (underfitting) enquanto uma arquitectura com muitos neurónios resulta num

convergência quando comparado com o algoritmo de Backpropagation. O parâmetro 'Momentum' é adicionado para acelerar o processo de aprendizagem e melhorar a eficiência do algoritmo.

Para definir a arquitectura e parametrização da rede neuronal:

 Com o botão direito do rato carregar no projecto e selecionar New -> Neural Network; • Definir o nome da rede e selecionar tipo de rede neuronal: Multi Layer Perceptron; • Especificar o número de neurónios da camada escondida (e.g. usar a regra r2);

• Selecionar as opções 'Use Bias Neurons' e 'Sigmoid' para a função de transferência (porque os dados estão normalizados entre 0 e 1); O Bias Neuron é obrigatório para uma rede neuronal treinada com o algoritmo backpropagation. Sem esta opção, a rede não aprende. • Os neurónios Bias permitem desviar a curva sigmóide ao longo das abcissasas. Um

Bias de input terá sempre o valor de 1. • Escolher o algoritmo Backpropagation with Momentum.

5. Treinar a Rede Neuronal

Após a criação do conjunto de treino e da rede neuronal pode-se passar à fase de treino da rede. Previamente, há que definir os seguintes parâmetros: • Learning Rate - avalia a contribuição relativa de cada peso no erro total. A seleção da taxa

de aprendizagem é de importância crítica na pesquisa do mínimo global da distância de erro. Se este parâmetro for demasiado pequeno, a progressão da aprendizagem será verdadeiramente lenta. Se a taxa de aprendizagem for muito elevada, prossegue de forma muito rápida mas pode potencialmente produzir oscilações entre soluções inadequadas. • Momentum Rate - pode ser útil para acelerar a convergência e evitar mínimos locais. A ideia base é estabilizar as alterações dos pesos evitando alterações radicais através de uma combinação entre o termo de decréscimo do gradiente com uma fração da alteração dos pesos anteriores. Quando o valor do Total Net Error descer abaixo do erro máximo, o treino está finalizado. Se o erro for menor do que o valor limiar pré-definido, então obterse-á uma melhor aproximação.

Para iniciar a fase de treino:

 Selecionar o conjunto de treino (drag and drop para a entrada da rede); • carregar no botão Train e definir os parâmetros indicados; os valores por defeito são: Max Error= 0.01; Learning Rate= 0.2; Momentum= 0.7

Ensaio 1

Treine a rede neuronal anterior com a arquitectura #i13#h17#03. A partir do gráfico produzido analise o número de iterações do algoritmo Backpropagation bem como o 'Total Net Error'. A rede aprendeu?

Analisar o Total Mean Square Error e todos os erros individuais;

Após o treino da rede, passar à fase de teste. Selecionar o botão Test;

Discutir os valores obtidos.

Pode selecionar aleatoriamente alguns (e.g. 5) exemplos do conjunto de dados e comparar a classificação aprendida pela rede com os valores reais de classificação.

A arquitetura escolhida neste ensaio é boa? <u>Ensaio 2</u>

O objectivo deste ensaio é analisar o papel experimental dos parâmetros 'taxa de aprendizagem' e 'momento'. Mantendo a arquitectura da rede do Ensaio 1, vamos incrementar a taxa de aprendizagem e

baixar o momento. Assim, iremos analisar como se comporta a rede neuronal quando treinada com uma taxa de aprendizagem mais elevada. • Instanciar os parâmetros: Learning Rate: 0.7; Momentum: 0.4;

 Carregar no botão Train; Analisar os resultados de treino.

Teste da rede neuronal

Selecionar o botão Test, para usar a rede neuronal em modo teste. Discutir os resultados de teste.

Ensaio 3 Proceder à mesma análise do Ensaio 2, quando se decresce a 'taxa de aprendizagem' e se mantém

o 'momento'. • Instanciar os parâmetros: Learning Rate: 0.1; Momentum: 0.4; Carregar no botão Train;

Analisar os resultados de treino.

outros parâmetros: #i13#h4#03;

Ensaio 4

O objectivo deste ensaio é treinar uma rede com menor número de neurónios na camada escondida e analisar o desempenho da rede. • Criar uma nova rede neuronal com 4 neurónios na camada escondida e manter todos os

• Treinar esta rede com os parâmetros de aprendizagem pré-definidos por forma a poder comparar os resultados com os obtidos anteriormente; • Testar a rede neuronal treinada; • Analisar e discutir os resultados obtidos comparando com os anteriores.

Pode selecionar alguns (e.g. 5) exemplos do conjunto de dados e comparar a classificação aprendida pela rede com os valores reais de classificação.

Ensaio 5 O objectivo deste ensaio é verificar a influência do incremento da 'taxa de aprendizagem' nesta

arquitectura. • Voltar a treinar a rede do Ensaio 4 com o parâmetro Learning Rate= 0.4 e Momentum=0.6; Aplicar o modo Test à rede treinada.

 Analisar e discutir os resultados obtidos comparando com os anteriores. Ensaio 6

O objectivo deste ensaio é treinar uma rede com número de neurónios na camada escondida ainda menor, e analisar o desempenho da rede.

outros parâmetros: #i13#h2#03; • Treinar esta rede com os parâmetros de aprendizagem pré-definidos por forma a comparar os resultados com os obtidos anteriormente; Testar a rede neuronal treinada;

Criar uma nova rede neuronal com 2 neurónios na camada escondida e manter todos os

 Analisar e discutir os resultados obtidos e comparar com os anteriores. Repetir para 1 neurónio na camada escondida.

<u>Ensaio 7</u> O objectivo é deste ensaio é analisar o desempenho de uma arquitectura de rede com duas

seguido de espaco e 2;

Ensaio 11

Comentário Final

Momentum: 0.7

camadas escondidas. • Criar uma nova Rede Neuronal do tipo Multi Layer Perceptron (como nos casos anteriores). • Para criar a aguitetura #i13#h3#h2#o3, de duas camadas escondida, preencher o valor 3,

• Treinar esta rede com os parâmetros de aprendizagem pré-definidos por forma a comparar os resultados com os obtidos anteriormente; Testar a rede neuronal treinada;

• Analisar e discutir os resultados obtidos e comparar com os anteriores. 6. Avaliar o desempenho de generalização

aprendizagem mas que não são usados no processo de treino. Este aspecto é fundamental uma vez que para aplicações do mundo real os utilizadores apenas têm acesso a um número limitado de

Uma das grandes vantagens das redes neuronais é a sua capacidade de generalização. Isto significa que uma rede treinada é capaz de classificar dados da mesma classe dos dados de

dados, por vezes escassos, para a geração das redes neuronais.

O conjunto de dados original será agora dividido em dois sub-conjuntos dijuntos: o conjunto de treino e o conjunto de teste. O conjunto de teste servirá para obter uma medida independente de análise do desempenho da rede.

o conjunto de treino for reduzido; e (ii) a capacidade de generalização da rede para classificar padrões de entrada que não foram usados no treino.

O objectivo dos ensaios seguintes é analisar: (i) quão rápida uma rede neuronal pode aprender se

Nos ensaios seguintes vamos usar a rede neuronal #i13#h4#o3 com os parâmetros do Ensaio 4.

Ensaio 8

• 40% dos dados para conjunto de treino e 60% para conjunto de teste; 71 instâncias para treino e 107 instâncias de teste; Analisar a fase de treino e de teste, comparando com os resultados anteriores.

Ensaio 9 • 30% dos dados para conjunto de treino e 70% para conjunto de teste; 53 instâncias para

treino e 125 instâncias para testar a rede; Analisar a fase de treino e de teste, comparando com os resultados anteriores. Ensaio 10

• 20% dos dados para conjunto de treino e 80% para conjunto de teste; 36 instâncias para treino e 142 instâncias para teste; Analisar a fase de treino e de teste, comparando com os resultados anteriores.

• 10% dos dados para conjunto de treino e 90% para conjunto de teste; 17 instâncias para treino e 161 instâncias para teste. Uma vez que este é um número reduzido de instâncias, por forma a melhorar a eficácia e a performance da rede, altera-se taxa de aprendizagem para 0.4, mas mantem-se o momento inalterado: Max Error: 0.01 Learning Rate: 0.4

• Analisar a fase de treino e de teste, comparando com os resultados anteriores.

Com este exercício tutorial, pretendeu-se analisar experimentalmente: i) a influência da

dos diferentes parâmetros no processo de aprendizagem; iii) a influência do tamanho do conjunto de exeplos de treino na capacidade de aprendizagem e generalização. <u>Início | Objectivos | Programa | Bibliografia | Docentes | Avaliação | Horário | Aulas | Trabalhos |</u>

Comentários e sugestões para jleite@fct.unl.pt

arquitectura da rede neuronal na capacidade de aprendizagem e generalização; ii) a influência