Ficha 10 - DC

João Nunes (A82300)
Luís Braga (A82088)

24/04/2020

Conteúdo

1 Parte I

1.1 De onde surgiu o nome redes neuronais? Quais são as características do modelo que o tornam "neuronal"?

As redes denominam-se de neuronais devido ao processo biológico inerente ao ser humano, no que toca à conexão entre os neurónios e a maneira em como a informação é trocada entre estes. Os neurónios das redes neuronais tratam-se de unidades de processamento, e usam estes neurónios com o propósito de comparar os atributos e encontrar conexões fortes.

1.2 Que vantagem(ns)apresentam as redes neuronais relativamente a outros modelos de previsão?

As redes neuronais artificiais possuem a vantagem de conseguirem aprender e modelar relações não lineares, para além disso as RNAs têm a capacidade .de generalizar pelo que após aprenderem os *inputs* estas irão conseguir inferir relações em dados novos.

1.3 Como é que as percentagens de confiança devem ser usadas em conjunto com as previsões de uma rede neural?

As percentagens de confiança deverão ser sempre consultadas em conjunto com a previsão da rede neural, uma vez que a percentagem de confiança indica se a previsão está sustentada com uma base forte ou com uma base fraca, ou seja, percentagem de confianças baixas poderão não ser um bom indicador uma vez que a previsão tanto poderá estar acertada como errada, contudo com percentagem de confianças o algoritmo indica que a previsão está correcta.

1.4 Se quiser ver os detalhes de um nó num gráfico de uma rede neural no RapidMiner, o que pode fazer?

Para ver os detalhes de um nó do gráfico baste aceder ao gráfico e de seguida pressionar por cima do nó que se pretende obter informações.

1.5 Quais as camadas que constituem as redes neuronais e o que representam?

Uma rede neuronal possui sempre duas camadas *standard* com um número variável de nodos, a camada de *input* e a camada de *output*. De seguida existem também as camadas intermédias (*hidden layers*) podendo existir também várias camadas intermédias. Uma rede simples possui apenas uma camada de *input*, uma *hidden layer* e uma camada de *output*.

2 Parte II

2.1 Faça o download do dataset de treino fornecido, denominado credit-training.csv, e importe-o para o repositório do RapidMiner. Execute a fase de Data Understanding.

2.1.1 Quais os níveis de risco de crédito existentes?

Existe quatro níveis de crédito distintos, muito baixo, baixo, moderado e alto. Para além disso existe também mais um nível para não permitir que seja emprestado dinheiro ao cliente.

Figura 1: Níveis de crédito..

2.1.2 Qual a média da quantidade de empréstimo?

Figura 2: Quantidade média de empréstimo.

Como se pode verificar na figura acima a quantidade média de empréstimo é de 189545.633.

2.2 Crie o seu próprio conjunto de dados de teste usando os atributos no conjunto de dados de treino como um guia. Digite pelo menos 20 observações. Pode inserir dados para pessoas que conhece (talvez seja necessário estimar alguns valores de atributos, por exemplo, a pontuação de crédito) ou pode simplesmente testar valores diferentes para cada um dos atributos. Por exemplo, pode optar por inserir quatro observações consecutivas com os mesmos valores em todos os atributos, exceto na pontuação de crédito, onde pode aumentar a pontuação de crédito de cada observação em 100, de 400 a 800.

2.3 Efectue a etapa de Data Preparation. Não se esqueça de colocar o operador Set Role nos atributos que justifiquem a sua aplicação, tendo em conta que o objetivo é a previsão do risco de crédito.

Criou-se então o dataset necessário com 20 observações. De seguida introduziram-se os dados no RapidMiner, como se pode comprovar na seguinte figura.

Row No.	Applicant_ID	Credit_Score	Late_Payme...	Months_In_...	Debt_Incom...	Loan_Amt	Liquid_Asse...	Num_Credit...
1	1	341	7	50	8	51761	12190	9
2	2	301	1	43	9	229092	24697	3
3	3	278	13	96	9	184530	4339	7
4	4	457	15	25	1	129018	844	7
5	5	590	4	6	8	361007	23911	8
6	6	409	3	20	10	39260	8446	8
7	7	671	5	100	9	447433	3138	2
8	8	705	6	98	9	323444	4224	5
9	9	491	2	76	8	89977	18807	1
10	10	443	3	43	5	103229	1197	6
11	11	368	7	41	5	165542	3876	12
12	12	490	11	102	5	339743	21499	12
13	13	521	10	52	7	381222	23560	3
14	14	545	8	48	9	125606	10392	4
15	15	399	0	49	1	152298	2350	9
16	16	415	1	102	8	322204	21115	4
17	17	427	6	19	5	46957	2404	11
18	18	567	4	3	2	286501	23550	6

Figura 3: Dataset de scoring criado.

2.4 Num novo processo, repita os passos no RapidMiner tal como descritos nos slides da aula para aplicar o modelo de rede neuronal ao dataset de teste. Pode optar por fazer antes um processo para descobrir os valores otimizados dos parâmetros do operador das redes neuronais, tal como exemplificado na aula.

Decidiu-se por encontrar os valores óptimos antes de aplicar o modelo de rede neuronal. Para tal, executou-se um processo semelhante ao da aula anterior com certas mudanças, como seria de esperar, pois o modelo é diferente.

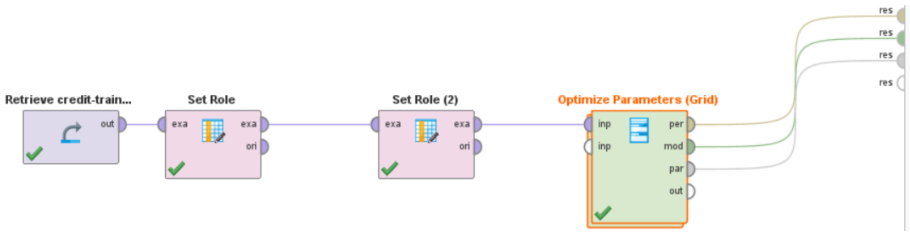

Figura 4: Processo para descobrir os valores óptimos.

Em *Optimize Parameters* especificou-se também o seguinte subprocesso. Por sua vez, neste processo, escolheram-se os parâmetros a optimizar e, em semelhança com os slides, escolheram-se os parâmetros *learning rate* e *training cycles*.

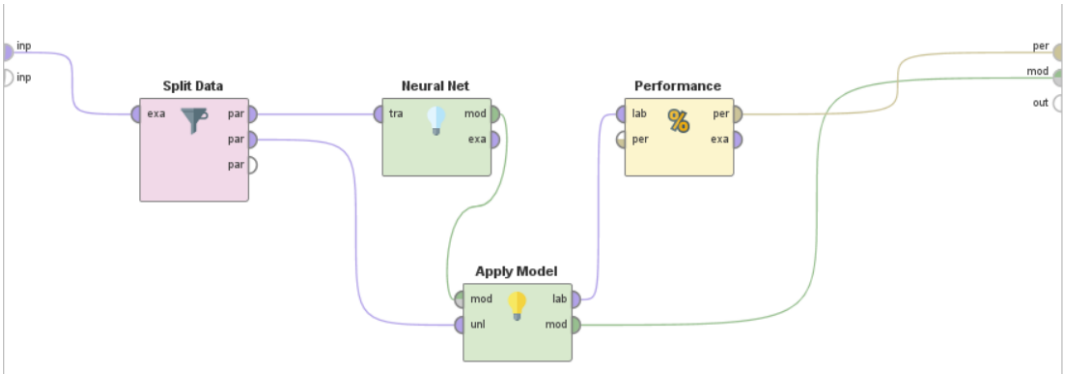

Figura 5: Subprocesso.

Obtiveram-se valores com certa de 98% de *accuracy*, o que se considerou bastante satisfatório. Atentando no valor com maior *accuracy* os valores *learning rate* é de 0.4 e o valor de *training cycles* é de 100. Portanto, serão estes os valores escolhidos que serão substituídos nos parâmetros da rede neural.

iteration	Neural Net.learning_rate	Neural Net.training_cycles	accuracy ↓
115	0.400	100	0.978
49	0.400	41	0.970
68	0.100	60	0.970
59	0.300	51	0.963
60	0.400	51	0.963
79	0.100	70	0.963
103	0.300	90	0.963
105	0.500	90	0.963
57	0.100	51	0.956
26	0.300	21	0.956
64	0.800	51	0.956
106	0.600	90	0.956
16	0.400	11	0.948
53	0.800	41	0.948
24	0.100	21	0.948
114	0.300	100	0.948
71	0.400	60	0.948
82	0.400	70	0.948

Figura 6: .

De forma alternativa os valores poderiam ser retirados do *Parameter Set*. Pode-se ver que os últimos dois campos são os valores que se irão usar na rede neural.

ParameterSet

```
Parameter set:

Performance:
PerformanceVector [
-----accuracy: 97.78%
ConfusionMatrix:
True:   Moderate        High    Low     DO NOT LEND     Very Low
Moderate:       36      2       0       0       0
High:   1       38      0       0       0
Low:    0       0       50      0       0
DO NOT LEND:    0       0       0       0       0
Very Low:       0       0       0       0       8
]
Neural Net.learning_rate        = 0.4
Neural Net.training_cycles      = 100
```

Figura 7: Parâmetros óptimos.

2.5 Execute o modelo e analise as previsões para cada uma das suas observações de pontuação. Relate os seus resultados, incluindo resultados interessantes ou inesperados.

Após executar o modelo obtiveram-se as previsões para o risco de crédito.

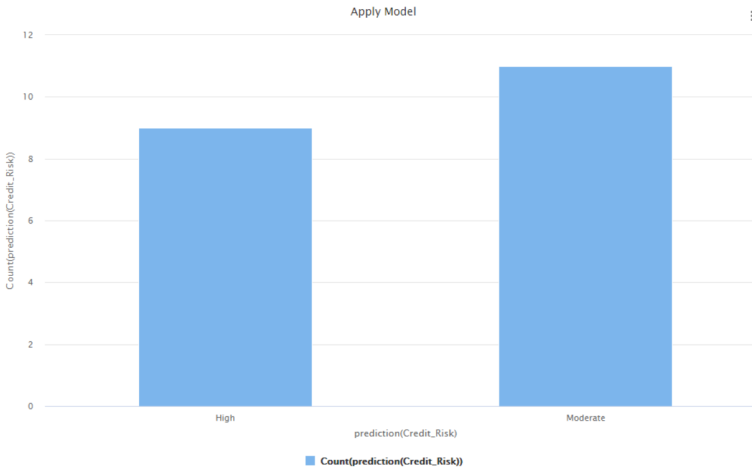

Figura 8: Histograma com as previsões feitas.

As previsões geradas apenas apresentam risco de crédito dos tipos *High* e *Moderate*. Contudo os resultados não são inesperados. Isto porque se se analisar os dados com algum cuidado poder-se-á ver que os mesmo são um tanto equilibrados. Ou seja, por exemplo, apesar de o sujeito ter um *credit score* relativamente baixo por outro lado tem bastante meses de trabalho e, alguns casos acontece também o contrário.

Sem grande conhecimento do domínio da área, logicamente e com facilidade se poderá afirmar que um indivíduo sem muitos meses de trabalho e com o número elevado de pagamentos em atraso poderá ter um *credit score* elevado. O *credit score* é uma variável dependente de outras e gerado através de um cálculo, contudo, no dataset elaborado não

se tiveram em conta estes pormenores da área de estudo. Por isto é que os valores poderão não fazer sentido quando vistos pela primeira vez.

Row No.	Applicant_ID	prediction(C...	confidence(...	confidence(...	confidence(...	confidence(...	confidence(...	Credit_Score
1	1	High	0.001	0.993	0.000	0.005	0.001	341
2	2	High	0.023	0.975	0.000	0.002	0.000	301
3	3	High	0.001	0.993	0.000	0.005	0.001	278
4	4	High	0.001	0.993	0.000	0.005	0.001	457
5	5	Moderate	0.997	0.000	0.001	0.002	0.000	590
6	6	High	0.001	0.993	0.000	0.005	0.001	409
7	7	Moderate	0.635	0.340	0.003	0.013	0.009	671
8	8	Moderate	0.873	0.065	0.041	0.010	0.011	705
9	9	Moderate	0.998	0.000	0.001	0.001	0.000	491
10	10	High	0.001	0.993	0.000	0.005	0.001	443
11	11	High	0.001	0.993	0.000	0.005	0.001	368
12	12	Moderate	0.574	0.000	0.000	0.426	0.000	490
13	13	Moderate	0.998	0.000	0.001	0.001	0.000	521
14	14	Moderate	0.876	0.116	0.000	0.005	0.002	545
15	15	Moderate	0.638	0.351	0.000	0.008	0.003	399
16	16	Moderate	0.998	0.000	0.001	0.001	0.000	415
17	17	High	0.001	0.993	0.000	0.005	0.001	427

Figura 9: Previsões.