



Licenciatura em Engenharia de Sistemas Informáticos

No âmbito da UC de Inteligência Artificial

Financiamento de Projetos

Machine Learning

André Cardoso & Leonel Fernandes
18848 & 18850

Barcelos, 24 de janeiro de 2022

Contents

Introdução	2
Problema	2
Metodologia	3
Resultados e Discussão	9
Adversidades	12
Conclusão	12

Introdução

No Âmbito da Unidade Curricular de Inteligência Artificial foi-nos proposta a resolução de um problema de *Machine Learning* como segundo trabalho prático.

Foi-nos fornecido um conjunto de dados a partir dos quais teremos de desenvolver formas de resolução do problema apresentado.

O problema trata-se da avaliação de um conjunto de dados sobre diversos projetos, com os quais o modelo de aprendizagem terá de efetuar uma previsão e classificar numa determinada classe. A classe referida é a classe de financiamento, um conjunto de dados pode ser classificado como financiado ou não.

Para abordar este problema foi decidido fazer o tratamento prévio dos dados de forma a adequá-los à avaliação por parte do modelo, de seguida são utilizados vários tipos de modelos de aprendizagem e avaliados os seus resultados.

Será apresentada uma análise estatística dos dados, assim como dos diversos modelos, para ser avaliada qual a melhor opção tomar.

Problema

O problema deste projeto de *Machine Learning* é avaliação de dados e subsequente categorização dos mesmos.

Como referido anteriormente, os dados para resolução deste problema foram fornecidos como forma de um *dataset* onde estão presentes informações sobre projetos. Os projetos podem ser classificados como financiados ou não, através de uma variável presente no dataset chamada "*funded*". Os nossos modelos terão como objetivo tentar descobrir se um conjuntos de dados sobre um projeto o torna financiado ou não.

Terão de ser efetuadas alterações no *dataset* original de forma a preparar os dados para serem avaliados pelos modelos. Podem ser removidas algumas variáveis desnecessárias, ou até mesmo alguns exemplos de dados que não estejam de acordo com o necessário para avaliação.

Nome	Tipo	Nome	Tipo
Type	categórica	Pledge Levels	numérica
Has FB	categórica	Min Pledge Tiers	numérica
Backed Projects	numérica	Max Pledge Tiers	numérica
Previous Projects	numérica	Proj Desc Len	numérica
Creator Desc Len	numérica	Images	numérica
Title Len	numérica	Videos	numérica
Goal	numérica	Has Video	categórica
Duration	numérica	Funded	categórica

Figura 1: Variáveis do *dataset*

Na tabela acima estão presentes todas as variáveis presentes no *dataset*. Terão de ser utilizados diversos modelos de aprendizagem nestas variáveis, através de avaliação estatística obteremos as suas precisões no problema apresentado, para assim ser escolhido o melhor.

Metodologia

Para resolução do problema em mãos seria necessário empregar algumas estratégias, começando pelo tratamento dos dados fornecidos no *dataset*.

Inicialmente decidimos fazer o treino do modelo sem qualquer alteração ao *dataset* original, simplesmente fizemos as ligações necessárias para obter um resultado.

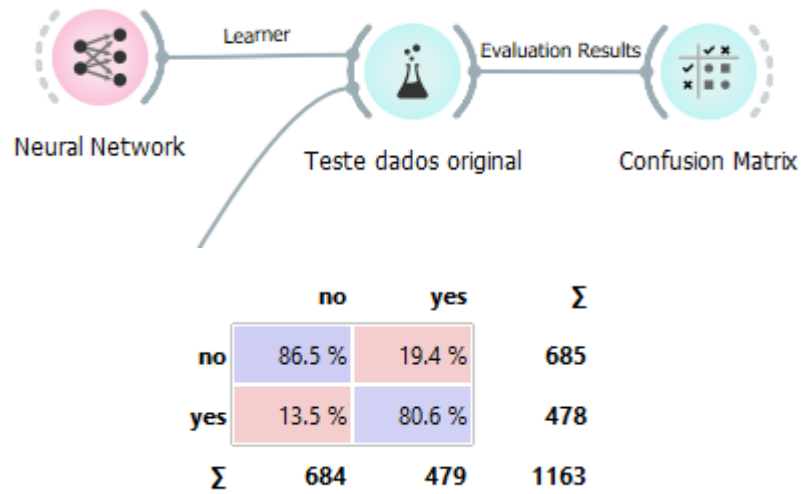


Figura 2: Teste *Dataset* original

Conseguimos verificar que existe uma disparidade considerável na percentagem de acertos para "no" e para "yes", o que nos levou a fazer algumas verificações nos dados.

Após uma breve verificação dos dados conseguimos chegar à conclusão que estes não se encontram simétricos, isto é, a distribuição da variável "*Funded*" é superior para o caso de "*no*", isto tornaria os resultados dos modelos desequilibrados.

		Funded		
		no	yes	
Funded	Count	no	yes	Total
	no	685.0	0.0	685.0
	yes	0.0	478.0	478.0
Total		685.0	478.0	1163.0

Figura 3: Distribuição da variável "*Funded*"

Como tal, decidimos utilizar uma amostra de 70% dos dados de "*no*", cerca de 480, um número bastante mais próximo dos 478 dos "*yes*". Assim sendo, a amostra final não teria 1163 dados, mas sim 968 após a remoção.

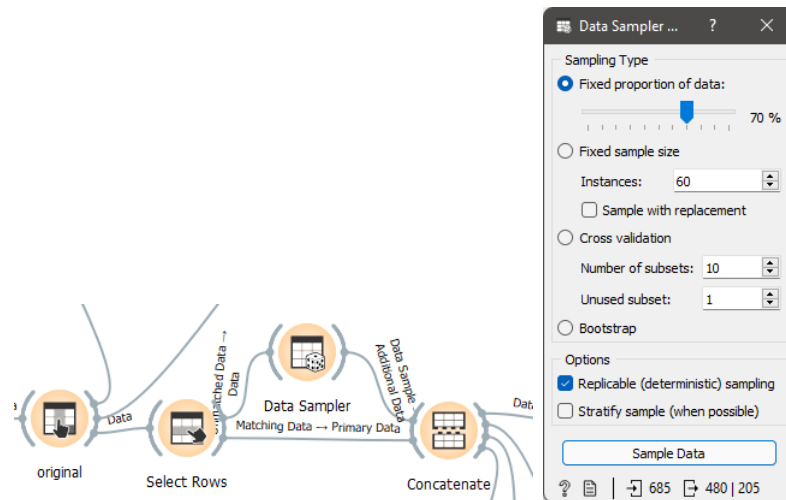


Figura 4: Distribuição dos Dados

Depois de distribuídos igualmente foi feita a normalização dos dados, para todos os valores numéricos estarem no intervalo entre $[0, 1]$. Foram ainda transformados todos os valores categóricos em numéricos.

Foram de seguida adicionadas duas formas de treino e teste, "Test on Train Data" e "Test on Test Data". Para o "Teste on Train Data" foram testados os dados de treino e para o "Test on Test Data" foram testados 30% dos dados originais que foram inseridos especificamente para treino, esta divisão foi feita através de um "Data Sampler".

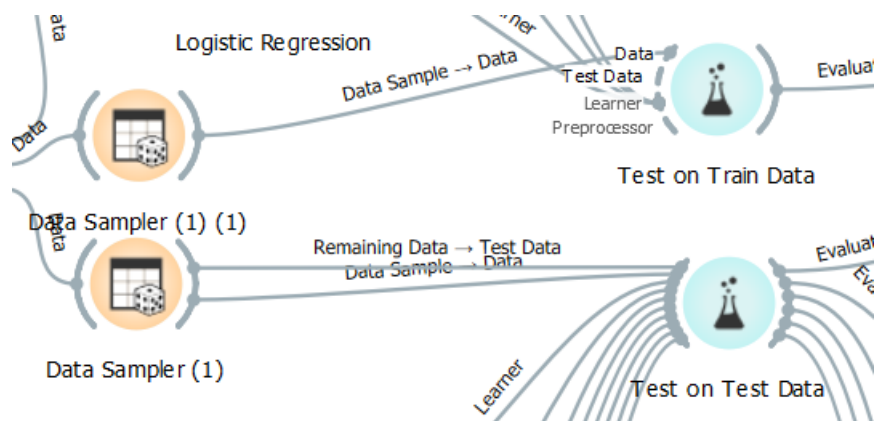


Figura 5: Treino dos modelos

Os dados para o "Test on Test Data" foram divididos como 70% dos dados para treino e os restantes para teste, já no "Test on Train Data" apenas os 70% foram utilizados, sendo que foram usados para treino e para teste.

Foram utilizados vários tipos de modelos de aprendizagem, os modelos utilizados foram os seguintes:

- Logistic Regression
- Neural Network
- Stochastic Gradient Descent
- Tree
- Random Forest
- Naive Bayes
- AdaBoost
- Gradient Boosting

Após obtenção dos resultados de teste dos modelos foram selecionados os Verdadeiros e Falsos Positivos e Negativos, estes foram separados e analisados através do nodo de estatística. Isto foi efetuado para verificar a existência de semelhanças entre Verdadeiros e Falsos Positivos ou Verdadeiros e Falsos Negativos.

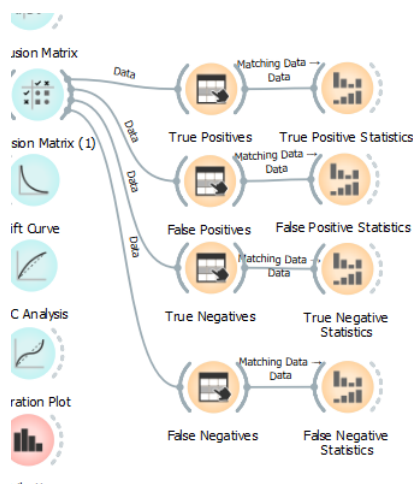


Figura 6: Estatísticas dos Dados

Depois de feita a ligação comparações foram feitas entre os dados, pelo que foram encontradas algumas semelhanças em algumas variáveis.

	Name	Distribution	Mean	Median	Dispersion	Min.	Max.	Missing
N	Backed Projects		0.56	0	3.72	0	18	0 (0%)
N	Previous Projects		0.34	0	2.43	0	4	0 (0%)
N	Max Pledge Tiers		1112.92	141	1.96	1	10000	0 (0%)
N	Proj Desc Len		3245.12	2229	0.93	440	20032	0 (0%)
N	Images		3.67	0	1.76	0	37	0 (0%)

Figura 7: Estatísticas dos Verdadeiros Negativos





	Name	Distribution	Mean	Median	Dispersion	Min.	Max.	Missing
N	Previous Projects		0.32	0	2.58	0	3	0 (0%)
N	Max Pledge Tiers		2214.86	171.50	1.72	7	10000	0 (0%)
N	Proj Desc Len		2998.86	2231	0.66	191	7855	0 (0%)
N	Images		7.73	3	1.50	0	40	0 (0%)

Figura 8: Estatística dos Falsos Negativos

Tanto "Previous Projects" como "Proj Desc Len" encontram-se com valores estatísticos semelhantes. Existiam outras variáveis com valores estatísticos semelhantes, pelo que todas essas variáveis foram removidas individualmente e foram verificados os resultados dos modelos para existência de alterações, pelo que foi verificado que a remoção dessas variáveis apenas afetava negativamente os resultados dos modelos.

Para finalizar o processo foram alterados os valores dos hiperparâmetros dos modelos de forma a conseguir melhores resultados.

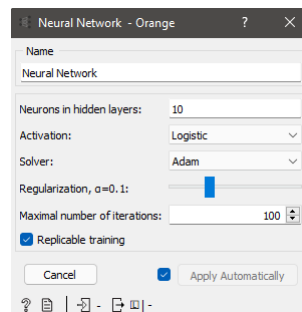


Figura 9: Hiperparâmetros da Rede Neural

Resultados e Discussão

Como referido anteriormente, foram utilizados dois tipos de teste, teste nos dados de treino e teste nos dados de teste. Ambos utilizam 70% dos dados originais. Mas no caso do teste em dados de teste, 30% dos dados são utilizados para teste.

Após o ajuste de hiperparâmetros, os melhores resultados obtidos para o teste nos dados de teste foram os seguintes:

Modelo	AUC	CA	F1	Precision	Recall
Tree	0.849	0.808	0.809	0.811	0.808
SGD	0.904	0.857	0.857	0.857	0.857
Random Forest	0.920	0.847	0.847	0.850	0.847
Neural Network	0.904	0.868	0.867	0.868	0.868
Naive Bayes	0.895	0.812	0.812	0.815	0.812
Logistic Regression	0.905	0.850	0.850	0.851	0.850
Gradient Boosting	0.916	0.819	0.819	0.820	0.819
AdaBoost	0.768	0.767	0.767	0.769	0.767

Figura 10: Resultados do Teste em dados de teste

Os modelos onde foram obtidos melhores resultados foram "SGD", "Random Forest", "Neural Network" e "Logistic Regression", sendo que o modelo onde se obteve mais precisão foi o "Neural Network". Podemos ainda verificar que "Random Forest" foi o modelo onde "AUC" é mais elevado, apesar da precisão mais baixa.

Podemos verificar através do seguinte gráfico uma comparação entre o "ROC" do "Neural Network" e do "Random Forest", estes resultados são para o target "no":

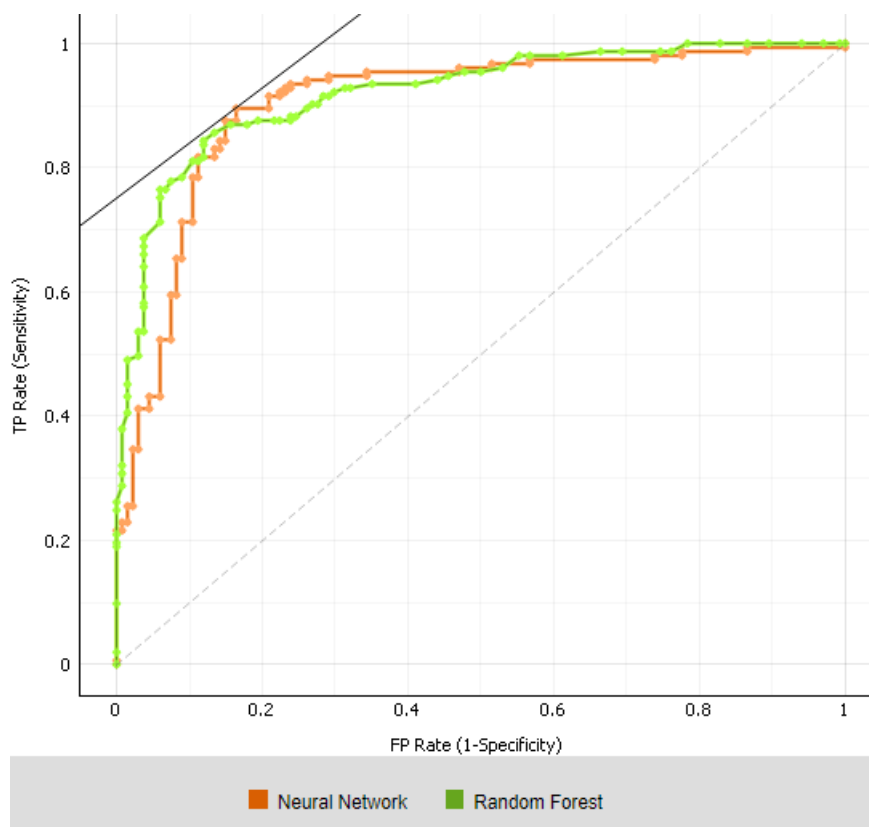


Figura 11: Comparação de ROC

Verificamos através deste gráfico que os resultados do "Random Forest" são melhores para valores de especificidade mais baixos. No entanto são melhores no "Neural Network" para valores mais elevados.

Nos resultados do teste nos dados de treino foram obtidos alguns resultados interessantes:

Modelo	AUC	CA	F1	Precision	Recall
Tree	0.874	0.855	0.855	0.858	0.855
SGD	0.915	0.833	0.833	0.833	0.833
Random Forest	1.000	1.000	1.000	1.000	1.000
Neural Network	0.912	0.832	0.832	0.832	0.832
Naive Bayes	0.892	0.820	0.819	0.821	0.820
Logistic Regression	0.908	0.826	0.826	0.826	0.826
Gradient Boosting	0.982	0.933	0.933	0.934	0.933
AdaBoost	1.000	1.000	1.000	1.000	1.000

Figura 12: Resultados do Teste em dados de treino

Podemos verificar que existem dois modelos, "Random Forest" e "AdaBoost", onde os valores são perfeitos. Temos ainda o "Gradient Boost" onde podemos observar valores também elevados.

Estes resultados, quando comparados com os resultados do teste em dados de teste podem parecer invulgares, no entanto, o mais provável é que se trate de *overfitting*. O modelo está demasiado ajustado aos dados de treino o que causa resultados medíocres quando recebe dados diferentes.

Conseguimos ainda verificar outro problema, os modelos, com a exceção dos 3 acima referidos, todos verificam precisões abaixo dos 90% quando testados com dados de treino, isto leva a querer que a quantidade de dados presente no *dataset* não é suficiente para treinar o modelo de forma eficiente.

Com isto tiramos duas conclusões, os modelos que apresentam precisão superior a 90% nos dados de treino apresentam *overfitting* e os restantes não têm dados suficientes para fazer um bom treino do modelo.

Adversidades

Durante o desenvolvimento deste projeto encontramos vários obstáculos que tivemos a necessidade de ultrapassar como:

- A disparidade dos dados em que para tal foi necessário selecionar 70% das linhas com valor ‘no’ para a variável *Funded*.
- A possível existência de overfitting pelo modelo ‘Neural Network’ que revelou menor pontuação para os dados de treino em comparação com os dados de teste.
- A dificuldade geral de ultrapassar uma precisão de 90% ao testar com dados de treino.

Consideramos que alguns dos problemas encontrados poderiam ser resolvidos com uma amostra de dados mais elevada.

Conclusão

Este trabalho revelou-se extremamente útil para a descoberta do ‘work-flow’ de machine learning, foi uma experiência diferente dos restantes projetos que temos vindo a desenvolver ao longo do curso.

Em suma, considerando o desafio proposto, acreditamos ter alcançado o objetivo esperado: uma análise cuidada dos resultados obtidos pelos modelos de aprendizagem de modo a compreender e ajustar os modelos em mãos, tentando alcançar a melhor precisão possível.