Aplicação e Avaliação de Algoritmos de Classificação

Guilherme A. A. D. Nascimento[1], Pedro Lucas Damasceno[1], Gabriel C. F. Oliveira[1], Rafael A. F. Oliveira[1], Robson N. Lobão[1]

[1]Universidade Federal de Ouro Preto (UFOP)
Ouro Preto – MG – Brazil

{guilherme.drummond, pedro.damasceno, gabriel.catizani, rafael.afo, robson.lobao}@aluno.ufop.edu.br

Abstract. *This report describes the experiments carried out to train the most appropriate classification model to determine the bankruptcy or not of the companies provided for evaluation 2 of the Artificial Intelligence course. The models tested were Logistic Regression, Decision Tree, and Random Forest.*

Resumo. *Este relatório descreve os experimentos realizados para treinar o modelo de classificação mais apropriado para definir a falência ou não das empresas fornecidas para a avaliação 2 da disciplina de Inteligência Artificial. Os modelos testados foram: Regressão Logística, Árvore de Decisão e Floresta Aleatória.*

1. Introdução

A partir de uma base de dados com informações rotuladas de 400 empresas, o modelo deve predizer a falência (1) ou não (0) de uma nova lista contendo 100 empresas. Utilizando os métodos de Regressão Logística, Árvore de Decisão e Floresta Aleatória, permutamos os parâmetros de cada método para buscar a configuração que melhor se encaixa à base de dados proposta. Uma vez encontrada a melhor configuração, analisamos a base de dados e fizemos modificações nos dados que julgamos promissoras para a classificação.

2. Metodologia de Testes

Para eleger o melhor modelo para o contexto do problema, pesquisamos os parâmetros de cada um e geramos todas as combinações que julgamos razoáveis. Para a validação cruzada, testamos o KFold e KFold estratificado, com partições de tamanho 10 e 20. O output dos testes pode ser consultado no diretório 'resultados'.

2.1. Regressão Logística

No modelo de regressão logística, testamos os solvers 'lbfgs', 'liblinear', 'newton-cg', 'newton-cholesky', 'sag' e 'saga'. As penalidades foram combinadas de acordo com cada solver, sendo elas 'l1', 'l2' e None. A penalidade 'elasticnet' foi desconsiderada devido a erros de incompatibilidade de versões.

Para o modelo em questão, determinamos que a melhor configuração consiste em: solver 'sag', penalidade 'l2' e KFold estratificado de 10 splits. A média de f1-measure de todo o conjunto de testes para o modelo foi de 0.4419, com desvio padrão 0.0496.

2.2. Árvore de Decisão

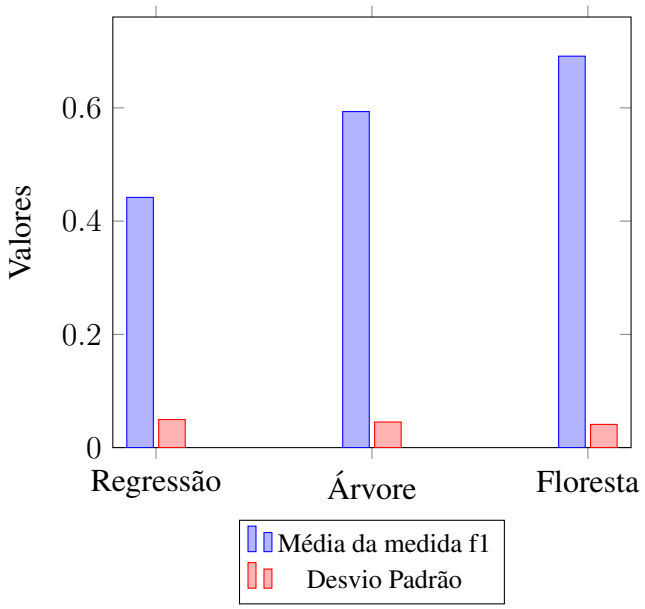
Para o modelo de árvore de decisão, testamos os critérios 'gini', 'entropy' e 'log_loss'. Para os parâmetros de splitter, utilizamos as opções 'best' e 'random', e para a profundidade máxima da árvore, estimamos 10 ou 20.

Nos testes realizados, observamos que o critério 'gini', com splitter 'best', profundidade máxima de 5 e KFold estratificado com 20 splits foi a configuração que produziu os melhores resultados. A média de f1-measure de todo o conjunto de testes para o modelo foi de 0.5934, com desvio padrão 0.0452.

2.3. Floresta Aleatória

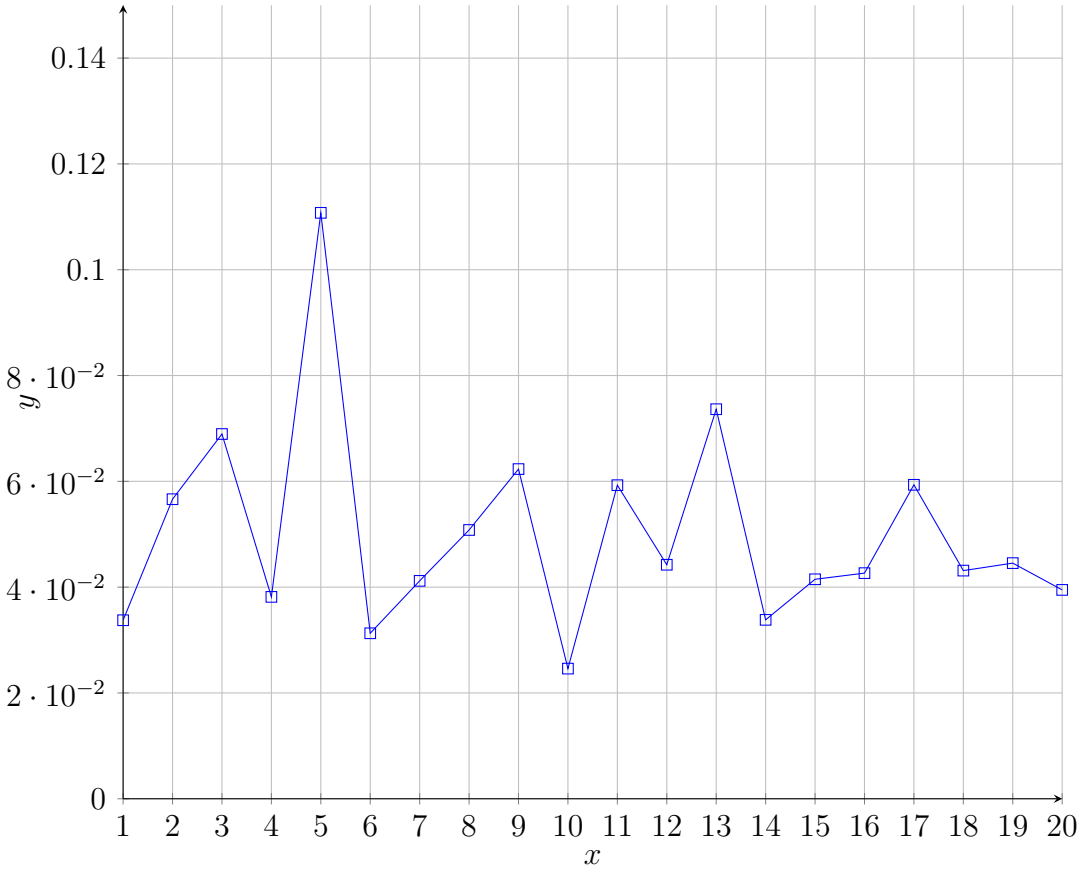
Para o modelo de floresta aleatória, variamos a quantidade de árvores entre 100 e 150. Entre os critérios, foram testados 'gini'e 'log_loss'. A profundidade máxima de cada árvore variou entre 10 e 15. Devido ao alto tempo computacional, restringimos os testes do modelo à metade.

Através dos testes, concluímos que os melhores resultados foram obtidos através da configuração com 150 árvores, critério de log_loss, profundidade máxima de 15 e KFold estratificado com 20 splits. A média de f1-measure de todo o conjunto de testes para o modelo foi de 0.6913, com desvio padrão 0.0410.



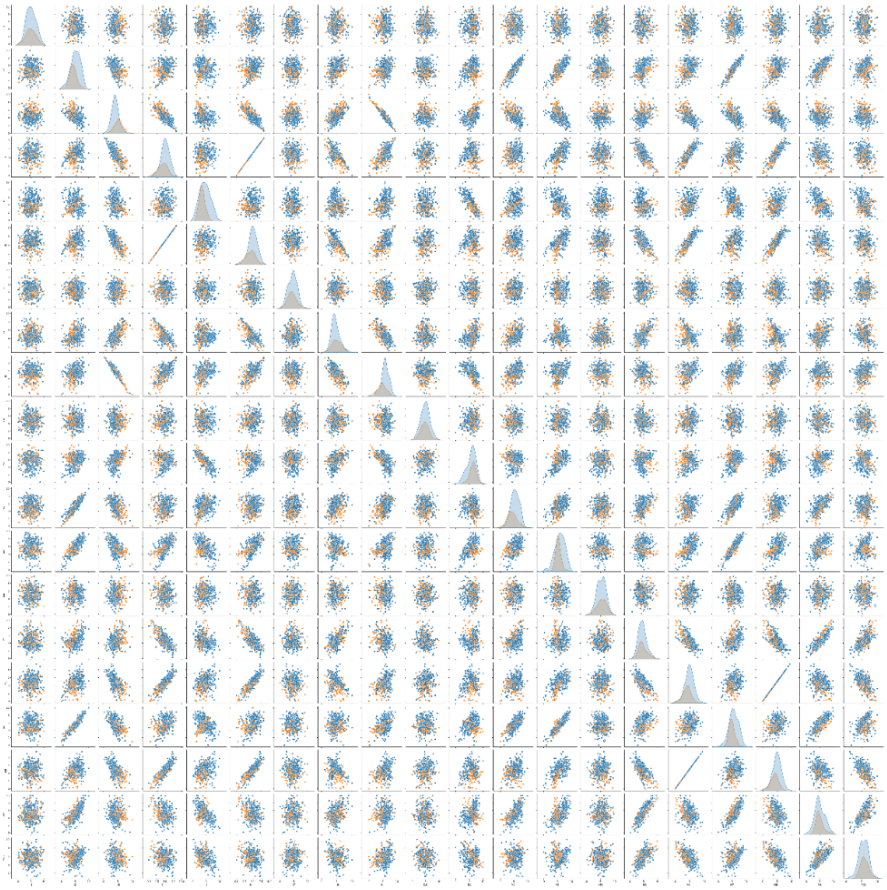
3. Subconjuntos de Atributos

Utilizamos o método de pairplot da biblioteca seaborn para visualizar a distribuição dos atributos do conjunto de dados. Dessa forma, observamos que alguns atributos possuem relação diretamente proporcional. Além disso, utilizamos a função de feature_importances para calcular a importância de cada atributo na tarefa de classificação do modelo.



A partir do gráfico acima, concluímos que alguns atributos, como o 5 e 13, são muito relevantes para a classificação do modelo. Já os atributos 10 e 14 possuem baixa relevância. Para fins de teste, removemos da classificação os atributos com relevância inferior a 0.04. Todavia, os resultados foram inexpressivos e até piores em relação ao conjunto completo.

Utilizamos o pairplot para visualizar a relação dos atributos e testamos a remoção daqueles cuja relação se mostrou linear.



Observando a imagem acima, pode-se notar que os atributos I4, I6, M6, M8 e alguns outros possuem a relação de linearidade mencionada anteriormente. Alguns desses atributos já haviam sido desconsiderados na análise anterior, pois apresentaram baixa relevância para a classificação do modelo. Removemos os atributos em questão e os resultados foram, novamente, inexpressivos.

4. Conclusão

Diante dos dados obtidos, concluímos que o modelo de floresta aleatória é o mais adequado para o problema, haja vista que apresentou os maiores índices de f1-measure e menor desvio padrão. Com base nos testes efetuados com a remoção dos atributos menos relevantes e/ou lineares, optamos por remover os atributos {I2, I3, I4, I6, M6, M7 e M9}. O subconjunto restante foi {I1, I5, I7, I8, I9, I10, M1, M2, M3, M4, M5, M8, M10}. Essa escolha de atributos não foi aleatória, mas arbitrária de acordo com quais linhas possuiam menos correlação com os outros atributos de acordo com o pairplot.

Uma validação cruzada do modelo utilizando f1 e f1-weighted foi realizada apenas para a submissão. A média de f1 foi 0.7174, com desvio padrão de 0.0075. E para f1-weighted, a média foi 0.8293 e o desvio padrão 0.0043. O resultado final está contido no arquivo 'submissao.csv'.