Vinicius Ferreira Santos
Data Science para Negócios
03/06/2018

Identificando Pessoas de Interesse da Enron

1. Summarize for us the goal of this project and how machine learning is useful in trying to accomplish it. As part of your answer, give some background on the dataset and how it can be used to answer the project question. Were there any outliers in the data when you got it, and how did you handle those?

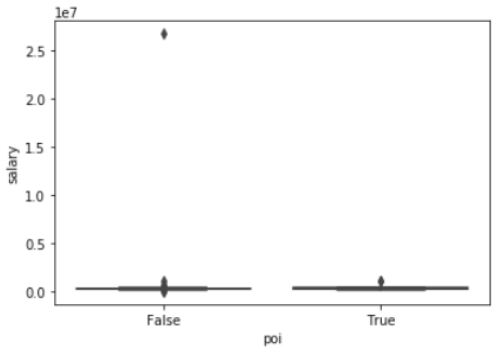
   O objetivo desse projeto é identificar as pessoas de interesse(POI) no caso de fraude da Enron. O projeto usa técnicas de Machine Learning para criar um modelo preditivo, que utiliza dados financeiros e de e-mails de funcionários investigados, o que significa que eles foram indiciados, fecharam acordos com o governo, ou testemunharam em troca de imunidade no processo.

   Na exploração dos dados, encontrei 146 funcionários, dos quais 18 eram POI. Cada registro do dataset possui 22 features, com os detalhes abaixo:

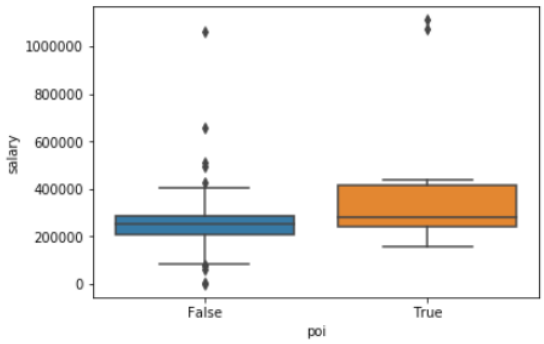
Feature	Tipo	Total non-Null	Total Null
nome	Object	146	0
bonus	Float64	82	64
deferral_payments	Float64	39	107
deferred_income	Float64	49	97
director_fees	Float64	17	129
email_address	Object	111	35
exercised_stock_options	Float64	102	44
expenses	Float64	95	51
from_messages	Float64	86	60
from_poi_to_this_person	Float64	86	60
from_this_person_to_poi	Float64	86	60
loan_advances	Float64	4	142
long_term_incentive	Float64	66	80
other	Float64	93	53
poi	Bool	146	0
restricted_stock	Float64	110	36
restricted_stock_deferred	Float64	18	128
salary	Float64	95	51

shared_receipt_with_poi	Float64	86	60
to_messages	Float64	86	60
total_payments	Float64	125	21
total_stock_value	Float64	126	20

Ao analisar os dados, foram encontrados alguns outliers interessantes que foram prontamente removidos, entre eles, está o funcionário *LOCKHART EUGENE E* que não possui dados em nenhuma feature de e-mail e financeira. Outro ruído encontrado foi um funcionário com nome de empresa *THE TRAVEL AGENCY IN THE PARK*. Por último, ao observar graficamente salário dos funcionários, me deparei com um outlier aberrante de $ 26.704.229 para um funcionário com nome *TOTAL,* que mais parecia ser o totalizador dos salários, causado por algum erro na criação do dataset.



Salários com Outlier



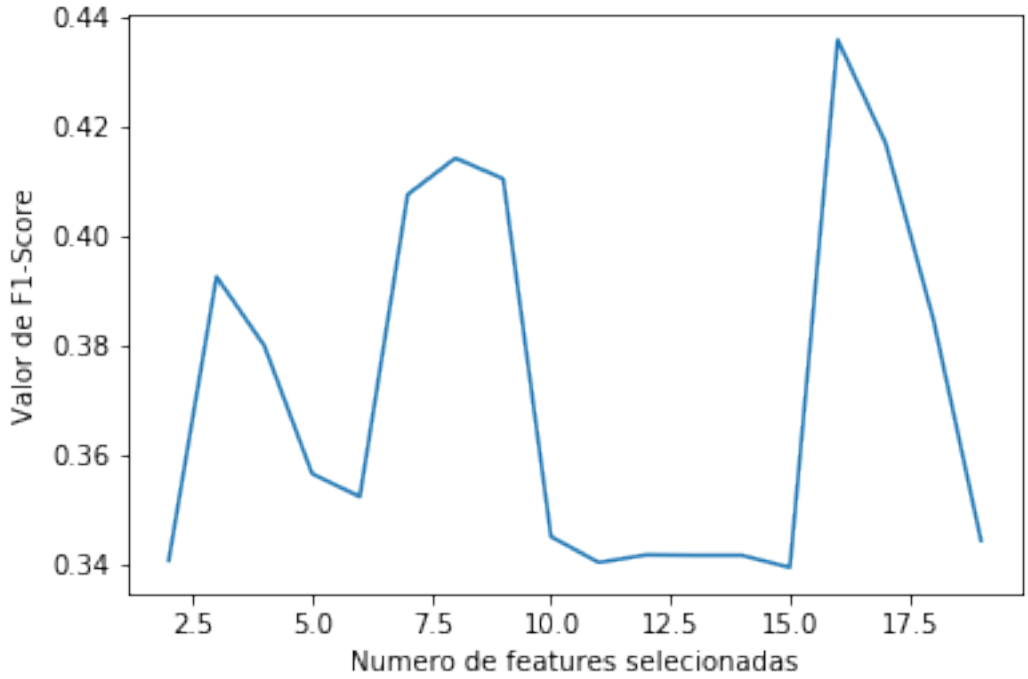
Salários sem Outlier

2. What features did you end up using in your POI identifier, and what selection process did you use to pick them? Did you have to do any scaling? Why or why not? As part of the assignment, you should attempt to engineer your own feature that does not come ready-made in the dataset -- explain what feature you tried to make, and the rationale behind it. (You do not necessarily have to use it in the final analysis, only engineer and test it.) In your feature selection step, if you used an algorithm like a decision tree, please also give the feature importances of the features that you use, and if you used an automated feature selection function like SelectKBest, please report the feature scores and reasons for your choice of parameter values.

Para selecionar as features do modelo final, inicialmente foi realizado um processo de engenharia para aumentar a probabilidade de identificar um POI. Como resultado da engenharia, surgiram três novas features:

- **fraction_to_poi**: que é a fração de e-mails enviados para um POI(from_poi_to_this_person/to_messages).
- **fraction_from_poi**: que é a fração de e-mails que foram recebidos de um POI(from_this_person_to_poi/from_messages)
- **income**: que é o rendimento de um funcionário(salary+bônus).

Após a engenharia de features, foi utilizado o SelectKBest para indicar a melhor quantidade de features que serão utilizadas no modelo final. Para a escolha do número final, foi observado a variação do Score F1. Ao final do processo, como podemos perceber no gráfico abaixo, o valor ideal definido são17 features:



Foi utilizado MinMaxScaler para definir o menor valor para 0 e o maior como 1 para todas as features. Foi necessário escalar as features, devido a quantidade de outliers que poderiam prejudicar o modelo sendo escolhidos como preditores principais. Outro motivo importante para o escalonamento, é que as features com valores ausentes foram preenchidos com 0 e o MinMaxScaler define como 0 os menores valores.

3. What algorithm did you end up using? What other one(s) did you try? How did model performance differ between algorithms?

O modelo com o melhor resultado de validação utilizava Adaboost com uma DecisionTree balanceada. Outros algoritmos testados foram DecisionTree e RandonForest. A DecisionTree padrão teve resultados bem próximos aos da Adabost, e a RandonForest, que gera uma série de DecisionTree, teve o pior resultado dos três algoritmos. Os testes iniciais ocorreram com os algoritmos sem hiperparâmetros, sendo escolhido o de melhor validação.

Algoritmo	**Accuracy**	**Precision**	**Recall**	**F1**	**Tempo**
DecisionTree	0.81400	0.29744	0.29000	0.29367	0.108s

RandonForest	0.8646	0.48000	0.18000	0.26182	2.489s
Adaboost	0.84000	0.36111	0.26000	0.30233	12.362s

4. What does it mean to tune the parameters of an algorithm, and what can happen if you don't do this well? How did you tune the parameters of your particular algorithm? What parameters did you tune? (Some algorithms do not have parameters that you need to tune -- if this is the case for the one you picked, identify and briefly explain how you would have done it for the model that was not your final choice or a different model that does utilize parameter tuning, e.g. a decision tree classifier).

   Tunar um algoritmo se refere ao processo de buscar o melhor ajuste para obter os melhores resultados. É preciso tomar bastante cuidado ao tunar um algoritmo, porque é fácil causar overfitting, fazendo com que ele não generalize bem para novos dados.

   No modelo final que selecionei, o tuning foi realizado com o GridSearchCV que a partir de um conjunto de parâmetros escolhidos, realiza um cruzamento com todas as combinações possíveis para encontrar o melhor ajuste dos hiperparâmetros. O melhor ajuste encontrado pelo GridSearchCV para a Adaboost possuía learning_rate=0.1, n_estimators=50 e uma DecisionTreeClassifier com os seguintes parâmetros: class_weight='balanced', max_depth = 1, min_samples_leaf=2 e n_estimators=50

5. What is validation, and what's a classic mistake you can make if you do it wrong? How did you validate your analysis?

   Validação é o processo de verificar o qual bem o seu modelo irá generalizar para novos dados. Um erro clássico é treinar o modelo com todos os dados, fazendo com que o modelo memorize a classificação e não aprenda a generalizar.

   A melhor estratégia para generalizar bem os dados é dividir os dados em dois conjuntos: conjunto de testes e conjunto de treinamento, onde treina o modelo com o conjunto de treinamento e valida com o conjunto de teste.

   Para validar meu modelo final, foi utilizado validação cruzada com StratifiedShuffleSplit devido ao tamanho do dataset ser muito pequeno, com 146 registros, dos quais 18 são POI.

6. Give at least 2 evaluation metrics and your average performance for each of them. Explain an interpretation of your metrics that says something human-understandable about your algorithm's performance.

   O modelo final selecionado possui precision de 34%, Recall de 54% e F1 de aproximadamente 48%. Precision significa que em 34% dos casos, o modelo classifica os funcionários como POI. O recall significa que para todos os casos de POI reais, o modelo classifica 54% como POI. O F1 é a média harmônica de precision e recall onde F1= 2 * (precision * recall) / (precision + recall).