# **Nanodegree Engenheiro de Machine Learning**

1. **– Definição**

**Visão geral do Projeto**

Ao longo desses anos o telemarketing sofreu uma série de mudanças, ou melhor, transformações que estão relacionadas diretamente com a evolução da tecnologia. Surgiram conceitos como Omnichannel (multicanal), Callback, Atendentes Virtuais, Chatbot, Portais de Autoatendimento e etc.

O cliente mudou, atualmente ele está no controle, não quer esperar ‘pendurado’ na linha e também não quer ser importunado por ligações oferecendo produtos que não são do seu interesse. Ele pode a qualquer momento deixar de comprar na sua loja, por exemplo, e escolher o mesmo produto em outra loja utilizando qualquer meio (ex: smartphone, tablet). Por outro lado, existe um potencial público de clientes que pode contratar o produto, desde que, seja realizado um contato e ofertado o produto adequado.

Nesse cenário é que o projeto se enquadra, utilizando machine learning para selecionar o público que será ofertada a Campanha. O modelo de machine learning treinado e testado conseguirá prever quais são os clientes mais propensos a contratar um empréstimo bancário.

O conjunto de dados utilizado para treinar o modelo é referente a uma campanha de marketing realizada por uma institução bancária portuguesa. Essa campanha foi baseada em chamadas telefonicas. O produto ofertado foi um empréstimo a prazo.

Esse conjunto de dados é público, disponivel para pesquisa. Os detalhes estão descritos em [Moro et al., 2014]. O conjunto foi obtido no site da UCI (https://archive.ics.uci.edu/ml/datasets/bank+marketing)

[Moro et al., 2014] S. Moro, P. Cortez and P. Rita. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems, Elsevier, 62:22-31, June 2014

A solução para esse problema é a execução de um modelo de aprendizagem supervisionado com esse conjunto de dados. Será necessário efetuar uma um processo de preparação dos dados. Esse modelo será treinado com esse conjunto (já preparado), gerando no final do seu processamento, um modelo matemático que consegue prever (a partir dos dados do cliente) se o cliente irá contratar o produto/serviço bancário.

Será feito um split do conjunto de dados, usando 80% para aprendizagem (treinamento) e 20% para testes. Os resultados dos testes serão comparados com os dados reais e dessa forma será possível medir a efetividade do modelo gerado.

A partir desse momento, nas próximas campanhas, a seleção do público será feita apresentando o novo conjunto para o modelo já treinado. Ao término da execução do modelo teremos a informação de quais clientes provavelmente contratarão o produto.

**Descrição do Problema**

O problema que estamos propondo solucionar é selecionar de forma mais assertiva o público que será ofertada campanhas de marketing do produto empréstimo a prazo de uma instituição bancária. A seleção não assertiva implica um alto gasto (estudo e seleção do público, de tempo, em pessoas, em telefonia, etc) contatando clientes que não tem interesse em contratar o produto.

Dessa formar será dado um foco nos clientes que realmente desejam contratar o produto, com isso a instituição financeira poderá investir mais ‘energia’ em melhor atender e oferecer serviços mais atraentes aos seus clientes.

**Métricas**

As métricas utilizadas serão (do Pacote Scikit.metrics):

accuracy\_score (acurácia): Proporção de casos que foram corretamente previstos, sejam eles verdadeiro positivo ou verdadeiro negativo. Com isso, mede a frequência que o modelo faz a previsão correta.

f1\_score (score): Média ponderada dos scores de precisão (precision) e sensibilidade (recall). Varia entre 0 e 1, sendo 1 a melhor pontuação possível.

precision\_score (precisão): Proporção de casos positivos que o modelo classificou como positivo e eram positivas, ou seja, é a capacidade de não rotular como positiva uma amostra que é negativa (Verdadeiro Positivo / (Verdadeiro Positivo + Falso Positivo))

recall\_score (sensibilidade): Proporção de casos realmente positivos e que foram classificados pelo modelo como positivo, ou seja, é a capacidade do modelo de encontrar todas as amostras positivas. ((Verdadeiro Positivo / (Verdadeiro Positivo + Falso Negativo))

- [confusion\_matrix](https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html#sklearn.metrics.confusion_matrix) (Matriz de confusão): Essa métrica calcula a quantidade de falso positivo e falso negativo, e de verdadeiro positivo e verdadeiro negativo

**2 – Análise**

**Exploração de Dados**

Primeiramente, segue abaixo as informações de cada feature do conjunto:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Label** | **Descrição** | **Tipo** | **Dominio** |
| 1 | Age | Idade | Numérica |  |
| 2 | Job | Tipo de Emprego | Categórica | admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown' |
| 3 | Marital | Estado Civil | Categórica | divorced','married','single','unknown'; note: 'divorced' means divorced or widowed |
| 4 | Education | Escolaridade | Categórica | basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown' |
| 5 | Default | Tem Crédito Default | Categórica | no','yes','unknown' |
| 6 | Housing | Tem Empréstimo Habitacional | Categórica | no','yes','unknown' |
| 7 | Loan | Tem Empréstimo Pessoal | Categórica | no','yes','unknown' |
| 8 | Contact | Tipo de Contato | Categórica | cellular','telephone' |
| 9 | Month | Mês do último contato | Categórica | jan', 'feb', 'mar', ..., 'nov', 'dec' |
| 10 | day\_of\_week | Último dia de contato da semana | Categórica | 'mon','tue','wed','thu','fri' |
| 11 | Duration | Duração do último contato em segundos | Numérica | Nota importante: este atributo afeta altamente a meta de saída (por exemplo, se a duração for = 0, então y = 'não'). No entanto, a duração não é conhecida antes de uma chamada ser executada. Além disso, após o término da chamada, é obviamente conhecido. Assim, essa entrada deve ser incluída apenas para fins de benchmark e deve ser descartada se a intenção for ter um modelo preditivo realista. |
| 12 | campaign | Número de contatos da última camapanha | Numérica |  |
| 13 | Pdays | Número de dias que se passaram depois que o cliente foi contatado pela última vez de uma campanha anterior. 999 indica que cliente não foi contato em campanha anterior | Numérica |  |
| 14 | Previous | Número de contatos realizados antes desta campanha | Numérica |  |
| 15 | poutcome | Resultado da campanha de marketing anterior | Categorica | failure','nonexistent','success' |
| 16 | emp.var.rate | Taxa de variação de emprego | Numérica |  |
| 17 | cons.price.idx | Indice de preços ao consumidor | Numérica |  |
| 18 | cons.conf.idx | Indice de confiança do consumidor | Numérica |  |
| 19 | euribor3m | Euro | Numérica |  |
| 20 | nr.employed | Número de empregos | Numérica |  |
| 21 | Target | Cliente contratou produto bancário | Binaria | yes','no' |

Para ilustrar, segue duas amostras do conjunto, cada uma com 10 registros:

10 casos que não contrataram os produtos

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| age | job | marital | Education | default | housing | loan | contact | month |
| 56 | housemaid | married | basic.4y | no | no | no | telephone | may |
| 57 | services | married | high.school | unknown | no | no | telephone | may |
| 37 | services | married | high.school | no | yes | no | telephone | may |
| 40 | admin. | married | basic.6y | no | no | no | telephone | may |
| 56 | services | married | high.school | no | no | yes | telephone | may |
| 45 | services | married | basic.9y | unknown | no | no | telephone | may |
| 59 | admin. | married | professional.course | no | no | no | telephone | may |
| 41 | blue-collar | married | Unknown | unknown | no | no | telephone | may |
| 24 | technician | single | professional.course | no | yes | no | telephone | may |
| 25 | services | single | high.school | no | yes | no | telephone | may |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| day\_of\_week | duration | campaign | pdays | previous | poutcome | emp.var.rate |
| mon | 261 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 149 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 226 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 151 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 307 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 198 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 139 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 217 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 380 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 50 | 1 | 999 | 0 | nonexistent | 1.1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| cons.price.idx | cons.conf.idx | euribor3m | nr.employed | y |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |
| 93.994 | -36.4 | 4.857 | 5191 | no |

10 casos que contrataram produtos

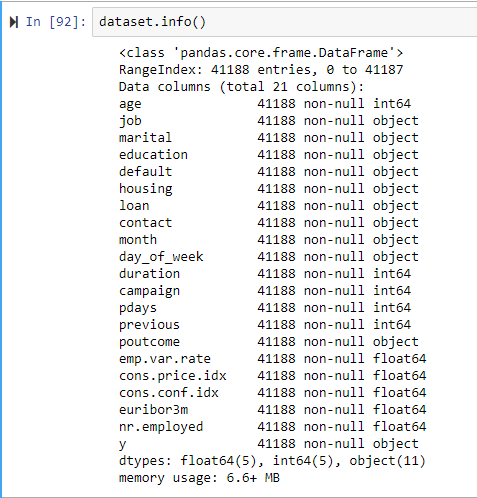
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| age | job | marital | Education | default | housing | loan | contact | month |
| 41 | blue-collar | divorced | basic.4y | unknown | yes | no | telephone | may |
| 49 | entrepreneur | married | university.degree | unknown | yes | no | telephone | may |
| 49 | technician | married | basic.9y | no | no | no | telephone | may |
| 41 | technician | married | professional.course | unknown | yes | no | telephone | may |
| 45 | blue-collar | married | basic.9y | unknown | yes | no | telephone | may |
| 42 | blue-collar | married | basic.9y | no | yes | yes | telephone | may |
| 39 | housemaid | married | basic.9y | no | yes | no | telephone | may |
| 28 | unknown | single | Unknown | unknown | yes | yes | telephone | may |
| 44 | services | married | high.school | no | yes | no | telephone | may |
| 42 | technician | married | professional.course | no | no | no | telephone | may |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| day\_of\_week | duration | campaign | pdays | previous | poutcome | emp.var.rate |
| mon | 1575 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 1042 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 1467 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 579 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 461 | 1 | 999 | 0 | nonexistent | 1.1 |
| mon | 673 | 2 | 999 | 0 | nonexistent | 1.1 |
| mon | 935 | 3 | 999 | 0 | nonexistent | 1.1 |
| tue | 1201 | 1 | 999 | 0 | nonexistent | 1.1 |
| tue | 1030 | 1 | 999 | 0 | nonexistent | 1.1 |
| tue | 1623 | 1 | 999 | 0 | nonexistent | 1.1 |

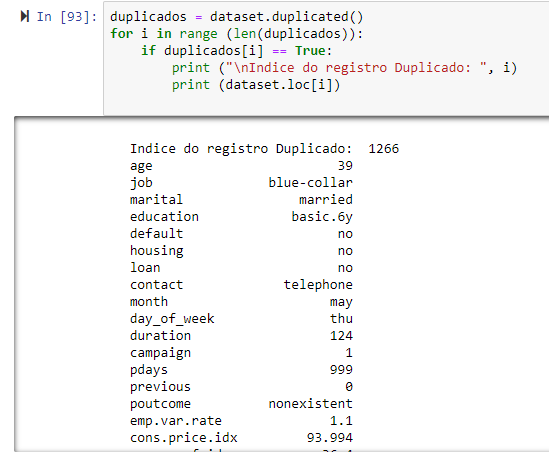
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| cons.price.idx | cons.conf.idx | euribor3m | nr.employed | y |
| 93.994 | -36.4 | 4.857 | 5191 | yes |
| 93.994 | -36.4 | 4.857 | 5191 | yes |
| 93.994 | -36.4 | 4.857 | 5191 | yes |
| 93.994 | -36.4 | 4.857 | 5191 | yes |
| 93.994 | -36.4 | 4.857 | 5191 | yes |
| 93.994 | -36.4 | 4.857 | 5191 | Yes |
| 93.994 | -36.4 | 4.857 | 5191 | Yes |
| 93.994 | -36.4 | 4.857 | 5191 | Yes |
| 93.994 | -36.4 | 4.857 | 5191 | Yes |
| 93.994 | -36.4 | 4.857 | 5191 | Yes |

Foram realizadas as seguintes análises nas features:

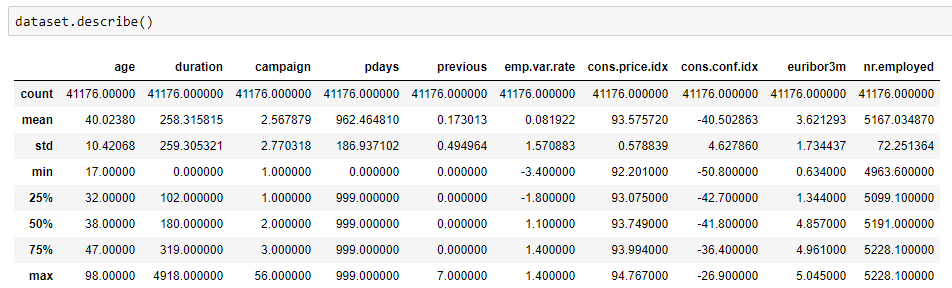
* Identificação de features com ausência de valores: Nenhuma feature tem valor ausente.



* Identificação de registros duplicados. Foi considerado duplicado, caso todas as features estejam com o mesmo conteúdo de outro registro

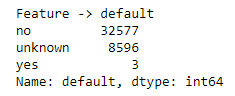


* Para as features numéricas foi visualizado e analisado a média, mediana, desvio padrão e máximo valor. Não foi identificado nenhuma grande variação. Um ponto importante identificado foi que as features possuem escalas de valores variadas

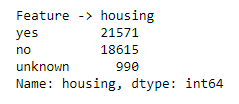


|  |  |  |
| --- | --- | --- |
| feature | mediana | media |
| age | 38 | 40,02380027 |
| campaign | 2 | 2,567879347 |
| pdays | 999 | 962,4648096 |
| previous | 0 | 0,173013406 |
| emp.var.rate | 1,1 | 0,081921508 |
| cons.price.idx | 93,749 | 93,57571989 |
| cons.conf.idx | -41,8 | -40,50286332 |
| euribor3m | 4,857 | 3,621293448 |
| nr.employed | 5191 | 5167,03487 |

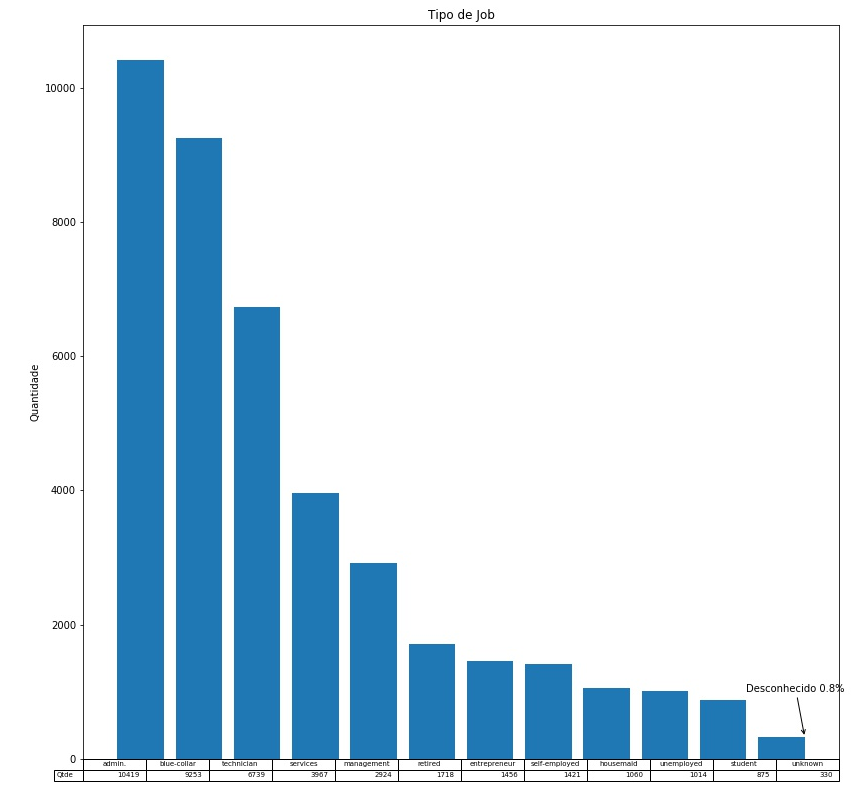
* Para as variáveis categóricas a visualização foi realizada através de um agrupamento para obter a quantidade (por domínio) de cada feature.
  + Foi constatado que a feature ‘default’ possui muitos registros ‘sem informação’, ou seja, é uma variável de pouca representatividade

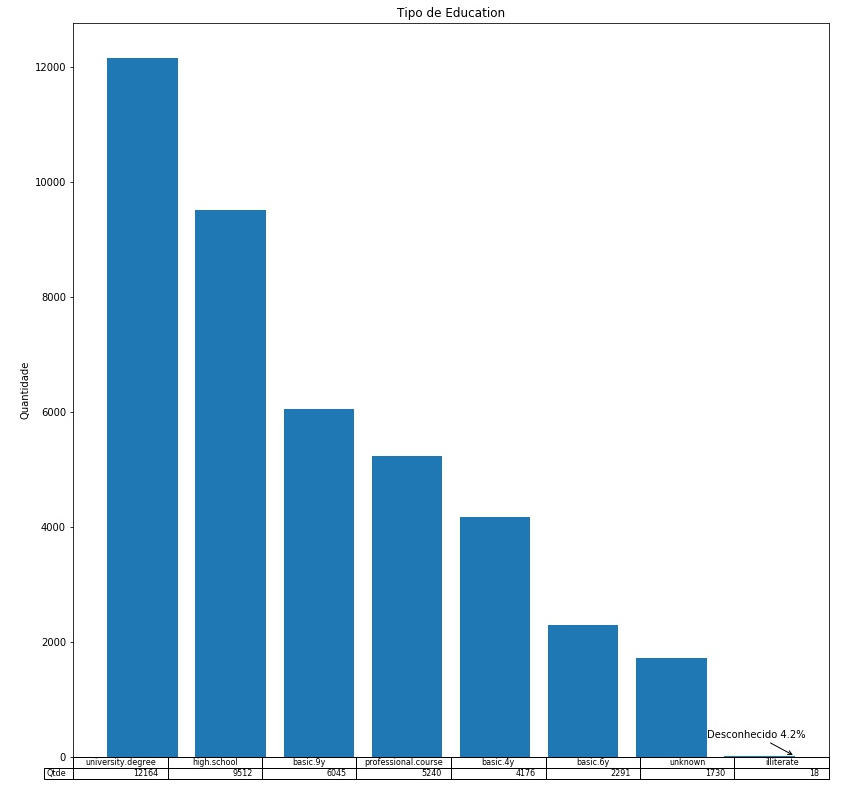


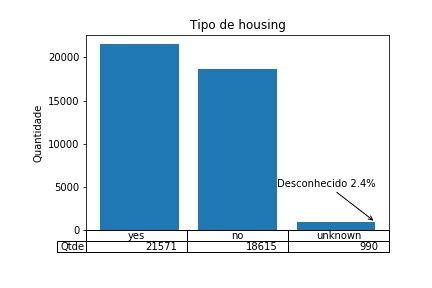
* + A feature ‘duration’ é obtida somente após a campanha ser realizada, ou seja, não pode ser utilizada para realizar previsões
  + A feature ‘housing’ não possui um valor dominante.

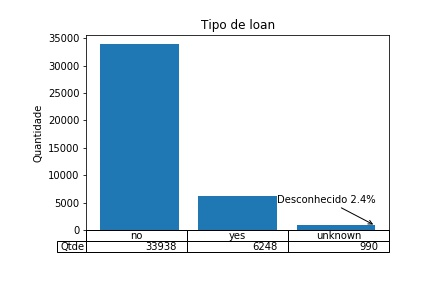
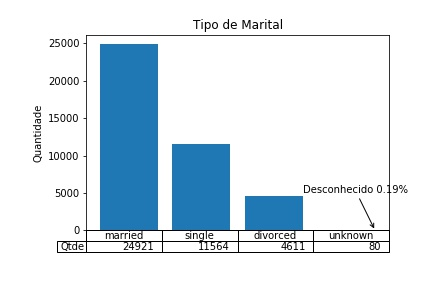


* + A maior parte das features categóricas possuem registros com registros com valores ‘unknown’, porém, não são valores dominantes. Para representar, os resultados foram plotados em gráficos (por feature)





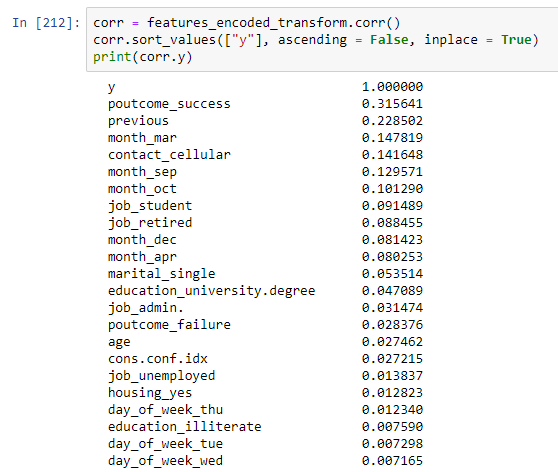




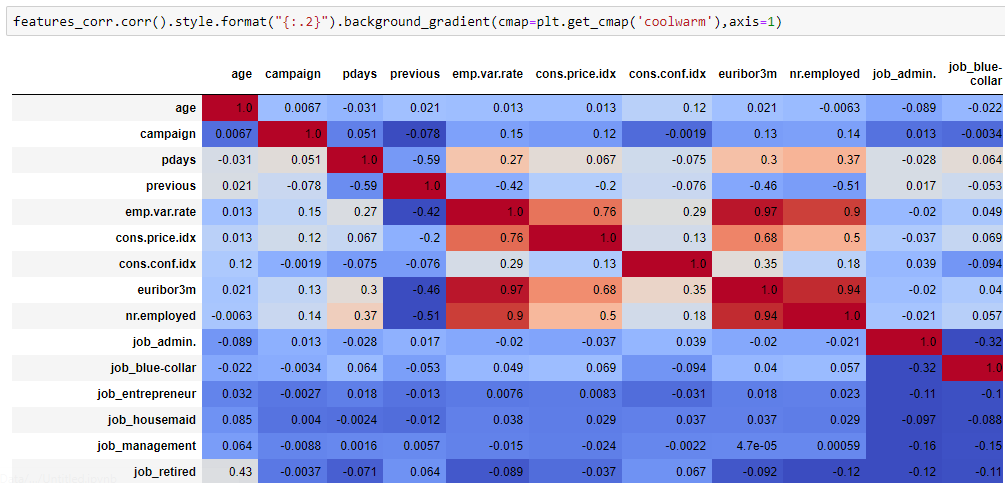
* Foi utilizado um método para identificar valores discrepantes (outliers)

**Visualização Exploratória**

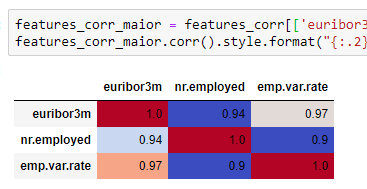
Buscando identificar features mais relevantes ao target, utilizei o método de correlação de Pearson. O resultado mostrou que não existe nenhuma feature com uma forte correlação com a variável target (y).

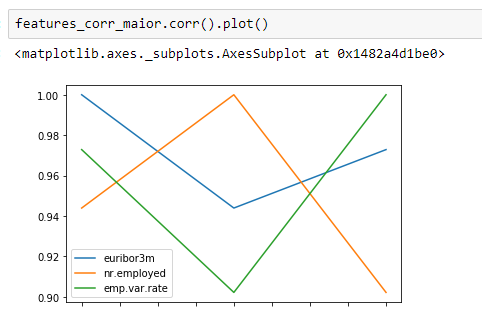


A mesma técnica foi aplicada para buscar a correlação entre todas as features (exceto a target).



As features que apresentaram uma correlação mais forte (mais próxima de 1) foram plotadas num gráfico.





Analisando os gráficos constatei que as features, euribour3m e emp.var.rate possuem uma forte correlação.

**Algoritmos e técnicas**

**Técnicas e métodos que serão aplicados:**

Primeiramente foram utilizadas funções para visualização das features:

* filtros /seleções
* agrupamento de valores
* contagem dos domínios
* geração de gráficos

Execução de Funções para identificar e/ou tratar:

* valores ausentes
* registros duplicados
* média, mediana e desvio padrão
* normalização da escala das variáveis numéricas
* codificação das variáveis categóricas
* exclusão de features
* atribuição de valores aleatórios

Para variáveis categóricas foi adotado o método para substituir o valor ‘unknown' pelo domínio que mais se repetiu.

Para reduzir o tamanho do conjunto, buscando gerar um modelo mais performático foram utilizados os métodos:

- Métodos de correlação

- PCA (para reduzir dimensionalidade)

E por fim, serão treinados, testados e refinados modelos preditivos de aprendizagem supervisionada

**Benchmark**

Encontrei alguns trabalhos no GitHub, eles serão como modelo de referência para o modelo que estou propondo.

Modelo de Referencia 1:

Fonte: <https://github.com/krishtanwani/bank-additional>

Modelo: LogisticRegression

* Acurácia: 0.896981467994
* Tamanho do Teste: 30%

Modelo de Referencia 2 (essa fonte possui 3 modelos):

Fonte: https://github.com/juliencohensolal/BankMarketing

Modelo: SVC

* Acurácia: 0.899193939983
* Tamanho do teste: 25%

Modelo: RandomForestClassifier

* Acurácia: 0.88798679227
* Tamanho do teste: 25%

Modelo: LogisticRegression

* Acurácia: 0.900320481694
* Tamanho do teste: 25%
  1. **– Metodologia**

**Pré-Processamento**

Para treinar e testar o modelo de machine learning foi necessário realizar um pré-processamento que envolveu:

- Exclusão dos registros duplicados

- Para todas as features categóricas, nos casos de valores ‘unknown’ foi atribuído o valor mais repetido

- Somente para a variável 'education' não foi possível adotar o método acima, pois, caso fosse adotado, todos os registros seriam marcados como ‘university.degree’ o que não faria sentido, pois, existem faixas de idades que não permitem. Para resolver esse problema foi realizado o seguinte processo:

* criação de uma feature nova para representar faixas de idades
* identificação, por faixa de idade, qual o domínio com a maior quantidade de registros
* para cada faixa, dos registros com valor ‘desconhecido’ foi atribuído o valor do domínio com maior quantidade
* - Transformar todas as variáveis categóricas em códigos para processamento do modelo

Como as variáveis numéricas possuíam valores com escalas diferentes, adotei o método para normalizar a escala dessas variáveis.

A feature duration foi excluída, pois, ela é obtida somente após a realização da campanha, ou seja, é um dado que nunca estará disponível no momento da execução do modelo preditivo

A feature ‘default’ também foi excluída, pois, ela possuía poucos registros com informações.

Para a feature housing foi adotado o método de atribuição aleatória para os registros com conteúdo ‘desconhecido’, pois, os dois valores possíveis possuem quantidade semelhantes.

Para identificar valores discrepantes (outliers) utilizei o método Turco para cada feature:

* cálculo do percentil 25 (Q1) e o percentil 75 (Q3)
* cálculo da amplitude interquantil x 1,5 (step)
* cálculo do menor valor (Q1 – step)
* cáclulo do maior valor (Q3 + step)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Feature | step = iqr \* 1.5 | Q1 | Q1-step | Q3 | Q3+step |
| Age | 22,5 | 32 | 9,5 | 47 | 69,5 |
| Campaign | 3 | 1 | -2 | 3 | 6 |
| Pdays | 0 | 999 | 999 | 999 | 999 |
| Previous | 0 | 0 | 0 | 0 | 0 |
| emp.var.rate | 4,8 | -1,8 | -6,6 | 1,4 | 6,2 |
| cons.price.idx | 1,3785 | 93,075 | 91,6965 | 93,994 | 95,3725 |
| cons.conf.idx | 9,45 | -42,7 | -52,15 | -36,4 | -26,95 |
| euribor3m | 5,4255 | 1,344 | -4,0815 | 4,961 | 10,3865 |
| nr.employed | 193,5 | 5099,1 | 4905,6 | 5228,1 | 5421,6 |

São considerados outiliers os valores de cada feature que forem menor que Q1 – step ou maior que Q3 + step.

Nessa fase foi fundamental entender o comportamento das features, pois, caso contrário, as variáveis abaixo teriam registros considerados outliers indevidamente:

* age: valores < 32 ou > 69,5 seriam considerados outliers (468 linhas), não faz sentido, pois, são valores dentro da normalidade
* campaign: valores > 6 seriam considerados outliers (2406 linhas), não faz sentido, pois, podem existir clientes que foram contatados em mais de 6 campanhas
* pdays: valores < 999 ou > 999 seriam considerados outliers (1515 linhas), isso aconteceu porque existem muitos registros com valores 999 (cliente não contatados)
* previous: valores < 0 e > 0 seriam considerados outliers (5625 linhas), isso aconteceu porque a maioria dos clientes não foi contatada em outras campanhas (previous = 0)

As features emp.var.rate , cons.price.idx , euribor3m e nr.employed não apresentaram valores considerados outliers.

Somente a feature ‘cons.conf.idx’ apresentou valores que podem ser considerados outliers.

Para reduzir a dimensionalidade do conjunto, utilizei a correlação de Pearson e o método PCA (Principal Component Analysis.

Os modelos abaixo foram treinados e testados:

- Modelo Decision Tree Classifier

- Modelo Linear SGDClassifier

- Modelo Logistic Regression

- Modelo SVC

- Modelo Random Forest Classifier

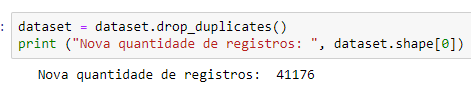
E por fim, foram escolhidos para refinamento, os modelos que apresentaram a melhor acurácia:

- Modelo Linear SGDClassifier

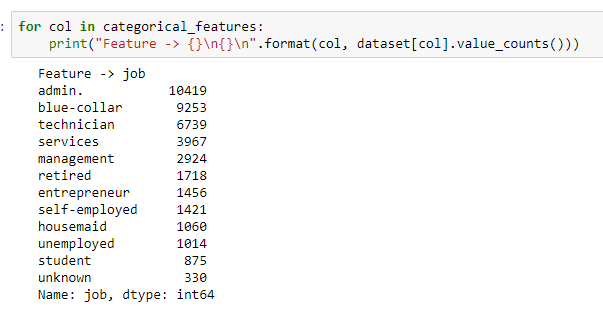
- Modelo Logistic Regression

**Implementação**

1 - Eliminação dos registros duplicados:



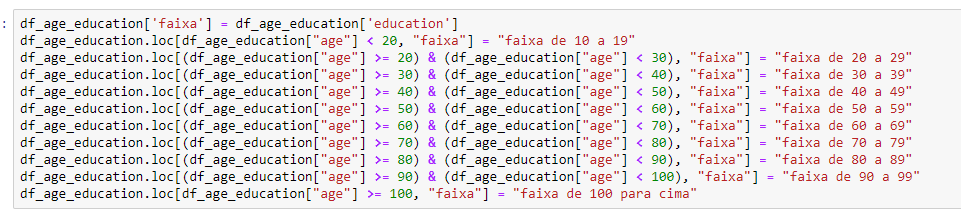
2 – Atribuição do valor para os casos ‘unknown’:



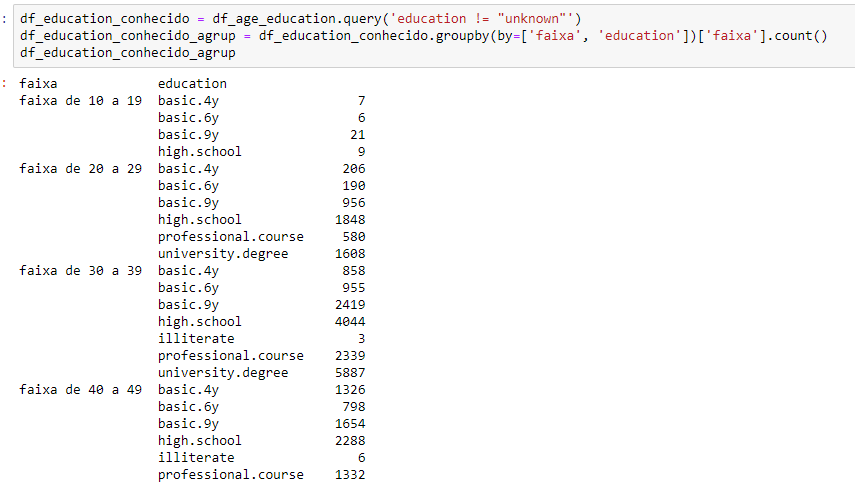


3 – Tratamento da Feature ‘education’:

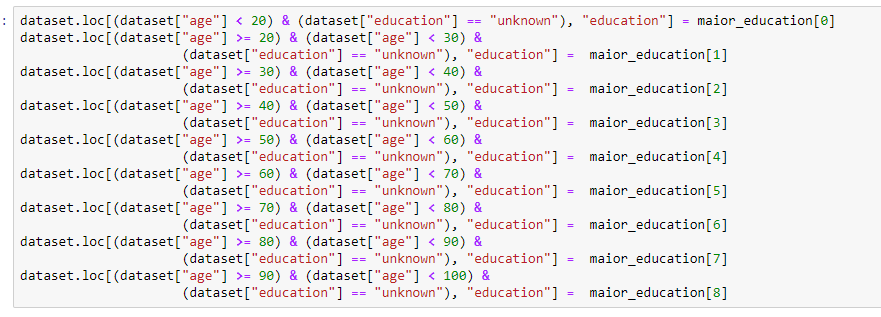
3.1 – Criação das faixas de idade



3.2 – Visualização dos valores mais repetidos por faixa de idade (excluindo os ‘unknown):



3.3 – Atribuição dos valores por faixa:



4 – Atribuição aleatória para a feature ‘housing’:

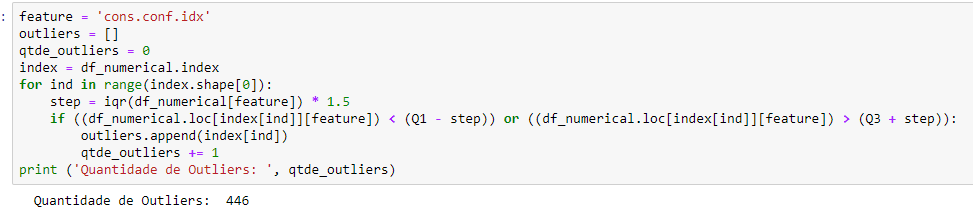


5 –Método Turco para identificação dos outliers:

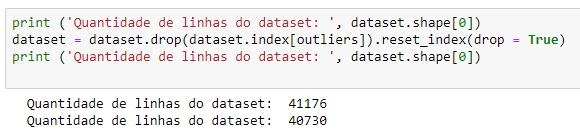
5.1. – Calculando e exibindo e armazenando os parâmetros:



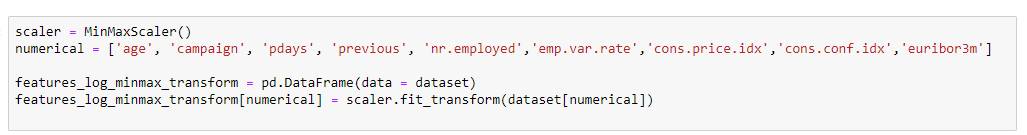
5.2 – Identificando e armazenando no lista os outliers para a feature ‘cons.conf.idx’



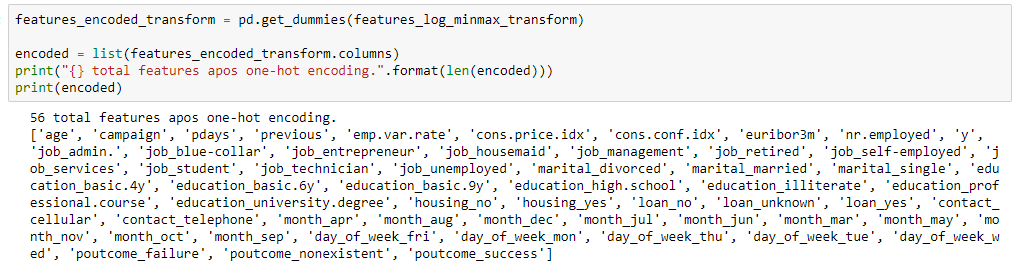
5.3 – Remoção dos outliers



6 – Normalizando a escala das variáveis numéricas:



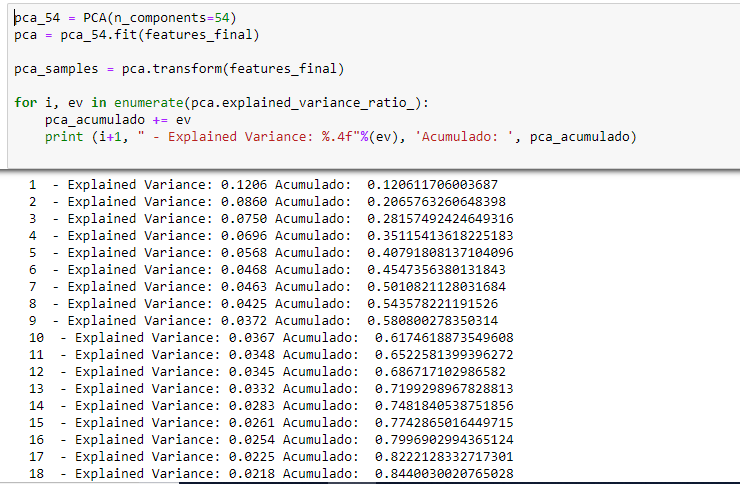
7 – Codificando as variáveis categóricas para funcionamento do modelo preditivo



8 – Calculando a correlação (método Pearson) e plotando num gráfico

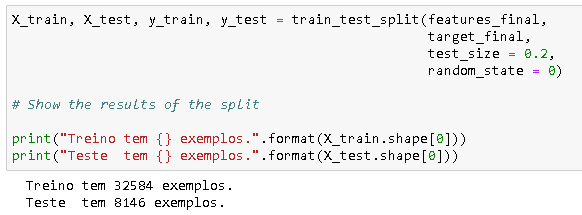


9 – Cálculo do PCA e exibição do resultado:

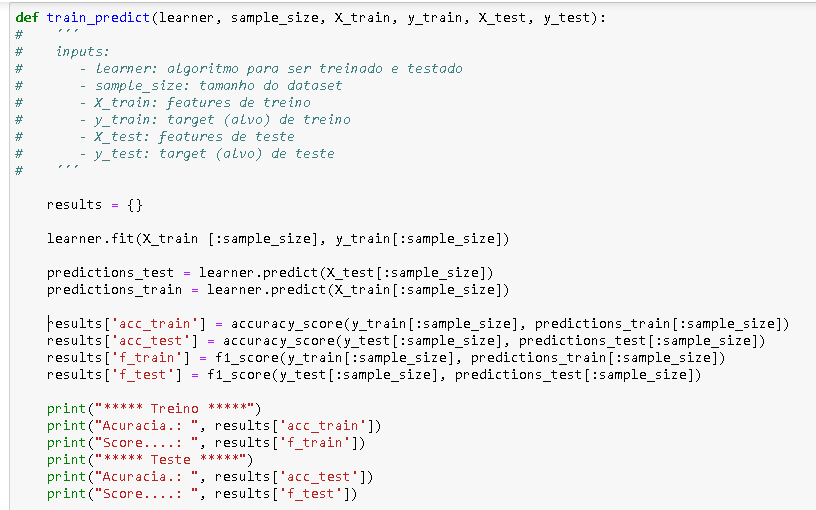


10 – Implementação dos Modelos de Machine Learning:

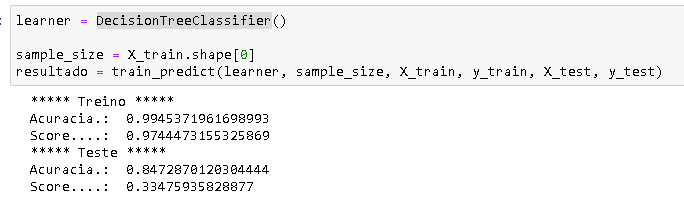
10.1 – Divisão do Dataset em treino (80%) e teste (20%)



10.2 – Função para treino do Modelo



10.3 – Exemplo de Treino do Modelo:



**Refinamento**

Buscando obter melhores resultados, para os dois modelos que apresentaram a melhor acurácia, fiz alguns ajustes nos parâmetros dos modelos, obtendo uma melhora nos resultados.

Parâmetros passados para otimizar o modelo:

|  |  |  |  |
| --- | --- | --- | --- |
| **Parâmetro** | **SGDClassifier** | **LogisticRegression** | **LogisticRegression** |
| random\_state | 15, 16, 17, 18, 19 | 1, 2, 3, 4 | 1, 2, 3, 4 |
| loss | hinge' 'log', 'modified\_huber',  'squared\_hinge', 'perceptron', 'squared\_loss',  'huber',  'epsilon\_insensitive', 'squared\_epsilon\_insensitive' | n/a | n/a |
| penalty | none', 'l2', 'l1', 'elasticnet' | l1' | newton-cg',  'lbfgs',  'liblinear',  'sag',  'saga' |
| C | n/a | 1, 2, 3, 4, 5, 6, 7, 8, 9 | 1, 2, 3, 4, 5, 6, 7, 8, 9 |

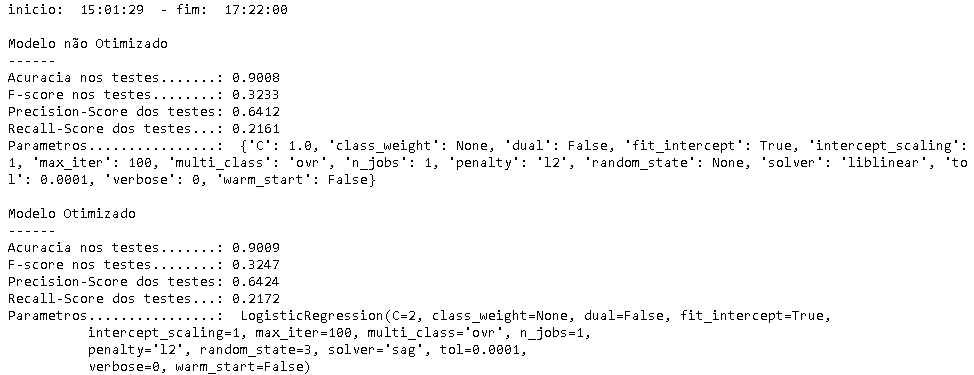
Melhores parâmetros retornados:

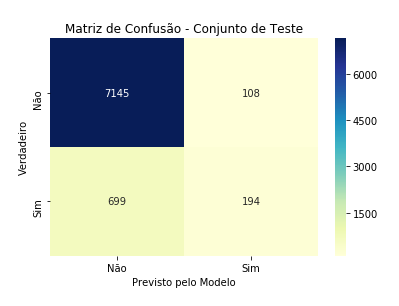
|  |  |  |  |
| --- | --- | --- | --- |
| **Parametro** | **SGDClassifier** | **LogisticRegression** | **LogisticRegression** |
| random\_state | 16 | 1 | 3 |
| loss | log' | n/a | n/a |
| penalty | l1' | l1' | l2' |
| C | n/a | 3 | 2 |

Modelo melhor otimizado:



Resultados obtidos:





* 1. **– Resultados**

**Modelo de avaliação e validação**

O modelo escolhido foi o que apresentou melhor acurácia, ou seja, o modelo que obteve o maior percentual de acertos.

Para garantir que o modelo é confiável fiz uma quebra no conjunto real, deixando 80% para treino e 20% para teste. Os resultados mostraram que o modelo treinado está generalizando muito bem, pois, ele obteve ótimos resultados com os 20% de dados de teste.

**Justificativa**

Pelos modelos executados obtive melhores resultados (considerando a acurácia) que os modelos utilizados como benchmark:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Modelo** | **Logistic**  **Regression** | **SVC** | **RandomForest**  **Classifier** | **DecisionTree**  **Classifier** | **SGDClassifier** |
| Modelo de Referencia 1 | 0.8969 | x | X | x | x |
| Modelo de Referencia 2 | 0.9003 | 0.8991 | 0.8879 | x | x |
| Meu Projeto | 0.9008  Otimizado:  0.9009 | 0.8997 | 0.8911 | 0.8465 | 0.8980  Otimizado:  0.9009 |

* 1. **– Conclusão**

A conclusão que eu cheguei foi que, para gerar um modelo de machine learning que consiga prever adequadamente um conjunto de dados é preciso:

1 – Primeiramente, entendimento claro do problema a ser resolvido

2 – Conhecimento do negócio relacionado ao problema. Esse conhecimento é fundamental para definir, localizar, avaliar e validar as melhores fontes de dados.

3– Conhecimento dos sistemas geradores das informações. É necessário entender como os sistemas armazenam os dados, a periodicidade que eles são gerados.

4 – Conhecimento de como funcionam os modelos e quais são os mais adequados para o problema a ser resolvido

5 – Conhecimento das métricas de avaliação dos modelos

6 – Conhecimento de linguagens de programação voltadas para machine learning (python, R, etc..)

7 – Conhecimento de métodos e técnicas tanto para análise, visualização e exploração de dados e principalmente para efetuar a tarefa de pré-processamento

**Melhorias**

Algumas melhorias podem ser adotadas nesse projeto:

- Aplicação de outras técnicas para detecção de outliers, redução de dimensionalidade do conjunto de dados e identificação de correlação entre variáveis

- Execução de outros modelos de machine learning

- Treino e testes com outros conjuntos de dados