Nanodegree Engenheiro de Machine Learning

Proposta de projeto final

Erivelton Marques de Carvalho
13 de fevereiro de 2019

Proposta

Histórico do assunto

Este projeto está ambientado na marcação de consultas e procedimentos médicos. O gerenciamento de instituições de saúde, como clínicas, hospitais e centrais de marcações de consultas, públicas ou particulares, visa a redução de custo no ato da execução da consulta ou procedimento. Por conseguinte, uma das formas que essas instituições podem fazer para essa redução é não gerar receita ao saber de forma antecipada se um cliente não vai comparecer, chamado de **no-show**. Mais informações podem ser encontradas no link **https://www.pixeon.com/blog/gestao-de-clinicas**, site de uma empresa de gerenciamento de clínicas.

O problema da falta dos pacientes às consultas é antigo e desde que a tecnologia se tornou capaz de fazer previsões a partir de dados cadastrados em sistemas automatizados, surgiram trabalhos importantes para esse fim, no qual podemos destacar alguns. No artigo **Previsão de No-Show no Agendamento de Serviços Médicos Baseada em Mineração de Dados**, (Adeodato, P. et al. 2008), disponível no link **http://www.cin.ufpe.br/~compint/aulas-IAS/kdd-112/material/Artigos_MSc+CIn/2008_CONTECSI_DataMining_em_NoShow_AtendimentoMedico.pdf**, acessado em 12 fev. 2019, os autores optaram pelo modelo de **rede neural Multilayer perceptron (MLP)** treinada com o **algoritmo de backpropagation**, (apud Haykin 1998), (apud Rumelhart et al. 1986). Nesse trabalho, os autores pretendem identificar, no momento do agendamento, as consultas com alto risco de no-show para auxiliar seu reagendamento em tempo real com o objetivo de estimar e minimizar a perda com a alocação de recursos. No artigo **Diferentes abordagens de Subamostragem para Balanceamento da Base de Dados aplicados ao estudo de caso da Classificação de Absenteísmo de Pacientes Clínicos**, (Darós, L. et al. 2018), disponível no link **http://portaldeconteudo.sbc.org.br/index.php/ersi-rj/article/view/4655/4572**, acessado em 12 fev. 2019, os autores escolheram os algoritmos **Naive Bayes** (apud Lewis 1998) e **K-Nearest Neighbors (KNN)** (apud Hwang e Wen 1998). Nesse trabalho, os autores também visam prever o no-show, mas com um diferencial no tratamento dos dados, considerando que há uma classe majoritária e outra minoritária, onde a primeira se refere aos pacientes que comparece às consultas e a segunda aos que não comparecem. Eles usam técnicas para minimizar a classe majoritária ou para maximizar a minoritária para melhorar o desempenho da previsão.

Descrição do problema

O problema numa visão macro é a **perda de receita** em casos de **no show**. Numa visão específica, o problema é a **incerteza sobre o comparecimento do cliente**. Isso pode deixar empregados ociosos, materiais comprados, equipamentos preparados, tudo contando com a presença dele. As causas podem ser muitas, como data da consulta muito distante, horário indesejado, etc. A empresa deve se preparar para uma possível substituição de cliente, liberação de empregados ou cancelamento de pedidos de materiais, mas para isso ela tem que saber de forma antecipada se o cliente vai comparecer. Essa possibilidade de previsão pode ser

conseguida através da análise de dados históricos registrados durante a marcação e o comparecimento ou não da consulta. São dados categóricos e numéricos no qual podemos quantificar e medir, além de replicar a solução para muitas empresas na mesma área, inclusive em áreas diferentes que tenha um processo de marcação de compromissos semelhante.

Conjuntos de dados e entradas

Os dados contém informações dos clientes que marcam as consultas ou procedimentos médicos nos sistemas de informação das empresas que têm esse serviço. Propriedades como identificação do cliente, data/hora e local da marcação, data/hora e local da consulta, se a consulta foi realizada, entre muitas outras são relevantes para a base de dados necessária a este projeto. Cada uma dessas empresas tem seu conjunto de dados com quantidade e propriedades diferentes, mas há sempre semelhanças, como o resultado, neste caso o comparecimento ou não do cliente, pois, é o foco do sistema a ser desenvolvido por este projeto.

A base de dados que vamos utilizar foi adquirida no site da Kaggle, seção Datasets, descrito como "Medical Appointment", disponibilizado por Alvaro Flores, do tipo público com licença GPL 2, podendo ser encontrada através do link **https://www.kaggle.com/afflores/medical-appointment**. Podemos visualizar e baixar os dados em arquivo do tipo ".CSV". Essa base de dados é uma tabela que possui **19 propriedades (colunas)** com **61214 registros (linhas)** de marcações de consultas ou procedimentos médicos. Dividimos as propriedades em dois grupos, sendo a primeira com 18 colunas que usaremos como **recursos** de entrada para processamento e a última coluna, intitulado "show" (se compareceu à consulta ou procedimento médico), como nosso **rótulo (objetivo)**. Essa base possui as propriedades importantes e necessários ao nosso sistema, já descritos antes, entre outros que são candidatos à melhoria do desempenho. Segundo as informações no site da Kaggle sobre essa base de dados, essas propriedades e o que representam são as seguintes:

- **especialidad**, tipo de especialista para a consulta. Ou seja, dematologista, oftalmologista, etc;
- **edad**, idade;
- **sexo**, sexo, 1=Homem, 2=Mulher;
- **reserva_mes_d**, valor discreto para o mês da consulta, 1 = Jan... 12 = Dez;
- **reserva_mes_c**, valor constínuo para o mês da consulta, a fórmula é COS(2*reserva_mes_d*Pi/12);
- **reserva_dia_d**, valor discreto para o dia da semana da consulta, 1 = Seg ... 7 = Sab;
- **reserva_dia_c**, valor contínuo do dia da semana da consulta, a fórmula é COS(2*reserva_dia_d*Pi/7);
- **reserva_hora_d**, valor discreto para a hora da consulta, 0h ... 23h;
- **reserva_hora_c**, valor contínuo para a hora da consulta, a fórmula é COS(2*reserva_hora_d*Pi/24);
- **creacion_mes_d**, valor discreto para o mês em que a consulta foi criada, 1 = Jan ... 12 = Dez;
- **creacion_mes_c**, valor contínuo para o mês em que a consulta foi criada, a fórmula é COS(2*creacion_mes_d*Pi/12);
- **creacion_dia_d**, mesmo que reserva_dia_d, mas considerando o dia em que a consulta foi criada;
- **creacion_dia_c**, mesmo que reserva_dia_c, mas considerando o dia em que a consulta foi criada;
- **creacion_hora_d**, houra em que a consulta foi criada, 0h ... 23h;
- **creacion_hora_c**, valor contínuo para creacion_hour_d, a fórmula é COS(2*creacion_hora_d*Pi/24);
- **latencia**, número de dias entre a consulta e a data em que foi criada;
- **canal**, canal usado para criação da consulta, 1 = call center, 2 = Personal, 3 = Web;
- **tipo**, tipo de consulta, 1 = consulta medica, 2 = procedimento médico;
- **show**, confirmação da consulta, 0 = no show, 1 = show.

Descrição da solução

Mediante à **incerteza sobre o comparecimento do cliente** na data/hora marcada para a consulta ou procedimento médico, a solução é a **previsão de comparecimento do cliente**. Para isso, usaremos os registros de marcações de consultas da base de dados do sistema da empresa como entrada em outro sistema, a ser desenvolvido por este projeto. Esse novo sistema será capaz de aprender com os registros históricos e responder se um determinado cliente vai comparecer à consulta. Trata-se de um **modelo de aprendizagem supervisionada**. Esse resultado deve ter as máximas **acuracidade, precisão e revocação**, com o tempo de resposta rápido, mas não sendo tão importante quanto às três primeiras medidas, pois, o uso do sistema não é constante. Quanto maior a quantidade de registros, maiores serão a acuracidade, precisão e revocação, e menor será a velocidade de processamento. Com esse novo sistema, a empresa poderá verificar os resultados de um grupo de clientes programados em um expediente ou turno de uma data específica, entrar em contato com eles para confirmações e se preparar antecipadamente para uma possível mudança de planos de modo a não gerar custo sem necessidade. Essa solução pode ser aplicada em qualquer computador com acesso ao sistema da empresa e a todos os sistemas de marcações de consultas ou de outras categorias de trabalho, com algumas modificações relevantes em áreas diferentes.

Modelo de referência (benchmark)

Os valores de benchmark serão determinados em três momentos. No primeiro, as referências para as métricas **F-score**, **acuracidade** e **tempo de atraso de processamento (delay)** serão determinadas na aplicação de um modelo bem simples de aprendizagem de máquina que chamaremos de **naive prediction**. Essas referências serão comparadas com os valores das mesmas métricas para outros modelos de aprendizagem com o objetivo de escolher entre eles o que apresentar o melhor desempenho, que chamaremos de **modelo não otimizado**. No segundo momento, teremos as referências para duas métricas, o F-score e a acuracidade, que serão determinadas na aplicação do modelo não otimizado, considerando apenas os parâmetros padrões dele. Ao modificar esses parâmetros, objetivando melhorar as métricas F-score e acuracidade, encontraremos os melhores resultados comparando com as referências deste segundo momento, determinando o **modelo otimizado**. No terceiro momento, a referência será a métrica delay, determinada pelo tempo de processamento do modelo otimizado. Essa referência será utilizada para comparação com o delay extraído no processamento do modelo otimizado após a remoção de alguns recursos de entrada de baixa importância em relação ao rótulo, que chamaremos de **modelo final**. A remoção desses recursos objetiva melhorar o delay, mas também devemos manter o F-score e a acuracidade ainda melhores que suas respectivas referências.

Métricas de avaliação

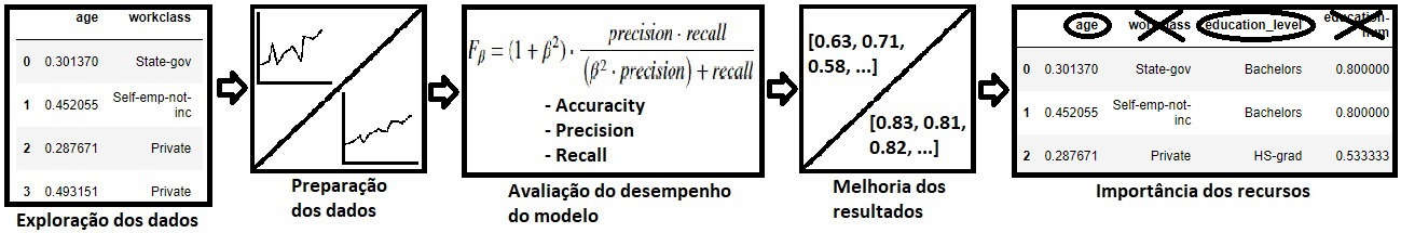
A **precisão** e a **revocação** são importantes para a tomada de decisão da empresa, porém, a primeira é mais importante. Por isso, a métrica **Fbeta** (pontuação F com fator beta) é a mais adequada, pois, seu valor é calculado com a precisão e a revocação. O Fbeta tem o parâmetro "beta" e o valor padrão que usaremos é "0,5", pois, é totalmente viável por dar mais ênfase à precisão e é chamado de pontuação F para simplicidade. O valor de Fbeta aplicado na comparação entre os resultados previstos e real na base de dados de teste será o fator de decisão que poderá julgar o treino do sistema entre finalizado ou não. Sua fórmula é **Fbeta = (1 + beta²) . precisão . revocação / ((beta² . precisão) + revocação)**. Para calcular essas métricas, devemos considerar os seguintes conceitos: **verdadeiros positivos (VP)**, sendo "previsão=verdadeiro" e "real=verdadeiro"; **verdadeiros negativos (VN)**, sendo "previsão=verdadeiro" e "real=falso"; **falsos positivos (FP)**, sendo "previsão=falso" e "real=falso"; e **falsos negativos (FN)**, sendo "previsão=falso" e "real=verdadeiro". Com isso, temos as fórmulas de **precisão = VP / (VP + FP)** e de **revocação = VP / (VP + FN)**. Vamos considerar que os clientes que compareceram e foram previstos como no-show, falsos negativos, podem ser atendidos com recursos extras programados para esses casos. Por isso a revocação, que considera os falsos negativos, é

menos importante que a precisão. Outra métrica importante para tomada de decisão que vamos utilizar é a **acuracidade** que mede com que frequência o classificador faz a previsão correta. Sua fórmula é **acuradidade = VP / total_registros_teste**, considerando que todas as métricas serão aplicadas aos registros separados para teste que serão submetidos à previsão.

Se houver necessidade de aumento da velocidade do processamento, a **quantidade de recursos** poderá ser reduzida, desconsiderando aquelas com menores **importâncias no rótulo**. O sistema será capaz de calcular um número de importância no resultado para cada recurso. A forma de escolha desses recursos a serem utilizados no modelo de previsão é a soma dos "N" maiores em importância até obter um valor superior à metade do total. Nesse sentido, não necessariamente temos que usar todos os recursos, pois, o aumento da quantidade deles pode não ter muita significância para a melhoria do Fbeta.

Design do projeto

O escopo deste projeto foi dividido nas etapas mostradas na figura a seguir.



Na **etapa de exploração dos dados** iremos conhecer as propriedades e quantidades de registros. Na **etapa de preparação dos dados** faremos um pré-processamento para limpar, formatar e reestruturar os dados para eliminar possíveis enviesamentos e aumentar o poder de processamento e resultado. Técnicas de transformação logarítmicas de dados contínuos e normalização para colunas numéricas serão utilizadas caso sejam necessárias. Para finalizar essa etapa, faremos a separação do conjunto de dados em dois, um para treinamento com 80% e outro para teste com 20%, tanto para as propriedades quanto para o rótulo. Na **etapa de avaliação do desempenho do modelo** faremos inicialmente a aplicação de um modelo de previsão ingênuo que chamaremos de **naive prediction** para extrairmos as métricas necessárias. Tomaremos os valores dessas métricas como referência para os calculados nos modelos candidatos de algoritmos de aprendizagem supervisionada a serem escolhidos. A princípio, esses algoritmos terão seus parâmetros com valores padrões, e a comparação entre eles por métricas determinará o melhor para se tornar o oficial do sistema. Os algoritmos que podem ser usados são:

- Gaussian Naive Bayes;
- Decision Trees;
- Ensemble Methods (Bagging, AdaBoost, Random Forest, Gradient Boosting);
- K-Nearest Neighbors;
- Stochastic Gradient Descent Classifier;
- Support Vector Machines;
- Logistic Regression.

Dentre esses algoritmos, três serão escolhidos para os testes, visando as características de cada um em relação aos dados já analisados na primeira e segunda etapa. Na **etapa de melhoria dos resultados** faremos vários testes para ajustar os parâmetros do algoritmo escolhido como oficial do sistema. Finalmente, na **etapa de importância dos recursos** faremos a verificação da importância de cada recurso para escolher as maiores entre eles de forma que as métricas não tenham piora considerável, sendo aceitável uma queda menor ou igual a 2 pontos percentuais em todas as métricas.