Previsão de Custo de Viagens por Aplicativo

Projeto Final Bi Master - Business Intelligence Master

Aluno: Carlos Henrique Albuquerque Da Silva

Orientador: Leonardo Forero Mendoza

Data: 21/06/2021

Trabalho apresentado ao curso BI MASTER como pré-requisito para conclusão de curso e obtenção de crédito na disciplina "Projetos de Sistemas Inteligentes de Apoio à Decisão".

Resumo

A utilização de meios de transporte por aplicativo já faz parte do nosso cotidiano. A proposta desse trabalho é utilizar uma base de dados do Kaggle da empresa Chh-OLA de Nova Delhi para gerar estimativa de corridas.

A idéia é avaliar o ganho de conhecimento do aluno no BI Master e verificar o quão próximo dos dados reais e das primeiras colocações da competição o mesmo consegue chegar.

Introdução

Utilizamos uma base do Kaggle (https://www.kaggle.com/c/chh-ola) de 2019 para iniciarmos esse estudo.

O trabalho envolveu a análise de 4 modelos diferentes via Python no Google Colab, foram considerados as etapas: análise exploratória de dados, missing values, conversão de dados, verificação de correlações, ajuste dos modelos e exportação dos resultados para submissão no Kaggle.

Fundamentação Teórica

Dado que o valor a ser previsto é uma variável contínua não limitada a ser “descoberta” através de variáveis independentes de vários tipos diferentes, entendeu-se que a regressão linear em aprendizado supervisionado era a ferramenta que melhor se enquadrava nesse problema.

A base de dados oferecida possui um grande volumo de dados e possibilita um bom treinamento dos modelos desenvolvidos.

A regressão é utilizada em vários campos de conhecimento, como economia, ciências sociais, medicina e muitas outras.

Modelagem

As seguintes etapas foram executadas:

Importação da base de treino do Kaggle.

Análise do tipo e conteúdo da informação importada.

```
ID                          int64
vendor+AFS-id              object
pickup+AFS-loc            float64
drop+AFS-loc              float64
driver+AFS-tip             object
mta+AFS-tax                object
distance                  float64
pickup+AFS-time            object
drop+AFS-time              object
numt+AFS-passengers       float64
toll+AFS-amount            object
payment+AFS-method        float64
rate+AFS-code             float64
stored+AFS-flag            object
extra+AFS-charges          object
improvement+AFS-charge     object
total+AFS-amount           object
dtype: object
```

Importação da base de testes.

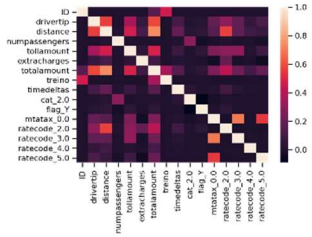
Renomeação das colunas de ambas as bases. Identificação e junção de ambas as bases para tratamento único da informação.

Busca e exclusão de valores anômalos.

Conversão de colunas erroneamente identificadas como objeto para numérico (float64) ou categórico (eliminando uma das colunas geradas devido à alta correlação entre elas), conforme o tipo.

Converter colunas “pickuptime” e “droptime” de objeto para formato “datetime” com posterior geração de novas informações a partir das mesmas, como dia da semana, dia e mês, por exemplo.

Analisar dataframe resultante.



Automaticamente verificar quais colunas possuem alta correlação entre si e eliminar uma delas.

Automaticamente verificar e eliminar colunas que possuem pouca correlação com a variável dependente (“totalamount”).

Separar novamente treino e teste. Separar variável dependente em outro dataframe.

Normalização das variáveis entre zero e um.

Criação de seletor para um dos quatro modelos a serem usados nas previsões.

Treino do modelo escolhido. Dependendo do modelo, múltiplos treinos com variações de parâmetros foram executadas em busca do melhor ajuste.

Avaliação do modelo escolhido via erro médio absoluto.

Aplicar modelo no grupo de testes e realizar previsões.

Exportar dados no formato padrão do Kaggle para análise.

Resultados

Várias experiências e variações foram tentadas; com e sem colunas correlacionadas entre si, com e sem colunas pouco correlacionadas com variável dependente e com os quatro modelos escolhidos. A sequência que obteve melhores resultados foi a relatada na sessão anterior.

Em relação aos quatro modelos diferentes, obtivemos os seguintes resultados:

Random Forest – o melhor resultado foi obtido com um estimador com erro médio absoluto de 0.32777566369841976.

Ada Boost - o melhor resultado foi obtido com 100 estimadores, learning rate de 0.1 e com erro médio absoluto de 4.540812809829317, cujo valor alto representa anomalia. Ao se verificar as previsões, as mesmas se agrupavam entre apenas alguns valores, não sendo viável, mesmo assim, seus dados foram exportados para a competição.

Gradient Boost – após várias tentativas, o menor erro médio absoluto foi obtido com 500 estimadores e profundidade igual a 12 resultando em um valor de 0.3604175543475147. Ainda foi feita uma tentativa com 700 estimadores e 12 nós de profundidade obtendo erro médio absoluto ainda menor, mas obtendo uma pior colocação na base do Kaggle, representando uma pior capacidade de generalização para outros dados diferentes dos dados utilizados para treino e teste e possivelmente configurando-se como um overfitting.

XgBoost – erro médio de 0.6952345451039076.

Abaixo pontuação dos quatro modelos no Kaggle. Quanto menor, melhor, assim como idealmente, ambos os resultados devem ter seu valor próximo um do outro.

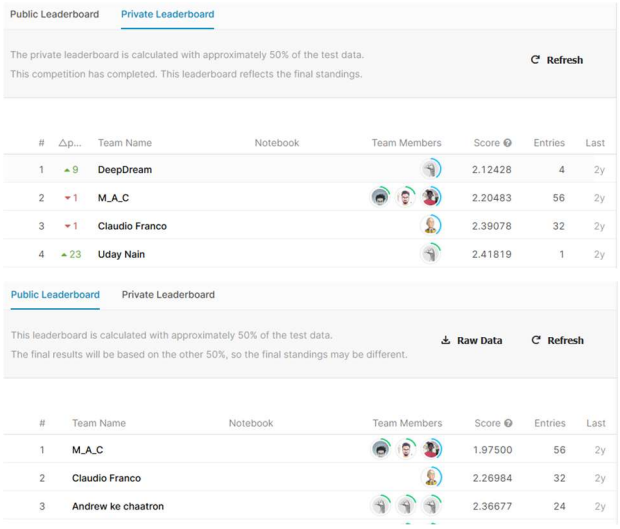
Submission and Description	Private Score	Public Score
Monografia v02 - RandomForest.csv 21 hours ago by Carlos Albuquerque Monografia v02 - RandomForest	2.53224	2.32114
Monografia v02 - AdaBoost.csv 21 hours ago by Carlos Albuquerque AdaBoost	7.89131	7.46399
Monografia v02 - XGBoost.csv a day ago by Carlos Albuquerque Regressor - XGBoost	2.78122	3.09522
Monografia v02 - Gradiente Boost - 500est 12depth.csv a few seconds ago by Carlos Albuquerque Monografia v02 - Gradiente Boost - 500est 12depth	2.37378	2.34567

Conclusão

O modelo com o melhor desempenho foi o Gradient Boost com o Random Forest muito próximo.

Observando-se os resultados públicos e privados, caso a competição estivesse aberta, o presente trabalho estaria colocado entre a segunda e terceira posições nas pontuações privada e pública.

Interessante observar que ambas as pontuações são coerentes entre si e com o treinamento, o que exclui a possibilidade de overfitting para esta configuração.



Considerações finais

Gostaria de agradecer a todos os professores do curso e em especial ao professor Leonardo pelo apoio e constante interesse em ensinar.

Período: 2019.2

Matrícula: 192.671.131

Pontifícia Universidade Católica do Rio de Janeiro

Curso de Pós Graduação Business Intelligence Master