Walmart Recruiting - Store Sales Forecasting

Leandro Ferreira, William Walder

October 29, 2017

1 A competição

Nessa competição de recrutamento, os participantes receberam dados históricos de vendas para 45 lojas do Walmart localizadas em diferentes regiões. Cada loja possui vários departamentos, e os participantes deviam projetar as vendas para cada departamento em cada loja, também foi adicionado ao dados a marcação se sequela semana teve algum feriado, esses eventos são conhecidos por afetar as vendas.

1.1 Base de dados

1.1.1 Stores.csv

Este arquivo contem dados sobre as 45 lojas, como tamanho e tipo.

1.1.2 Train.csv e Test.csv

Este arquivo contem para o treinamento do modelo como, ID da loja, departamentos, onde podemos ter ate 99 departamentos para uma loja, vendas semanais e se naquela semana era feriado. O arquivo de teste os mesmos dados, excerto o de vendas semanais.

1.1.3 Features.csv

Este arquivo provem informações sobre as lojas e seus departamentos, entre outras informações relevantes para as vendas.

- Date - Data da semana
- Temperature - Temperatura media na região
- Fuel_Price - Preço de combustível na região da loja
- CPI - indice de consumo por pessoa
- Unemployment - Taxa de desemprego
- IsHoliday - Marcador de quando aquela semana foi feriado
- Markdown 1 a 5 - Marcadores que indicam que a data esta relacionada com algum tipo de promoção no período. Estes marcadores estão disponíveis apos novembro de 2011 mas nem todas as lojas possuem.

1.2 Avaliação

A avaliação para esta competição é erro absoluto médio ponderado (WMAE), dado pela formula:

$$WMAE = \frac{1}{\sum w_i} \sum_{i=1}^{n} w_i |y_i - y'_i| \tag{1}$$

onde:

$$\begin{gathered} n \text{ numero de linhas} \\ y'_i \text{ as vendas preditas} \\ y_i \text{ são os valores verdadeiros} \\ w_i = \begin{cases} 5 & \text{Se } isHolliday = 1 \\ 1 & \text{Demais casos} \end{cases} \end{gathered}$$

2 Analise e processamento dos dados

Table 1: Quantidade de dados ausentes

	Train	**Test**
CPI	0	38162
Taxa de Desemprego	0	38162
MarkDown1	270889	149
MarkDown2	310322	28627
MarkDown3	284479	9829
MarkDown4	286603	12888
MarkDown5	270138	0
Total de Instancias	421570	115064

2.1 Dados retirados

Como demostrado na Tabela 1 havia muitos dados ausentes principalmente para os MarkDown na base de treino, o uso deles poderia ocasionar uma inconsistência no treinamento do nosso modelo. Os dados de CPI e a taxa de desemprego estão ausentes na base de teste, para que fiquemos com os mesmos dados na base de treinamento e de teste eles também foram retirados.

3 Algoritmos utilizados

3.1 Estrategia

Nos optamos fazer o treinamento filtrando os dados da base para cada departamento de cada loja, assim, criando um modelo especializando em apenas um departamento de uma loja especifica.

Reforçamos também os dados de feriado relacionados a cada departamento, replicando os dados de feriados.

3.2 Algoritmos

A árvore de decisão, uma vez construida, seu uso é imediato e muito rápido computacionalmente por isso escolhemos utilizar os algoritmos derivados da mesma, *Random Forest* e *Ada Boost*, pois em nossa estrategia construimos vários modelos que podem fazer a predição para muitos dados

4 Resultados

A especialização em cada departamento resultou em resultados positivos para grande parte dos modelos treinados, como pode ser visto nas Figuras 1 e 2, que demostram a acuraria para os modelos criados para cada departamento. Valores negativos podem ter sido causados por erros de calculos pela função utilizada. Podemos ver uma maior variação de acurácia no algoritmo *Ada Boost* isso pode ser atribuído ao seu treinamento que é feito por algoritmos fracos e suas predições são baseadas no peso médio desses algoritmos, neste caso o algoritmo utilizado foi arvore de decisão, podendo ter causando uma variação maior nos pesos utilizados para a predição.

A Tabela 2 mostra a pontuação conseguida pelos dois algoritmos, que é baseada na Equação 1. Na pontuação Publica do Kaggle o *Ada Boost* teve uma melhor pontuação por ser menos suscetível a perda da capacidade de generalização após o aprendizado de muitos padrões de treino (*overfitting*) do que a maioria dos algoritmos de aprendizado de máquina.

A pontuação do Kaggle é baseada em 50% da base de teste da competição para o publico e a outra metade é utilizada para o privado, ou seja, o algoritmo *Random Forest* teve uma variação menor em seus resultados do que o *Ada Boost*, como foi visto anteriormente pela acurácia obtida.

Table 2: Pontuação na Competição

	Public	**Private**
Ada Boosting	2986.87303	223168108934.14600
Random Forest	3060.09786	3128.09802

5 Conclusão

A estrategia utilizada conseguiu ótimos resultados, provando ser efetiva para a predição de vendas semanais com os dados utilizados. O Algoritmo Random Forest se demostrou ser mais estável para o objetivo.

Em trabalhos futuros pode ser estudado o uso de outros algoritmos de predição para completar a base de teste e assim conseguir trabalhar com mais dados para melhorar o resultado.

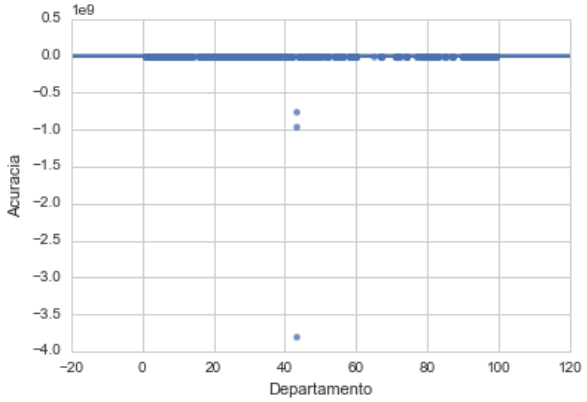

Figure 1: Acuraciao do algoritmo Random Forest.

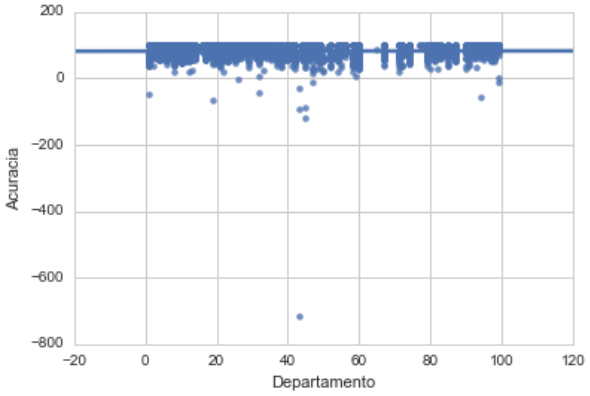

Figure 2: Acuracia do algoritmo Ada Boost.