**PROJETO II – Pós-Graduação Analytics and Data Science Empresarial (ISLA GAIA)**

14-07-2023

Paulo Nunes | Igor Tavares | Gabriel Marchi

1. Enquadramento / Business Understanding

Foi disponibilizada informação de uma empresa cuja atividade é de comércio de roupa em lojas físicas presentes em diversas cidades da Turquia.

A empresa precisa de atualizar o armazenamento de roupa em cada loja para Outubro de 2019 por isso o objetivo principal passa por prever com precisão as vendas semanais de cada loja considerando sazonalidade, tendência e variáveis explicativas complementando com uma análise dos dados que serão utilizados no modelo.

Neste sentido, e seguindo a metodologia CRISP-DM, pretendemos criar modelo (s) de previsão para as lojas e prever as vendas para Outubro de 2019 para cada loja, avaliando se a previsão é próxima do número de vendas real para que a gestão do armazenamento possa ser o mais eficiente possível e potenciar o aumento das vendas/receitas.

1. Data Understanding

Os dados disponibilizados são em formato “csv” e correspondem a três tabelas (“sales”, “product” e “cities”).

Para a análise de dados utilizamos as ferramentas Excel, Python (Spyder e Google Colab) e Rstudio para extrair o melhor entendimento possível da informação presente nos dados.

Primeiro verificamos as variáveis presentes em cada uma das tabelas e o possível relacionamento existente entre elas. Neste caso, as tabelas “cities” e “product” estão relacionadas diretamente com a tabela “sales” pelas variáveis “store\_id” e “product\_id” respectivamente, podendo estas colunas serem consideradas chaves primárias visto que apenas contêm dados únicos (sendo chaves estrangeiras na tabela “sales”).

Para apoiar nesta análise recorremos à biblioteca “pandas\_profiling” do Python (relatórios de cada tabela no repositório) e percebemos o seguinte:

- A tabela “cities” contém 6 variáveis (1 numérica e 6 categóricas) e 63 observações, sem valores em falta:

- Existem 63 lojas classificadas em 4 tipos e com 32 tamanhos entre elas sendo que nos tipos das lojas conseguimos perceber que são pequenas, médias e grandes, havendo uma especial pois é a única presente no tipo ST02.

- As lojas são na Turquia e dispersas por 19 cidades diferentes, no entanto, 32 delas estão na cidade de Istanbul.

- A tabela “product” contém 10 variáveis (3 numéricas e 7 categóricas) e 699 observações, com 100 valores em falta (1,4%)

- Existem registados 699 produtos diferentes segmentados em 10 clusters pela coluna “cluster\_id” (havendo 50 produtos sem segmento e sendo o “cluster\_0” o mais representativo com 450 produtos – 64,4%) que não conseguimos perceber com estes dados os critérios.

- As colunas “product\_length”, “product\_depth” e “product\_width” caracterizam as dimensões do produto e possuem valores em falta (nem todos comuns às 3 colunas) e um registo zero.

- As 5 colunas “hierarchy…” classificam os produtos em vários níveis e nenhuma possui valores em falta.

- A tabela “sales” contém 14 variáveis (6 numéricas e 8 categóricas) e 8.886.058 observações, com 35.271.795 valores em falta (28,4%), mas sem duplicados.

- Possui uma coluna com números sequenciais e que não se repetem.

- “store\_id” com registos de todas as 63 lojas.

- “product\_id” com registos de apenas 615 produtos dos 699 registados na tabela “product”.

- “date” com datas compreendidas entre o dia 02-01-2017 e 31-10-2019.

- “sales” com as quantidades vendidas (mínimo de 0 e máximo de 43301), 3,4% de valores em falta e 79,3% de zeros

- “revenue” com a receita da venda, 3,4% de valores em falta (igual a “sales”) e 79,3% de zeros (1072 zeros a mais que “sales”)

- “stock”com 3,4% de valores em falta(igual a “sales” e “revenue”) e 0,7% de zeros.

- “price” com 606 valores distintos (ou seja, existem produtos com o mesmo preço) e 91381 valores em falta (1972 não apresentam valores em “sales”, “revenue” e “stock” | 69035 com “sales”=0 e “revenue”=0 | 69057 com “revenue”=0 | 22 com “revenue”=0 mas “sales”>0 de produtos, lojas e datas diferentes)

- Características das variáveis “promo”:

* “type\_1” com 17 tipos sendo a PR14 a que se destaca com 86,1% das vendas
* “bin\_1” com 5 características mas havendo 86,1% das vendas sem nada registado
* “type\_2” com 4 tipos sendo 99,9% dos valores do tipo PR03
* “bin\_2” com 3 características mas 99,9% das vendas sem registo
* “discount\_2” com 6 valores diferentes de desconto atribuídos às vendas com registo de “promo\_type\_2”/ “promo\_bin\_2” (ou seja, em 99,9% dos registos não tem valor)
* “discount\_type\_2” com uma classificação em 4 tipos dos descontos/ “promo\_2” anteriormente indicados (mantém os 99,9% de registos sem valor)

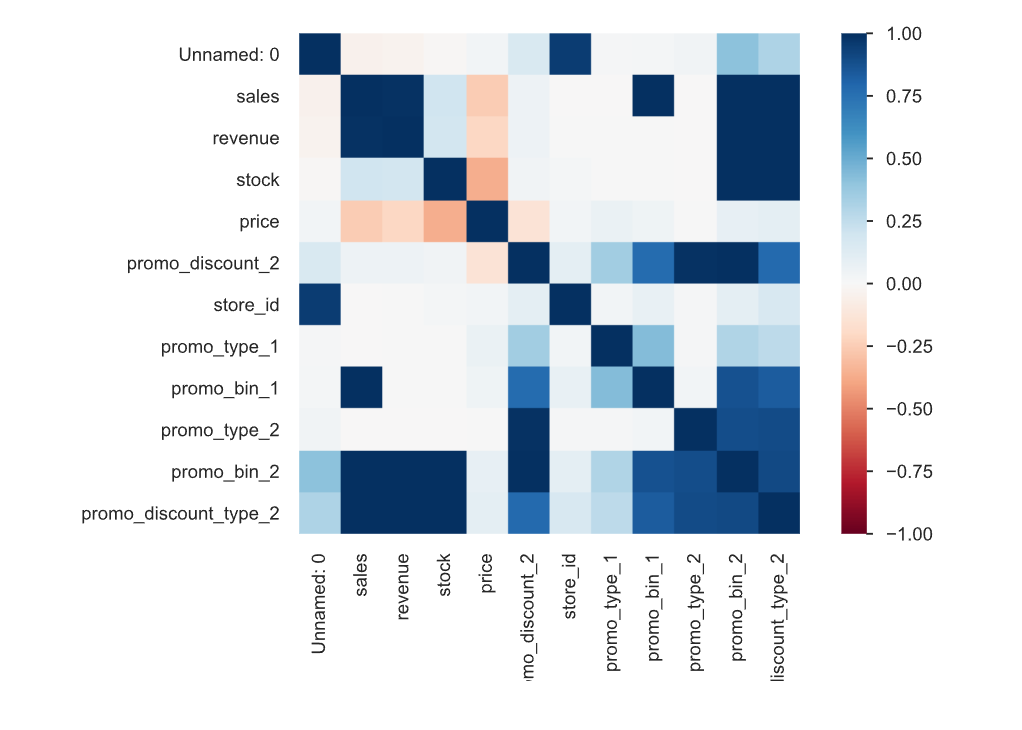
- Correlações:

- As ‘sales’ têm uma forte correlação positiva com a ‘revenue’, o que já seria de esperar

- A correlação de ‘sales’ com ‘stock’ é positiva mas pouco significativa

- O mesmo acontece com ‘sales’ e ‘price’ mas neste caso com correlação negativa

- A correlação entre ‘stock’ e ‘price’ é negativa mas não muito significativa



1. Data Preparation

Para o tratamento e agregação dos dados todos que escolhemos foi utilizado o Python:

- Correção dos nomes das cidades ("?" - substituir por "i")

- Preencher os dados em falta nas colunas “promo\_bin\_1”, “promo\_bin\_2” e “promo\_discount\_type\_2” com “NA” e na coluna “promo\_discount\_2” com “zero”.

- Agregação dos dados das 3 tabelas e eliminação das variáveis que não vamos utilizar: “city\_id\_old”, “Unnamed: 0”, “country\_id”, “hierarchy1\_id”, “hierarchy2\_id”, “hierarchy3\_id”, “hierarchy4\_id”, “hierarchy5\_id”, “product\_length”, “product\_depth”, “product\_width”, “cluster\_id”

- Verificação de nulos e eliminação das linhas que tinham nulos de “sales”, “revenue”, “stock”.

- Criação de uma coluna com o número contínuo de semanas com base na data.

- Eliminação da semana nº144 (última) por apenas ter dados de 1 dia

Como temos lojas com valores de semana muito abaixo de 143 decidimos fazer uma análise particular e verificamos que algumas lojas destas têm o primeiro registo de vendas na semana 53 e depois têm um período sem vendas (sendo esse período diferente de loja para loja) o que nos leva a crer que o primeiro registo se tratou de um teste e que as semanas seguintes sem registos se deveram à preparação da loja para abertura definitiva e por isso eliminamos a semana 53 dessas lojas e utilizamos os dados apenas das semanas seguintes que tinham registos.

Existem também 2 lojas (S0007 e S0059) que começaram a vender em semanas diferentes pelo que utilizamos os dados apenas das semanas seguintes que tinham registos.

Temos ainda o caso especial da loja S0136 que percebemos que não está aberta todo o ano e, pela análise efetuada a loja terá fechado em setembro como habitualmente fez nos períodos homólogos de 2017 e 2018, pelo que a previsão para outubro de 2019 é que esteja fechada e por isso não haja vendas.

Depois deste tratamento dos dados agregamos tudo numa variável e “csv” final.

1. Modeling

Para a criação de modelos iniciamos com a média e média móvel (1trimestre – 13semanas) que testamos para algumas lojas e analisamos os resultados com base no R2, MAE e RMSE, no entanto tendo em conta os resultados e considerando que iremos avançar para um modelo ARIMA que já nos dará indicação de modelos Auto-Regressivos e de Média Móvel, assim como o tempo disponível, resolvemos começar com a criação de modelo ARIMA com a parametrização automática dos parâmetros (auto-arima) e avaliar os resultados.

De seguida começamos a efetuar modelo ARIMA mas com cálculo e definição manual de parâmetros, no entanto, esta opção implica efetuar manualmente por loja e mais uma vez tendo o tempo limitado optamos por seguir com a inclusão da sazonalidade (SARIMA) utilizando o auto-arima e o parâmetro de sazonalidade m=52. De salientar que inicialmente corremos os modelos utilizando as variáveis “sales” e “revenue” de forma a comparar qual delas poderia ser a melhor a utilizar para o fim proposto, no entanto, pela análise efectuada até ao momento optamos por nos centrarmos na “revenue”.

Com estes dados decidimos criar clusters com base na store\_type e nos modelos gerados pelo modelo SARIMA para cada loja agrupando por store\_type e de seguida por modelo gerado e identificamos 36 clusters diferentes havendo a possibilidade de agrupar lojas do mesmo tipo e com o mesmo modelo.

Para além de tornar mais eficiente correr o modelo pretendemos avaliar se o modelo do cluster poderá ser mais preciso do que o de cada loja individual.

A análise gráfica dos erros de cada loja parece revelar uma tendência para a sub-estimação da revenue, ressalvando que em praticamente todas as lojas existem semanas sub e sobre estimadas e ainda que na semana 138 existe um pico que foge bastante à tendência em praticamente todas as lojas, sendo relevante na análise final das previsões e tomada de decisão quanto ao armazenamento a ser efectuado.

Com estes dados pensamos então em acrescentar variáveis exógenas que pudessem ajudar a melhorar a precisão do modelo e criamos/utilizamos as seguintes variáveis:

- stock inicial: calculado….

- Feriados: inseridos ….

- Estações do ano: classificadas como variáveis dummy, sendo o inverno a variável subentendida e classificadas da seguinte forma…

- …

Com o modelo SARIMAX verificamos que, considerando um p-value<0,10:

- O stock inicial é a variável que impacta em mais lojas , de todos os tipos, tamanhos e cidades, mas com coeficientes reduzidos

- A Primavera e o Verão são as que causam um efeito de maior amplitude na revenue, e essencialmente na cidade de Istanbul, sendo o efeito negativo o de maior amplitude.

- Não têm impacto relevante na única loja do tipo ST02

- O impacto, significativo, restringe-se a 7 cidades mas que são geograficamente dispersas

- A maioria das lojas onde tem efeito são de tamanho pequeno e médio/baixo, no entanto, o maior impacto negativo é nas lojas de tamanho maior

- As variáveis Probin1\_low, Probin1\_very\_low, Outono, Probin1\_moderate são as que causam apenas impacto positivo nas lojas também de vários tipos, tamanhos e cidades, em que foram consideradas significativas

Esta análise permitiu perceber que algumas variáveis não são significativas e por isso retiramos do modelo para testar num cluster específico e avaliar e comparar com os resultados do modelo das lojas individuais…

Através das variáveis exógenas foi possível retirar informações que permitem saber como estas influenciam a revenue.A empresa apartir desta informação pode tentar manipular algumas das variáveis de forma a tentar aumentar a revenue.

O modelo Sarimax para o cluster de 8 lojas permite fazer previsõesmais acertadas em relação aos modelos individuais de cada loja.

Num futuro projeto para este tipo de dados seria interessante utilizar um modelo de dados em painel