

Introdução

Neste projeto, deseja-se compreender as operações de uma empresa de e-commerce e criar um sistema automatizado de tomada de decisões que possa ser escalável. A empresa utiliza muitos sistemas para colher e monitorar dados. Entretanto, estes dados se encontram segregados e desconexos. Com isso, torna-se difícil a empresa obter *business insights* relevantes para seu negócio.

Para este projeto foram disponibilizados três conjuntos de dados: *Inventory*, *Orders* e *Shipments*. As principais tarefas desejadas são, integrar os dados, definir uma ontologia para aplicar a lógica de negócio, automatizar o processo de recomendação de reposição de estoque, gerar *insights* de negócio com visualização de tendência e identificar os produtos mais vendidos pela empresa.

Ontologia

Inicialmente, é realizada uma exploração destes conjuntos de dados, como visualização das tabelas, colunas e suas linhas. Esta exploração será utilizada como base para criar a ontologia com as regras de negócio.

Após a exploração, foi definida a ontologia mostrada na Figura 1 utilizando o software Power BI. Conforme as regras de negócio, *order_id* no dataframe *Orders* tem cardinalidade 1 para 1 para *order_id* no dataframe *Shipments*. Esta relação é justificada pelo fato de que cada entrega (*shipment*) é relacionada a uma única ordem. Também foi possível estabelecer uma relação entre os dataframes *Orders* e *Inventory* através da coluna *product_id*. Esta relação tem cardinalidade muitos para muitos, pois a coluna *product_id* não é chave primária em nenhum dos dois dataframes.

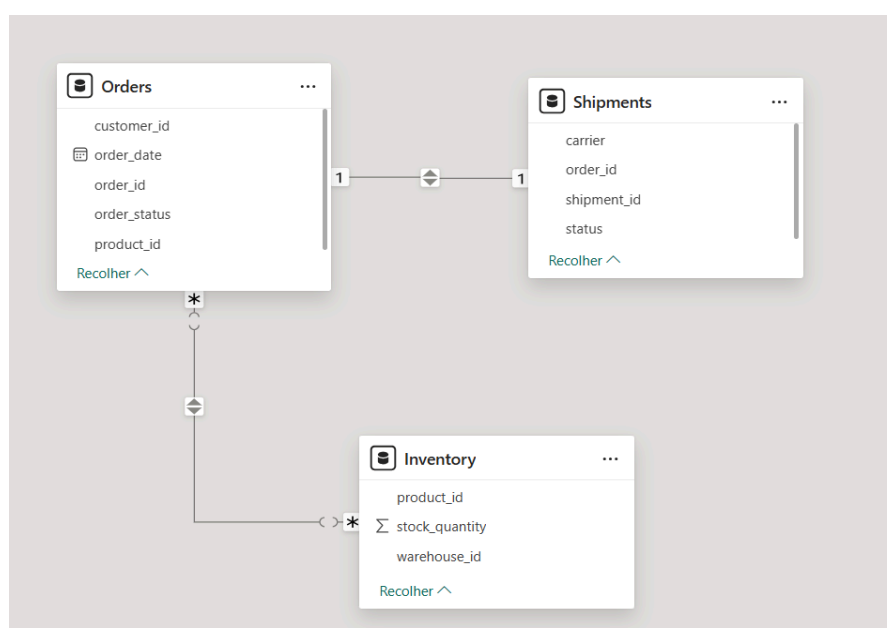


Figura 1 - Ontologia

Tratamento de dados e procura de inconsistências

Com a ontologia previamente definida, deve-se procurar por inconsistência nos dados. Primeiramente, foi realizada uma transformação nos dados de data para que eles se adaptassem ao tipo de variável data da linguagem de programação. Logo após, foi realizada exploração inicial dos dados buscando valores nulos ou entradas duplicadas, não sendo encontrada nenhuma inconsistência deste tipo nos dados.

Logo após, foram realizadas diversas comparações entre as colunas de diferentes dataframes para verificar inconsistências. Nesta etapa foi verificada se existe realmente a relação de 1 para 1 entre os dados do dataframe *Orders* e do dataframe *Shipments*. Foi verificado que existem diversos dados inconsistentes, sendo que existem diversas ordens sem registro de entrega. Além disso, também foi relacionado o *status* entre as ordens e as entregas. Verificou-se também inconsistência nos dados, existindo diferenças entre o *status* de ordens e entregas relacionadas.

Todas estas inconsistências foram apontadas mas não corrigidas. Para sua correção, será necessário trabalhar no sistema de registro de ordens e entregas e verificar a origem de tais erros.

Automação

A próxima etapa do projeto foi automatizar o pedido de repor estoque. De acordo com a regra de negócio, era pedido para analisar as vendas dos últimos 30 dias. Para isto, foi criado um dataframe temporário contendo somente os últimos 30 dias de ordens e contou-se o número de ordens por cada produto. Como não existe a quantidade de produtos em cada ordem, assumiu-se que cada ordem é específica para um único produto.

Com esta informação, foi analisado o dataframe *Inventory*. Nele, procurava-se por faltas de produtos nos diferentes *warehouses*. Caso o estoque de um *warehouse* fosse menor que a quantidade de vendas de um produto nos últimos 30 dias, foi criada uma ordem de compra deste produto de tal forma a garantir, em estoque, a quantidade de vendas do período anterior. Aqui só foi identificada a necessidade de compra de dois produtos:

- Compra de uma unidade do produto P002 para o *warehouse* H01
- Compra de uma unidade do produto P016 para o *warehouse* H03

Business insights

Nesta última etapa do projeto, foi solicitado um ranking dos 5 produtos mais vendidos no trimestre e uma visualização de tendência com os dados de cada estoque e suas demandas. O ranking dos 5 produtos mais vendidos foi criado levando-se em consideração as vendas realizadas nos últimos 3 meses (90 dias) e assumindo novamente que cada ordem é relativa a venda de um único produto. Desta forma, foi realizada a contagem do número de ordens de

cada produto, sendo esta informação salva em uma lista. Para obtermos os produtos mais vendidos, a lista foi ordenada e então suas 5 primeiras posições foram exibidas. O resultado desta ordenação é exibido na Figura 2.

product_id	
P001	8
P003	8
P007	8
P005	7
P012	6

Figura 2 - Ranking dos 5 produtos mais vendidos no trimestre

Para finalizar o projeto, foi criada a visualização de tendência de estoque utilizando um gráfico de linhas. Ele faz uma representação visual que ajuda a identificar padrões e tendências em dados ao longo do tempo, no caso queremos insights de dados ao longo do tempo de estoque nos depósitos ao decorrer de um trimestre. Um exemplo de visualização de estoque para o produto P001 pode ser observado na Figura 3. Esta visualização foi criada como exemplo, mas as visualizações de todos os outros produtos podem ser criadas utilizando-se um pequeno *loop* no código.

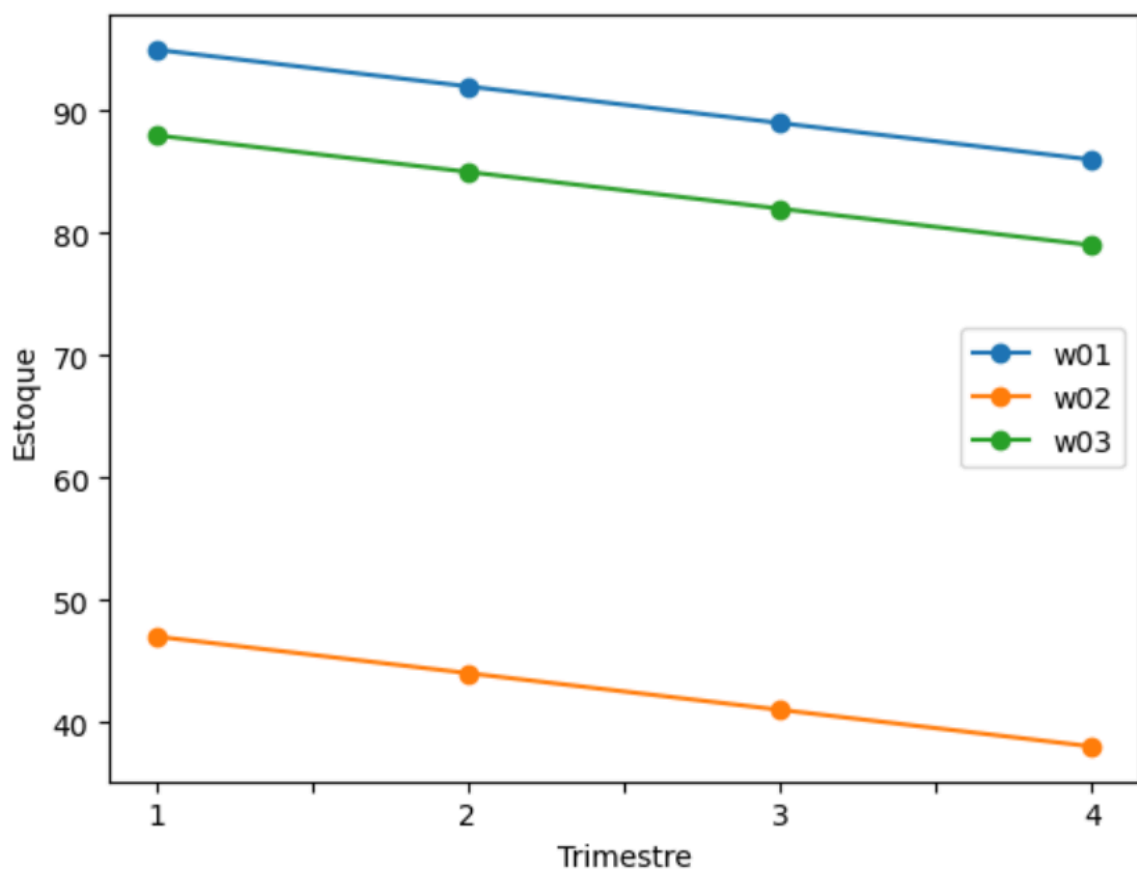


Figura 3 - Projeção de estoque para trimestres posteriores

Na Figura 3 pode-se observar três diferentes linhas, sendo que cada linha corresponde a um *warehouse*. No trimestre 1 representa-se o estado atual de inventário, enquanto nos trimestres posteriores pode ser observado a projeção de estoque de cada *warehouse*. Esta projeção levou em consideração o número de vendas do produto P001 no trimestre anterior dividido por 3 (pelos três *warehouses*). O gráfico foi construído conforme a Tabela 1, onde *media_vendas* corresponde a um terço do número de vendas de P001 no trimestre anterior.

Trimestre	depósito w01	depósito w02	depósito w03
1	estoque	estoque	estoque
2	estoque - <i>media_vendas</i>	estoque - <i>media_vendas</i>	estoque - <i>media_vendas</i>
3	estoque - 2* <i>media_vendas</i>	estoque - 2* <i>media_vendas</i>	estoque - 2* <i>media_vendas</i>
4	estoque - 3* <i>media_vendas</i>	estoque - 3* <i>media_vendas</i>	estoque - 3* <i>media_vendas</i>

Tabela 1 - Lógica para popular o dataframe *dados_trimestre*

Com esta visualização se finaliza o projeto proposto. Com a criação da ontologia e o tratamento de dados, pode-se obter insights, automação do processo de controle e reposição de estoque e um maior controle de status de entregas e atrasos. A visualização de previsões pode preparar outras áreas da empresa para tomadas de decisões mais perspicazes.

Dificuldades e melhorias

Apesar de solicitado, não foi possível avaliar o prazo médio de entrega das diferentes transportadoras, pois o tempo de entrega não estava contido nos conjuntos de dados enviados. Também não foi possível gerar melhores previsões de pedido de recompra de produtos devido ao curto histórico dos dados fornecidos.

Pode-se propor algumas melhorias na ontologia. Por exemplo, o *product_id* no dataframe *inventory* poderia ser uma chave primária. Deste modo, seria possível estabelecer uma relação de 1 para muitos entre os dataframes *orders* e *inventory*, tendo assim melhor controle sobre os dados.

Em geral, faltam mais colunas de datas. Na tabela *shipments*, pode-se criar a coluna de data de entrega para poder ter mais controle de datas quando o produto foi entregue ou em atraso. No *inventory*, poderia existir uma coluna da data que o produto entrou em determinado depósito a data que saiu do depósito. E por fim, deveria ser tratado quando o cliente solicita a devolução de uma ordem. Esta informação não existe na tabela *orders* e nem na tabela *shipments*.