



## Projeto Final

### Integrantes:

**Antero de Vasconcelos Alves Neto  
Jonatas Vasconcelos da Costa Vieira  
Lucas B. A. Machado  
Victor Hugo P Gomes**

### Tutores:

**Luís Roque  
Tiago Otto  
João Reis**

**Data de entrega 05/10/2023**

**Apresentação 07/10/2023**

**Program de Mestrado em Data Science**



## Índice

<b>1. Capa .....</b>	<b>p1</b>
<b>2. Introdução.....</b>	<b>p4</b>
<b>3. Contextualização.....</b>	<b>p19</b>
<b>4. Objetivos.....</b>	<b>p20</b>
<b>5. Metodologia Utilizada.....</b>	<b>p21</b>
<b>6. Resultados e Conclusões.....</b>	<b>p22</b>

Programa de Mestrado em Data Science



#### **Colaboradores DS Market:**

**Nicole, Senior DS  
Justin  
Martin  
Paul Rogers**

#### **Colaboradores Núclio:**

**Antero Neto  
Jonatas Vasconcelos da Costa Vieira  
Lucas B. A. Machado  
Victor Hugo P Gomes**

**NUCLIO**  
**DIGITAL SCHOOL**



## Introdução

### Enquadramento

A DSMarket é uma empresa varejista estabelecida no mercado americano, atualmente passando por um processo de transformação digital e rebranding. A empresa possui operações em três estados: Nova York, Boston e Filadélfia, focando no atendimento ao público por meio de suas lojas. A DSMarket tem como objetivo se tornar a loja de próxima geração nos próximos 5 anos, buscando se destacar no cenário do comércio.

**DSMarket – Caso de Estudo** Este projeto tem como propósito criar um cenário de trabalho realista para um cientista de dados, utilizando três conjuntos de dados. O objetivo é realizar análises, agrupamentos, desenvolver um modelo de previsão de vendas e implementar estratégias de reabastecimento de estoque na loja utilizando técnicas de Machine Learning e operações de Machine Learning (MLops).

- Análise
- Agrupamento
- Modelo de Previsão de Vendas
- Reabastecimento de loja - com MLops



## Data

Tal como referido anteriormente, foram disponibilizados três ficheiros de dados:

- `calendar_with_events.csv`
- `item_prices.csv`
- `item_sales.csv`

Vamos agora ver as variáveis e os seus significados nos respetivos ficheiros.

### Ficheiro `calendar_with_events.csv`

Dados referentes ao calendário

- `date` - data no formato y-m-d
- `weekday` – dia da semana
- `weekday_int` – dia numérico da semana (1-Sábado, 7-Sexta)
- `d` – Identificador do dia
- `event` – o nome do evento, caso tenha

### Ficheiro `item_prices.csv`

Dados referentes aos preços dos itens

- `item` – id do produto
- `category` – categoria do produto
- `store_code` – Código da loja alfanumérico
- `yearweek` – data do período do preço (formato ano-semana)
- `sell_price` – preço de venda do produto referente ao yearweek.



**Observação:** Os valores dos itens são preparados para cada semana, em intervalos de 7 dias. Se não houver um valor listado para um produto, isso significa que não houve vendas desse produto na semana mencionada (ano e semana).

#### Ficheiro item\_sales.csv Dados referentes às vendas

- id – identificador das vendas ( combinação do item + store\_code)
- item – id do produto
- category – categoria do produto
- department – departamento do produto
- store – loja do produto
- store\_code – código da loja
- region – região do produto

#### Tarefa 1 – Análise

De forma a começar a explorar os ficheiros de dados, fomos averiguar quantas observações tinham em cada ficheiro e as suas respetivas dimensões. Constatouse que:

- Dataset de vendas: (30490, 1920)
- Dataset de preços: (6965706, 5)
- Dataset dos eventos: (1913, 5)



De seguida, veremos as primeiras 5 observações de cada dataset:

#### Dataset de Eventos

	date	weekday	weekday_int	d	event
0	2011-01-29	Saturday	1	d_1	NaN
1	2011-01-30	Sunday	2	d_2	NaN
2	2011-01-31	Monday	3	d_3	NaN
3	2011-02-01	Tuesday	4	d_4	NaN
4	2011-02-02	Wednesday	5	d_5	NaN

#### Dataset de preços

	item	category	store_code	yearweek	sell_price
0	ACCESORIES_1_001	ACCESORIES	NYC_1	201328.0	12.7414
1	ACCESORIES_1_001	ACCESORIES	NYC_1	201329.0	12.7414
2	ACCESORIES_1_001	ACCESORIES	NYC_1	201330.0	10.9858
3	ACCESORIES_1_001	ACCESORIES	NYC_1	201331.0	10.9858
4	ACCESORIES_1_001	ACCESORIES	NYC_1	201332.0	10.9858

#### Dataset de vendas

	id	item	category	department	store	store_code	region	d_1	d_2	d_3	...
0	ACCESORIES_1_001_NYC_1	ACCESORIES_1_001	ACCESORIES	ACCESORIES_1	Greenwich_Village	NYC_1	New York	0	0	0	...
1	ACCESORIES_1_002_NYC_1	ACCESORIES_1_002	ACCESORIES	ACCESORIES_1	Greenwich_Village	NYC_1	New York	0	0	0	...
2	ACCESORIES_1_003_NYC_1	ACCESORIES_1_003	ACCESORIES	ACCESORIES_1	Greenwich_Village	NYC_1	New York	0	0	0	...
3	ACCESORIES_1_004_NYC_1	ACCESORIES_1_004	ACCESORIES	ACCESORIES_1	Greenwich_Village	NYC_1	New York	0	0	0	...
4	ACCESORIES_1_005_NYC_1	ACCESORIES_1_005	ACCESORIES	ACCESORIES_1	Greenwich_Village	NYC_1	New York	0	0	0	...



## Observações retiradas dos ficheiros:

### 1. Calendar:

Tem features das datas. É apresentado uma coluna `event` que denomina a presença de um determinado evento no dia ou não. No total existem a presença de 5 eventos: SuperBowl, Ramadan Starts, Thanksgiving, Newyear, Easter.

### 2. Prices:

Têm os preços de venda dos items, como o respetivo código da loja e uma coluna com o ano\_semana.

### 3. Sales:

Estão incluídos todos os id's dos items, como também a categoria, departamento, loja, região e uma determinada coluna de vendas para cada dia, desde 2011-01-29 até 2016-04-24. Temos, portanto, informação referente a 1913 dias.

## Redução do tamanho de dados

### Redução de Tipo (Downcasting)

Nesta seção, vou fazer a redução de tipo dos dataframes para diminuir a quantidade de armazenamento utilizado por eles e também para acelerar as operações realizadas sobre eles.



## Colunas Numéricas:

Dependendo do seu ambiente, o pandas cria automaticamente colunas int32, int64, float32 ou float64 para as numéricas. Se você conhece o valor mínimo ou máximo de uma coluna, você pode usar um subtipo que consome menos memória. Você também pode usar um subtipo sem sinal (unsigned) se não houver valor negativo.

Aqui estão os diferentes subtipos que você pode usar:

int8 / uint8: consome 1 byte de memória, varia entre -128/127 ou 0/255

bool: consome 1 byte, verdadeiro ou falso

float16 / int16 / uint16: consome 2 bytes de memória, varia entre -32768 e 32767 ou 0/65535

float32 / int32 / uint32: consome 4 bytes de memória, varia entre -2147483648 e 2147483647

float64 / int64 / uint64: consome 8 bytes de memória

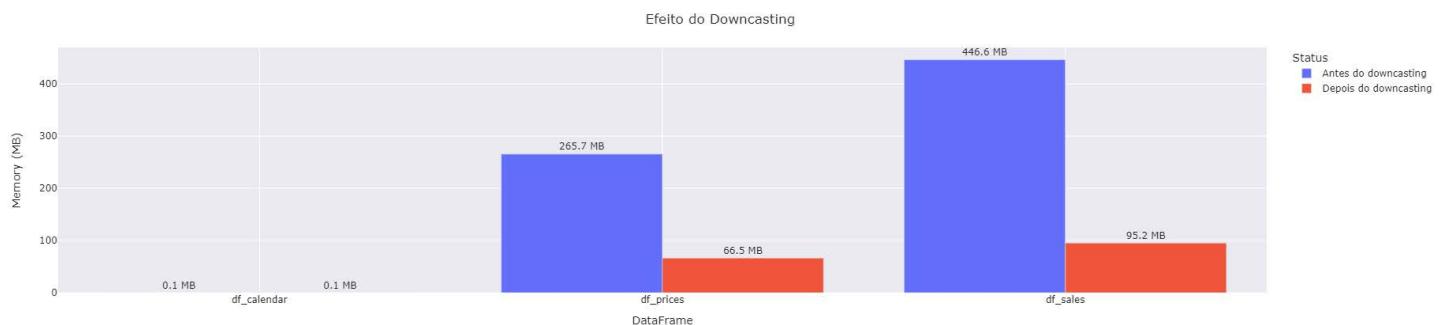
Se uma de suas colunas tem valores entre 1 e 10, por exemplo, você reduzirá o tamanho dessa coluna de 8 bytes por linha para 1 byte, o que é uma economia de memória de mais de 85% nessa coluna!

## Colunas Categóricas:

O pandas armazena colunas categóricas como objetos. Uma das razões pelas quais esse armazenamento não é otimizado é que ele cria uma lista de ponteiros para o endereço de memória de cada valor da sua coluna. Para colunas com baixa cardinalidade (a quantidade de valores únicos é menor que 50% da contagem desses valores), isso pode ser otimizado forçando o pandas a usar uma tabela de mapeamento virtual, onde todos os valores únicos são mapeados através de um número inteiro, em vez de um ponteiro. Isso é feito usando o tipo de dado "category".



O gráfico abaixo mostra o quanto o downcasting afetou o uso de memória dos DataFrames. Claramente, conseguimos reduzir sales e prices para menos de 1/4 de seu uso de memória real. calendar já é um DataFrame pequeno.



## Dicionário de Dados

File 1 – item\_sales.csv

Name	Description
id	sales series id (combination of item + store_code)
item	product id
category	product category
department	department id (different identifier for different stores)
store	store name
store_code	store id
region	region
d_1,d_2,d_...	number of units sold per day

+ Code + Markdown

File 2 – item\_prices.csv

Name	Description
item	product id
category	product category
store_code	alphanumeric code of the store
yearweek	date period for the price (year-week format)
sell_price	price for the product "item" for the period in "yearweek". *

\* Prices are provided per week (average across 7 days). If not available, there were no sales for the product during that week (includes an event, the name of this event (only a few are included)).

File 3 – daily\_calendar\_with\_events.csv

Name	Description
date	date in y-m-d format
weekday	day of the week
weekday_int	numeric day of the week (Saturday day 1, Friday day 7)
d	day identifier
event	if the date includes an event, the name of this event (only a few are included)



## Pré-processamento e exploração de dados

### Observação para coluna 'item':

- Tipo de Dados: object
- Quantidade de Valores Únicos: 3049  
(Mostrando 10 dos 3049 valores únicos)
- Valores Únicos: ['ACCESORIES\_1\_001' 'ACCESORIES\_1\_002' 'ACCESORIES\_1\_003'  
'ACCESORIES\_1\_004' 'ACCESORIES\_1\_005' 'ACCESORIES\_1\_006'  
'ACCESORIES\_1\_007' 'ACCESORIES\_1\_008' 'ACCESORIES\_1\_009'  
'ACCESORIES\_1\_010'] ...
- Quantidade de Valores Nulos: 0
- Contagem de Valores:  
SUPERMARKET\_3\_587 2870  
HOME\_&\_GARDEN\_1\_177 2870  
HOME\_&\_GARDEN\_2\_283 2870  
ACCESORIES\_1\_337 2870  
SUPERMARKET\_1\_032 2870  
...  
HOME\_&\_GARDEN\_1\_308 652  
HOME\_&\_GARDEN\_1\_159 633  
HOME\_&\_GARDEN\_1\_242 610  
SUPERMARKET\_3\_296 602  
SUPERMARKET\_2\_379 543

Name: item, Length: 3049, dtype: int64

### Observações para a coluna 'category':

- Tipo de Dados: object
- Quantidade de Valores Únicos: 3
- Valores Únicos: ['ACCESORIES' 'HOME\_&\_GARDEN' 'SUPERMARKET']
- Quantidade de Valores Nulos: 0
- Contagem de Valores:  
SUPERMARKET 3239821  
HOME\_&\_GARDEN 2418627  
ACCESORIES 1307258

Name: category, dtype: int64



### **Observações para a coluna 'store\_code':**

- Tipo de Dados: object
- Quantidade de Valores Únicos: 10
- Valores Únicos: ['NYC\_1' 'NYC\_2' 'NYC\_3' 'NYC\_4' 'BOS\_1' 'BOS\_2' 'BOS\_3' 'PHI\_1' 'PHI\_2' 'PHI\_3']
- Quantidade de Valores Nulos: 0
- Contagem de Valores:

```
BOS_2    713960
BOS_1    712527
NYC_1    711073
PHI_3    708747
NYC_3    706585
BOS_3    703682
NYC_4    691375
PHI_2    690546
PHI_1    678209
NYC_2    649002
Name: store_code, dtype: int64
```

### **Observações para a coluna 'date':**

- Tipo de Dados: datetime64[ns]
- Quantidade de Valores Únicos: 1913
- Valores Únicos: ['2011-01-29T00:00:00.000000000' '2011-01-30T00:00:00.000000000'
 '2011-01-31T00:00:00.000000000' ... '2016-04-22T00:00:00.000000000'
 '2016-04-23T00:00:00.000000000' '2016-04-24T00:00:00.000000000']
- Quantidade de Valores Nulos: 0

```
- Contagem de Valores:
2011-01-29    1
2014-07-23    1
2014-08-04    1
2014-08-03    1
2014-08-02    1
..
2012-10-24    1
2012-10-23    1
2012-10-22    1
2012-10-21    1
2016-04-24    1
Name: date, Length: 1913, dtype: int64
```



### Observações para a coluna 'sell\_price':

- Tipo de Dados: float64
- Quantidade de Valores Únicos: 1892  
(Mostrando 10 dos 1892 valores únicos)
- Valores Únicos: [12.7414 10.9858 11.1454 5.2801 3.9501 5.7722 6.1712 3.9634 3.2984 4.0964] ...

- Quantidade de Valores Nulos: 0

- Estatísticas Básicas:

```
count    6.965706e+06
mean     5.518273e+00
std      4.387861e+00
min      1.200000e-02
25%     2.620100e+00
50%     4.200000e+00
75%     7.176000e+00
max     1.341500e+02
Name: sell_price, dtype: float64
```

- Contagem de Valores:

```
2.3760    221088
3.5760    217873
...
16.5750      1
2.2800      1
Name: sell_price, Length: 1892, dtype: int64
```

### Tratamento da coluna yearweek

Temos uma grande quantidade de dados faltantes na coluna yearweek.

A porcentagem de dados faltantes para a coluna 'yearweek' é 3.5%.

Isso corresponde a 243920 de 6965706 registros.

A coluna yearweek possui datas faltando no ano de 2016, que vai da semana 18 a semana 25. no total de 8 datas por item, como são 3090 items e os items se repetem pelas 10 lojas, por isso temos um total de 243920 datas em falta.



## Total de vendas por lojas

Podemos observar que a região de Nova York se destaca tanto no aspecto de vendas mais altas por loja quanto nas vendas mais baixas. As lojas que registram o maior número de vendas são NYC\_3 e NYC\_1, respectivamente, enquanto NYC\_4 mostra um desempenho inferior.

Além disso, é notável que as lojas em Filadélfia e Boston têm números de vendas comparáveis.



## Vendas totais por categorias

### Categoria ACCESORIES:

ACCESSORIES\_1 apresenta um nível de vendas superior ao ACCESSORIES\_2  
ACCESSORIES\_2 apresenta um comportamento constante e a tender para zero

### Categoria HOME\_&\_GARDEN:

HOME\_&\_GARDEN\_2 apresenta um crescimento exponencial de vendas e superior ao HOME\_&\_GARDEN\_1

HOME\_&\_GARDEN\_1 apresenta um comportamento constante e com pouca variabilidade no que toca ao número de vendas realizado

### Categoria SUPERMARKET:

Todos os departamentos do SUPERMARKET apresentam um comportamento crescente ao longo dos anos

SUPERMARKET\_2 e SUPERMARKET\_1 apresentam um comportamento semelhante e com pouco variabilidade

SUPERMARKET\_3 é o departamento que apresenta uma maior taxa de crescimento, no entanto também apresenta a maior variabilidade nas vendas



## Observações:

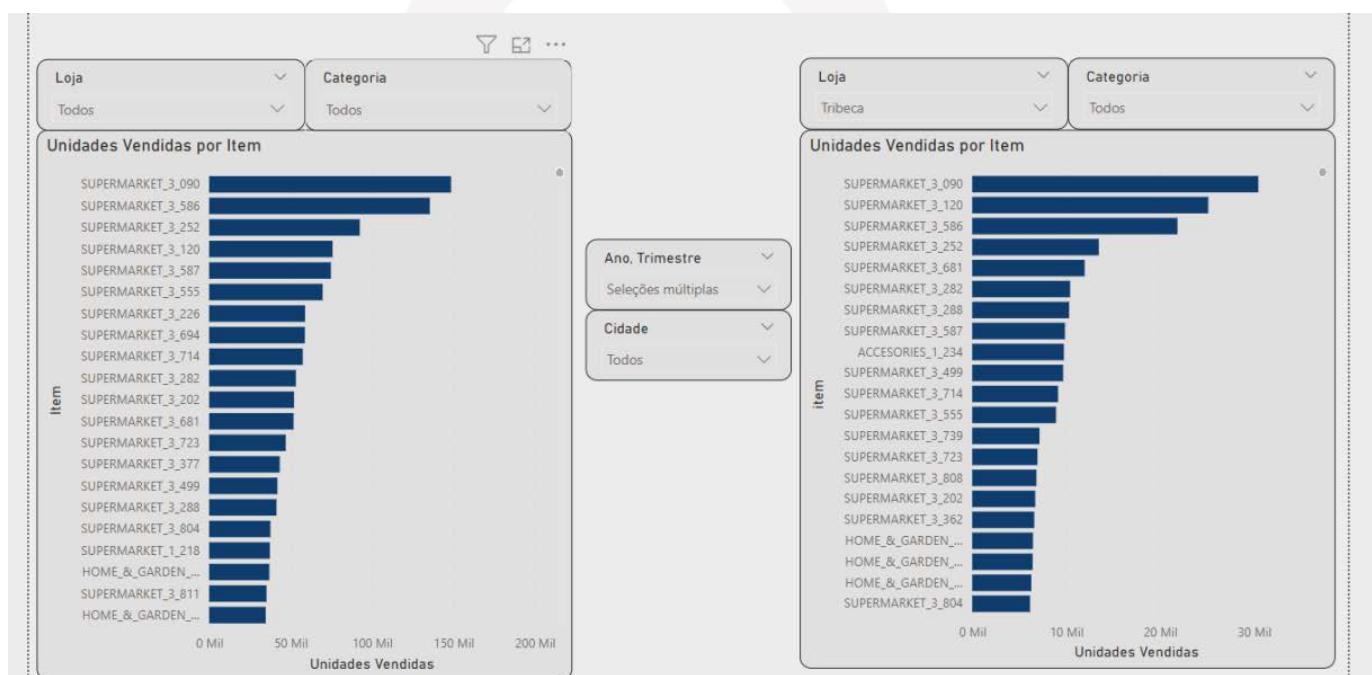
A região de New York apresenta o maior número de vendas e a região de Philadelphia o menor número de vendas

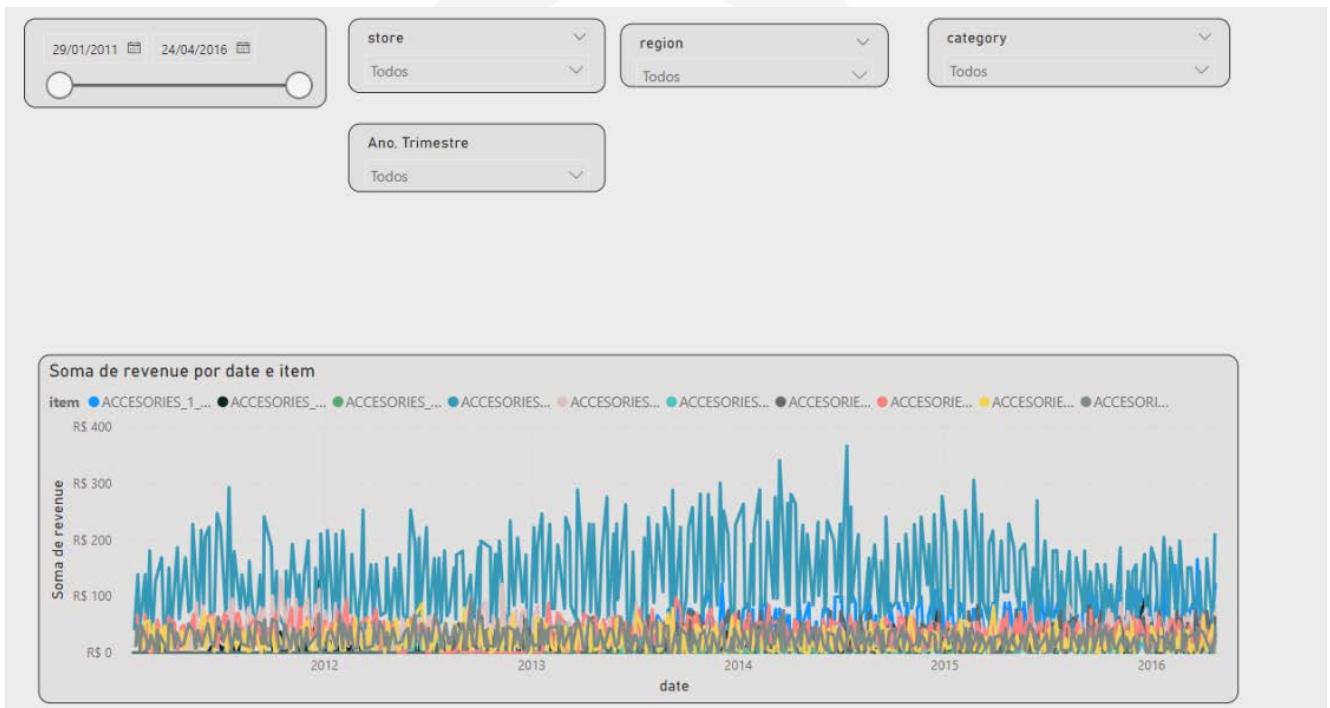




## Business Intelligence









## Contextualização

O presente trabalho de mestrado tem como foco a DSMarket e consiste na criação de um cenário realista de trabalho para um cientista de dados. Para alcançar esse objetivo, foram fornecidos três conjuntos de dados que servirão de base para a realização de diversas tarefas, todas relacionadas à análise de dados e aprendizado de máquina.

Este trabalho de mestrado visa não apenas criar soluções analíticas para a DSMarket, mas também preparar um cenário prático para um cientista de dados, onde ele ou ela enfrentará desafios reais relacionados à análise de dados, modelagem preditiva e implementação de sistemas de aprendizado de máquina em operações de negócios.



## Objetivos

### Tarefa 1

Análise de Dados Exploratórios

Analítica

### Tarefa 2

Identifique grupos de produtos que comportam-se de uma maneira semelhante

Agrupamento

### Tarefa 3

Previsão de vendas

Serries temporais

### Tarefa 4

Reabastecimento de Loja

Regressão



## Metodologia Utilizada

**Análise de Dados:** Esta fase envolve a exploração dos dados fornecidos para entender melhor o perfil do cliente, as tendências de compra e outros insights que podem ser úteis para a empresa. A análise de dados pode incluir a criação de visualizações, estatísticas descritivas e identificação de padrões.

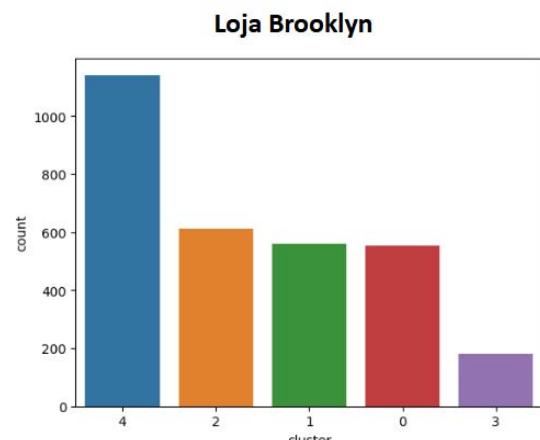
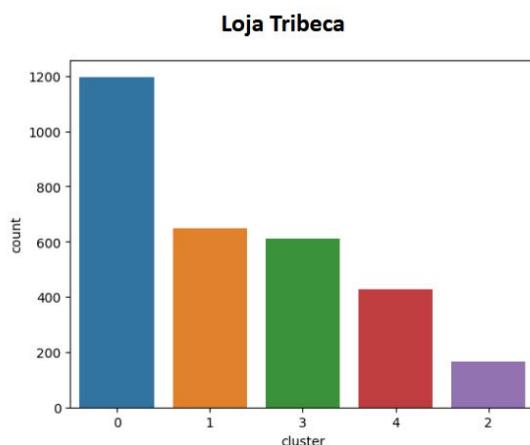
**Agrupamento:** Nesta etapa, o cientista de dados pode aplicar técnicas de agrupamento, como clusterização, para segmentar os clientes ou produtos em grupos com características semelhantes. Isso pode ajudar a empresa a personalizar estratégias de marketing e oferecer produtos mais direcionados.

**Modelo de Previsão de Vendas:** O desenvolvimento de um modelo de previsão de vendas é fundamental para a DSMarket. Usando os dados históricos de vendas e outras variáveis relevantes, o cientista de dados pode construir um modelo de aprendizado de máquina capaz de prever as vendas futuras. Isso auxilia na gestão de estoque e planejamento financeiro.

**Reabastecimento de Loja com MLOps:** A implementação de um sistema de reabastecimento de loja baseado em Machine Learning Operations (MLOps) pode otimizar o estoque da DSMarket. Isso envolve a automação do processo de reabastecimento com base nas previsões de vendas, evitando estoques excessivos ou insuficientes.

## Resultados e Conclusões:

A loja Tribeca parece ser mais bem estabelecida com os clusters de produtos dentro do esperado. A loja Brooklyn parece ser mais nova já que o catálogo possui muitos itens recentes.



Após um profundo estudo e implantação de modelos preditivos para otimizar o abastecimento de lojas, identificamos diversos pontos críticos e soluções para aprimorar a eficiência no setor varejista. A utilização de dados históricos de vendas como base para a previsão de demandas demonstrou ser uma estratégia altamente eficaz, permitindo uma gestão de estoque mais precisa e minimizando os custos associados a excedentes e faltas de produtos.

Ao iniciar o projeto piloto na loja Tribeca, conseguimos capturar insights valiosos que poderão ser aplicados em outras lojas, devido ao seu alto volume de vendas e receita. A segmentação de produtos com base em recência e frequência mostrou-se uma ferramenta valiosa para estratégias de venda, precificação e catálogo.

Além disso, enfatizamos a importância da explicabilidade dos modelos. É vital que os tomadores de decisão compreendam como os modelos funcionam, quais variáveis são mais influentes e como interpretar as previsões. Esta transparência não só fortalece a confiança nas decisões baseadas em dados, mas também facilita a adaptação e ajuste das estratégias de negócios conforme necessário.



**Cluster 1 - Itens Ganhando ou perdendo popularidade:**

Cluster com recência média e uma ampla variação na frequência: eles podem ser produtos emergentes (parte inferior esquerda do cluster) ou produtos em declínio (parte superior direita do cluster).

**Cluster 0 - Demais Itens:**

Cluster com frequência média e baixa recência: esses são produtos comuns, devendo ser a maioria dos produtos.

**Cluster 3 - Itens Mais Populares:**

Cluster com alta frequência e baixa recência: esses são os produtos mais quentes/melhor vendidos.

**Cluster 2 - Itens de baixa popularidade:**

Cluster com alta recência e uma ampla variação na frequência: eles podem ser produtos ruins que nunca venderam (parte inferior do cluster) ou produtos antigos (parte esquerda do cluster) que costumavam vender, mas agora não vendem mais.

**Cluster 4 - Itens Novos:**

Cluster com baixa frequência e baixa recência: são produtos novos/que estão aparecendo.



A integração das ferramentas sugeridas promete aprimorar ainda mais a eficiência na implantação e gerenciamento dos modelos. Contudo, é crucial considerar a segurança dos dados e dos próprios modelos de ML, garantindo a privacidade dos dados e protegendo-os contra possíveis ataques adversários.

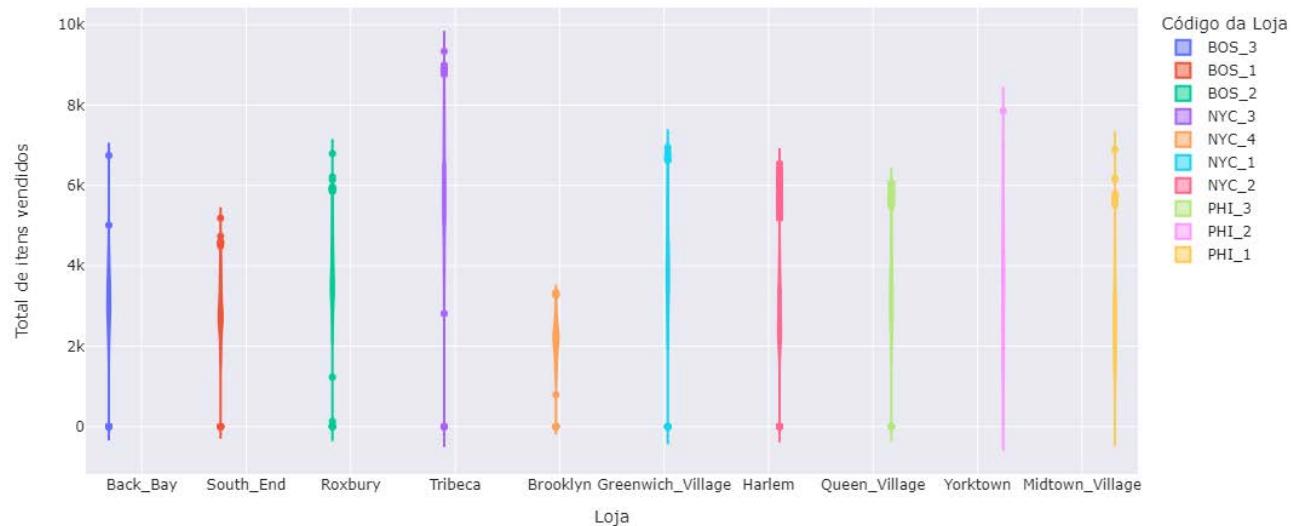
Em suma, a aplicação de modelos preditivos no abastecimento de lojas promete revolucionar a forma como o setor varejista opera, levando a um aumento nas vendas, melhorias na eficiência operacional e uma experiência do cliente mais satisfatória.

New York: Tribeca vendeu o maior número de itens, enquanto Brooklyn vendeu o menor número de itens.  
 Boston : Back\_Bay e Roxbury venderam o maior número de itens. South\_End vendeu o menor número de itens.

Philadelphia : Yorktown vendeu o maior número de itens, enquanto Queen\_Village vendeu o menor número de itens.

EUA: Tribeca vendeu o maior número de itens, enquanto Brooklyn vendeu o menor número de itens.

Distribuição de Itens Vendidos por Loja



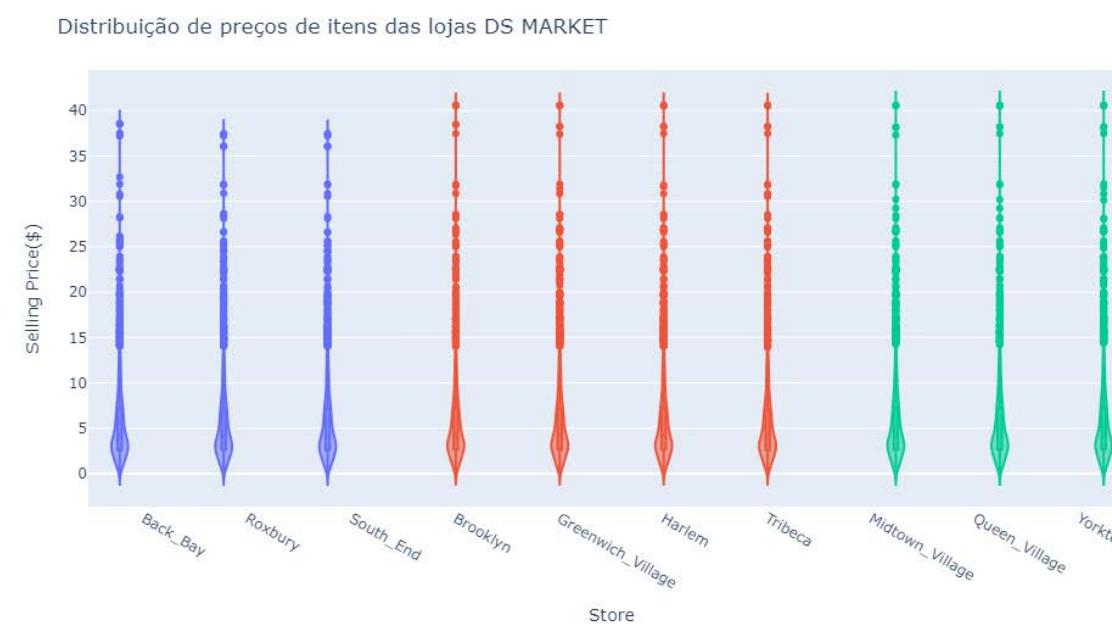
Abaixo estão algumas das observações a partir do gráfico acima:-

A distribuição dos preços dos itens é quase uniforme para todas as lojas em Boston, Nova York e Philadelphia.

O item ACESSORIES\_1\_225 precificado em cerca de 38,54 dólares é o item mais caro sendo vendido nas DS MARKET em BOSTON, nas lojas de Roxbury e South\_End o item mais caro muda

O item ACESSORIES\_1\_060 precificado em cerca de 40,58 dólares é o item mais caro sendo vendido nas DS MARKET em Nova York.

O item ACESSORIES\_1\_225 precificado em cerca de 40,57 dólares é o item mais caro sendo vendido em Midtown\_village ,Queen\_Village e Yorktown na Philadelphia





Detalhas da receita acumulada por cidades:



revenue (\$ millions)

store	region		
Tribeca	New York	3.944224e+07	\$39.44MM
Greenwich_Village	New York	2.786773e+07	\$27.87MM
Roxbury	Boston	2.539238e+07	\$25.39MM
Back_Bay	Boston	2.206267e+07	\$22.06MM
Yorktown	Philadelphia	2.174855e+07	\$21.75MM
Harlem	New York	2.159434e+07	\$21.59MM
Queen_Village	Philadelphia	2.089039e+07	\$20.89MM
South_End	Boston	1.945117e+07	\$19.45MM
Midtown_Village	Philadelphia	1.829981e+07	\$18.30MM
Brooklyn	New York	1.510968e+07	\$15.11MM

Tendência semelhante nas três regiões os picos de venda acontecem no começo da semana (Segunda-feira e Terça-feira).





Maiores vendas observadas em Agosto (Aug), com pequenas quedas no mês de Novembro (Nov) e Maio (May).

#### Observações

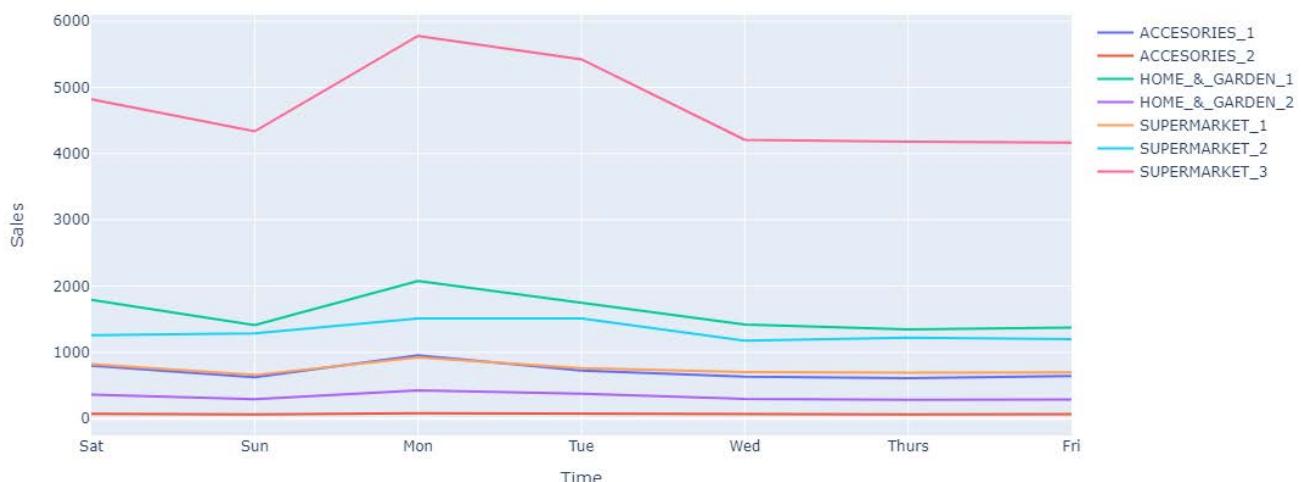
As tendências semanais para todas as regiões e departamentos foram quase as mesmas

As vendas nos finais de semana foram maiores para todos os departamentos

É possível observar uma queda nas vendas nos meses de Maio (May) em todos os departamentos independente de região

Em Boston e New York as maiores vendas foram observadas em Agosto (Aug) e para Philadelphia foi observada a maior quantidade de vendas em Fevereiro (Feb)

Weekly Seasonalities of depts in Philadelphia





#### Avaliação sobre a Predição Média da Loja (Average Store Prediction):

Na análise do modelo de predição média da loja, observamos que a característica `rollin_sold_mean_7` emergiu como a mais dominante, contribuindo com a maior porcentagem para a importância das features. Isso indica que a média móvel de vendas de 7 dias é um indicador crucial para entender e prever o comportamento das vendas na loja. Esse padrão sugere que as vendas da semana anterior são um bom indicador para as vendas futuras, possivelmente devido à consistência nos padrões de compra do cliente ou a promoções e eventos recorrentes.

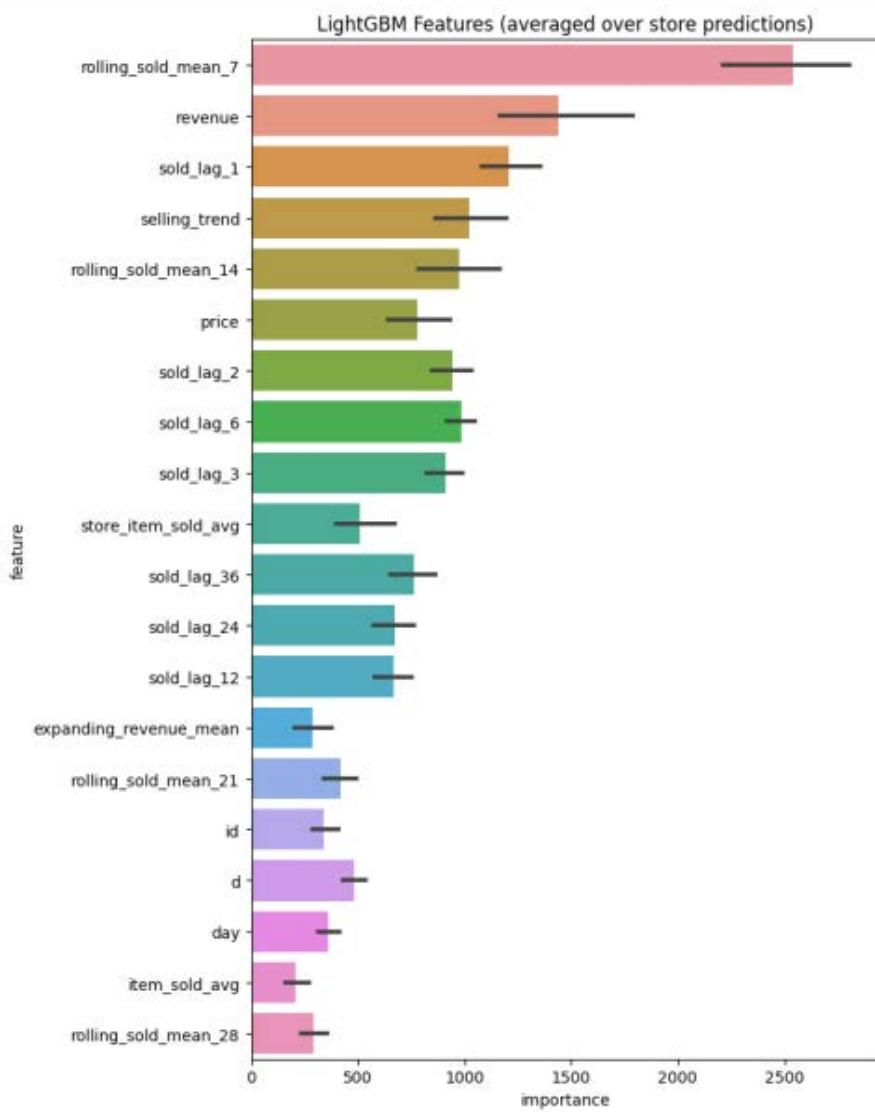
Revenue, ou receita, apareceu como a segunda característica mais importante. Esta é uma constatação intuitiva, já que a receita geralmente é um reflexo direto das vendas e tendências da loja. Um alto volume de vendas, geralmente, traduz-se em receitas elevadas, o que torna este fator uma métrica significativa na previsão.

A característica `sold_log_1` também demonstrou importância significativa, indicando que as vendas logarítmicas do dia anterior são relevantes para a predição. Esta transformação logarítmica pode ajudar a estabilizar variações e é especialmente útil quando lidamos com picos extremos nas vendas.

Por fim, selling trend mostrou-se relevante, sugerindo que as tendências de venda ao longo do tempo, possivelmente influenciadas por fatores sazonais, promoções ou eventos, desempenham um papel vital nas previsões de vendas.

Em conclusão, para melhorar a precisão das previsões médias da loja, é essencial dar um peso adequado a estas características identificadas, adaptando-se continuamente às novas tendências e padrões emergentes no comportamento das vendas.

## Média das previsões da loja





### Resultado modelo LGBMRegression

	Store	MAE	R2	RMSE	Department
0	Greenwich_Village	0.078575	0.990361	0.340967	NaN
1	Harlem	0.077275	0.981523	0.418990	NaN
2	Tribeca	0.116118	0.985211	0.610474	NaN
3	Brooklyn	0.031887	0.990642	0.182215	NaN
4	South_End	0.056478	0.986005	0.362765	NaN
5	Roxbury	0.066198	0.988868	0.374152	NaN
6	Back_Bay	0.060464	0.985733	0.393595	NaN
7	Midtown_Village	0.055099	0.981426	0.344455	NaN
8	Yorktown	0.108375	0.977937	0.725880	NaN
9	Queen_Village	0.078382	0.983795	0.487264	NaN
10	NaN	0.032575	0.985948	0.274856	ACCESORIES_1
11	NaN	0.010619	0.986433	0.092536	ACCESORIES_2
12	NaN	0.054973	0.991420	0.260687	HOME_&_GARDEN_1
13	NaN	0.011660	0.950858	0.216822	HOME_&_GARDEN_2
14	NaN	0.064910	0.948959	0.671722	SUPERMARKET_1
15	NaN	0.065104	0.973577	0.496967	SUPERMARKET_2
16	NaN	0.097264	0.991456	0.510381	SUPERMARKET_3



## Caso de uso de abastecimento da loja MLOps

### Resumo Executivo:

Nossa proposta visa melhorar a eficiência no abastecimento de lojas por meio da aplicação de modelos preditivos de vendas. Utilizando dados históricos de vendas, pretendemos desenvolver um sistema automatizado que preveja as demandas de produtos para cada loja, otimizando assim os estoques e reduzindo custos associados ao excesso ou falta de produtos.

Para iniciar, propomos o piloto na loja Tribeca que é a loja que tem o maior volume de vendas e receita.

### Introdução:

O abastecimento de lojas é uma operação crítica para o sucesso de qualquer varejista. O desafio reside em equilibrar estoques para evitar a falta de produtos, que pode resultar na perda de vendas, e evitar excessos que levam a custos adicionais de armazenamento e deterioração. Para enfrentar esse desafio, propomos a implementação de preditores de vendas.

### Objetivos:

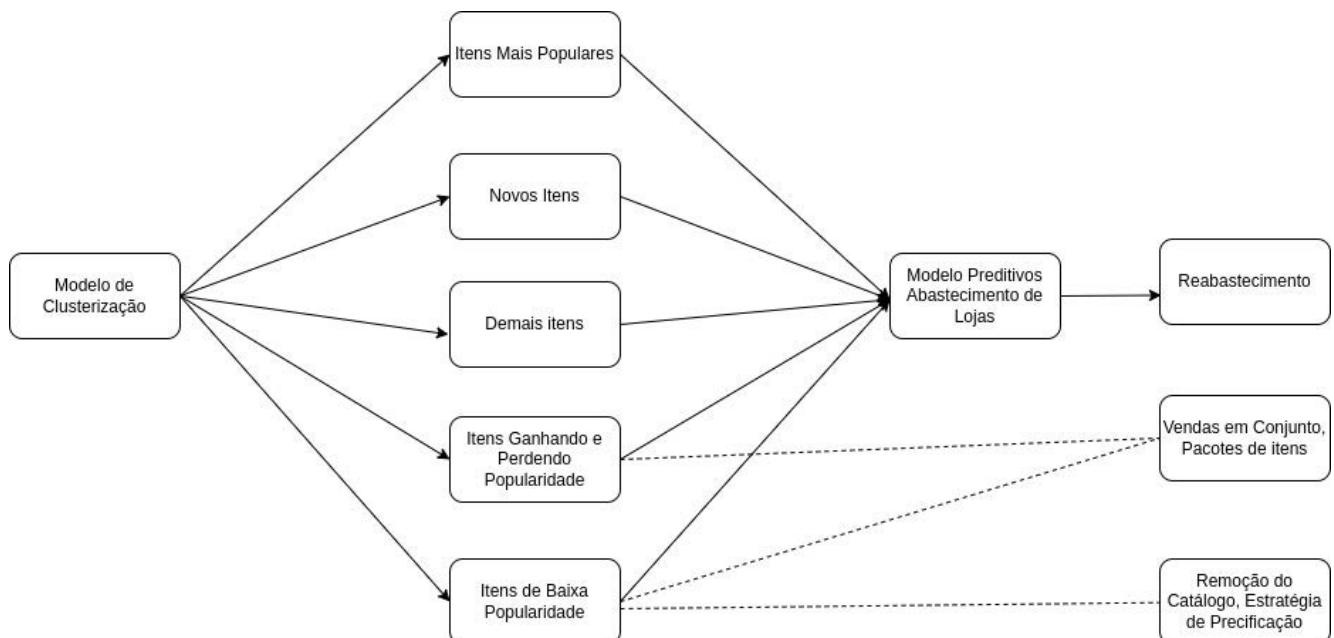
Desenvolver modelos preditivos de vendas personalizados para cada loja, levando em consideração históricos de vendas, eventos especiais, localização da loja e outros fatores relevantes.

Criar um sistema automatizado que gerencie o abastecimento com base nas previsões, otimizando os níveis de estoque.

Reducir os custos associados a estoques em excesso e à perda de vendas por falta de produtos. Melhorar a satisfação do cliente, garantindo que os produtos estejam disponíveis quando e onde são necessários.

### Fluxograma do caso de uso:

No diagrama abaixo podemos ver o fluxo da aplicação dos modelos neste caso de uso:



Utilizamos os modelos de clusterização para seguimentar os produtos de acordo com a recencia e a frequência, já extraíndo insights para orientar estratégias de vendas precificação e catálogo. Depois utilizamos o modelo de previsão de vendas para orientar o reabastecimento, também utilizando as informação de seguimentação.

#### Cronograma previsto:

Name	Progress %	Sep, 2023		Oct, 2023					Nov, 2023					Dec, 2023		
		25 Sep	01 Oct	08 Oct	15 Oct	22 Oct	29 Oct	05 Nov	12 Nov	19 Nov	26 Nov	03 Dec	10 Dec			
Planejamento	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Segurança de ML	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Integração Contínua (CI)	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Entrega Contínua (CD)	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Testes e Validação Automatizados	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Versionamento de Modelos	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Implantação Gradual (Rollout)	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Monitoramento Contínuo	50		<div style="width: 50%; background-color: blue;"></div>		<div style="width: 50%; background-color: green;"></div>		<div style="width: 50%; background-color: yellow;"></div>		<div style="width: 50%; background-color: orange;"></div>		<div style="width: 50%; background-color: red;"></div>		<div style="width: 50%; background-color: blue;"></div>			
Retroalimentação e Aprendizado Contínuo	0		<div style="width: 0%; background-color: blue;"></div>		<div style="width: 0%; background-color: green;"></div>		<div style="width: 0%; background-color: yellow;"></div>		<div style="width: 0%; background-color: orange;"></div>		<div style="width: 0%; background-color: red;"></div>		<div style="width: 50%; background-color: green;"></div>			



### **Benefícios Esperados:**

Redução de custos de estoque em excesso.  
Aumento das vendas devido à disponibilidade adequada de produtos.  
Melhorias na eficiência operacional.

### **Implantação:**

#### **Explicabilidade dos modelos:**

Neste caso vamos levar em consideração as seguintes

Interpretação de Características: Utilizaremos técnicas que destacam quais características ou variáveis estratégicas de vendas precificação e catálogo, mais influenciando as previsões de vendas, permitindo uma visão clara dos fatores-chave que afetam as decisões de abastecimento.

Visualização de Modelos: Criaremos gráficos e visualizações interativas que permitirão aos usuários explorar como as previsões são geradas e como elas se relacionam com os dados históricos.

Documentação Detalhada: Prepararemos documentação abrangente que descreve o funcionamento dos modelos, suas limitações e como as previsões são calculadas. Isso garantirá que todos os envolvidos possam compreender e confiar nas previsões.

#### **Implantação dos modelos:**

Após a fase de desenvolvimento dos modelos, inicia-se a fase de implantação observando-se:

#### **Performance dos modelos:**

Neste caso são utilizadas técnicas de compressão como, Quantization, Pruning e Distillation, bem como paralelização da execução dos treinamentos sempre que possível.

#### **Ferramentas que podem ser utilizadas:**

A seleção dos modelos dependerá da compatibilidade com as tecnologias atualmente utilizadas na empresa e recursos como orçamento previsto, cronograma previsto e tempo para implantação.

#### **1. Orquestração e Gerenciamento de Pipelines:**

Apache Airflow: Uma plataforma de orquestração de fluxo de trabalho que pode ser usada para agendar e monitorar tarefas de treinamento e implantação de modelos.

Kubeflow Pipelines: Um sistema baseado em Kubernetes para a criação, execução e gerenciamento de pipelines de machine learning.



MLflow: Uma plataforma de código aberto para gerenciar o ciclo de vida de modelos de machine learning, incluindo treinamento, rastreamento de experimentos e implantação.

## **2. Gerenciamento de Modelos:**

MLflow: Além da funcionalidade de pipeline, o MLflow oferece recursos de gerenciamento de modelos, incluindo registro e versionamento de modelos.

DVC (Data Version Control): Uma ferramenta para rastrear e gerenciar versões de dados e modelos de machine learning.

ModelDB: Uma plataforma de gerenciamento de modelos de código aberto que ajuda a rastrear e gerenciar modelos em produção.

## **3. Treinamento de Modelos:**

TensorFlow: Uma popular biblioteca de código aberto para treinamento de modelos de machine learning e deep learning.

PyTorch: Outra biblioteca de deep learning amplamente usada para treinamento de modelos.

Scikit-learn: Uma biblioteca de machine learning em Python que fornece ferramentas simples e eficazes para treinamento de modelos tradicionais de machine learning.

## **4. Implantação de Modelos:**

Kubernetes: Uma plataforma de orquestração de contêineres que é frequentemente usada para implantar modelos em contêineres para produção.

Docker: Uma plataforma de contêineres que pode ser usada para empacotar e implantar modelos de machine learning em contêineres isolados.

Seldon: Uma plataforma de implantação de modelos em Kubernetes que simplifica a implantação e o gerenciamento de modelos em produção.

## **5. Monitoramento e Monitoramento de Modelos:**

Prometheus: Um sistema de monitoramento e alerta de código aberto que pode ser usado para rastrear o desempenho de modelos em produção.

Grafana: Uma plataforma de análise e visualização que pode ser integrada ao Prometheus para criar painéis de monitoramento personalizados.



Kubeflow Katib: Uma ferramenta de otimização automatizada que pode ser usada para otimizar hiperparâmetros de modelos em tempo real.

#### **6. Gerenciamento de Implantação e Entrega:**

Jenkins: Uma ferramenta de automação de código aberto que pode ser usada para criar pipelines de CI/CD para modelos de machine learning.

GitLab CI/CD: Uma funcionalidade integrada no GitLab para criar pipelines de CI/CD para implantação de modelos.

CircleCI: Uma plataforma de CI/CD que também pode ser usada para automatizar a implantação de modelos de machine learning.

#### **Segurança de ML:**

Levando em consideração a Privacidade dos dados: A privacidade é uma preocupação fundamental quando se trata de machine learning. Os modelos de ML muitas vezes dependem de grandes conjuntos de dados para aprender e tomar decisões, e garantir que esses dados sejam protegidos contra acesso não autorizado e vazamento é essencial. Ataques adversários, Adversários podem tentar manipular deliberadamente os modelos de ML. Isso pode incluir ataques de envenenamento de dados, onde dados maliciosos são inseridos nos conjuntos de treinamento, ou ataques de falsificação, onde tentativas são feitas para enganar o modelo. Segurança do modelo, os próprios modelos de machine learning podem ser alvos de ataques. Isso pode incluir ataques de força bruta para descobrir pesos do modelo ou ataques de injeção de código para explorar vulnerabilidades.

Treinamento e conscientização, as equipes que desenvolvem e implantam sistemas de machine learning devem ser devidamente treinadas e conscientizadas sobre os riscos de segurança e as melhores práticas de mitigação.

#### **Integração contínua (Continuous Integration - CI) e entrega contínua (Continuous Delivery – CD)**

Automatização de Pipelines: Ao adotar o CI/CD, as organizações criam pipelines de ML automatizados que incorporam etapas de treinamento, validação, implantação e monitoramento. Isso permite que as atualizações de modelos sejam implementadas de forma eficiente e repetível.



**Integração Contínua (CI):** A CI em ML Ops envolve a integração frequente de novos códigos, atualizações de dados e ajustes de modelos em um repositório compartilhado. Isso desencadeia a automação de testes de qualidade, treinamento de modelos e avaliações, ajudando a identificar problemas precocemente.

**Entrega Contínua (CD):** A CD em ML Ops concentra-se na entrega automatizada de modelos treinados em produção. Isso garante que os modelos sejam implantados rapidamente e sem interrupções, mantendo a consistência entre o ambiente de desenvolvimento e produção.

**Testes e Validação Automatizados:** A automação de testes e validações é essencial para garantir que os modelos funcionem conforme o esperado e atendam aos requisitos de negócios. Isso inclui testes de unidade, testes de regressão e validação em dados de produção simulados.

**Versionamento de Modelos:** Assim como o controle de versão de código-fonte é crucial na CI/CD tradicional, o versionamento de modelos é igualmente importante em ML Ops. Isso permite rastrear as mudanças nos modelos e retroceder a versões anteriores, se necessário.

**Implantação Gradual (Rollout):** A CD em ML Ops geralmente envolve a implantação gradual de modelos atualizados para minimizar riscos. Isso permite que problemas sejam detectados antes que afetem todo o ambiente de produção.

**Monitoramento Contínuo:** Após a implantação, o monitoramento contínuo é essencial. Métricas de desempenho, qualidade e segurança dos modelos são monitoradas em tempo real, ajudando a identificar desvios e problemas de forma proativa.

**Retroalimentação e Aprendizado Contínuo:** A CI/CD em ML Ops promove um ciclo de feedback constante. As informações obtidas durante a operação dos modelos são usadas para ajustar os pipelines, melhorar os modelos e aprimorar o processo de desenvolvimento.

### **Governança em MLOps:**

#### **Políticas e Normas:**

Estabelecer políticas e normas claras que definam como os modelos de machine learning devem ser desenvolvidos, implantados e monitorados. Isso pode incluir requisitos de privacidade, segurança e conformidade regulatória.

#### **Aprovação e Revisão:**



Implementar um processo de aprovação e revisão para modelos de machine learning antes de serem implantados em produção. Isso garante que os modelos atendam aos padrões estabelecidos.

**Auditoria de Modelos:**

Realizar auditorias regulares nos modelos para garantir que eles estejam em conformidade com regulamentações, padrões éticos e requisitos de qualidade.

**Controle de Versão:**

Mantenha um controle rigoroso de versões de modelos, código e dados. Isso permite rastrear alterações, facilitar reversões em caso de problemas e manter a transparência.

**Ética:**

Estabeleça diretrizes éticas para o desenvolvimento de modelos de machine learning, garantindo que eles não perpetuem preconceitos, discriminação ou qualquer outro tipo de injustiça.

**Transparência:**

Forneça documentação detalhada sobre os modelos, incluindo informações sobre como foram treinados, as métricas de desempenho, os dados de treinamento e outros aspectos relevantes.

**Conclusão:**

A aplicação de preditores de vendas no abastecimento de lojas é uma estratégia fundamental para varejistas modernos que buscam otimizar suas operações. Este projeto proposto tem o potencial de melhorar significativamente a eficiência e a lucratividade, garantindo que os produtos certos estejam no lugar certo e no momento certo. Estamos ansiosos para discutir esta proposta em mais detalhes e colaborar com sua equipe para tornar esse projeto uma realidade.