Recompra Bázica

Kaiki Mello dos Santos

Análise exploratória dos dados

Verificando informações sobre o Dataset

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17028 entries, 0 to 17027
Data columns (total 10 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   ID_Cliente         17026 non-null  Int64
 1   Data               17028 non-null  datetime64[ns]
 2   ID_Produto         16625 non-null  Int64
 3   Descrição_Produto  17028 non-null  object
 4   Quantidade         17028 non-null  int64
 5   Preço_Unitário     16804 non-null  float64
 6   ID_Pedido          17028 non-null  int64
 7   Desconto           16984 non-null  float64
 8   Frete              17028 non-null  float64
 9   Total_do_Pedido    17028 non-null  float64
dtypes: Int64(2), datetime64[ns](1), float64(4), int64(2), object(1)
memory usage: 1.3+ MB
```

Como é percebido, existem dois pedidos com id de cliente nulos que equivalem a 0.01%, então irei analisar se existem outros pedidos de venda com o mesmo código do pedido para substituir o ID de cliente nulo desses pedidos, caso contrário irei optar por remover esses dados devido que a quantidade pedidos não irá influenciar tanto no resultado final.

Verificando informações sobre os dados de Data

	Data
count	17026
unique	636
top	2022-10-14 00:00:00
freq	484
first	2020-08-19 00:00:00
last	2022-12-31 00:00:00

Vemos que as compras online foram feitas no períodode 19-08-2020 a 31-12-2022.

Prevendo compra do cliente

O objetivo desta seção é criar um modelo usando o dataframe df_data fornecido, para estimar se um determinado cliente comprará algo novamente na loja online.

O dataframe é dividido em dois:

- O primeiro sub-dataframe atribuído à variável Python cliente_comp_dt contém compras feitas por clientes de 19-08-2020 a 30-09-2022. Este dataframe será usado para estudar as compras comportamentais dos clientes online.
- O segundo sub-dataframe atribuído à variável Python cliente_prox_tri será usado para estudar as compras comportamentais dos clientes no próximo trimestre. Ou seja, de 01-10-2022 a 31-12-2022, com ênfase de localizar os clientes que irão comprar no período **01-01-2023** e **14-01-2023**.

Descobrimos a primeira data de compra do trimestre e a ultima data de compra anterior ao trimestre pra calcular o próximo dia de compra do usuário.

	ID_Cliente	UltDataCompraAnteriorAoTrismestre	PrimeiraDataCompraTrimestre
0	1	2021-05-27	NaT
1	2	2021-05-26	NaT
2	3	2021-05-17	NaT
3	4	2021-05-17	NaT
4	5	2021-05-16	NaT

Então obtemos o próximo dia de compra:

	ID_Cliente	ProximoDiaCompra
0	12429082030	9999.0
1	12793619292	9999.0
2	1	9999.0
3	2	9999.0
4	12732289533	27.0

Em seguida, definiremos alguns recursos e os adicionaremos ao dataframe dados_clientes para construir nosso modelo de aprendizado de máquina.

Usaremos o método de segmentação Recência - Frequência - Valor Monetário. Ou seja, vamos colocar os clientes em grupos com base no seguinte:

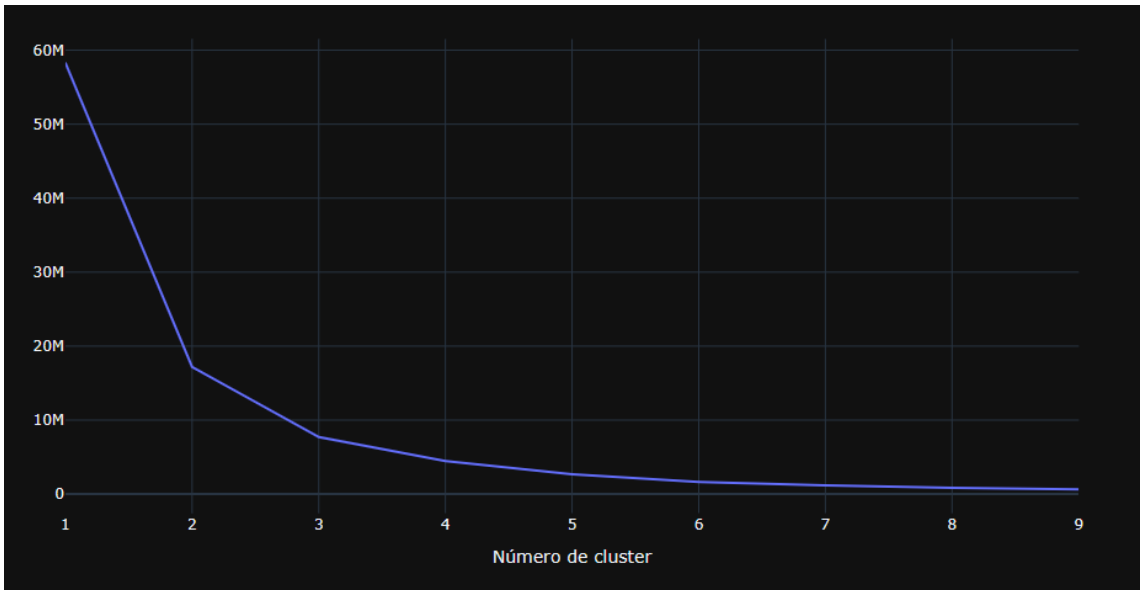
- Recência: o comportamento de compra dos clientes com base na data de compra mais recente e quantos dias eles ficaram inativos desde a última compra.
- Frequência: Comportamento de compra dos clientes com base no número de vezes que compram na loja de varejo online.
- Valor/receita monetária: comportamento de compra dos clientes com base na receita que geram.

Depois, aplicaremos o agrupamento de K-means para atribuir aos clientes uma pontuação para cada um dos recursos.

	Recencia
count	2448.000000
mean	160.996732
std	154.353439
min	0.000000
25%	50.000000
50%	109.000000
75%	256.250000
max	772.000000

A recência média é de aproximadamente 161 dias, enquanto a mediana é de 109 dias.

No passo seguinte, apliquei o agrupamento K-means para atribuir uma pontuação de recência. No entanto, precisamos saber quantos clusters para usar o algoritmo K-means. Aplicaremos o método Elbow para determinar quantos clusters precisaremos. O método Elbow simplesmente informa o número de cluster ideal para a inércia ideal.



Irei usar o número de clusters igual a 4. Após obter a pontuação de recência verifico os dados com describe.

RecenciaCluster	count	mean	std	min	25%	50%	75%	max
0	143.0	591.657343	90.296078	455.0	507.0	613.0	672.0	772.0
1	506.0	316.063241	52.516475	231.0	281.0	291.5	346.0	452.0
2	787.0	138.179161	39.473052	90.0	108.0	125.0	169.0	226.0
3	1012.0	40.353755	26.243050	0.0	15.0	44.0	60.0	89.0

Observe acima que 3 cobre os clientes mais recentes, enquanto 0 tem os clientes mais inativos. O mesmo é calculado para Frequência e Receitas.

FrequenciaCluster	count	mean	std	min	25%	50%	75%	max
0	55.0	33.727273	10.124145	26.0	27.0	31.0	36.5	86.0
1	163.0	17.748466	3.077885	14.0	15.0	17.0	20.0	25.0
2	591.0	8.585448	2.220641	6.0	7.0	8.0	10.0	13.0
3	1639.0	2.273948	1.278162	1.0	1.0	2.0	3.0	5.0

Como foi o caso da Recência, maior número de frequência significa melhores clientes.

Mesmo raciocínio é aplicado para Receita.

TotalPedidoCluster	count	mean	std	min	25%	50%	75%	max
0	1721.0	274.236729	165.961921	0.0	119.01	238.00	376.00	672.00
1	573.0	1070.700489	305.723094	675.0	803.00	1013.89	1300.88	1785.95
2	129.0	2510.312946	507.539641	1797.0	2098.00	2370.41	2894.24	3892.70
3	25.0	5340.499600	1466.003006	4124.0	4559.20	4998.00	5405.80	11313.90

Agora iremos finalmente somar os recursos do cluster.

PontuacaoGeral	Recencia	Frequencia	Total_do_Pedido
2	382.250000	13.000000	462.632500
3	538.267442	3.436047	319.029128
4	295.368321	3.721374	406.600840
5	126.289786	5.731591	631.033076
6	42.660292	6.780652	809.213003
7	49.062500	6.062500	1045.003750
8	190.000000	5.000000	5593.000000

A pontuação acima nos mostra claramente que os clientes com pontuação menor que 4 tem valor baixo, os que tem valor médio ficam entre 4 e 6 e os de valor alto entre 7 e 8.

Como nosso objetivo é estimar se um cliente fará uma compra nas próximas duas semanas, criaremos uma nova coluna NextPurchaseDayRange com valores como 1 ou 0 definidos da seguinte forma:

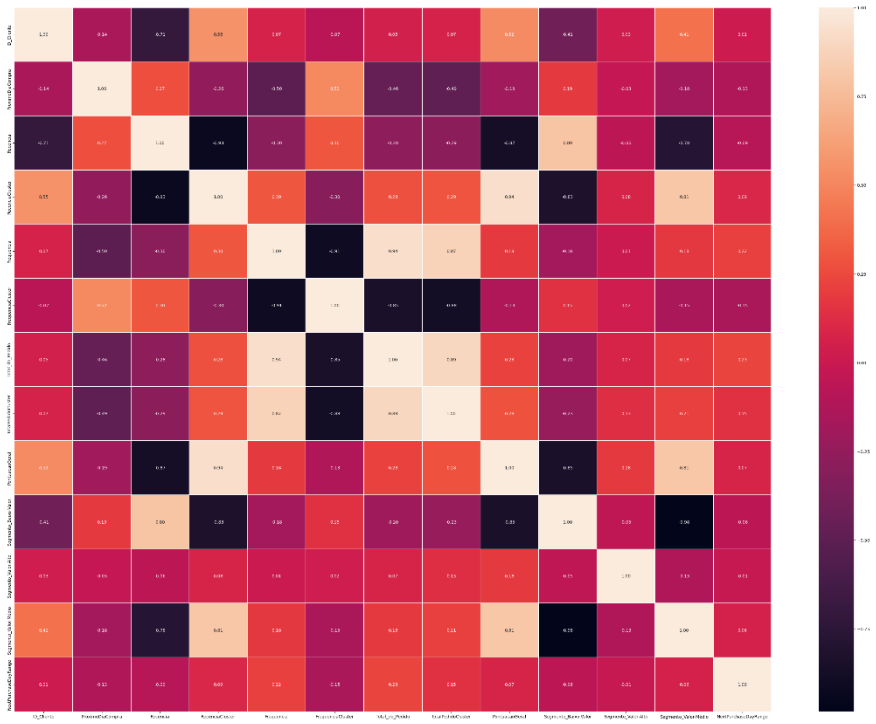
- Se o valor for 1, então indica que o cliente vai comprar algo nas próximas 2 semanas, ou seja, 14 dias a partir da última compra.
- O valor 0 indica que o cliente comprará algo em mais de 14 dias a partir de sua última compra.

Finalmente nesta seção, vamos ver a correlação entre nossos recursos e rótulo

	MinCorrelationCoeff	MaxCorrelationCoeff
ID_Cliente	-0.706256	0.553122
ProximoDiaCompra	-0.503995	0.517740
Recencia	-0.931819	0.797785
RecenciaCluster	-0.931819	0.935684
Frequencia	-0.913483	0.937441
FrequenciaCluster	-0.913483	0.517740
Total_do_Pedido	-0.845218	0.937441
TotalPedidoCluster	-0.883088	0.892710
PontuacaoGeral	-0.868093	0.935684
Segmento_Baixo-Valor	-0.983257	0.797785
Segmento_Valor-Alto	-0.129933	0.179651
Segmento_Valor-Médio	-0.983257	0.811554
NextPurchaseDayRange	-0.146663	0.231941

Na saída acima, observamos que a Total_do_Pedido e Frequencia tem a maior correlação positiva de 0,94, e com Segmento_Baixo-Valor e Segmento_Valor-Médio tem a maior correlação negativa de -0,99.

visualização da matriz de coeficientes abaixo.



Construindo modelo de aprendizado de máquina

Nesta seção, tenho o que é preciso em relação aos pré-requisitos necessários para construir o modelo de aprendizado de máquina. Depois, divido X e y para obter os conjuntos de dados de treinamento e teste e, em seguida, meço a precisão, F_1-score, recall e precisão dos diferentes modelos.

```
LogisticRegression [0.9912854  0.99346405]
GaussianNB [0.9912854  0.99346405]
RandomForestClassifier [0.99237473 0.9956427 ]
SVC [0.9912854  0.99346405]
DecisionTreeClassifier [0.98474946 0.99346405]
xgb.XGBClassifier [0.99019608 0.99455338]
KNeighborsClassifier [0.9912854  0.99346405]
```

A partir dos resultados da Figura acima, vemos que o modelo Random Forest está entre os melhores em termos de precisão de métricas e F_1-score.

```
Precisão do classificador Random Forest no conjunto de treinamento: 1.00
Precisão do classificador Random Forest no conjunto de teste: 0.99
              precision    recall  f1-score   support

           0       0.99      1.00      1.00       607
           1       0.00      0.00      0.00         5

    accuracy                           0.99       612
   macro avg       0.50      0.50      0.50       612
weighted avg       0.98      0.99      0.99       612
```

Por fim realizo as previsões com o modelo. Abaixo a lista com a previsão.

	ID_Cliente	predicoes
0	12429082030	0
1	12920724665	0
2	13148663300	0
3	12706822040	0
4	12383340726	0
...	...	...
2443	15662353010	0
2444	15692298038	0
2445	15758948769	1
2446	15762013256	0