

projeto1

April 2, 2024

Lucas Szavara Nusp 12690087

Maria Victória Brandão Barros 12608692

<https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce>

```
[2]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import pingouin as pg
import seaborn as sns
import scipy.stats as stats
from sklearn import datasets, svm, naive_bayes, metrics, model_selection, tree, \
    neighbors, ensemble
from sklearn.model_selection import train_test_split
from statsmodels.miscmodels.ordinal_model import OrderedModel
plt.rcParams['figure.figsize'] = (15, 8)
```

```
[3]: df_reviews = pd.read_csv('./archive/olist_order_reviews_dataset.csv')
df_items = pd.read_csv('./archive/olist_order_items_dataset.csv')
df_payments = pd.read_csv('./archive/olist_order_payments_dataset.csv')
df_orders = pd.read_csv('./archive/olist_orders_dataset.csv')
df_customer = pd.read_csv('./archive/olist_customers_dataset.csv')
df_products = pd.read_csv('./archive/olist_products_dataset.csv')
df_geolocation = pd.read_csv('./archive/olist_geolocation_dataset.csv')
df_sellers = pd.read_csv('./archive/olist_sellers_dataset.csv')
```

0.1 Perguntas:

0.1.1 1 – Quantas categorias de produtos foram comercializadas e quais os atributos mais relevantes?

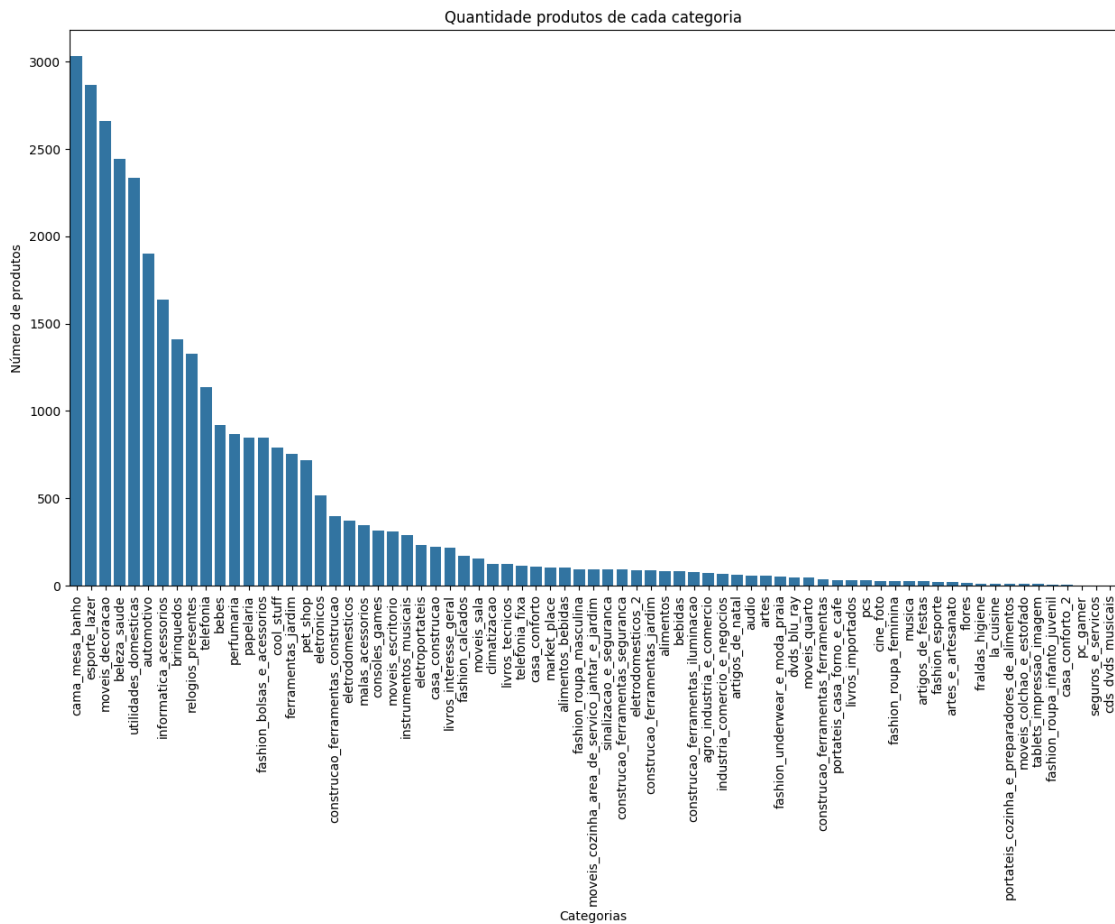
```
[ ]: df_products['product_category_name'].nunique()
```

```
[ ]: 73
```

0.1.2 2 – Qual a categoria de produtos mais vendida?

```
[3]: plt.xticks(rotation=90)
sns.countplot(df_products, x='product_category_name', order =_
↳df_products['product_category_name'].value_counts().index)
plt.title("Quantidade produtos de cada categoria")
plt.xlabel("Categorias")
plt.ylabel("Número de produtos")
```

```
[3]: Text(0, 0.5, 'Número de produtos')
```



```
[ ]: df_order_products = (df_orders.merge(df_items, on='order_id')
    .merge(df_products, on='product_id'))
df_order_products['product_category_name'].value_counts(dropna=False)
```

```
[ ]: product_category_name
cama_mesa_banho      11115
beleza_saude          9670
esporte_lazer         8641
```

```

moveis_decoracao      8334
informatica_acessorios 7827
...
cds_dvds_musicais      14
la_cuisine             14
pc_gamer               9
fashion_roupa_infanto_juvenil 8
seguros_e_servicos     2
Name: count, Length: 74, dtype: int64

```

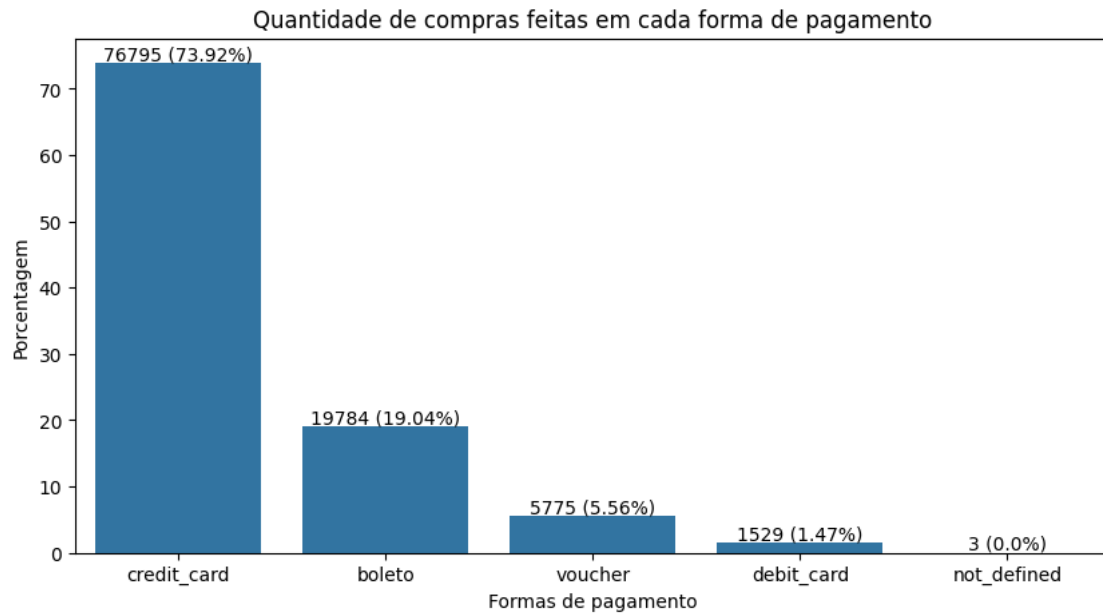
```
[ ]: del df_order_products
```

0.1.3 3 – Qual a porcentagem de compras feitas com cartão de crédito? Neste caso, qual é o número médio de parcelas?

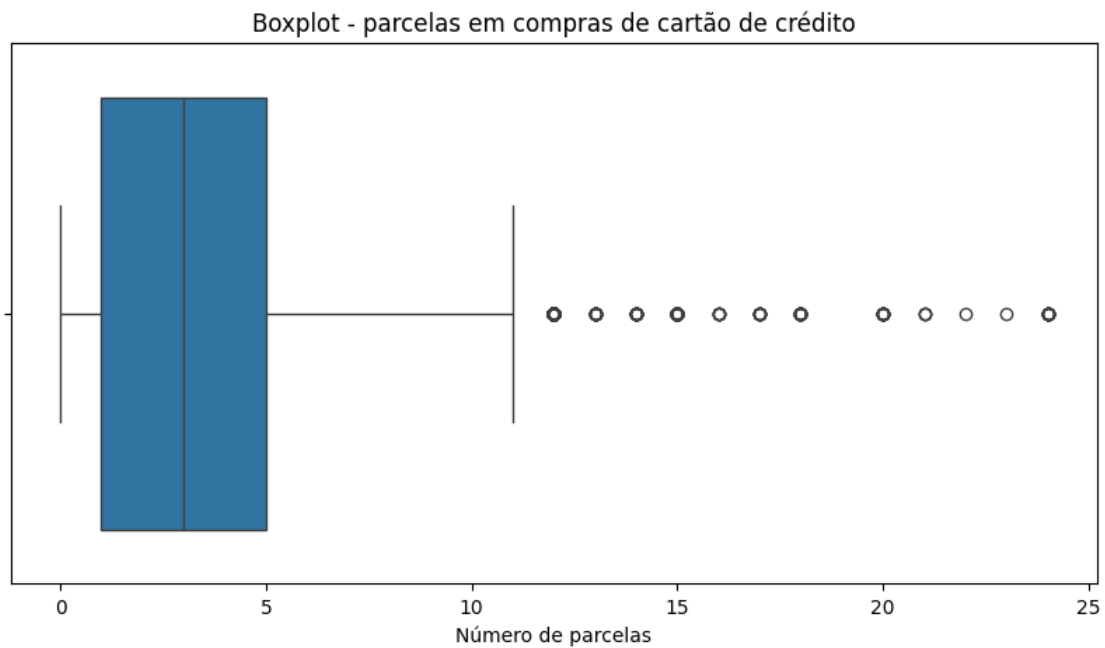
```
[4]: plt.rcParams['figure.figsize'] = (10, 5)
```

```
[5]: df_order_payments = df_orders.merge(df_payments, on='order_id')
df_pay = pd.DataFrame(df_order_payments['payment_type'].value_counts()).
    ↪reset_index()
```

```
[6]: ax = sns.countplot(df_order_payments, x='payment_type', stat='percent', order =
    ↪df_order_payments['payment_type'].value_counts().index)
i = 0
for p in ax.patches:
    height = p.get_height()
    text = (str(df_pay['payment_type'][i]) + ' (' + str(p.get_height().round(2))
    ↪+ '%)')
    ax.text(p.get_x()+p.get_width()/2., height + 0.5, text, ha="center")
    i+=1
plt.title("Quantidade de compras feitas em cada forma de pagamento")
plt.xlabel("Formas de pagamento")
plt.ylabel("Porcentagem")
plt.show()
```



```
[7]: sns.boxplot(df_order_payments[df_order_payments['payment_type'] == 'credit_card'], x='payment_installments')
plt.title("Boxplot - parcelas em compras de cartão de crédito")
plt.xlabel("Número de parcelas")
plt.show()
```



```
[ ]: df_order_payments[df_order_payments['payment_type'] ==  
    ↳ 'credit_card']['payment_installments'].describe()
```

```
[ ]: count    76795.000000  
    mean      3.507155  
    std       2.850990  
    min       0.000000  
    25%       1.000000  
    50%       3.000000  
    75%       5.000000  
    max       24.000000  
    Name: payment_installments, dtype: float64
```

0.1.4 4 – O valor das compras fica em média em torno de qual valor em reais?

```
[ ]: df_order_payments['payment_value'].describe()
```

```
[ ]: count    103886.000000  
    mean      154.100380  
    std       217.494064  
    min       0.000000  
    25%       56.790000  
    50%       100.000000  
    75%       171.837500  
    max       13664.080000  
    Name: payment_value, dtype: float64
```

```
[8]: sns.histplot(df_order_payments, x='payment_value')  
    plt.title("Valor das compras")  
    plt.xlabel("Valor")  
    plt.ylabel('')  
    plt.show()
```



```
[9]: del df_order_payments
```

0.1.5 5 – Qual a porcentagem de clientes satisfeitos com a realização da compra (review score = 4 ou 5)

```
[10]: df_order_reviews = df_orders.merge(df_reviews, on='order_id', how='left')
df_order_reviews['review_score'] = df_order_reviews['review_score'].astype(str)
df_order_reviews.loc[df_order_reviews['review_score'] == 'nan', 'review_score'] =
    ↳ 'Not reviewed'
df_ord = pd.DataFrame(df_order_reviews['review_score'].value_counts()).
    ↳ reset_index()
```

```
[ ]: df_order_reviews.shape, df_orders.shape
```

```
[ ]: ((99992, 14), (99441, 8))
```

```
[11]: ax = sns.countplot(df_order_reviews.sort_values('review_score'),
    ↳ x='review_score', stat='percent', order = df_order_reviews['review_score'].
    ↳ value_counts().index)

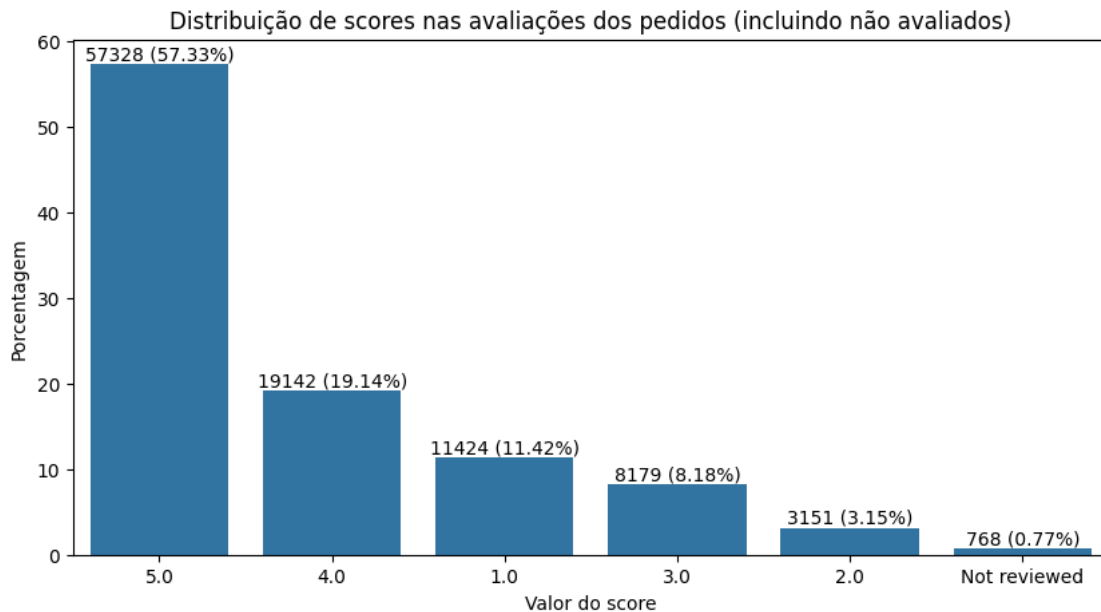
i = 0
for p in ax.patches:
    height = p.get_height()
    text = (str(df_ord['review_score'][i]) + ' (' + str(p.get_height().round(2))
    ↳ + '%)')
    ax.text(p.get_x()+p.get_width()/2., height + 0.5, text, ha="center")
```

```

i+=1

plt.title("Distribuição de scores nas avaliações dos pedidos (incluindo não_
↪avaliados)")
plt.xlabel("Valor do score")
plt.ylabel('Porcentagem')
plt.show()

```



```

[13]: df_rev = pd.DataFrame(df_reviews['review_score'].value_counts()).reset_index()

```

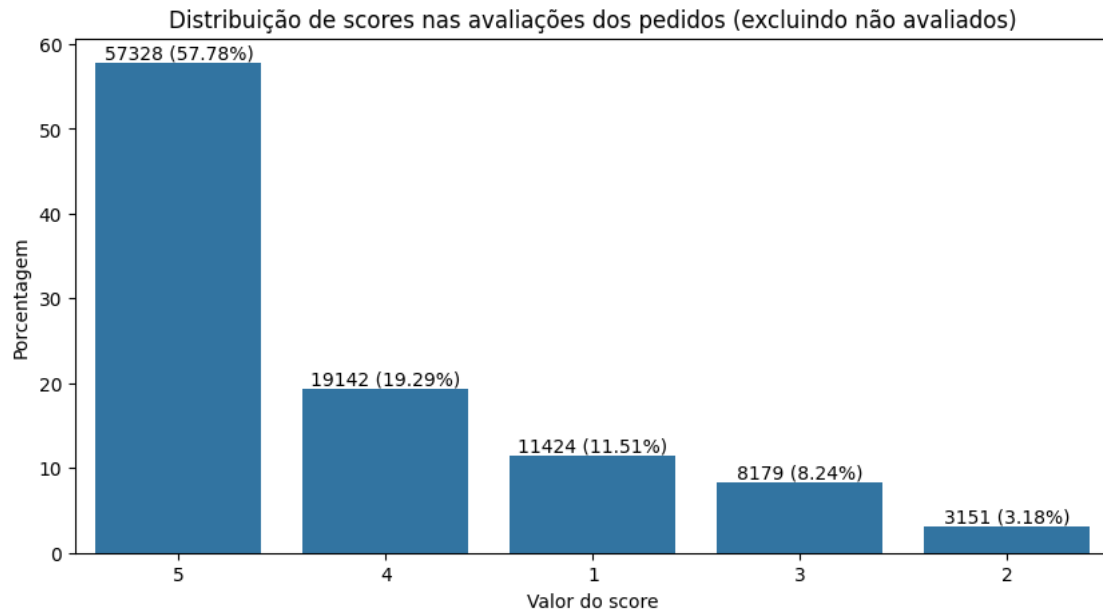
```

[14]: ax = sns.countplot(df_reviews.sort_values('review_score'), x='review_score',
↪stat='percent', order = df_reviews['review_score'].value_counts().index)

i = 0
for p in ax.patches:
    height = p.get_height()
    text = (str(df_rev['review_score'][i]) + ' (' + str(p.get_height().round(2))
↪+ '%)')
    ax.text(p.get_x()+p.get_width()/2., height + 0.5, text,ha="center")
    i+=1

plt.title("Distribuição de scores nas avaliações dos pedidos (excluindo não_
↪avaliados)")
plt.xlabel("Valor do score")
plt.ylabel('Porcentagem')
plt.show()

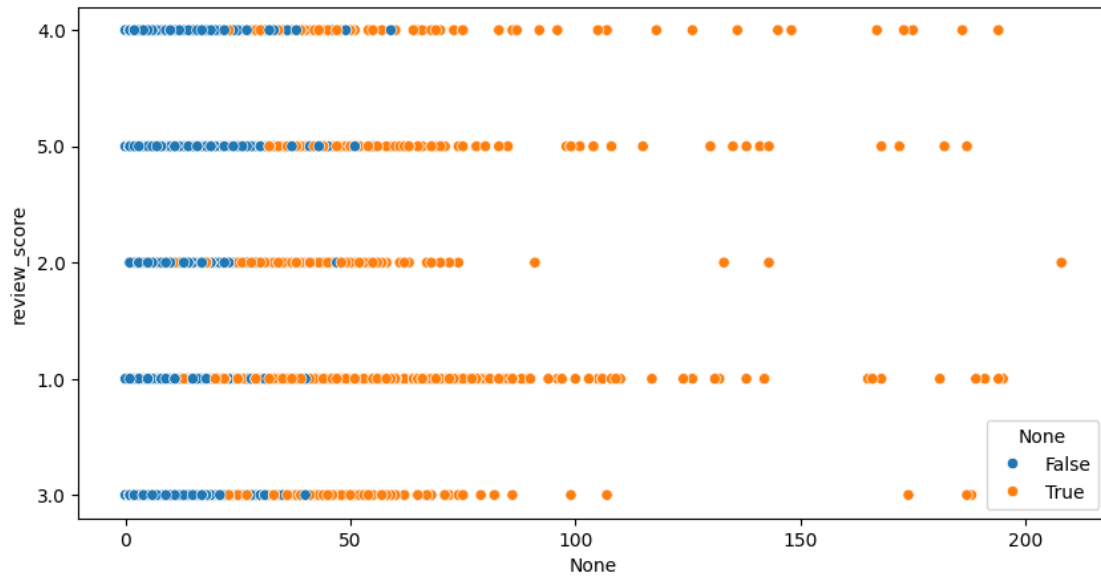
```



0.1.6 6 – Existe relação entre o tempo de entrega e avaliação feita pelo cliente?

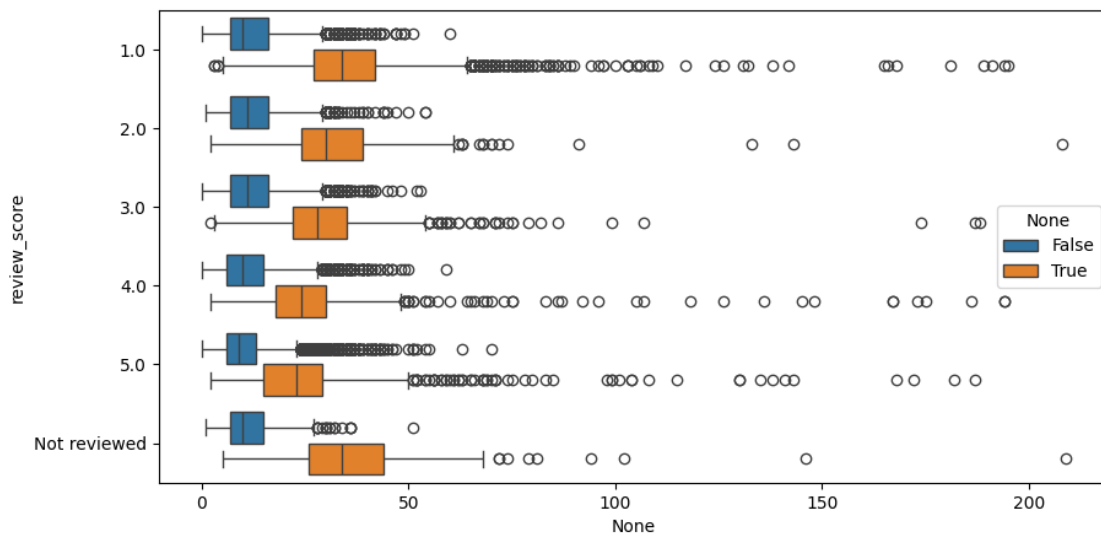
```
[ ]: sns.scatterplot(x=(pd.
    ↳to_datetime(df_order_reviews['order_delivered_customer_date']) - pd.
    ↳to_datetime(df_order_reviews['order_purchase_timestamp'])).dt.
    ↳days[df_order_reviews['review_score'] != 'Not reviewed'],
    y=df_order_reviews['review_score'][df_order_reviews['review_score'] != 'Not_
    ↳reviewed'], hue=pd.
    ↳to_datetime(df_order_reviews['order_delivered_customer_date'])[df_order_reviews['review_sco
    ↳!= 'Not reviewed'] > pd.
    ↳to_datetime(df_order_reviews['order_estimated_delivery_date'])[df_order_reviews['review_sco
    ↳!= 'Not reviewed'])
```

```
[ ]: <Axes: xlabel='None', ylabel='review_score'>
```

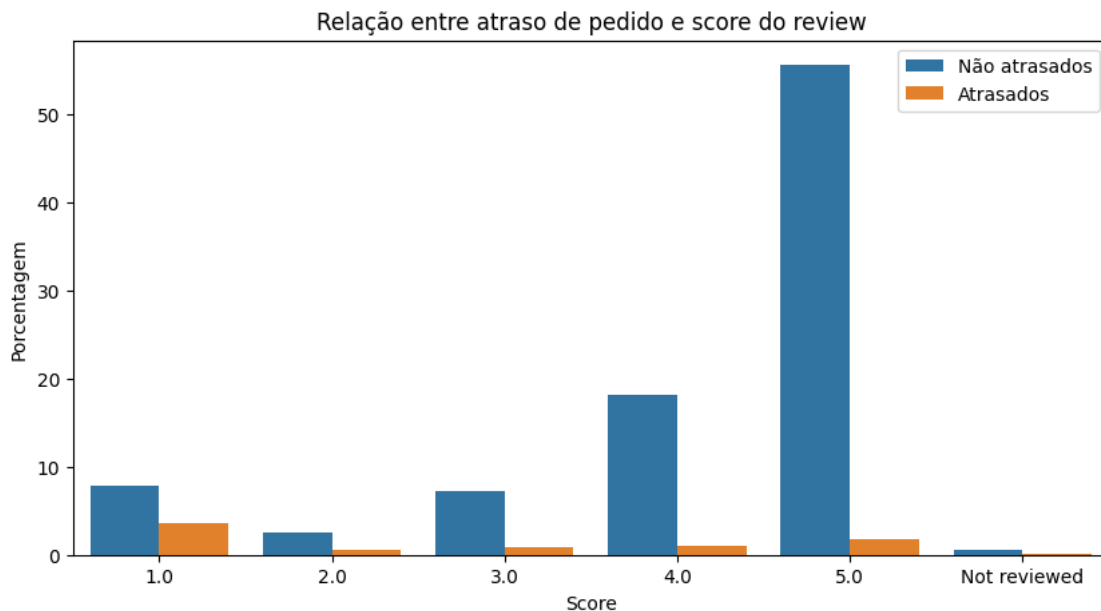
```
[ ]: sns.boxplot(x=((pd.to_datetime(df_order_reviews.
    ↳sort_values('review_score')['order_delivered_customer_date']) - pd.
    ↳to_datetime(df_order_reviews.
    ↳sort_values('review_score')['order_purchase_timestamp']))).dt.days,
    ↳y=df_order_reviews.sort_values('review_score')['review_score'], orient='h',
    ↳hue=pd.to_datetime(df_order_reviews.
    ↳sort_values('review_score')['order_delivered_customer_date']) > pd.
    ↳to_datetime(df_order_reviews.
    ↳sort_values('review_score')['order_estimated_delivery_date']))
```

[]: <Axes: xlabel='None', ylabel='review_score'>



```
[15]: sns.countplot(df_order_reviews.sort_values('review_score'), x='review_score',
    ↳ stat='percent', hue=pd.to_datetime(df_order_reviews.
    ↳ sort_values('review_score')['order_delivered_customer_date']) > pd.
    ↳ to_datetime(df_order_reviews.
    ↳ sort_values('review_score')['order_estimated_delivery_date']))
plt.legend(labels=['Não atrasados', 'Atrasados'])
plt.title('Relação entre atraso de pedido e score do review')
plt.xlabel('Score')
plt.ylabel('Porcentagem')
```

```
[15]: Text(0, 0.5, 'Porcentagem')
```



0.1.7 7 – Quais estados do Brasil que possuem mais clientes insatisfeitos?

```
[16]: df_geo_reviews = (df_reviews.merge(df_orders, on='order_id'))
df_geo_reviews = df_geo_reviews.merge(df_customer, on='customer_id')
df_geo_reviews_low = df_geo_reviews[df_geo_reviews.review_score < 4]
df_geo_reviews_low['customer_state'].value_counts()
```

```
[16]: SP      8564
      RJ      3697
      MG      2515
      RS      1181
      PR      1010
```

BA	971
SC	853
DF	493
GO	491
ES	482
PE	405
CE	395
PA	286
MA	236
MT	202
MS	158
PB	136
PI	132
AL	128
RN	107
SE	105
RO	66
TO	61
AM	28
AC	21
RR	18
AP	13

Name: customer_state, dtype: int64

```
[17]: df = pd.DataFrame(df_geo_reviews_low['customer_state'].value_counts()).
      ↪reset_index()
      df = df[df['customer_state'] > 1000]
      df
```

```
[17]:   index  customer_state
0     SP             8564
1     RJ             3697
2     MG             2515
3     RS             1181
4     PR             1010
```

```
[18]: ax = plt.bar(df['index'], df['customer_state'])

i = 0
for p in ax:
    width = p.get_width()
    height = p.get_height()
    text = (str(df['customer_state'][i]))
    x, y = p.get_xy()
    plt.text(x+width/2,
             y+height+100,
             text,
```

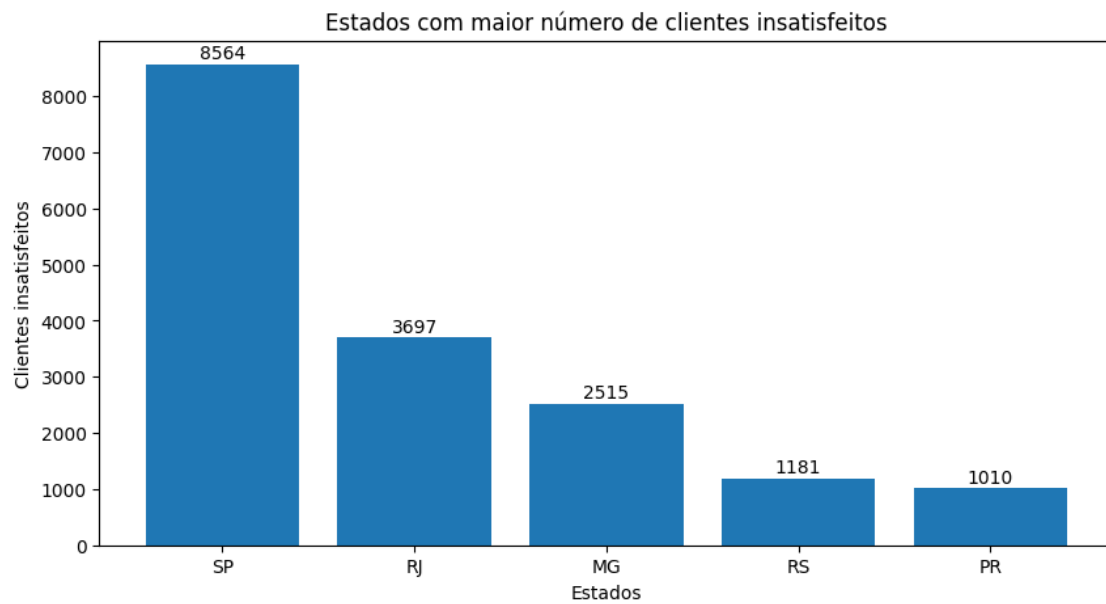
```

        ha='center')

    i+=1

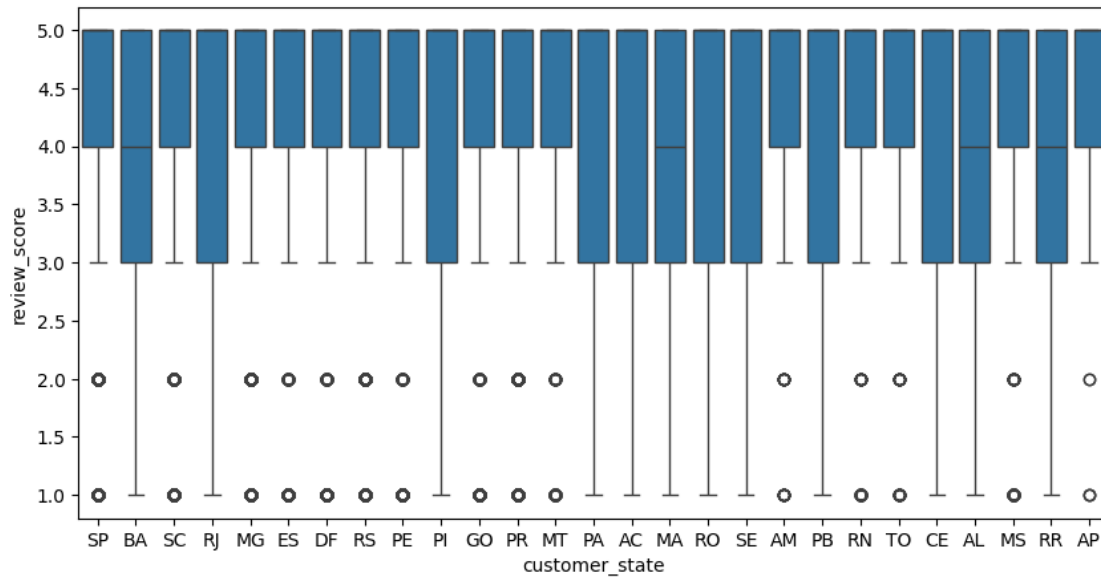
plt.xlabel("Estados")
plt.ylabel("Clientes insatisfeitos")
plt.title("Estados com maior número de clientes insatisfeitos")
plt.show()

```



```
[ ]: sns.boxplot(df_geo_reviews, x='customer_state', y='review_score')
```

```
[ ]: <Axes: xlabel='customer_state', ylabel='review_score'>
```



```
[19]: df2 = pd.DataFrame(((df_geo_reviews_low['customer_state'].value_counts())/
    ↪(df_geo_reviews['customer_state'].value_counts())) *100)
df2 = df2.sort_values(by=['customer_state'], ascending=False).reset_index()
df2
```

```
[19]:
```

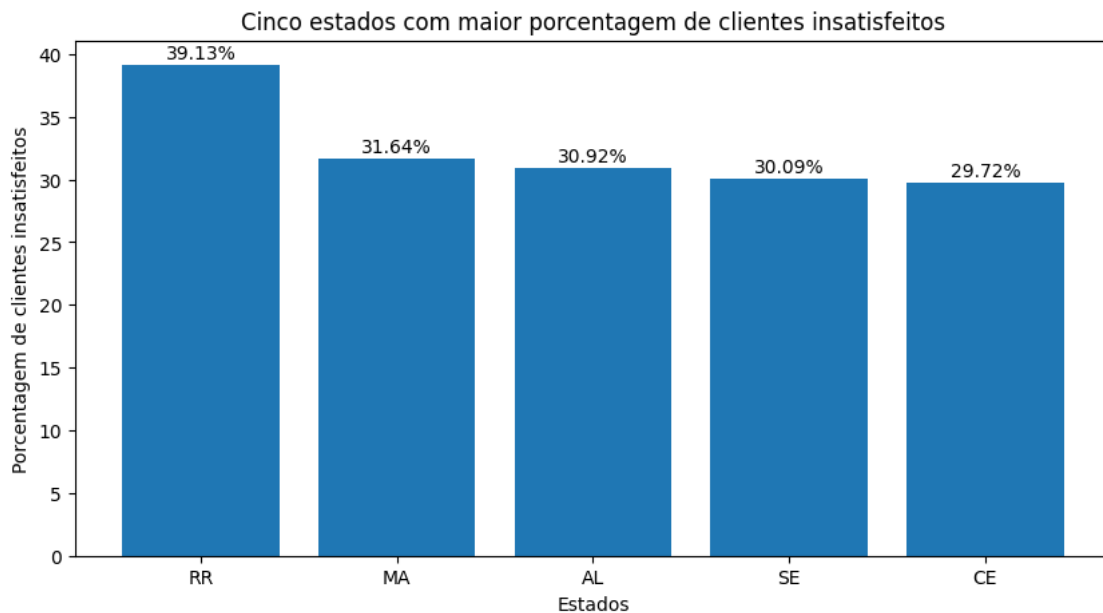
	index	customer_state	
0	RR	39.130435	
1	MA	31.635389	
2	AL	30.917874	
3	SE	30.085960	
4	CE	29.721595	
5	PA	29.545455	
6	RJ	28.962005	
7	BA	28.924635	
8	PI	26.883910	
9	RO	26.190476	
10	AC	25.925926	
11	PB	25.612053	
12	PE	24.605103	
13	GO	24.258893	
14	ES	23.908730	
15	SC	23.544024	
16	DF	22.951583	
17	MT	22.369878	
18	RN	22.199170	
19	TO	21.863799	
20	MS	21.823204	

21	MG	21.634409
22	RS	21.539303
23	SP	20.542096
24	PR	20.047638
25	AP	19.402985
26	AM	19.047619

```
[20]: df2 = df2[df2['customer_state'] > 29.72]
ax = plt.bar(df2['index'], df2['customer_state'])

i = 0
for p in ax:
    width = p.get_width()
    height = p.get_height()
    text = (str(round(df2['customer_state'][i],2)) + "%")
    x, y = p.get_xy()
    plt.text(x+width/2,
             y+height+0.5,
             text,
             ha='center')
    i+=1

plt.xlabel("Estados")
plt.ylabel("Porcentagem de clientes insatisfeitos")
plt.title("Cinco estados com maior porcentagem de clientes insatisfeitos")
plt.figure(figsize=(8,6))
plt.show()
```



<Figure size 800x600 with 0 Axes>

0.1.8 Média do preço do pedido por categoria de produtos vendidos

```
[21]: df_item_order = df_items.merge(df_products, on='product_id')
categorias_mais_vendidas = pd.DataFrame(df_item_order['product_category_name'].
    ↳value_counts())
categorias_mais_vendidas =
    ↳categorias_mais_vendidas[categorias_mais_vendidas['product_category_name'] >
    ↳7800]
categorias_mais_vendidas
```

```
[21]:
```

	product_category_name
cama_mesa_banho	11115
beleza_saude	9670
esporte_lazer	8641
moveis_decoracao	8334
informatica_acessorios	7827

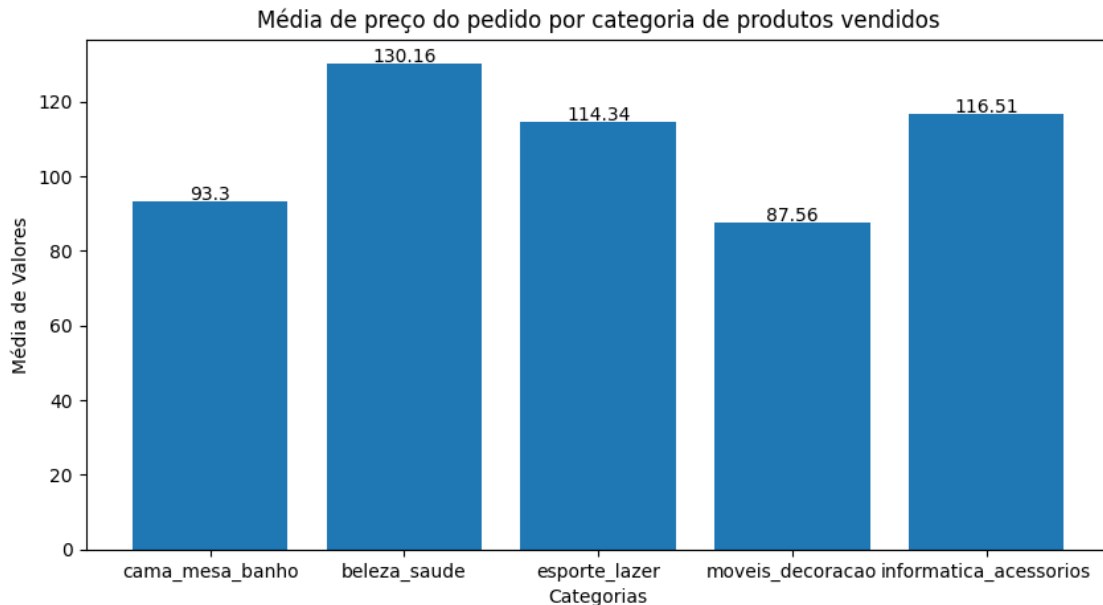
```
[22]: lista_valores = []
for i in categorias_mais_vendidas.index.to_list():
    media = round(df_item_order[df_item_order['product_category_name'] ==
    ↳i]['price'].mean(),2)
    lista_valores.append(media)
    print(i, media)
```

```
cama_mesa_banho 93.3
beleza_saude 130.16
esporte_lazer 114.34
moveis_decoracao 87.56
informatica_acessorios 116.51
```

```
[23]: ax = plt.bar(categorias_mais_vendidas.index.to_list(), lista_valores)

i = 0
for p in ax:
    width = p.get_width()
    height = p.get_height()
    text = (str(lista_valores[i]))
    x, y = p.get_xy()
    plt.text(x+width/2,
            y+height+0.5,
            text,
            ha='center')
    i+=1
```

```
plt.xlabel("Categorias")
plt.ylabel("Média de Valores")
plt.title("Média de preço do pedido por categoria de produtos vendidos")
plt.show()
```



1 Ajuste dos modelos:

1.1 Tratamento das covariáveis

```
[4]: df_payments = pd.get_dummies(df_payments, columns=['payment_type'])
df_products = pd.get_dummies(df_products, columns=['product_category_name'])
```

```
[5]: df_items['total_price'] = df_items['price'] * df_items['order_item_id']
df_items['total_freight'] = df_items['freight_value'] *
    ↪ df_items['order_item_id']
df_sellers_customers = (df_orders.merge(df_customer, on='customer_id',
    ↪ validate='many_to_one')
    .merge(df_items, on='order_id',
    ↪ validate='one_to_many')
    .merge(df_sellers, on='seller_id',
    ↪ validate='many_to_one')
    .merge(df_products, on='product_id',
    ↪ validate='many_to_one')
    )
```



```
df_sellers_customers['same_state'] = df_sellers_customers['customer_state'] == df_sellers_customers['seller_state']
df_sellers_customers.head()
```

```
[5]:
```

	order_id	customer_id
0	e481f51cbdc54678b7cc49136f2d6af7	9ef432eb6251297304e76186b10a928d
1	53cdb2fc8bc7dce0b6741e2150273451	b0830fb4747a6c6d20dea0b8c802d7ef
2	47770eb9100c2d0c44946d9cf07ec65d	41ce2a54c0b03bf3443c3d931a367089
3	949d5b44dbf5de918fe9c16f97b45f8a	f88197465ea7920adcdbec7375364d82
4	ad21c59c0840e6cb83a9ceb5573f8159	8ab97904e6daea8866dbdbc4fb7aad2c

	order_status	order_purchase_timestamp	order_approved_at
0	delivered	2017-10-02 10:56:33	2017-10-02 11:07:15
1	delivered	2018-07-24 20:41:37	2018-07-26 03:24:27
2	delivered	2018-08-08 08:38:49	2018-08-08 08:55:23
3	delivered	2017-11-18 19:28:06	2017-11-18 19:45:59
4	delivered	2018-02-13 21:18:39	2018-02-13 22:20:29

	order_delivered_carrier_date	order_delivered_customer_date
0	2017-10-04 19:55:00	2017-10-10 21:25:13
1	2018-07-26 14:31:00	2018-08-07 15:27:45
2	2018-08-08 13:50:00	2018-08-17 18:06:29
3	2017-11-22 13:39:59	2017-12-02 00:28:42
4	2018-02-14 19:46:34	2018-02-16 18:17:02

	order_estimated_delivery_date	customer_unique_id
0	2017-10-18 00:00:00	7c396fd4830fd04220f754e42b4e5bff
1	2018-08-13 00:00:00	af07308b275d755c9edb36a90c618231
2	2018-09-04 00:00:00	3a653a41f6f9fc3d2a113cf8398680e8
3	2017-12-15 00:00:00	7c142cf63193a1473d2e66489a9ae977
4	2018-02-26 00:00:00	72632f0f9dd73dfee390c9b22eb56dd6

	customer_zip_code_prefix
0	3149
1	47813
2	75265
3	59296
4	9195

	product_category_name_portateis_casa_forno_e_cafe
0	False
1	False
2	False
3	False
4	False

	product_category_name_portateis_cozinha_e_preparadores_de_alimentos
--	---

0	False
1	False
2	False
3	False
4	False

product_category_name_relogios_presentes \	
0	False
1	False
2	False
3	False
4	False

product_category_name_seguros_e_servicos \	
0	False
1	False
2	False
3	False
4	False

product_category_name_sinalizacao_e_seguranca \	
0	False
1	False
2	False
3	False
4	False

product_category_name_tablets_impressao_imagem \	
0	False
1	False
2	False
3	False
4	False

product_category_name_telefonia product_category_name_telefonia_fixa \		
0	False	False
1	False	False
2	False	False
3	False	False
4	False	False

product_category_name_utilidades_domesticas same_state		
0	True	True
1	False	False
2	False	False
3	False	False
4	False	True

[5 rows x 104 columns]

```
[32]: df_payments_unique = df_payments.groupby('order_id').agg({
        'payment_sequential': 'count',
        'payment_value': 'sum',
        'payment_type_boleto': 'mean',
        'payment_type_credit_card': 'mean',
        'payment_type_debit_card': 'mean',
        'payment_type_not_defined': 'mean',
        'payment_type_voucher': 'mean',
        'payment_installments': 'mean'
    }).reset_index()
df_sellers_unique = df_sellers_customers.groupby('order_id').agg(dict({
        'product_id': 'nunique',
        'seller_id': 'nunique',
        'same_state': 'mean',
        'total_price': 'mean',
        'total_freight': 'mean'
    }), **{
        nome: 'mean' for nome in df_sellers_customers.columns if 'product' in nome
        and nome != 'product_id'
    })
df_reg = (df_reviews.merge(df_orders, on='order_id', validate='many_to_one')
        .merge(df_customer, on='customer_id',
        validate='many_to_one')
        .merge(df_payments_unique, on='order_id',
        validate='many_to_one')
        .merge(df_sellers_unique, on='order_id',
        validate='many_to_one')
        )
df_reg.head()
```

```
[32]:
```

	review_id	order_id \
0	7bc2406110b926393aa56f80a40eba40	73fc7af87114b39712e6da79b0a377eb
1	80e641a11e56f04c1ad469d5645fdfdde	a548910a1c6147796b98fdf73dbeba33
2	228ce5500dc1d8e020d8d1322874b6f0	f9e4b658b201a9f2ecdecbb34bed034b
3	e64fb393e7b32834bb789ff8bb30750e	658677c97b385a9be170737859d3511b
4	f7c4243c7fe1938f181bec41a392bdeb	8e6bfb81e283fa7e4f11123a3fb894f1

	review_score	review_comment_title \
0	4	NaN
1	5	NaN
2	5	NaN
3	5	NaN
4	5	NaN

	review_comment_message	review_creation_date	\
0		NaN 2018-01-18 00:00:00	
1		NaN 2018-03-10 00:00:00	
2		NaN 2018-02-17 00:00:00	
3	Recebi bem antes do prazo estipulado.	2017-04-21 00:00:00	
4	Parabéns lojas lannister adorei comprar pela I...	2018-03-01 00:00:00	

	review_answer_timestamp	customer_id	order_status	\
0	2018-01-18 21:46:59	41dcb106f807e993532d446263290104	delivered	
1	2018-03-11 03:05:13	8a2e7ef9053dea531e4dc76bd6d853e6	delivered	
2	2018-02-18 14:36:24	e226dfed6544df5b7b87a48208690feb	delivered	
3	2017-04-21 22:02:06	de6dff97e5f1ba84a3cd9a3bc97df5f6	delivered	
4	2018-03-02 10:26:53	5986b333ca0d44534a156a52a8e33a83	delivered	

	order_purchase_timestamp	order_approved_at	order_delivered_carrier_date	\
0	2018-01-11 15:30:49	2018-01-11 15:47:59	2018-01-12 21:57:22	
1	2018-02-28 12:25:19	2018-02-28 12:48:39	2018-03-02 19:08:15	
2	2018-02-03 09:56:22	2018-02-03 10:33:41	2018-02-06 16:18:28	
3	2017-04-09 17:41:13	2017-04-09 17:55:19	2017-04-10 14:24:47	
4	2018-02-10 10:59:03	2018-02-10 15:48:21	2018-02-15 19:36:14	

	order_delivered_customer_date	order_estimated_delivery_date	\
0	2018-01-17 18:42:41	2018-02-02 00:00:00	
1	2018-03-09 23:17:20	2018-03-14 00:00:00	
2	2018-02-16 17:28:48	2018-03-09 00:00:00	
3	2017-04-20 09:08:35	2017-05-10 00:00:00	
4	2018-02-28 16:33:35	2018-03-09 00:00:00	

	customer_unique_id	customer_zip_code_prefix	\
0	68a5590b9926689be4e10f4ae2db21a8	6030	
1	64190b91b656ab8f37eb89b93dc84584	13380	
2	1d47144362c14e94ccdd213e8ec277d5	44571	
3	c8cf6cb6b838dc7a33ed199b825e8616	88735	
4	d16000272660a1fef81482ad75ba572a	89520	

	customer_city	customer_state	payment_sequential	payment_value	\
0	osasco	SP	1	397.26	
1	nova odessa	SP	1	88.09	
2	santo antonio de jesus	BA	1	194.12	
3	gravatal	SC	1	222.84	
4	curitibanos	SC	1	1333.25	

	payment_type_boleto	payment_type_credit_card	payment_type_debit_card	\
0	0.0	1.0	0.0	
1	0.0	1.0	0.0	
2	0.0	1.0	0.0	
3	0.0	1.0	0.0	

4	0.0	1.0	0.0
---	-----	-----	-----

	payment_type_not_defined	payment_type_voucher	payment_installments	\
0	0.0	0.0	8.0	
1	0.0	0.0	1.0	
2	0.0	0.0	1.0	
3	0.0	0.0	1.0	
4	0.0	0.0	10.0	

	product_id	seller_id	same_state	total_price	total_freight	\
0	1	1	1.0	277.50	20.445	
1	1	1	1.0	79.79	8.300	
2	1	1	0.0	149.00	45.120	
3	1	1	0.0	179.99	42.850	
4	1	1	0.0	1199.00	134.250	

	product_name_lenght	product_description_lenght	product_photos_qty	\
0	42.0	858.0	1.0	
1	47.0	493.0	1.0	
2	59.0	1893.0	1.0	
3	33.0	2188.0	2.0	
4	48.0	562.0	5.0	

	product_weight_g	product_length_cm	product_height_cm	product_width_cm	\
0	1300.0	30.0	30.0	35.0	
1	245.0	19.0	14.0	14.0	
2	6550.0	20.0	20.0	20.0	
3	7650.0	20.0	20.0	20.0	
4	9850.0	105.0	35.0	50.0	

	product_category_name_agro_industria_e_comercio	\
0	0.0	
1	0.0	
2	0.0	
3	0.0	
4	0.0	

	product_category_name_alimentos	product_category_name_alimentos_bebidas	\
0	0.0	0.0	
1	0.0	0.0	
2	0.0	0.0	
3	0.0	0.0	
4	0.0	0.0	

	product_category_name_artes	product_category_name_artes_e_artesanato	\
0	0.0	0.0	
1	0.0	0.0	

2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_artigos_de_festas \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_artigos_de_natal	product_category_name_audio \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_automotivo	product_category_name_bebes \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_bebidas	product_category_name_beleza_saude \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_brinquedos	product_category_name_cama_mesa_banho \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_casa_conforto	product_category_name_casa_conforto_2 \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_casa_construcao \
--	---

0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_cds_dvds_musicais	product_category_name_cine_foto \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_climatizacao	product_category_name_consoles_games \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_construcao_ferramentas_construcao \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_construcao_ferramentas_ferramentas \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_construcao_ferramentas_iluminacao \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_construcao_ferramentas_jardim \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_construcao_ferramentas_seguranca \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_cool_stuff	product_category_name_dvds_blu_ray \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_eletrrodomesticos \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_eletrrodomesticos_2 \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_eletronicos	product_category_name_eletrportateis \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_esporte_lazer \
0	1.0
1	0.0
2	0.0
3	0.0
4	1.0

	product_category_name_fashion_bolsas_e_acessorios \
0	0.0
1	0.0
2	0.0

3	0.0
4	0.0

	product_category_name_fashion_calcados \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_fashion_esporte \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_fashion_roupa_feminina \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_fashion_roupa_infanto_juvenil \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_fashion_roupa_masculina \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_fashion_underwear_e_moda_praia \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_ferramentas_jardim	product_category_name_flores \
0	0.0	0.0

1	0.0	0.0
2	0.0	0.0
3	1.0	0.0
4	0.0	0.0

product_category_name_fraldas_higiene \	
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

product_category_name_industria_comercio_e_negocios \	
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

product_category_name_informatica_acessorios \	
0	0.0
1	1.0
2	1.0
3	0.0
4	0.0

product_category_name_instrumentos_musicais \	
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

product_category_name_la_cuisine		product_category_name_livros_importados \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

product_category_name_livros_interesse_geral \	
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_livros_tecnicos \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_malas_acessorios	product_category_name_market_place \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_moveis_colchao_e_estofado \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_moveis_cozinha_area_de_servico_jantar_e_jardim \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_moveis_decoracao \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_moveis_escritorio \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_moveis_quarto	product_category_name_moveis_sala \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0

4	0.0	0.0
---	-----	-----

	product_category_name_musica	product_category_name_papelaria \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_pc_gamer	product_category_name_pcs \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_perfumaria	product_category_name_pet_shop \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_portateis_casa_forno_e_cafe \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_portateis_cozinha_e_preparadores_de_alimentos \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_relogios_presentes \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_seguros_e_servicos \
0	0.0
1	0.0

2	0.0
3	0.0
4	0.0

	product_category_name_sinalizacao_e_seguranca \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_tablets_impressao_imagem \
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

	product_category_name_telefonia	product_category_name_telefonia_fixa \
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.0	0.0
4	0.0	0.0

	product_category_name_utilidades_domesticas
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

```
[33]: df_reg.columns.values
```

```
[33]: array(['review_id', 'order_id', 'review_score', 'review_comment_title',
        'review_comment_message', 'review_creation_date',
        'review_answer_timestamp', 'customer_id', 'order_status',
        'order_purchase_timestamp', 'order_approved_at',
        'order_delivered_carrier_date', 'order_delivered_customer_date',
        'order_estimated_delivery_date', 'customer_unique_id',
        'customer_zip_code_prefix', 'customer_city', 'customer_state',
        'payment_sequential', 'payment_value', 'payment_type_boleto',
        'payment_type_credit_card', 'payment_type_debit_card',
        'payment_type_not_defined', 'payment_type_voucher',
        'payment_installments', 'product_id', 'seller_id', 'same_state',
        'total_price', 'total_freight', 'product_name_lenght',
        'product_description_lenght', 'product_photos_qty',
```

'product_weight_g', 'product_length_cm', 'product_height_cm',
'product_width_cm',
'product_category_name_agro_industria_e_comercio',
'product_category_name_alimentos',
'product_category_name_alimentos_bebidas',
'product_category_name_artes',
'product_category_name_artes_e_artesanato',
'product_category_name_artigos_de_festas',
'product_category_name_artigos_de_natal',
'product_category_name_audio', 'product_category_name_automotivo',
'product_category_name_bebes', 'product_category_name_bebidas',
'product_category_name_beleza_saude',
'product_category_name_brinquedos',
'product_category_name_cama_mesa_banho',
'product_category_name_casa_conforto',
'product_category_name_casa_conforto_2',
'product_category_name_casa_construcao',
'product_category_name_cds_dvds_musicais',
'product_category_name_cine_foto',
'product_category_name_climatizacao',
'product_category_name_consoles_games',
'product_category_name_construcao_ferramentas_construcao',
'product_category_name_construcao_ferramentas_ferramentas',
'product_category_name_construcao_ferramentas_iluminacao',
'product_category_name_construcao_ferramentas_jardim',
'product_category_name_construcao_ferramentas_seguranca',
'product_category_name_cool_stuff',
'product_category_name_dvds_blu_ray',
'product_category_name_eletrodomesticos',
'product_category_name_eletrodomesticos_2',
'product_category_name_eletronicos',
'product_category_name_eletoportateis',
'product_category_name_esporte_lazer',
'product_category_name_fashion_bolsas_e_acessorios',
'product_category_name_fashion_calcados',
'product_category_name_fashion_esporte',
'product_category_name_fashion_roupa_feminina',
'product_category_name_fashion_roupa_infanto_juvenil',
'product_category_name_fashion_roupa_masculina',
'product_category_name_fashion_underwear_e_moda_praia',
'product_category_name_ferramentas_jardim',
'product_category_name_flores',
'product_category_name_fraldas_higiene',
'product_category_name_industria_comercio_e_negocios',
'product_category_name_informatica_acessorios',
'product_category_name_instrumentos_musicais',
'product_category_name_la_cuisine',

```

'product_category_name_livros_importados',
'product_category_name_livros_interesse_geral',
'product_category_name_livros_tecnicos',
'product_category_name_malas_acessorios',
'product_category_name_market_place',
'product_category_name_moveis_colchao_e_estofado',
'product_category_name_moveis_cozinha_area_de_servico_jantar_e_jardim',
'product_category_name_moveis_decoracao',
'product_category_name_moveis_escritorio',
'product_category_name_moveis_quarto',
'product_category_name_moveis_sala',
'product_category_name_musica', 'product_category_name_papelaria',
'product_category_name_pc_gamer', 'product_category_name_pcs',
'product_category_name_perfumaria',
'product_category_name_pet_shop',
'product_category_name_portateis_casa_forno_e_cafe',
'product_category_name_portateis_cozinha_e_preparadores_de_alimentos',
'product_category_name_relogios_presentes',
'product_category_name_seguros_e_servicos',
'product_category_name_sinalizacao_e_seguranca',
'product_category_name_tablets_impresao_imagem',
'product_category_name_telefonia',
'product_category_name_telefonia_fixa',
'product_category_name_utilidades_domesticas'], dtype=object)

```

```

[34]: del df_reg['review_id'], df_reg['order_id'], df_reg['review_comment_title'],
      ↪df_reg['review_comment_message'], df_reg['review_creation_date']
del df_reg['review_answer_timestamp'], df_reg['customer_id'],
      ↪df_reg['order_approved_at'], df_reg['order_delivered_carrier_date']
del df_reg['customer_unique_id'], df_reg['customer_zip_code_prefix'],
      ↪df_reg['customer_city']

```

```

[35]: df_reg['delayed_order'] = pd.
      ↪to_datetime(df_reg['order_estimated_delivery_date']) < pd.
      ↪to_datetime(df_reg['order_delivered_customer_date'])
df_reg['delayed_days'] = (pd.
      ↪to_datetime(df_reg['order_estimated_delivery_date']) - pd.
      ↪to_datetime(df_reg['order_delivered_customer_date'])).dt.days
df_reg.loc[df_reg['delayed_days'] < 0, 'delayed_days'] = 0
df_reg['delivery_time'] = (pd.
      ↪to_datetime(df_reg['order_delivered_customer_date']) - pd.
      ↪to_datetime(df_reg['order_purchase_timestamp'])).dt.days
df_reg['delivery_interaction'] = df_reg['delayed_order'] *
      ↪df_reg['delivery_time']
df_reg['installment_value'] = df_reg['payment_value'] /
      ↪df_reg['payment_installments']

```

```

df_reg.loc[df_reg['payment_installments'] == 0, 'installment_value'] = 1
df_reg['payment_value'][df_reg['payment_installments'] == 0]
df_reg['cancelado'] = df_reg['order_status'] == 'canceled'
del df_reg['order_delivered_customer_date'], df_reg['order_status'],
df_reg['order_estimated_delivery_date'], df_reg['order_purchase_timestamp']
df_reg = df_reg.dropna()
df_reg = pd.get_dummies(df_reg, columns=['customer_state'])
df_reg_train, df_reg_test = train_test_split(df_reg, test_size=0.1)
df_reg_train.shape

```

[35]: (85522, 127)

```

[36]: pd.set_option('display.max_columns', 504)
df_reg_train.describe()

```

```

[36]:      review_score  payment_sequential  payment_value  payment_type_boleto \
count    85522.000000      85522.000000    85522.000000      85522.000000
mean         4.156439         1.044597     159.710865         0.199118
std          1.283842         0.372678     217.443128         0.399340
min           1.000000         1.000000         9.590000         0.000000
25%           4.000000         1.000000         61.862500         0.000000
50%           5.000000         1.000000        104.940000         0.000000
75%           5.000000         1.000000        176.760000         0.000000
max           5.000000        26.000000    13664.080000         1.000000

      payment_type_credit_card  payment_type_debit_card \
count          85522.000000      85522.000000
mean             0.757734         0.015435
std              0.422109         0.123274
min              0.000000         0.000000
25%              0.500000         0.000000
50%              1.000000         0.000000
75%              1.000000         0.000000
max              1.000000         1.000000

      payment_type_not_defined  payment_type_voucher  payment_installments \
count          85522.0      85522.000000      85522.000000
mean             0.0         0.027713         2.916517
std              0.0         0.146786         2.700434
min              0.0         0.000000         0.000000
25%              0.0         0.000000         1.000000
50%              0.0         0.000000         2.000000
75%              0.0         0.000000         4.000000
max              0.0         1.000000        24.000000

      product_id  seller_id  same_state  total_price  total_freight \
count    85522.000000  85522.000000  85522.000000  85522.000000  85522.000000

```


mean	1.039417	1.014020	0.359498	130.956537	21.490824
std	0.231774	0.124246	0.479215	195.030985	17.721015
min	1.000000	1.000000	0.000000	0.850000	0.000000
25%	1.000000	1.000000	0.000000	44.990000	13.680000
50%	1.000000	1.000000	0.000000	81.965000	17.060000
75%	1.000000	1.000000	1.000000	146.990000	23.080000
max	8.000000	5.000000	1.000000	7560.000000	1047.060000

	product_name_lenght	product_description_lenght	product_photos_qty	\
count	85522.000000	85522.000000	85522.000000	
mean	48.867933	791.696753	2.250635	
std	9.946547	651.060332	1.741977	
min	5.000000	4.000000	1.000000	
25%	43.000000	350.625000	1.000000	
50%	52.000000	606.000000	2.000000	
75%	57.000000	995.000000	3.000000	
max	76.000000	3992.000000	20.000000	

	product_weight_g	product_length_cm	product_height_cm	\
count	85522.000000	85522.000000	85522.000000	
mean	2099.823941	30.150127	16.450975	
std	3748.408030	16.088447	13.254694	
min	0.000000	7.000000	2.000000	
25%	300.000000	18.000000	8.000000	
50%	700.000000	25.000000	13.000000	
75%	1813.000000	38.000000	20.000000	
max	40425.000000	105.000000	105.000000	

	product_width_cm	product_category_name_agro_industria_e_comercio	\
count	85522.000000	85522.000000	
mean	23.044490	0.001906	
std	11.687071	0.043616	
min	6.000000	0.000000	
25%	15.000000	0.000000	
50%	20.000000	0.000000	
75%	30.000000	0.000000	
max	118.000000	1.000000	

	product_category_name_alimentos	\
count	85522.000000	
mean	0.004593	
std	0.067453	
min	0.000000	
25%	0.000000	
50%	0.000000	
75%	0.000000	
max	1.000000	

	product_category_name_alimentos_bebidas	product_category_name_artes \
count	85522.000000	85522.000000
mean	0.002356	0.001985
std	0.048392	0.044385
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_artes_e_artesanato \
count	85522.000000
mean	0.000246
std	0.015481
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_artigos_de_festas \
count	85522.000000
mean	0.000380
std	0.019415
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_artigos_de_natal	product_category_name_audio \
count	85522.000000	85522.000000
mean	0.001317	0.003701
std	0.036236	0.060650
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_automotivo	product_category_name_bebes \
count	85522.000000	85522.000000
mean	0.040192	0.028779
std	0.196157	0.166455
min	0.000000	0.000000
25%	0.000000	0.000000

50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_bebidas	product_category_name_beleza_saude \
count	85522.000000	85522.000000
mean	0.002930	0.090389
std	0.053898	0.286420
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_brinquedos \
count	85522.000000
mean	0.039791
std	0.195131
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_cama_mesa_banho \
count	85522.000000
mean	0.096606
std	0.294540
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_casa_conforto \
count	85522.000000
mean	0.003732
std	0.059785
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_casa_conforto_2 \
count	85522.000000
mean	0.000234

std	0.015098
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_casa_construcao \	
count	85522.000000
mean	0.005031
std	0.070291
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_cds_dvds_musicais \	
count	85522.000000
mean	0.000129
std	0.011340
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_cine_foto		product_category_name_climatizacao \	
count	85522.000000		85522.000000
mean	0.000678		0.002572
std	0.025921		0.050596
min	0.000000		0.000000
25%	0.000000		0.000000
50%	0.000000		0.000000
75%	0.000000		0.000000
max	1.000000		1.000000

product_category_name_consoles_games \	
count	85522.000000
mean	0.010504
std	0.101852
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_construcao_ferramentas_construcao \
count	85522.000000
mean	0.007625
std	0.086742
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000
	product_category_name_construcao_ferramentas_ferramentas \
count	85522.000000
mean	0.001029
std	0.032061
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000
	product_category_name_construcao_ferramentas_iluminacao \
count	85522.000000
mean	0.002464
std	0.049122
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000
	product_category_name_construcao_ferramentas_jardim \
count	85522.000000
mean	0.001912
std	0.043358
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000
	product_category_name_construcao_ferramentas_seguranca \
count	85522.000000
mean	0.001657
std	0.040471
min	0.000000
25%	0.000000
50%	0.000000

75%	0.000000
max	1.000000

	product_category_name_cool_stuff	product_category_name_dvds_blu_ray \
count	85522.000000	85522.000000
mean	0.036891	0.000575
std	0.188020	0.023855
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_eletrodomesticos \
count	85522.000000
mean	0.008162
std	0.089973
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_eletrodomesticos_2 \
count	85522.000000
mean	0.002362
std	0.048543
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_eletronicos \
count	85522.000000
mean	0.026200
std	0.159585
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_eleetroportateis \
count	85522.000000
mean	0.006356
std	0.079372

min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_esporte_lazer \	
count	85522.000000
mean	0.079207
std	0.269715
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_bolsas_e_acessorios \	
count	85522.000000
mean	0.019073
std	0.136596
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_calcados \	
count	85522.000000
mean	0.002456
std	0.049433
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_esporte \	
count	85522.000000
mean	0.000281
std	0.016574
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_roupa_feminina \

count	85522.000000
mean	0.000403
std	0.020008
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_roupa_infanto_juvenil \	
count	85522.000000
mean	0.000058
std	0.007254
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_roupa_masculina \	
count	85522.000000
mean	0.001099
std	0.033135
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_fashion_underwear_e_moda_praia \	
count	85522.000000
mean	0.001228
std	0.035018
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_ferramentas_jardim		product_category_name_flores \
count	85522.000000	85522.000000
mean	0.035792	0.000276
std	0.185286	0.016493
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000

max	1.000000	1.000000
-----	----------	----------

	product_category_name_fraldas_higiene \
count	85522.000000
mean	0.000246
std	0.015668
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_industria_comercio_e_negocios \
count	85522.000000
mean	0.002342
std	0.048255
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_informatica_acessorios \
count	85522.000000
mean	0.068540
std	0.252435
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_instrumentos_musicais \
count	85522.000000
mean	0.006271
std	0.078820
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_la_cuisine \
count	85522.000000
mean	0.000134
std	0.011469
min	0.000000

25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_livros_importados \
count	85522.000000
mean	0.000526
std	0.022933
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_livros_interesse_geral \
count	85522.000000
mean	0.004999
std	0.070421
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_livros_tecnicos \
count	85522.000000
mean	0.002806
std	0.052790
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_malas_acessorios \
count	85522.000000
mean	0.010442
std	0.101452
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_market_place \
count	85522.000000

mean	0.002751
std	0.052083
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_moveis_colchao_e_estofado \	
count	85522.000000
mean	0.000386
std	0.019640
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_moveis_cozinha_area_de_servico_jantar_e_jardim \	
count	85522.000000
mean	0.002403
std	0.048811
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_moveis_decoracao \	
count	85522.000000
mean	0.065267
std	0.246000
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

product_category_name_moveis_escritorio \	
count	85522.000000
mean	0.013077
std	0.113500
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_moveis_quarto	product_category_name_moveis_sala \
count	85522.000000	85522.000000
mean	0.000930	0.004233
std	0.030331	0.064617
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_musica	product_category_name_papelaria \
count	85522.000000	85522.000000
mean	0.000421	0.023647
std	0.020370	0.151661
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_pc_gamer	product_category_name_pcs \
count	85522.000000	85522.000000
mean	0.000053	0.001847
std	0.007049	0.042943
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_perfumaria	product_category_name_pet_shop \
count	85522.000000	85522.000000
mean	0.032647	0.017729
std	0.177501	0.131828
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_portateis_casa_forno_e_cafe \
count	85522.000000
mean	0.000772
std	0.027769
min	0.000000
25%	0.000000

50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_portateis_cozinha_e_preparadores_de_alimentos \
count	85522.000000
mean	0.000129
std	0.011340
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_relogios_presentes \
count	85522.000000
mean	0.057136
std	0.231893
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_seguros_e_servicos \
count	85522.000000
mean	0.000023
std	0.004836
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_sinalizacao_e_seguranca \
count	85522.000000
mean	0.001479
std	0.038393
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_tablets_impressao_imagem \
count	85522.000000
mean	0.000836

std	0.028750
min	0.000000
25%	0.000000
50%	0.000000
75%	0.000000
max	1.000000

	product_category_name_telefonia	product_category_name_telefonia_fixa \
count	85522.000000	85522.000000
mean	0.042884	0.002198
std	0.202444	0.046772
min	0.000000	0.000000
25%	0.000000	0.000000
50%	0.000000	0.000000
75%	0.000000	0.000000
max	1.000000	1.000000

	product_category_name_utilidades_domesticas	delayed_days \
count	85522.000000	85522.000000
mean	0.059406	11.693342
std	0.235799	7.880904
min	0.000000	0.000000
25%	0.000000	6.000000
50%	0.000000	11.000000
75%	0.000000	16.000000
max	1.000000	146.000000

	delivery_time	delivery_interaction	installment_value
count	85522.000000	85522.000000	85522.000000
mean	12.067059	2.478006	82.079108
std	9.480654	9.695137	131.882894
min	0.000000	0.000000	3.767000
25%	6.000000	0.000000	28.630000
50%	10.000000	0.000000	53.641250
75%	15.000000	0.000000	93.300000
max	208.000000	208.000000	13664.080000

```
[37]: q,r = np.linalg.qr(np.asarray(df_reg_train.loc[:, df_reg_train.columns != 'review_score'], dtype='float'))
      print(df_reg_train.columns[~np.insert(np.abs(np.diag(r))>=1e-8, 0, True)])
      df_reg_train = df_reg_train.iloc[:, np.insert(np.abs(np.diag(r))>=1e-8, 0, True)]
      np.insert(np.abs(np.diag(r))>=1e-8, 0, True)
```

```
Index(['payment_type_not_defined', 'customer_state_T0'], dtype='object')
```

```
[12]: df_reg_train.shape
```

```
[38]: payment_type = [
        i for i in df_reg_train.columns if 'payment_type' in i
    ]
    payment_type.pop()

    product_category_name = [
        i for i in df_reg_train.columns if 'product_category_name' in i
    ]
    product_category_name.pop()

    customer_state = [
        i for i in df_reg_train.columns if 'customer_state' in i
    ]

    covariaveis_unicas = [
        'payment_sequential',
        'payment_installments',
        'payment_value',
        'delayed_order',
        'delivery_interaction',
        'delivery_time',
        'delayed_days',
        'installment_value',
        'cancelado',
        'product_id',
    ]
```

```

'seller_id',
'same_state',
'product_name_lenght',
'product_description_lenght',
'product_photos_qty',
'product_weight_g',
'product_length_cm',
'product_height_cm',
'product_width_cm',
'total_price',
'total_freight'
]
covariaveis = covariaveis_unicas + payment_type + product_category_name +
↳customer_state

```

1.2 Regressão ordinal logistica

```

[39]: modf_logit = OrderedModel(df_reg_train['review_score'],
↳df_reg_train[covariaveis].astype(float),
                                distr='logit')
resf_logit = modf_logit.fit(method='bfgs')
resf_logit.summary()

```

/home/lucas/.cache/pypoetry/virtualenvs/ecommerce-classification-xG4A-NiM-py3.10/lib/python3.10/site-packages/scipy/optimize/_optimize.py:1397:

OptimizeWarning: Maximum number of iterations has been exceeded.

```
res = _minimize_bfgs(f, x0, args, fprime, callback=callback, **opts)
```

```
Current function value: 1.095104
```

```
Iterations: 500
```

```
Function evaluations: 510
```

```
Gradient evaluations: 510
```

/home/lucas/.cache/pypoetry/virtualenvs/ecommerce-classification-xG4A-NiM-py3.10/lib/python3.10/site-packages/statsmodels/base/model.py:607:

ConvergenceWarning: Maximum Likelihood optimization failed to converge. Check mle_retvals

```
warnings.warn("Maximum Likelihood optimization failed to "
```

[39]:

Dep. Variable:	review_score	Log-Likelihood:	-93656.
Model:	OrderedModel	AIC:	1.876e+05
Method:	Maximum Likelihood	BIC:	1.887e+05
Date:	Tue, 02 Apr 2024		
Time:	20:30:39		
No. Observations:	85522		
Df Residuals:	85396		
Df Model:	122		

	coef	std
payment__sequential	0.0281	0.02
payment__installments	0.0043	0.00
payment__value	-0.0035	0.00
delayed__order	-1.0655	0.00
delivery__interaction	-0.0100	0.00
delivery__time	-0.0499	0.00
delayed__days	0.0024	0.00
installment__value	9.631e-05	9.41e-05
cancelado	-2.8569	0.93
product__id	-0.2310	0.03
seller__id	-1.4554	0.00
same__state	-0.0148	0.02
product__name__lenght	-0.0036	0.00
product__description__lenght	4.63e-06	1.18e-06
product__photos__qty	-0.0082	0.00
product__weight__g	-6.311e-07	3.13e-07
product__length__cm	0.0001	0.00
product__height__cm	-0.0012	0.00
product__width__cm	0.0002	0.00
total__price	0.0036	0.00
total__freight	0.0036	0.00
payment__type__boleto	0.1759	0.03
payment__type__credit_card	0.1122	0.03
payment__type__debit_card	0.1900	0.07
product__category__name__agro_industria_e_comercio	-0.3169	0.13
product__category__name__alimentos	0.2100	0.13
product__category__name__alimentos_bebidas	0.2883	0.13
product__category__name__artes	-0.1485	0.13
product__category__name__artes_e_artesanato	-0.1880	0.43
product__category__name__artigos_de_festas	-0.3204	0.33
product__category__name__artigos_de_natal	-0.0946	0.13
product__category__name__audio	-0.3961	0.13
product__category__name__automotivo	-0.0510	0.03
product__category__name__bebes	-0.0428	0.03
product__category__name__bebidas	0.0681	0.13
product__category__name__beleza_saude	0.1022	0.03
product__category__name__brinquedos	0.0764	0.03
product__category__name__cama_mesa_banho	-0.1631	0.03
product__category__name__casa_conforto	-0.0467	0.13
product__category__name__casa_conforto_2	-0.0658	0.43
product__category__name__casa_construcao	-0.1317	0.03
product__category__name__cds_dvds_musicais	0.6044	0.63
product__category__name__cine_foto	0.5954	0.33
product__category__name__climatizacao	-0.1068	0.13
product__category__name__consoles_games	0.0482	0.07
product__category__name__construcao_ferramentas_construcao	-0.0442	0.03
product__category__name__construcao_ferramentas_ferramentas	0.2822	0.23
product__category__name__construcao_ferramentas_iluminacao	-0.0833	0.13
product__category__name__construcao_ferramentas_jardim	0.2639	0.10
product__category__name__construcao_ferramentas_seguranca	-0.3973	0.10
product__category__name__cool_stuff	0.0109	0.03
product__category__name__dvds_blu_ray	0.0802	0.23

```
[44]: modf_logit.transform_threshold_params(resf_logit.params[-4:])
```

```
[44]: array([      -inf, -5.16416689, -4.79333175, -4.0788251 , -3.01209947,
           inf])
```

```
[45]: def amostra_dist_truncada(dist, a, b, loc, scale):
    u = np.random.uniform()
    return dist.ppf(
        u * (dist.cdf(b, loc=loc, scale=scale) - dist.cdf(a, loc=loc,
        ↪scale=scale)) +
        dist.cdf(a, loc=loc, scale=scale),
        loc=loc, scale=scale
    )

def residuo_substituto(modelo, y, dist, new_data=None, y_min=1):
    """Calcula o residuo substituo proposto por Liu and Zhang (2017)."""
    response_values = np.sort(np.unique(y))
    classes = len(response_values)
    media = modelo.model.predict(modelo.params, exog=new_data, which="linpred")
    cut_points = modelo.model.transform_threshold_params(modelo.
    ↪params[1-classes:])
    deslocamento = 0
    if dist is stats.gumbel_r:
        deslocamento = np.euler_gamma
    s = [amostra_dist_truncada(dist, cut_points[y[i] - y_min],
    ↪cut_points[y[i]+1 - y_min], media[i] - deslocamento, 1) for i in
    ↪range(len(y))] # Currently assuming that y takes sequential integer values
    return s - media
```

```
[46]: def plota_residuos(data, y_name, features_names, dist, size=100):
    import math
    plot_size=7
    num_plots_x=5 # No. of plots in every row
    num_plots_y = math.ceil(len(features_names)/num_plots_x) # No. of plots
    ↪in y direction

    '''
    for i in range(num_plots_y):
        start = i * num_plots_x
        end = start + num_plots_x
        sns.pairplot(x_vars=features_names[start:end], y_vars=y_name, data=data)
    '''

    g = sns.FacetGrid(pd.DataFrame(features_names), col=0, col_wrap=4,
    ↪sharey=False, sharex=False)
    for ax, x_var in zip(g.axes, features_names):
```

```

    if x_var == y_name:
        params = dist.fit(data[y_name])
        arg = params[:-2]
        loc = params[-2]
        scale = params[-1]
        x = np.linspace(data[y_name].min(), data[y_name].max(), num=size)
        if arg:
            pdf_fitted = dist.pdf(x, *arg, loc=loc, scale=scale)
        else:
            pdf_fitted = dist.pdf(x, loc=loc, scale=scale)
        sns.histplot(data=data, x=y_name, ax=ax, stat='density')
        sns.lineplot(x=x, y=pdf_fitted, ax=ax, color='r')
    else:
        sns.regplot(data=data, x=x_var, y=y_name, ax=ax,
↪line_kws=dict(color="r"), scatter_kws={'alpha': 0.1})
    g.tight_layout()

    plt.show()

def reporta_predicoes(modelo, endog, exog=None):
    mult = np.diag(np.sort(endog.unique()))
    probs = modelo.model.predict(modelo.params, exog=exog)
    predicted = np.argmax(probs, axis=1) + 1
    expected = np.sum(probs @ mult, axis=1)
    mse_expected = ((endog - expected)** 2).mean()
    mse_predicted = ((endog - predicted)** 2).mean()
    accuracy = (endog == predicted).mean()
    print(f'MSE de acordo com os valores esperados: {mse_expected:.2f}')
    print(f'MSE de acordo com a nota mais provável: {mse_predicted:.2f}')
    print(f'Acurácia das notas mais provaveis: {100*accuracy:.2f}%)

```

```

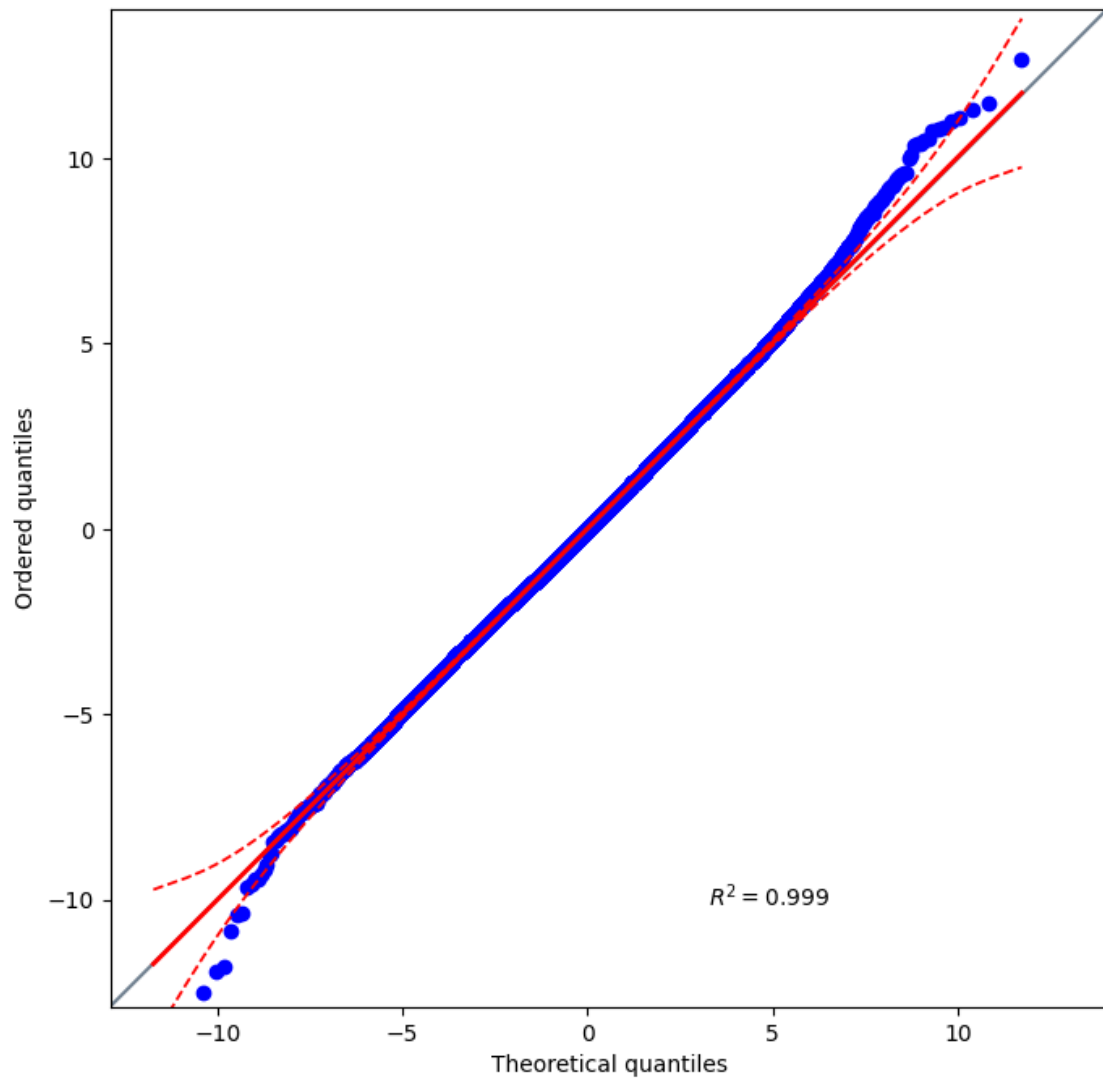
[47]: r_logit = residuo_substituto(resf_logit, df_reg_train.review_score.values,
↪stats.logistic)
df_reg_train['r_logit'] = r_logit
df_reg_train['predicted_logit'] = np.argmax(resf_logit.model.predict(resf_logit.
↪params), axis=1) + 1
mult = np.diag(np.sort(df_reg_train.review_score.unique()))
df_reg_train['expected_logit'] = np.sum(resf_logit.model.predict(resf_logit.
↪params) @ mult, axis=1)
pg.qqplot(r_logit, dist=stats.logistic, confidence=.95)

```

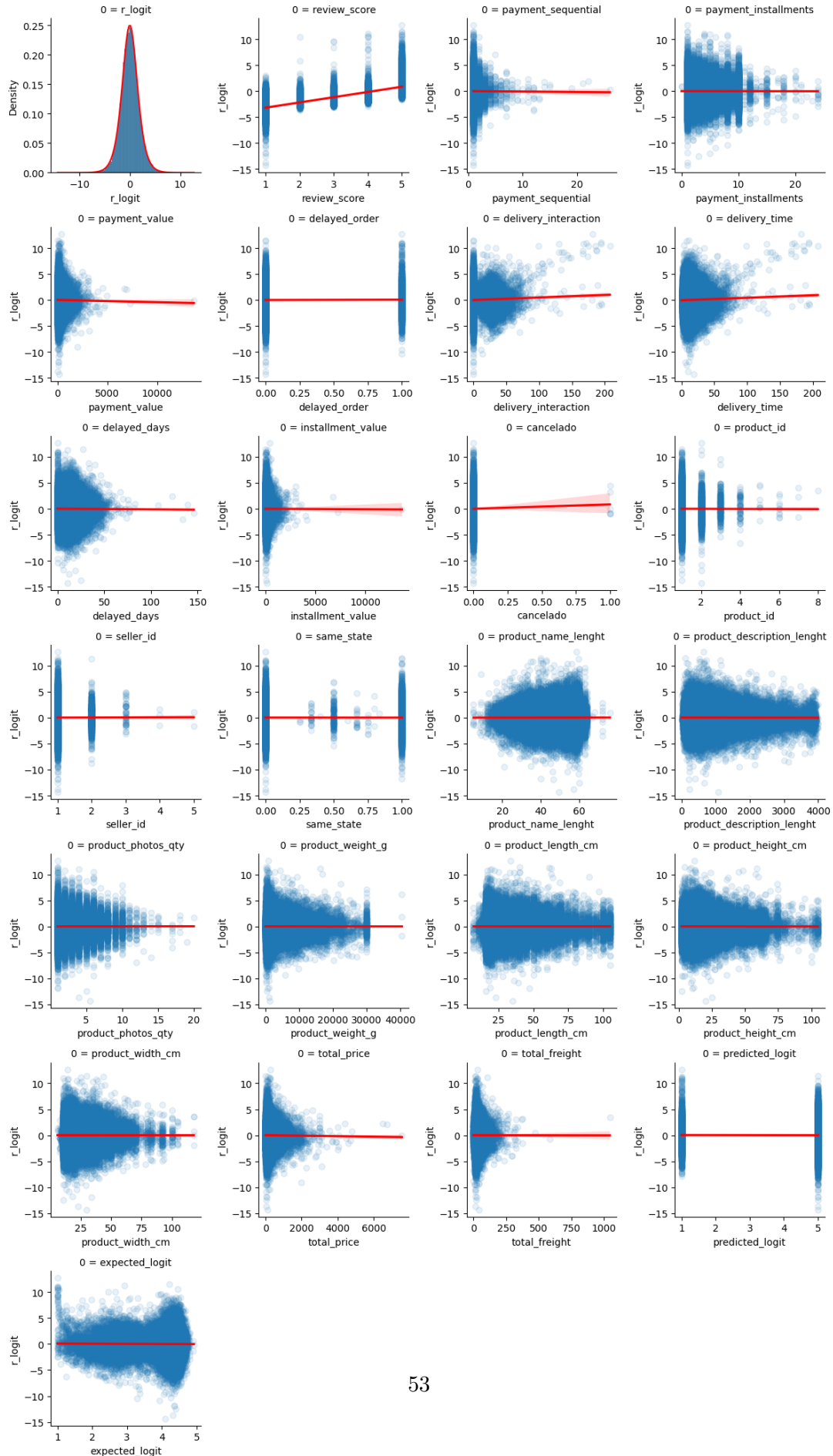
```

[47]: <Axes: xlabel='Theoretical quantiles', ylabel='Ordered quantiles'>

```



```
[48]: plota_residuos(df_reg_train, 'r_logit', ['r_logit', 'review_score'] +  
    ↪ covariaveis_unicas + ['predicted_logit', 'expected_logit'], stats.logistic)
```



```
[49]: reporta_predicoes(resf_logit, df_reg_train.review_score)
```

MSE de acordo com os valores esperados: 1.33
MSE de acordo com a nota mais provável: 2.00
Acurácia das notas mais prováveis: 61.56%

1.3 Regressão ordinal probit

```
[50]: modf_probit = OrderedModel(df_reg_train['review_score'],  
    ↪df_reg_train[covariaveis].astype(float),  
                                     distr='probit')  
resf_probit = modf_probit.fit(method='bfgs')  
resf_probit.summary()
```

```
/home/lucas/.cache/pypoetry/virtualenvs/ecommerce-classification-xG4A-NiM-  
py3.10/lib/python3.10/site-packages/scipy/optimize/_optimize.py:1397:  
OptimizeWarning: Maximum number of iterations has been exceeded.
```

```
res = _minimize_bfgs(f, x0, args, fprime, callback=callback, **opts)
```

```
Current function value: 1.095928
```

```
Iterations: 500
```

```
Function evaluations: 510
```

```
Gradient evaluations: 510
```

```
/home/lucas/.cache/pypoetry/virtualenvs/ecommerce-classification-xG4A-NiM-  
py3.10/lib/python3.10/site-packages/statsmodels/base/model.py:607:
```

```
ConvergenceWarning: Maximum Likelihood optimization failed to converge. Check  
mle_retvals
```

```
warnings.warn("Maximum Likelihood optimization failed to "
```

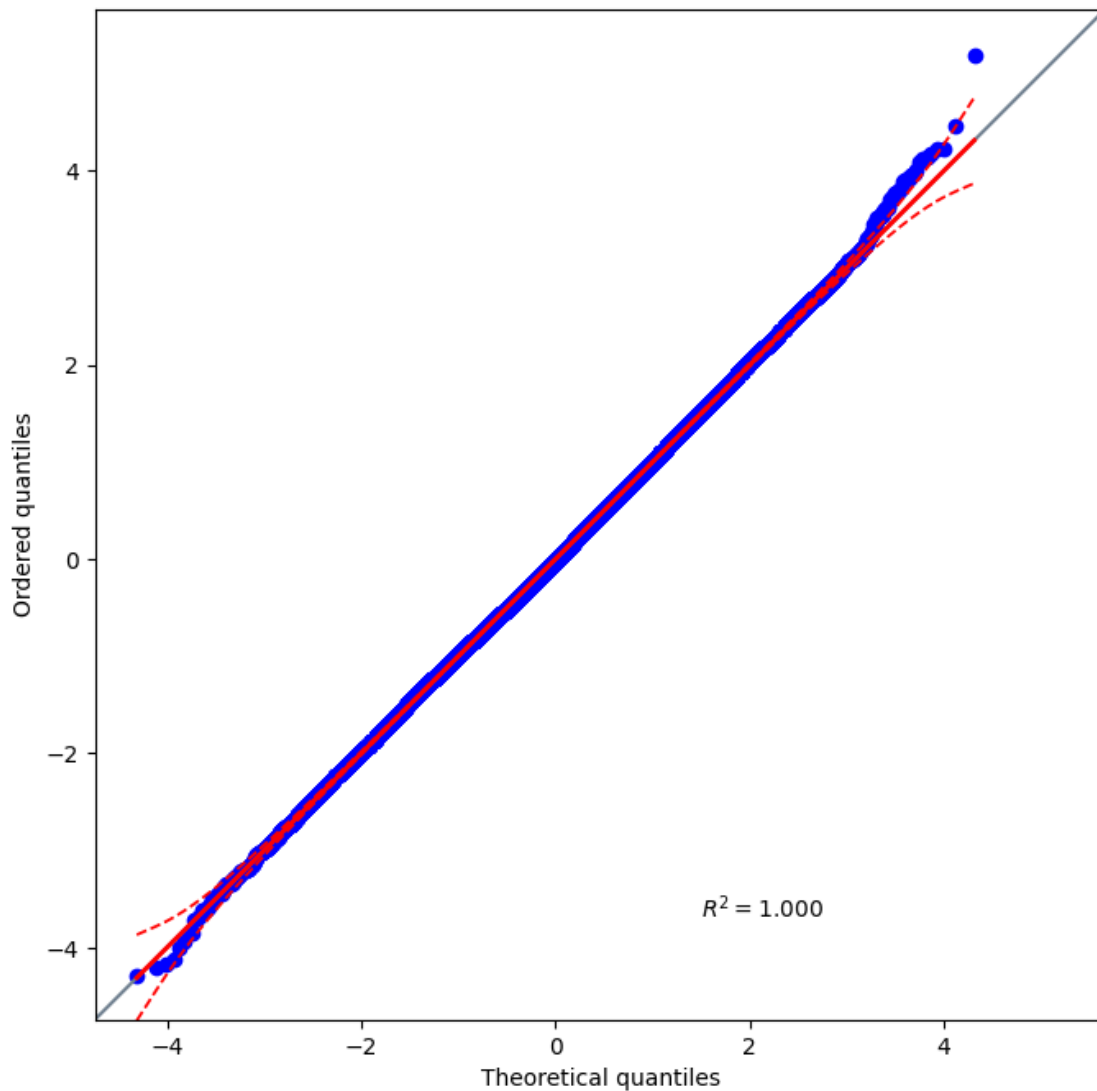
```
[50]:
```

Dep. Variable:	review_score	Log-Likelihood:	-93726.
Model:	OrderedModel	AIC:	1.877e+05
Method:	Maximum Likelihood	BIC:	1.889e+05
Date:	Tue, 02 Apr 2024		
Time:	21:58:19		
No. Observations:	85522		
Df Residuals:	85396		
Df Model:	122		

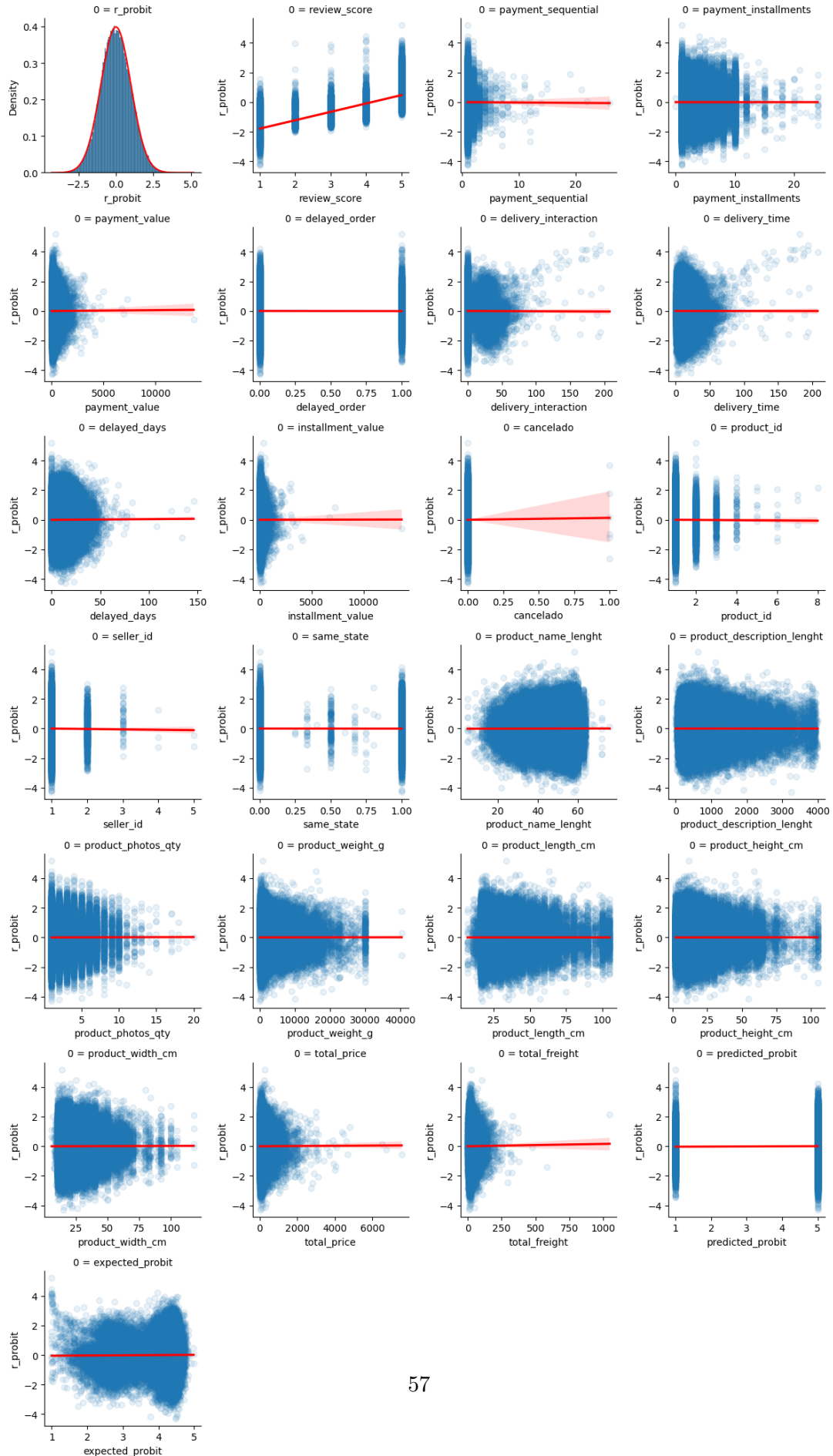
	coef	std e
payment__sequential	0.0153	0.01
payment__installments	0.0014	0.00
payment__value	-0.0021	0.00
delayed__order	-0.9176	0.03
delivery__interaction	0.0047	0.00
delivery__time	-0.0280	0.00
delayed__days	0.0015	0.00
installment__value	6.306e-05	5.36e-
cancelado	-1.5632	0.49
product__id	-0.1444	0.02
seller__id	-0.7997	0.03
same__state	0.0001	0.01
product__name__lenght	-0.0021	0.00
product__description__lenght	3.401e-06	7e-0
product__photos__qty	-0.0040	0.00
product__weight__g	-5.69e-07	1.82e-
product__length__cm	0.0001	0.00
product__height__cm	-0.0008	0.00
product__width__cm	0.0003	0.00
total__price	0.0021	0.00
total__freight	0.0020	0.00
payment__type__boleto	0.1065	0.03
payment__type__credit_card	0.0731	0.03
payment__type__debit_card	0.1160	0.04
product__category__name__agro_industria_e_comercio	-0.1894	0.09
product__category__name__alimentos	0.1215	0.06
product__category__name__alimentos_bebidas	0.1944	0.09
product__category__name__artes	-0.0904	0.09
product__category__name__artes_e_artesanato	-0.0737	0.27
product__category__name__artigos_de_festas	-0.2518	0.20
product__category__name__artigos_de_natal	-0.0607	0.11
product__category__name__audio	-0.2557	0.06
product__category__name__automotivo	-0.0306	0.02
product__category__name__bebes	-0.0265	0.03
product__category__name__bebidas	0.0392	0.07
product__category__name__beleza_saude	0.0551	0.02
product__category__name__brinquedos	0.0408	0.02
product__category__name__cama_mesa_banho	-0.1046	0.02
product__category__name__casa_conforto	-0.0520	0.07
product__category__name__casa_conforto_2	-0.0804	0.26
product__category__name__casa_construcao	-0.0777	0.05
product__category__name__cds_dvds_musicais	0.2974	0.38
product__category__name__cine_foto	0.3027	0.17
product__category__name__climatizacao	-0.0886	0.08
product__category__name__consoles_games	0.0269	0.04
product__category__name__construcao_ferramentas_construcao	-0.0253	0.05
product__category__name__construcao_ferramentas_ferramentas	0.1593	0.13
product__category__name__construcao_ferramentas_iluminacao	-0.0191	0.08
product__category__name__construcao_ferramentas_jardim	0.1482	0.09
product__category__name__construcao_ferramentas_seguranca	-0.2467	0.09
product__category__name__cool_stuff	0.0093	0.02
product__category__name__dvds_blu_ray	0.0332	0.17

```
[51]: r_probit = residuo_substituto(resf_probit, df_reg_train.review_score.values,
    ↪ stats.norm)
df_reg_train['r_probit'] = r_probit
df_reg_train['predicted_probit'] = np.argmax(resf_probit.model.
    ↪ predict(resf_probit.params), axis=1) + 1
df_reg_train['expected_probit'] = np.sum(resf_probit.model.predict(resf_probit.
    ↪ params) @ mult, axis=1)
pg.qqplot(r_probit, dist=stats.norm, confidence=.95)
```

```
[51]: <Axes: xlabel='Theoretical quantiles', ylabel='Ordered quantiles'>
```



```
[52]: plota_residuos(df_reg_train, 'r_probit', ['r_probit', 'review_score'] +
    ↪ covariaveis_unicas + ['predicted_probit', 'expected_probit'], stats.norm)
```

```
[53]: reporta_predicoes(resf_probit, df_reg_train.review_score)
```

MSE de acordo com os valores esperados: 1.33
MSE de acordo com a nota mais provável: 2.01
Acurácia das notas mais prováveis: 61.52%

1.4 Regressão ordinal LogLog

```
[ ]: modf_loglog = OrderedModel(df_reg_train['review_score'],  
    df_reg_train[covariaveis].astype(float),  
                                distr=stats.gumbel_r)  
resf_loglog = modf_loglog.fit(method='bfgs')  
resf_loglog.summary()
```

/home/lucas/.cache/pypoetry/virtualenvs/ecommerce-classification-xG4A-NiM-
py3.10/lib/python3.10/site-packages/scipy/optimize/_optimize.py:1397:
OptimizeWarning: Maximum number of iterations has been exceeded.

```
res = _minimize_bfgs(f, x0, args, fprime, callback=callback, **opts)
```

```
Current function value: 1.092161
```

```
Iterations: 500
```

```
Function evaluations: 510
```

```
Gradient evaluations: 510
```

/home/lucas/.cache/pypoetry/virtualenvs/ecommerce-classification-xG4A-NiM-
py3.10/lib/python3.10/site-packages/statsmodels/base/model.py:607:

ConvergenceWarning: Maximum Likelihood optimization failed to converge. Check
mle_retvals

```
warnings.warn("Maximum Likelihood optimization failed to "
```

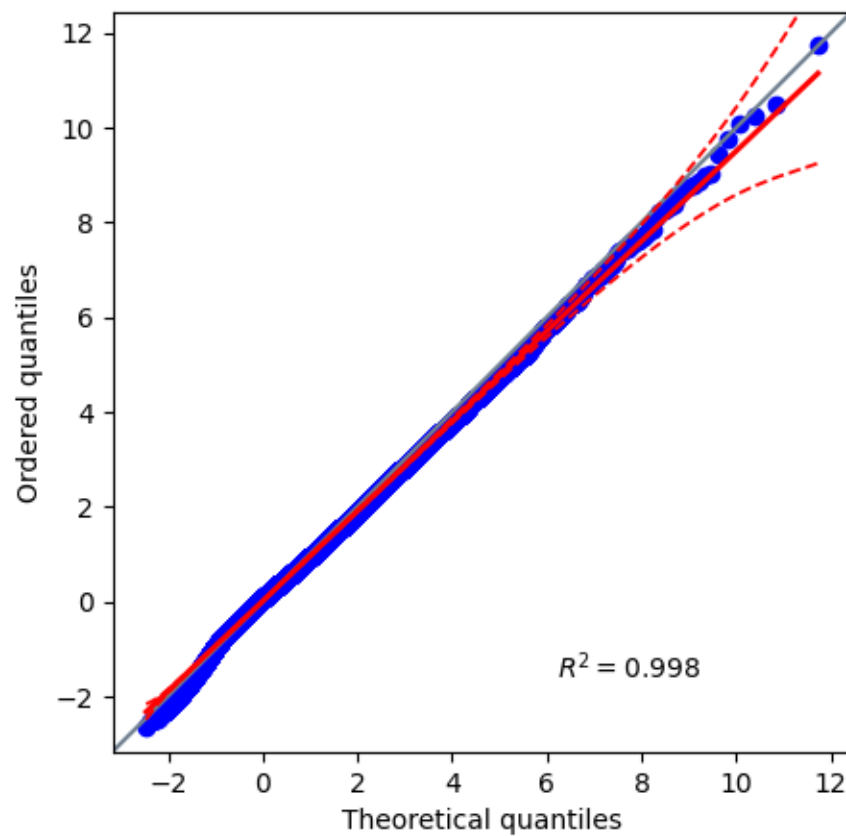
```
[ ]:
```

Dep. Variable:	review_score	Log-Likelihood:	-93404.
Model:	OrderedModel	AIC:	1.871e+05
Method:	Maximum Likelihood	BIC:	1.882e+05
Date:	Tue, 02 Apr 2024		
Time:	10:18:21		
No. Observations:	85522		
Df Residuals:	85397		
Df Model:	121		

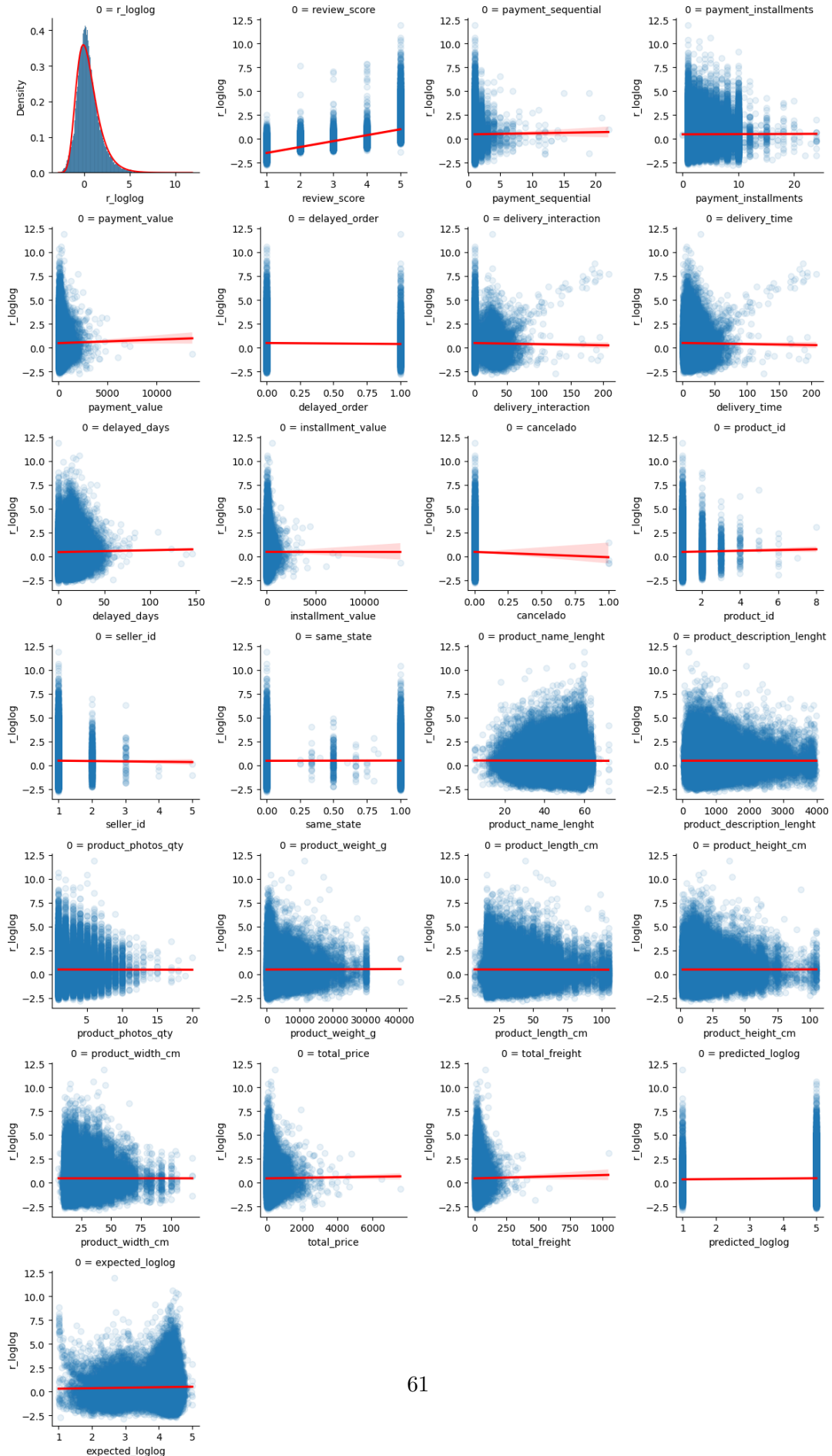
	coef	std
payment__sequential	0.0032	0.0000
payment__installments	-0.0008	0.0000
payment__value	-0.0028	0.0000
delayed__order	-0.4265	0.0000
delivery__interaction	-0.0157	0.0000
delivery__time	-0.0252	0.0000
delayed__days	-0.0001	0.0000
installment__value	8.042e-05	4.87e-05
cancelado	-4.9147	3.6000
product__id	-0.1968	0.0000
seller__id	-0.6743	0.0000
same__state	-0.0288	0.0000
product__name__lenght	-0.0017	0.0000
product__description__lenght	6.975e-06	6.38e-06
product__photos__qty	-0.0041	0.0000
product__weight__g	-1.276e-06	1.68e-06
product__length__cm	0.0004	0.0000
product__height__cm	-0.0004	0.0000
product__width__cm	0.0004	0.0000
total__price	0.0029	0.0000
total__freight	0.0029	0.0000
payment__type__boleto	0.0958	0.0000
payment__type__credit_card	0.0849	0.0000
payment__type__debit_card	0.1090	0.0000
product__category__name__agro_industria_e_comercio	-0.1926	0.0000
product__category__name__alimentos	0.1098	0.0000
product__category__name__alimentos_bebidas	0.2090	0.0000
product__category__name__artes	-0.1973	0.0000
product__category__name__artes_e_artesanato	0.1379	0.2000
product__category__name__artigos_de_festas	-0.2937	0.1000
product__category__name__artigos_de_natal	-0.0604	0.1000
product__category__name__audio	-0.2159	0.0000
product__category__name__automotivo	-0.0249	0.0000
product__category__name__bebes	-0.0166	0.0000
product__category__name__bebidas	0.0762	0.0000
product__category__name__beleza_saude	0.0528	0.0000
product__category__name__brinquedos	0.0398	0.0000
product__category__name__cama_mesa_banho	-0.0862	0.0000
product__category__name__casa_conforto	-0.0924	0.0000
product__category__name__casa_conforto_2	0.0981	0.2000
product__category__name__casa_construcao	-0.0617	0.0000
product__category__name__cds_dvds_musicais	0.2839	0.3000
product__category__name__cine_foto	0.2202	0.1000
product__category__name__climatizacao	-0.0618	0.0000
product__category__name__consoles_games	0.0402	0.0000
product__category__name__construcao_ferramentas_construcao	-0.0373	0.0000
product__category__name__construcao_ferramentas_ferramentas	0.2163	0.1000
product__category__name__construcao_ferramentas_iluminacao	0.0168	0.0000
product__category__name__construcao_ferramentas_jardim	0.2059	0.0000
product__category__name__construcao_ferramentas_seguranca	-0.2373	0.0000
product__category__name__cool_stuff	0.0195	0.0000
product__category__name__dvds_blu_ray	0.0479	0.1000

```
[ ]: r_loglog = residuo_substituto(resf_loglog, df_reg_train.review_score.values,
    ↪stats.gumbel_r)
df_reg_train['r_loglog'] = r_loglog
df_reg_train['predicted_loglog'] = np.argmax(resf_loglog.model.
    ↪predict(resf_loglog.params), axis=1) + 1
df_reg_train['expected_loglog'] = np.sum(resf_loglog.model.predict(resf_loglog.
    ↪params) @ mult, axis=1)
pg.qqplot(r_loglog, dist=stats.gumbel_r, confidence=.95)
```

```
[ ]: <Axes: xlabel='Theoretical quantiles', ylabel='Ordered quantiles'>
```



```
[ ]: plota_residuos(df_reg_train, 'r_loglog', ['r_loglog', 'review_score'] +
    ↪covariaveis_unicas + ['predicted_loglog', 'expected_loglog'], stats.gumbel_r)
```



```
[ ]: reporta_predicoes(resf_loglog, df_reg_train.review_score)
```

MSE de acordo com os valores esperados: 1.32
MSE de acordo com a nota mais provável: 1.99
Acurácia das notas mais prováveis: 61.64%

```
[ ]: r_probit.mean(), r_logit.mean(), r_loglog.mean()
```

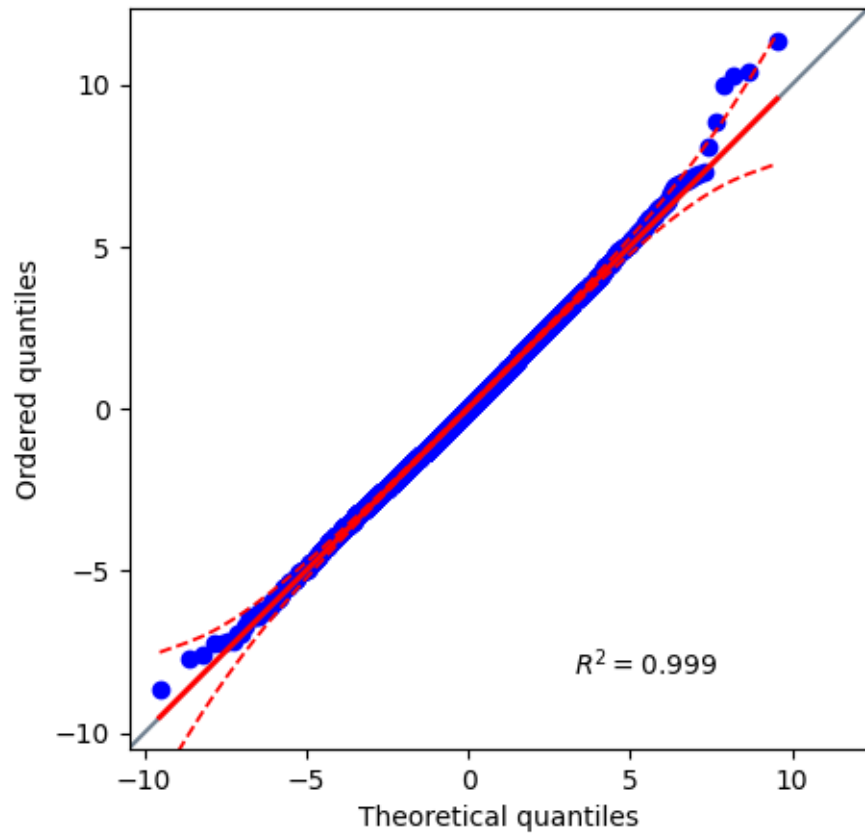
```
[ ]: (-0.0007963687629919248, 0.015365549142877034, 0.475517297550056)
```

Apesar das métricas de previsão melhores para a regressão loglog, o seu resíduo tem valor distante de 0, indicando um ajuste não tão bom

1.5 Teste Regressão ordinal logit

```
[ ]: probs_preditas_logit = resf_logit.model.predict(resf_logit.params,
    ↪exog=df_reg_test[covariaveis].astype(float))
df_reg_test['predicted_logit'] = np.argmax(probs_preditas_logit, axis=1) + 1
df_reg_test['expected_logit'] = np.sum(probs_preditas_logit @ mult, axis=1)
r_probit = residuo_substituto(resf_probit, df_reg_test.review_score.values,
    ↪stats.logistic, new_data=df_reg_test[covariaveis].astype(float))
df_reg_test['r_logit'] = r_logit
pg.qqplot(r_logit, dist=stats.logistic, confidence=.95)
```

```
[ ]: <Axes: xlabel='Theoretical quantiles', ylabel='Ordered quantiles'>
```



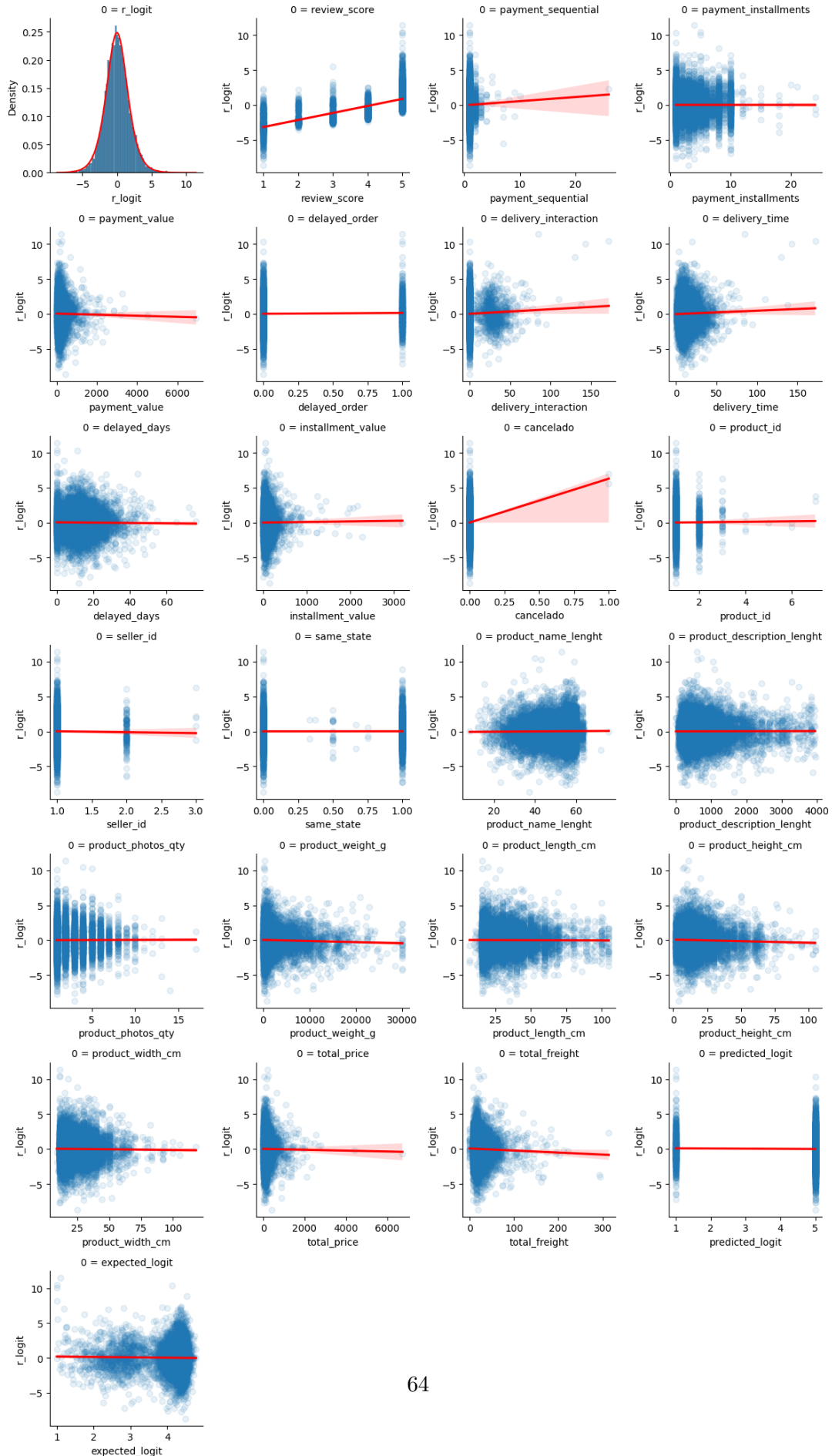
```
[ ]: reporta_predicoes(resf_logit, df_reg_test.review_score,
    ↪exog=df_reg_test[covariaveis].astype(float))
```

MSE de acordo com os valores esperados: 1.35

MSE de acordo com a nota mais provável: 2.05

Acurácia das notas mais prováveis: 61.09%

```
[ ]: plota_residuos(df_reg_test, 'r_logit', ['r_logit', 'review_score'] +
    ↪covariaveis_unicas + ['predicted_logit', 'expected_logit'], stats.logistic)
```



1.6 Random Forest

```
[28]: clf = ensemble.RandomForestClassifier()
param_grid = {
    'max_depth': range(1, 16),
    'min_samples_split': range(2, 5),
    'min_samples_leaf': range(1, 5),
    'n_estimators': range(100, 200, 20)
}

grid = model_selection.GridSearchCV(clf, param_grid)
grid.fit(df_reg_train[covariaveis], df_reg_train['review_score'])

# Obtém predições para a base de teste
predito = grid.predict(df_reg_test[covariaveis])
print(f'melhores parametros: {grid.best_params_}')
print(f"Acurácia do modelo de floresta aleatória: {metrics.
↳accuracy_score(df_reg_test['review_score'], predito):.2f}")
```

melhores parametros: {'max_depth': 14}
Acurácia do modelo de floresta aleatória: 0.62

```
[29]: pd.DataFrame({'importancia': grid.best_estimator_.feature_importances_,
↳'variavel': covariaveis}).sort_values('importancia', ascending=False).iloc[:
↳30]
```

```
[29]:
```

	importancia	variavel
4	0.132800	delivery_interaction
5	0.130031	delivery_time
6	0.112665	delayed_days
3	0.086712	delayed_order
20	0.042212	total_freight
7	0.037709	installment_value
2	0.037540	payment_value
19	0.035136	total_price
13	0.034209	product_description_lenght
15	0.032871	product_weight_g
12	0.029304	product_name_lenght
17	0.029268	product_height_cm
16	0.026941	product_length_cm
18	0.026779	product_width_cm
10	0.018686	seller_id
9	0.017976	product_id
1	0.017331	payment_installments
14	0.016630	product_photos_qty

11	0.009102	same_state
114	0.006836	customer_state_RJ
22	0.006385	payment_type_credit_card
79	0.005528	product_category_name_moveis_escritorio
37	0.005515	product_category_name_cama_mesa_banho
21	0.005140	payment_type_boleto
106	0.003938	customer_state_MG
0	0.003709	payment_sequential
78	0.003561	product_category_name_moveis_decoracao
100	0.003359	customer_state_BA
35	0.003142	product_category_name_beleza_saude
94	0.003001	product_category_name_telefonia

1.7 Naive-Bayes

```
[ ]: # Cria um classificador Bayes ingênuo (Naive Bayes)
gnb = naive_bayes.GaussianNB()

# Treina o classificador na base de treinamento
gnb.fit(df_reg_train[covariaveis], df_reg_train['review_score'])

# Predição dos rótulos na base de teste
predito = gnb.predict(df_reg_test[covariaveis])

print(f"Acurácia do modelo de Naive-Bayes: {metrics.
      ↳accuracy_score(df_reg_test['review_score'], predito):.2f}")
```

Acurácia do modelo de Naive-Bayes: 0.57

1.8 KNN

```
[43]: from sklearn.preprocessing import StandardScaler

classifier = model_selection.GridSearchCV(neighbors.KNeighborsClassifier(),
                                          {'n_neighbors': range(30, 35)},
                                          scoring='accuracy')

# treina o classificador com os dados de treinameto
scaler = StandardScaler()
X_train_scale = scaler.fit_transform(df_reg_train[covariaveis])
X_test_scale = scaler.transform(df_reg_test[covariaveis])
classifier.fit(X_train_scale, df_reg_train['review_score'])

# Obtém predições para a base de teste
predito = classifier.predict(X_test_scale)
print(f'melhores parametros: {classifier.best_params_}')
```

```
print(f"Acurácia do modelo de KNN {metrics.  
↳accuracy_score(df_reg_test['review_score'], predito):.2f}")
```

melhores parametros: {'n_neighbors': 33}
Acurácia do modelo de KNN 0.61

2 Análise do modelo de regressão ordinal Logístico

```
[59]: parametros = pd.DataFrame({'param': resf_logit.params, 'order': np.  
↳absolute(resf_logit.params), 'RC': np.exp(resf_logit.params), 'p':  
↳resf_logit.pvalues}).sort_values('order', ascending=False)  
parametros[parametros['p'] <= 0.05]
```

```
[59]:
```

	param	order \
1/2	-5.164167	5.164167
cancelado	-2.856894	2.856894
seller_id	-1.455406	1.455406
delayed_order	-1.065513	1.065513
2/3	-0.991998	0.991998
product_category_name_livros_interesse_geral	0.702292	0.702292
customer_state_AM	0.553305	0.553305
product_category_name_livros_tecnicos	0.506331	0.506331
product_category_name_fashion_calçados	0.410149	0.410149
product_category_name_fashion_roupa_masculina	-0.409666	0.409666
product_category_name_construcao_ferramentas_se...	-0.397313	0.397313
product_category_name_audio	-0.396051	0.396051
product_category_name_moveis_escritorio	-0.370449	0.370449
3/4	-0.336163	0.336163
product_category_name_telefonia_fixa	-0.318910	0.318910
product_category_name_agro_industria_e_comercio	-0.316950	0.316950
customer_state_DF	-0.308306	0.308306
customer_state_SP	-0.300133	0.300133
product_category_name_papelaria	0.258483	0.258483
product_category_name_market_place	-0.255910	0.255910
customer_state_MG	-0.251376	0.251376
product_id	-0.231003	0.231003
product_category_name_malas_acessorios	0.224628	0.224628
product_category_name_telefonia	-0.196524	0.196524
payment_type_debit_card	0.190028	0.190028
payment_type_boleto	0.175918	0.175918
product_category_name_perfumaria	0.165890	0.165890
product_category_name_cama_mesa_banho	-0.163139	0.163139
product_category_name_pet_shop	0.155692	0.155692
product_category_name_informatica_acessorios	-0.117670	0.117670
payment_type_credit_card	0.112193	0.112193
product_category_name_beleza_saude	0.102152	0.102152

product_category_name_relogios_presentes	-0.085814	0.085814
product_category_name_esporte_lazer	0.076118	0.076118
4/5	0.064594	0.064594
delivery_time	-0.049918	0.049918
delivery_interaction	-0.010034	0.010034
total_price	0.003631	0.003631
product_name_lenght	-0.003617	0.003617
total_freight	0.003577	0.003577
payment_value	-0.003491	0.003491
delayed_days	0.002368	0.002368

	RC	P
1/2	0.005718	2.014396e-210
cancelado	0.057447	3.588707e-03
seller_id	0.233306	6.172873e-101
delayed_order	0.344551	5.041127e-62
2/3	0.370835	0.000000e+00
product_category_name_livros_interesse_geral	2.018373	4.971047e-09
customer_state_AM	1.738991	1.324183e-02
product_category_name_livros_tecnicos	1.659192	8.410585e-04
product_category_name_fashion_calcados	1.507042	5.872696e-03
product_category_name_fashion_roupa_masculina	0.663872	4.896240e-02
product_category_name_construcao_ferramentas_se...	0.672124	1.738321e-02
product_category_name_audio	0.672972	5.558317e-04
product_category_name_moveis_escritorio	0.690424	4.078485e-08
3/4	0.714507	1.402905e-187
product_category_name_telefonia_fixa	0.726941	2.907588e-02
product_category_name_agro_industria_e_comercio	0.728367	3.727399e-02
customer_state_DF	0.734690	2.067791e-02
customer_state_SP	0.740720	1.770389e-02
product_category_name_papelaria	1.294964	2.228226e-06
product_category_name_market_place	0.774211	4.893097e-02
customer_state_MG	0.777730	4.721346e-02
product_id	0.793737	2.694543e-09
product_category_name_malas_acessorios	1.251857	3.319734e-03
product_category_name_telefonia	0.821582	1.483178e-05
payment_type_debit_card	1.209283	1.532836e-02
payment_type_boleto	1.192340	1.813896e-03
product_category_name_perfumaria	1.180444	9.653152e-04
product_category_name_cama_mesa_banho	0.849473	1.325918e-05
product_category_name_pet_shop	1.168467	1.028895e-02
product_category_name_informatica_acessorios	0.888990	3.183750e-03
payment_type_credit_card	1.118729	4.304680e-02
product_category_name_beleza_saude	1.107552	7.677115e-03
product_category_name_relogios_presentes	0.917765	4.557904e-02
product_category_name_esporte_lazer	1.079090	4.779715e-02
4/5	1.066726	3.308856e-19

delivery_time	0.951308	2.011927e-247
delivery_interaction	0.990016	1.608497e-05
total_price	1.003638	1.119194e-49
product_name_lenght	0.996390	8.593978e-07
total_freight	1.003584	1.108029e-06
payment_value	0.996515	1.965261e-47
delayed_days	1.002371	3.077013e-02

Vemos que a importância das variáveis é diferente entre o modelo de Floresta Aleatória e a regressão ordinal. Como na regressão ordinal podemos quantificar a incerteza, e pudemos validar o ajuste do modelo, é recomendado seguir com a interpretação desse modelo

[]: