Report

Best Model: `CatBoostRegressor`

Metric: R Squared

Test Size: 0.8

Score: 0.651

In [1]:
```
# Initialize Otter
import otter
grader = otter.Notebook("lab4.ipynb")
```

Lab 4: Putting it all together in a mini project

This lab is an optional group lab. You can choose to work alone of in a group of up to four students. You are in charge of how you want to work and who you want to work with. Maybe you really want to go through all the steps of the ML process yourself or maybe you want to practice your collaboration skills, it is up to you! Just remember to indicate who your group members are (if any) when you submit on Gradescope. If you choose to work in a group, you only need to use one of your GitHub repos.

Submission instructions

rubric={mechanics}

You receive marks for submitting your lab correctly, please follow these instructions:

- Follow the general lab instructions.
- Click here to view a description of the rubrics used to grade the questions
- Make at least three commits.
- Push your `.ipynb` file to your GitHub repository for this lab and upload it to Gradescope.
  - Before submitting, make sure you restart the kernel and rerun all cells.
- Also upload a `.pdf` export of the notebook to facilitate grading of manual questions (preferably WebPDF, you can select two files when uploading to gradescope)
- Don't change any variable names that are given to you, don't move cells around, and don't include any code to install packages in the notebook.
- The data you download for this lab **SHOULD NOT BE PUSHED TO YOUR REPOSITORY** (there is also a `.gitignore` in the repo to prevent this).

- Include a clickable link to your GitHub repo for the lab just below this cell
  - It should look something like this https://github.ubc.ca/MDS-2020-21/DSCI_531_labX_yourcwl.

Points: 2

Public repo link (under github.com/UBC-MDS):
https://github.com/UBC-MDS/AirBNB_Luke_HanChen_573lab4

Authors:
Luke Yang
HanChen Wang
Dec 10 2022

Introduction

In this lab you will be working on an open-ended mini-project, where you will put all the different things you have learned so far in 571 and 573 together to solve an interesting problem.

A few notes and tips when you work on this mini-project:

Tips

1. Since this mini-project is open-ended there might be some situations where you'll have to use your own judgment and make your own decisions (as you would be doing when you work as a data scientist). Make sure you explain your decisions whenever necessary.
2. **Do not include everything you ever tried in your submission** -- it's fine just to have your final code. That said, your code should be reproducible and well-documented. For example, if you chose your hyperparameters based on some hyperparameter optimization experiment, you should leave in the code for that experiment so that someone else could re-run it and obtain the same hyperparameters, rather than mysteriously just setting the hyperparameters to some (carefully chosen) values in your code.
3. If you realize that you are repeating a lot of code try to organize it in functions. Clear presentation of your code, experiments, and results is the key to be successful in this lab. You may use code from lecture notes or previous lab solutions with appropriate attributions.

Assessment

We don't have some secret target score that you need to achieve to get a good grade. **You'll be assessed on demonstration of mastery of course topics, clear presentation, and the quality of your analysis and results.** For example, if you just

have a bunch of code and no text or figures, that's not good. If you instead do a bunch of sane things and you have clearly motivated your choices, but still get lower model performance than your friend, don't sweat it.

A final note

Finally, the style of this "project" question is different from other assignments. It'll be up to you to decide when you're "done" -- in fact, this is one of the hardest parts of real projects. But please don't spend WAY too much time on this... perhaps "several hours" but not "many hours" is a good guideline for a high quality submission. Of course if you're having fun you're welcome to spend as much time as you want! But, if so, try not to do it out of perfectionism or getting the best possible grade. Do it because you're learning and enjoying it. Students from the past cohorts have found such kind of labs useful and fun and we hope you enjoy it as well.

1. Pick your problem and explain the prediction problem

rubric={reasoning}

In this mini project, you will pick one of the following problems:

1. A classification problem of predicting whether a credit card client will default or not. For this problem, you will use Default of Credit Card Clients Dataset. In this data set, there are 30,000 examples and 24 features, and the goal is to estimate whether a person will default (fail to pay) their credit card bills; this column is labeled "default.payment.next.month" in the data. The rest of the columns can be used as features. You may take some ideas and compare your results with the associated research paper, which is available through the UBC library.

OR

2. A regression problem of predicting `reviews_per_month`, as a proxy for the popularity of the listing with New York City Airbnb listings from 2019 dataset. Airbnb could use this sort of model to predict how popular future listings might be before they are posted, perhaps to help guide hosts create more appealing listings. In reality they might instead use something like vacancy rate or average rating as their target, but we do not have that available here.

Your tasks:

1. Spend some time understanding the problem and what each feature means. Write a few sentences on your initial thoughts on the problem and the dataset.
2. Download the dataset and read it as a pandas dataframe.

3. Carry out any preliminary preprocessing, if needed (e.g., changing feature names, handling of NaN values etc.)

Points: 3

1.Airbnb is a popular alternative accommodation other than hotels or inns at guest cities provided by individual renters/owners. This data set describes Airbnb's accommodation listings in New Year City in 2019. It includes metrics such as the ID and name of the listing, host's ID and name, location and neighborhood, room type, price, amount of nights minimum, last review date, number of reviews, availability etc.

Particularly, it has our target column which is the number of review per month. This prediction helps Airbnb to identify the popular listings and help hosts to modify their listing in favour of more guests and reviews. Thus, in this analysis, we are trying to answer the following prediction question:

Given the metrics of an Airbnb accommodation listing such as the location, room type, pricing, minimum nights, availability, etc, what are the predicted number of reviews per month for that listing?

In [2]:
```
# 1.2 downloading and reading the dataset as a pandas dataframe.
import pandas as pd
import altair as alt
from altair_data_server import data_server

# Save a vega-lite spec and a PNG blob for each plot in the notebook
alt.renderers.enable('mimetype')
# Handle large data sets without embedding them in the notebook
alt.data_transformers.enable('data_server')

df = pd.read_csv('data/raw/AB_NYC_2019.csv')
df.head()
```

Out[2]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latit
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75
2	3647	THE VILLAGE OF HARLEM....NEW YORK !	4632	Elisabeth	Manhattan	Harlem	40.80
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79

In [3]:
```
# 3. preliminary preprocessing (removing nulls)
df = df.query('not reviews_per_month.isnull()')
df.head()
```

Out[3]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude
0	2539	Clean & quiet apt home by the park	2787	John	Brooklyn	Kensington	40.64749
1	2595	Skylit Midtown Castle	2845	Jennifer	Manhattan	Midtown	40.75362
3	3831	Cozy Entire Floor of Brownstone	4869	LisaRoxanne	Brooklyn	Clinton Hill	40.68514
4	5022	Entire Apt: Spacious Studio/Loft by central park	7192	Laura	Manhattan	East Harlem	40.79851
5	5099	Large Cozy 1 BR Apartment In Midtown East	7322	Chris	Manhattan	Murray Hill	40.74767

In [4]:
```
# Looking at the uniqueness and non-null counts of the data.
unique_df = pd.DataFrame()
unique_df['columns'] = df.columns
unique_df['valid_count'] = df.count(axis=0).reset_index()[0]
unique_df['unique_count'] = df.nunique().reset_index()[0]
unique_df
```

Out[4]:

	columns	valid_count	unique_count
0	id	38843	38843
1	name	38837	38269
2	host_id	38843	30251
3	host_name	38827	9886
4	neighbourhood_group	38843	5
5	neighbourhood	38843	218
6	latitude	38843	17443
7	longitude	38843	13641
8	room_type	38843	3
9	price	38843	581
10	minimum_nights	38843	89
11	number_of_reviews	38843	393
12	last_review	38843	1764
13	reviews_per_month	38843	937
14	calculated_host_listings_count	38843	47
15	availability_365	38843	366

On the large scale, this dataset contains 38843 records where the target column is not missing. It has 15 columns and the target column. After filtering on non-null targets, most of the columns do not contain `NaN`, except two columns which are listing name and host name.

2. Data splitting

rubric={reasoning}

Your tasks:

1. Split the data into train and test portions.

    > Make the decision on the `test_size` based on the capacity of your laptop.

Points: 1

In [5]:
```
# We will split the data on a 2:8 train:test ratio
# to facilitate the speed of model building.
from sklearn.model_selection import train_test_split
```

```
train_df, test_df = train_test_split(df, test_size=0.8, random_state=573)
# feature engineer from part 4 (before EDA)
latest_day = max(pd.to_datetime(train_df["last_review"]))
train_df["days_from_last_review"] = (
     latest_day - (pd.to_datetime(train_df["last_review"]))
).dt.days
```

In [6]:
```
train_df
```

Out[6]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhc
27438	21630372	Beautiful and confortable room	109146538	Marisol	Queens	Jack Heig
13825	10435733	Cute Spacious Williamsburg Apt	5927655	Ann	Brooklyn	Williamsb
473	166983	3 BR, Beautiful Brooklyn Duplex	795640	Jilly	Brooklyn	Carroll Gard
46103	35081477	West Village Gem! Like Paris	196890	Siobhan	Manhattan	West Vill
33450	26443376	Big 1BR in PRIME Bushwick! 2 blocks to L Train!	6293227	Nish	Brooklyn	Bushw
...	...	...	...	...	...	
24387	19634067	Gigantic, convenient loft in S Williamsburg!	13785996	David	Brooklyn	Williamsb
34600	27435989	**Studio Apartment 20 min. From MANHATTAN **	206758830	Chris And Jocelyn	Queens	Middle Vill
33735	26743564	Brand New Brooklyn Style Hostel RM3 #3	119669058	Melissa	Brooklyn	Bedfc Stuyves
4106	2636643	darling small studio	13501034	Tanya	Manhattan	Upper East S
11381	8844652	Quiet living room in Greenpoint 1BR	34738391	Rae	Brooklyn	Greenpc

7768 rows × 17 columns

3. EDA

rubric={viz,reasoning}

Perform exploratory data analysis on the train set.

Your tasks:

1. Include at least two summary statistics and two visualizations that you find useful, and accompany each one with a sentence explaining it.
2. Summarize your initial observations about the data.
3. Pick appropriate metric/metrics for assessment.

Points: 6

1. The dataset contains 15 columns and a target `reviews_per_month`. Looking at the training data, we have only two columns with small number of missing values, which are `name` and `host_name`.

   For the `train_df.describe()` summary statistics, we report the summary of numeric-like columns. It is clear that we need to perform `StandardScaler` on these columns and perhaps discretinization for the longitude and latitude columns.

2. For the `train_df['neighbourhood'].value_counts()` summary statistics and the `Number of Airbnb listings in neighbourhoods of New York` graph, we can see that there are 5 neighbourhood groups listed and there seems to be imbalance in the count of listings.

   For the `Correlation matrix`, we can see that `number_of_reviews`, and `availability_365` have the most **positive** correlations to target `reviews_per_month`, and `minimum_nights` has the most **negative** correlation to `reviews_per_month`.

3. We can use R2 score for the assessment of how well our data is predicting the Number of reviews per month an Airbnb listing will have.

In [7]:
```
train_df.head()
```

Out[7]:

	id	name	host_id	host_name	neighbourhood_group	neighbourho
27438	21630372	Beautiful and confortable room	109146538	Marisol	Queens	Jacks Heigh
13825	10435733	Cute Spacious Williamsburg Apt	5927655	Ann	Brooklyn	Williamsbu
473	166983	3 BR, Beautiful Brooklyn Duplex	795640	Jilly	Brooklyn	Carroll Garde
46103	35081477	West Village Gem! Like Paris	196890	Siobhan	Manhattan	West Villa
33450	26443376	Big 1BR in PRIME Bushwick! 2 blocks to L Train!	6293227	Nish	Brooklyn	Bushwi

In [8]:
```
train_df.tail()
```

Out[8]:

	id	name	host_id	host_name	neighbourhood_group	neighbourho
24387	19634067	Gigantic, convenient loft in S Williamsburg!	13785996	David	Brooklyn	Williamsb
34600	27435989	**Studio Apartment 20 min. From MANHATTAN **	206758830	Chris And Jocelyn	Queens	Middle Vill
33735	26743564	Brand New Brooklyn Style Hostel RM3 #3	119669058	Melissa	Brooklyn	Bedfo Stuyves
4106	2636643	darling small studio	13501034	Tanya	Manhattan	Upper East S
11381	8844652	Quiet living room in Greenpoint 1BR	34738391	Rae	Brooklyn	Greenpo

In [9]:
```
train_df.describe()
```

Out[9]:

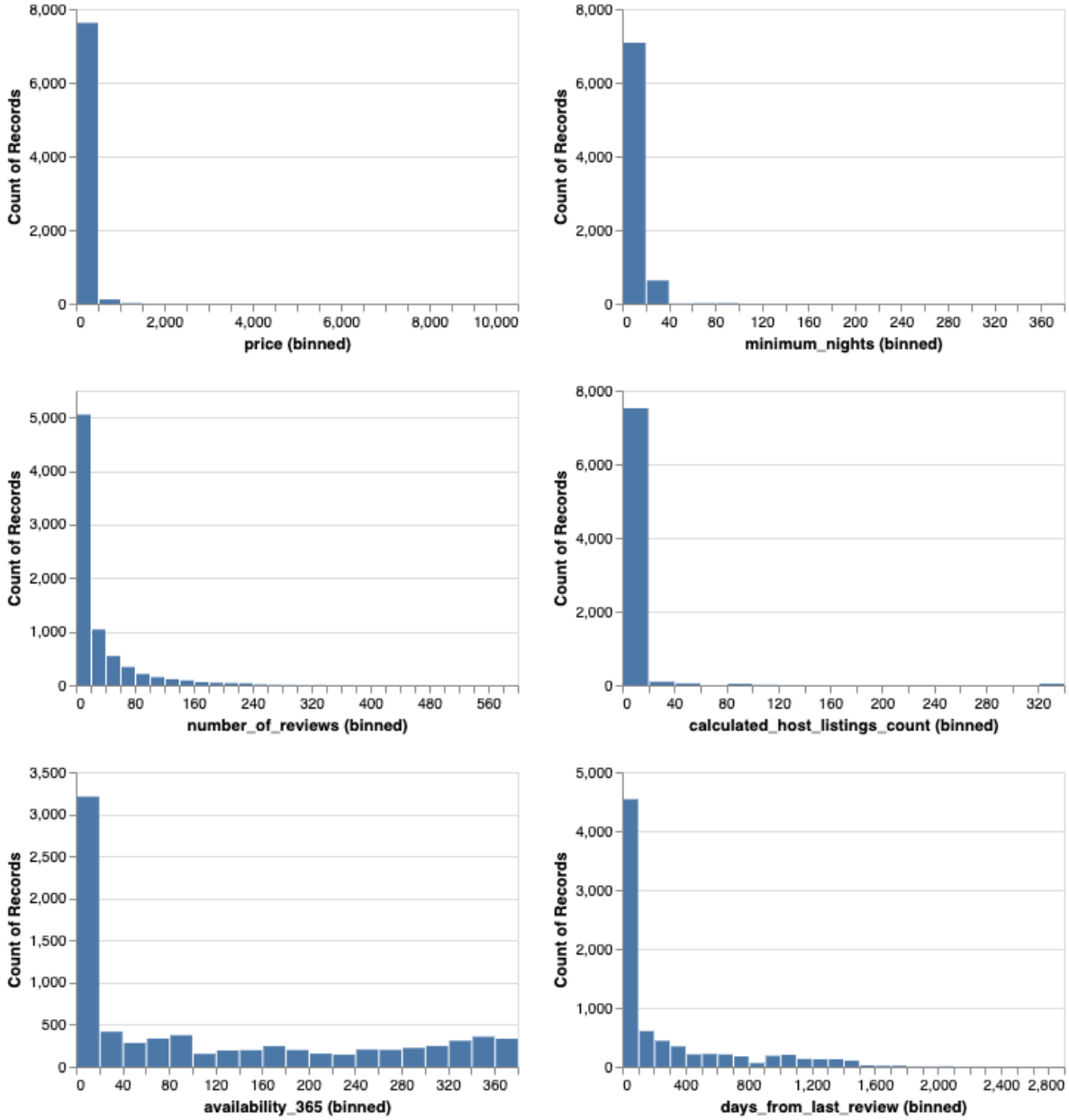
	id	host_id	latitude	longitude	price	minimum_nig
count	7.768000e+03	7.768000e+03	7768.000000	7768.000000	7768.000000	7768.000
mean	1.805075e+07	6.451151e+07	40.728054	-73.951523	142.229789	5.873
std	1.070566e+07	7.644186e+07	0.055116	0.046345	217.478782	15.288
min	5.178000e+03	2.438000e+03	40.508680	-74.244420	0.000000	1.000
25%	8.632134e+06	7.071581e+06	40.689068	-73.982930	68.000000	1.000
50%	1.871149e+07	2.767220e+07	40.721315	-73.955080	100.000000	2.000
75%	2.749310e+07	1.027564e+08	40.763040	-73.936577	165.000000	4.000
max	3.635154e+07	2.723278e+08	40.912340	-73.719280	9999.000000	365.000

In [10]:

```
import altair as alt
numeric_features = [
    "price",
    "minimum_nights",
    "number_of_reviews",
    "calculated_host_listings_count",
    "availability_365",
    "days_from_last_review"
]
alt.Chart(train_df).mark_bar().encode(
    x=alt.X(alt.repeat(), type='quantitative', bin=alt.Bin(maxbins=30)),
    y=alt.Y("count()")
).properties(
    width=300,
    height=200
).repeat(
    numeric_features,
    columns=2
)
```

```
/opt/miniconda3/lib/python3.10/site-packages/altair/utils/core.py:317: Futu
reWarning: iteritems is deprecated and will be removed in a future version.
Use .items instead.
  for col_name, dtype in df.dtypes.iteritems():
```

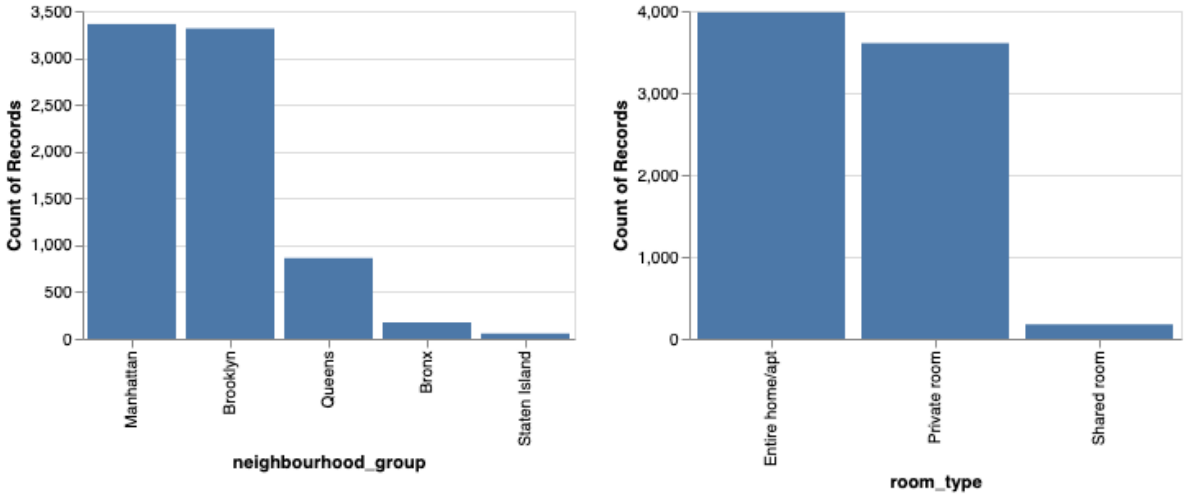
Out[10]:



In [11]:

```
categorical_features = [
    "neighbourhood_group",
    "room_type",
]
alt.Chart(train_df).mark_bar().encode(
    x=alt.X(alt.repeat(), type='nominal', sort='-y'),
    y=alt.Y("count()")
).properties(
    width=300,
    height=200
).repeat(
    categorical_features,
    columns=2
)
```

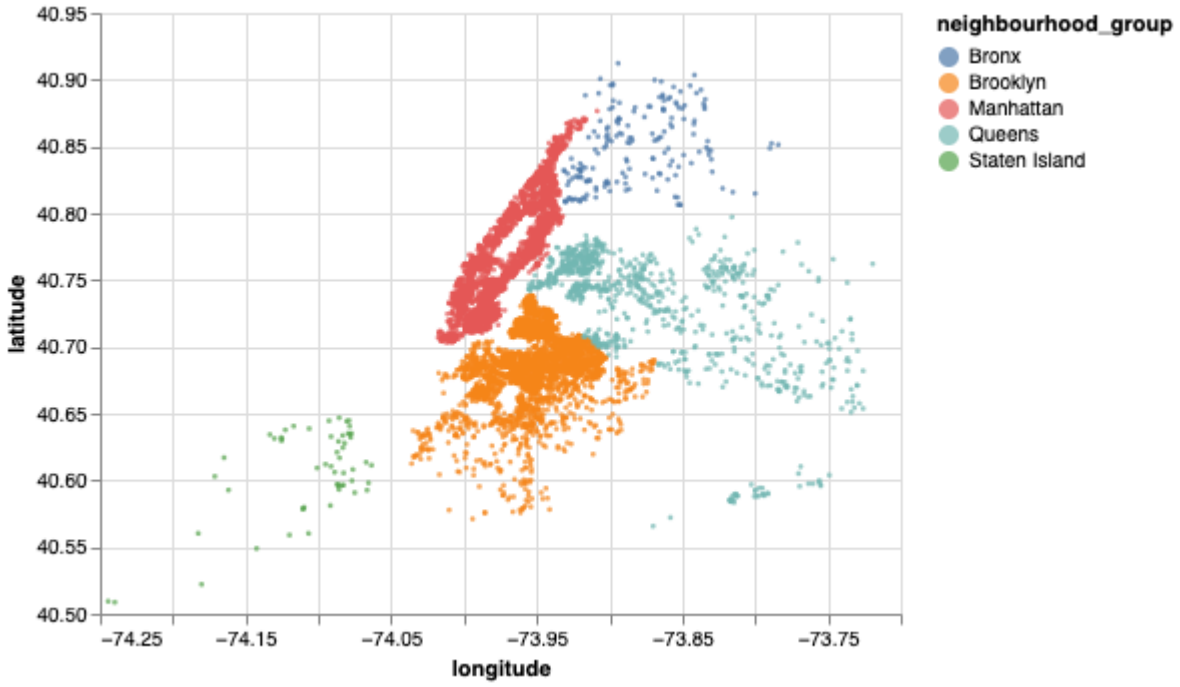
Out[11]:



In [12]:

```
alt.Chart(train_df).mark_circle(size=6).encode(
    y=alt.Y("latitude:Q", scale=alt.Scale(zero=False)),
    x=alt.X("longitude:Q", scale=alt.Scale(zero=False)),
    color=alt.Color("neighbourhood_group:N")
)
```

Out[12]:



In [13]:

```
train_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 7768 entries, 27438 to 11381
Data columns (total 17 columns):
 #   Column                            Non-Null Count  Dtype  
---  ------                            --------------  -----  
 0   id                                7768 non-null   int64  
 1   name                              7768 non-null   object 
 2   host_id                           7768 non-null   int64  
 3   host_name                         7765 non-null   object 
 4   neighbourhood_group               7768 non-null   object 
 5   neighbourhood                     7768 non-null   object 
 6   latitude                          7768 non-null   float64
 7   longitude                         7768 non-null   float64
 8   room_type                         7768 non-null   object 
 9   price                             7768 non-null   int64  
 10  minimum_nights                    7768 non-null   int64  
 11  number_of_reviews                 7768 non-null   int64  
 12  last_review                       7768 non-null   object 
 13  reviews_per_month                 7768 non-null   float64
 14  calculated_host_listings_count    7768 non-null   int64  
 15  availability_365                  7768 non-null   int64  
 16  days_from_last_review             7768 non-null   int64  
dtypes: float64(3), int64(8), object(6)
memory usage: 1.3+ MB
```

In [14]:
```
# Correlation matrix
# We can see that number_of_reviews and availability_365 have the highest po
# and minimum_nights has the most negative correlation to reviews_per_month.
train_df.corr('spearman').style.format(precision=2).background_gradient()
```

```
/var/folders/zc/bzykxkmd5b59v_pk442hsdnm0000gn/T/ipykernel_21611/254500833
7.py:4: FutureWarning: The default value of numeric_only in DataFrame.corr
is deprecated. In a future version, it will default to False. Select only v
alid columns or specify the value of numeric_only to silence this warning.
  train_df.corr('spearman').style.format(precision=2).background_gradient()
```

Out[14]:

	id	host_id	latitude	longitude	price	minimum_nights
id	1.00	0.57	-0.01	0.07	-0.04	-0.16
host_id	0.57	1.00	0.03	0.11	-0.10	-0.18
latitude	-0.01	0.03	1.00	0.04	0.11	0.02
longitude	0.07	0.11	0.04	1.00	-0.41	-0.12
price	-0.04	-0.10	0.11	-0.41	1.00	0.12
minimum_nights	-0.16	-0.18	0.02	-0.12	0.12	1.00
number_of_reviews	-0.30	-0.12	-0.02	0.07	-0.02	-0.14
reviews_per_month	0.37	0.26	-0.03	0.10	-0.02	-0.26
calculated_host_listings_count	0.08	0.11	-0.02	0.08	-0.16	0.01
availability_365	0.08	0.12	-0.03	0.07	0.07	0.05
days_from_last_review	-0.36	-0.20	0.05	-0.08	-0.02	0.16

4. Feature engineering (Challenging)

rubric={reasoning}

Your tasks:

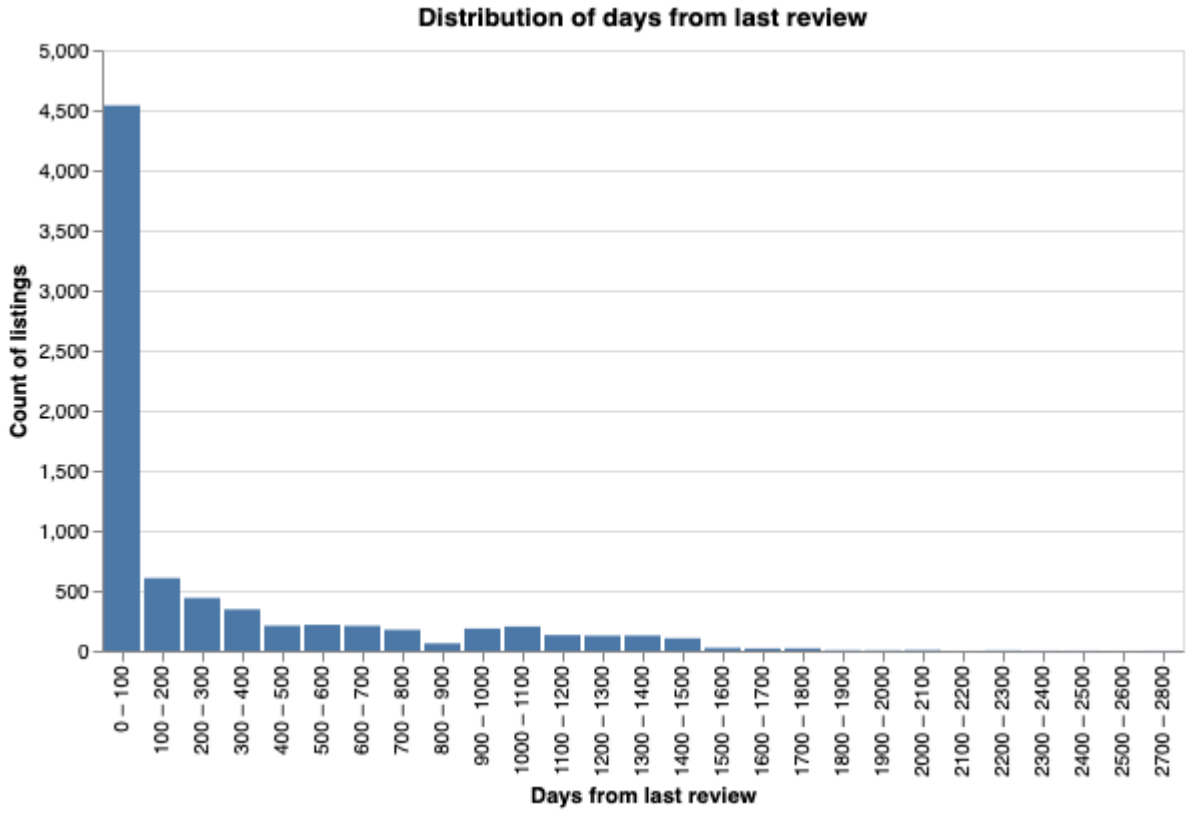
1. Carry out feature engineering. In other words, extract new features relevant for the problem and work with your new feature set in the following exercises. You may have to go back and forth between feature engineering and preprocessing.

Points: 0.5

In [15]:
```
from sklearn.preprocessing import KBinsDiscretizer

discretization_feats = ["latitude", "longitude"]

# We will add this code in the preprocessor section below.
# KBinsDiscretizer(n_bins=20, encode="onehot"), discretization_feats

# We will also add a new column called 'days_from_last_review'
# this feature calculates the timespan from the last review
# may be potentially helpful when predictive the review rate

# latest_day = max(pd.to_datetime(train_df["last_review"]))
# train_df["days_from_last_review"] = (
#      latest_day - (pd.to_datetime(train_df["last_review"]))
# ).dt.days

test_df["days_from_last_review"] = (
   latest_day - (pd.to_datetime(test_df["last_review"]))
).dt.days

alt.Chart(train_df, title="Distribution of days from last review").mark_bar(
    x=alt.X("days_from_last_review:N", title="Days from last review", bin=al
    y=alt.Y("count()", title="Count of listings")
)
```

```
/opt/miniconda3/lib/python3.10/site-packages/altair/utils/core.py:317: Futu
reWarning: iteritems is deprecated and will be removed in a future version.
Use .items instead.
  for col_name, dtype in df.dtypes.iteritems():
```

Out[15]:



In [16]:
```
train_df.head()
```

Out[16]:

	id	name	host_id	host_name	neighbourhood_group	neighbourho
27438	21630372	Beautiful and confortable room	109146538	Marisol	Queens	Jacks Heigh
13825	10435733	Cute Spacious Williamsburg Apt	5927655	Ann	Brooklyn	Williamsbu
473	166983	3 BR, Beautiful Brooklyn Duplex	795640	Jilly	Brooklyn	Carroll Garde
46103	35081477	West Village Gem! Like Paris	196890	Siobhan	Manhattan	West Villa
33450	26443376	Big 1BR in PRIME Bushwick! 2 blocks to L Train!	6293227	Nish	Brooklyn	Bushwi

5. Preprocessing and transformations

rubric={accuracy,reasoning}

Your tasks:

1. Identify different feature types and the transformations you would apply on each feature type.
2. Define a column transformer, if necessary.

Points: 4

In [17]:
```
X_train, y_train = (
    train_df.drop(columns=["reviews_per_month"]),
    train_df["reviews_per_month"],
)
X_test, y_test = (
    test_df.drop(columns=["reviews_per_month"]),
    test_df["reviews_per_month"],
)
```

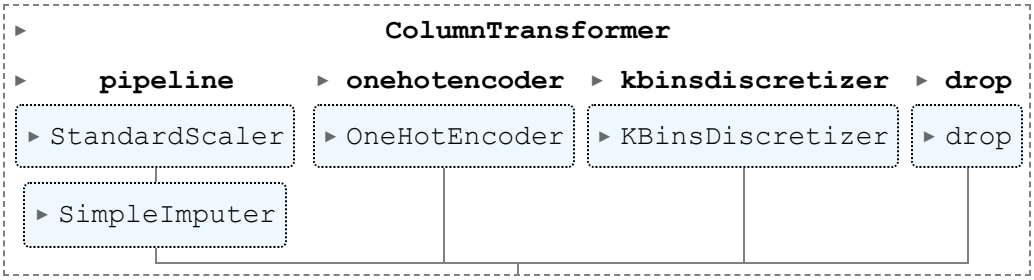
In [18]:
```
train_df.columns.to_list()
```

Out[18]:
```
['id',
 'name',
 'host_id',
 'host_name',
 'neighbourhood_group',
 'neighbourhood',
 'latitude',
 'longitude',
 'room_type',
 'price',
 'minimum_nights',
 'number_of_reviews',
 'last_review',
 'reviews_per_month',
 'calculated_host_listings_count',
 'availability_365',
 'days_from_last_review']
```

In [19]:
```
from sklearn.compose import ColumnTransformer, make_column_transformer
from sklearn.pipeline import Pipeline, make_pipeline
from sklearn.preprocessing import OneHotEncoder, OrdinalEncoder, StandardSca
from sklearn.impute import SimpleImputer

numeric_features = [
    "price",
    "minimum_nights",
    "number_of_reviews",
    "calculated_host_listings_count",
    "availability_365",
    "days_from_last_review"
]
categorical_features = [
    "neighbourhood_group",
    "neighbourhood",
```

```
    "room_type",
]
discretization_features = ["latitude", "longitude"]
drop = [
    "id",
    "last_review",
    "host_id",
    "host_name",
    "name"
]

preprocessor = make_column_transformer(
    (make_pipeline(StandardScaler(), SimpleImputer(strategy="most_frequent")
    (OneHotEncoder(handle_unknown="ignore", sparse=False), categorical_featu
    (KBinsDiscretizer(n_bins=30, encode="onehot"), discretization_feats),
    ("drop", drop),
)

preprocessor
```

Out[19]:



In [20]:

```
# This line nicely formats the feature names from `preprocessor.get_feature_
# so that we can more easily use them below
preprocessor.verbose_feature_names_out = False
# Create a dataframe with the transformed features and column names
preprocessor.fit(X_train)

# transformed data
X_train_transformed = preprocessor.transform(X_train)
ohe_features = (
    preprocessor.named_transformers_["onehotencoder"].get_feature_names_out(
)

discretization_features = (
    preprocessor.named_transformers_["kbinsdiscretizer"].get_feature_names_o
)

# Code to get all the feature names
feature_names = numeric_features + ohe_features + discretization_features

X_train_enc = pd.DataFrame(X_train_transformed, columns=feature_names)

# Show the transformed data
X_train_enc.head()
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoders.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version 1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave `sparse` to its default value.
  warnings.warn(
```

Out[20]:

	price	minimum_nights	number_of_reviews	calculated_host_listings_count	availabili
0	-0.378129	-0.318773	0.361461	-0.158247	-0.
1	-0.125215	-0.187950	-0.040219	-0.158247	-0.
2	0.955420	-0.057126	-0.484182	-0.158247	-0.
3	0.104708	-0.253361	-0.568746	-0.158247	-0
4	-0.240176	-0.187950	-0.420758	-0.158247	-0.

5 rows × 263 columns

In [21]:
```
X_train_enc.shape
```

Out[21]:
```
(7768, 263)
```

6. Baseline model

rubric={accuracy}

Your tasks:

1. Train a baseline model for your task and report its performance.

Points: 2

In [22]:
```
# Code from Varada, DSCI 573, UBC Master of Data Science course
from sklearn.model_selection import cross_validate
def mean_std_cross_val_scores(model, X_train, y_train, **kwargs):
    """
    Returns mean and std of cross validation

    Parameters
    ----------
    model :
        scikit-learn model
    X_train : numpy array or pandas DataFrame
        X in the training data
    y_train :
        y in the training data

    Returns
    ----------
        pandas Series with mean scores from cross_validation
    """
```

```
    scores = cross_validate(model, X_train, y_train, **kwargs)

    mean_scores = pd.DataFrame(scores).mean()
    std_scores = pd.DataFrame(scores).std()
    out_col = []

    for i in range(len(mean_scores)):
        out_col.append((f"%0.3f (+/- %0.3f)" % (mean_scores[i], std_scores[i

    return pd.Series(data=out_col, index=mean_scores.index)
```

In [23]:

```
from sklearn.dummy import DummyRegressor
from collections import defaultdict

cross_val_results = {}
cross_val_results["Dummy"] = mean_std_cross_val_scores(
    make_pipeline(preprocessor, DummyRegressor()), X_train, y_train, return_
)
pd.DataFrame(cross_val_results)
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

Out[23]:

	Dummy
fit_time	0.014 (+/- 0.002)
score_time	0.004 (+/- 0.000)
test_score	-0.001 (+/- 0.001)
train_score	0.000 (+/- 0.000)

7. Linear models

rubric={accuracy,reasoning}

Your tasks:

1. Try a linear model as a first real attempt.
2. Carry out hyperparameter tuning to explore different values for the regularization hyperparameter.
3. Report cross-validation scores along with standard deviation.
4. Summarize your results.

Points: 8

Type your answer here, replacing this text.

In [24]:
```
from sklearn.linear_model import LinearRegression, LogisticRegression, Ridge

cross_val_results["Ridge"] = mean_std_cross_val_scores(
    make_pipeline(preprocessor, Ridge()), X_train, y_train, return_train_scc
)
pd.DataFrame(cross_val_results)
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

Out[24]:

	Dummy	Ridge
fit_time	0.014 (+/- 0.002)	0.029 (+/- 0.011)
score_time	0.004 (+/- 0.000)	0.004 (+/- 0.000)
test_score	-0.001 (+/- 0.001)	0.395 (+/- 0.038)
train_score	0.000 (+/- 0.000)	0.439 (+/- 0.010)

8. Different models

rubric={accuracy,reasoning}

Your tasks:

1. Try out three other models aside from the linear model.
2. Summarize your results in terms of overfitting/underfitting and fit and score times. Can you beat the performance of the linear model?

Points: 10

2. The `RandomForest` model is the most overfitted model. It has a very high score in the training set but a large gap between the training score and the test score. `LGBMR`, and `CatBoost` as improved ensemble model are the less overfitted compared to `RandomForest`. Performance-wise, we would choose these two models over the `RandomForest` model. `SVR` and `Ridge` seem to be underfitted. The scores are not very good but the training score and the test score are roughly the same. Even though the ensemble models are overfitting they still have higher test scores than `SVR` and `Ridge`.

In [25]:
```
from sklearn.linear_model import LinearRegression, LogisticRegression, Ridge
from sklearn.ensemble import RandomForestRegressor
from sklearn.svm import SVR
from lightgbm.sklearn import LGBMRegressor
from catboost import CatBoostRegressor

cross_val_results["Random Forests"] = mean_std_cross_val_scores(
    make_pipeline(preprocessor, RandomForestRegressor(n_jobs=-1, random_stat
    X_train,
    y_train,
    return_train_score=True,
)

cross_val_results["SVR"] = mean_std_cross_val_scores(
    make_pipeline(preprocessor, SVR()),
    X_train,
    y_train,
```

```
    return_train_score=True,
)

cross_val_results["LGBMR"] = mean_std_cross_val_scores(
    make_pipeline(preprocessor, LGBMRegressor(random_state=573)),
    X_train,
    y_train,
    return_train_score=True,
)

cross_val_results["CatBoost"] = mean_std_cross_val_scores(
    make_pipeline(preprocessor, CatBoostRegressor(verbose=False,random_state
    X_train,
    y_train,
    return_train_score=True,
)

pd.DataFrame(cross_val_results)
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
```

```
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

Out[25]:

	Dummy	Ridge	Random Forests	SVR	LGBMR	CatBoost
fit_time	0.014 (+/- 0.002)	0.029 (+/- 0.011)	1.035 (+/- 0.311)	2.254 (+/- 0.417)	0.253 (+/- 0.003)	1.595 (+/- 0.062)
score_time	0.004 (+/- 0.000)	0.004 (+/- 0.000)	0.020 (+/- 0.001)	0.829 (+/- 0.111)	0.006 (+/- 0.000)	0.059 (+/- 0.001)
test_score	-0.001 (+/- 0.001)	0.395 (+/- 0.038)	0.607 (+/- 0.035)	0.472 (+/- 0.029)	0.614 (+/- 0.044)	0.628 (+/- 0.037)
train_score	0.000 (+/- 0.000)	0.439 (+/- 0.010)	0.945 (+/- 0.002)	0.535 (+/- 0.008)	0.811 (+/- 0.009)	0.815 (+/- 0.003)

9. Feature selection (Challenging)

rubric={reasoning}

Your tasks:

Make some attempts to select relevant features. You may try `RFECV`, forward selection or L1 regularization for this. Do the results improve with feature selection? Summarize your results. If you see improvements in the results, keep feature selection in your pipeline. If not, you may abandon it in the next exercises unless you think there are other benefits with using less features.

Points: 0.5

As shown in the cross validation results table below, `RFECV` did not improve the train or test score. Thus, we will not used this reduced model for the next step.

In [26]:
```
from sklearn.feature_selection import RFECV
cross_val_results["CatBoost RFE"] = mean_std_cross_val_scores(
    make_pipeline(
        preprocessor,
        RFECV(Ridge(), cv=10),
        CatBoostRegressor(verbose=False, random_state=573),
    ),
    X_train,
    y_train,
    return_train_score=True,
)
pd.DataFrame(cross_val_results)
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

Out[26]:

	Dummy	Ridge	Random Forests	SVR	LGBMR	CatBoost	CatBoost RFE
fit_time	0.014 (+/- 0.002)	0.029 (+/- 0.011)	1.035 (+/- 0.311)	2.254 (+/- 0.417)	0.253 (+/- 0.003)	1.595 (+/- 0.062)	17.229 (+/- 0.431)
score_time	0.004 (+/- 0.000)	0.004 (+/- 0.000)	0.020 (+/- 0.001)	0.829 (+/- 0.111)	0.006 (+/- 0.000)	0.059 (+/- 0.001)	0.012 (+/- 0.009)
test_score	-0.001 (+/- 0.001)	0.395 (+/- 0.038)	0.607 (+/- 0.035)	0.472 (+/- 0.029)	0.614 (+/- 0.044)	0.628 (+/- 0.037)	0.595 (+/- 0.034)
train_score	0.000 (+/- 0.000)	0.439 (+/- 0.010)	0.945 (+/- 0.002)	0.535 (+/- 0.008)	0.811 (+/- 0.009)	0.815 (+/- 0.003)	0.765 (+/- 0.008)

10. Hyperparameter optimization

rubric={accuracy,reasoning}

Your tasks:

Make some attempts to optimize hyperparameters for the models you've tried and summarize your results. In at least one case you should be optimizing multiple hyperparameters for a single model. You may use `sklearn`'s methods for hyperparameter optimization or fancier Bayesian optimization methods.

- GridSearchCV

- RandomizedSearchCV
- scikit-optimize

Points: 6

Type your answer here, replacing this text.

In [27]:
```
import numpy as np
param_grid = {
    "catboostregressor__learning_rate": np.arange(0.01,0.1,0.02),
    "catboostregressor__max_depth": np.arange(4,10,1)
}

from sklearn.model_selection import GridSearchCV
pipe_cat = make_pipeline(preprocessor, CatBoostRegressor(verbose=False, rand

grid_search = GridSearchCV(
    pipe_cat,
    param_grid,
    cv=5,
    n_jobs=-1,
    return_train_score=True,
)
grid_search.fit(X_train, y_train)
grid_search.best_score_
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
```

```
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
```

```
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
```

```
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
```

```
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
```

```
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
```

```
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
```

```
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
```

```
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
```

```
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
```

```
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
```

```
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

Out[27]:
```
0.6328814781300852
```

In [28]:
```
grid_search.best_params_
```

Out[28]:
```
{'catboostregressor__learning_rate': 0.03, 'catboostregressor__max_depth':
8}
```

11. Interpretation and feature importances

rubric={accuracy,reasoning}

Your tasks:

1. Use the methods we saw in class (e.g., `eli5`, `shap`) (or any other methods of your choice) to examine the most important features of one of the non-linear models.
2. Summarize your observations.

Points: 8

In [29]:
```
# Generate a table showing feature importances.
feat_importance = grid_search.best_estimator_.named_steps["catboostregressor
```

```
# Code to get all the feature names
feature_names = numeric_features + ohe_features + discretization_features

pd.DataFrame({"Feature": feature_names, "Feature Importance": feat_importanc
    by="Feature Importance", ascending=False
)
```

Out[29]:

	Feature	Feature Importance
5	days_from_last_review	31.728355
2	number_of_reviews	22.841847
1	minimum_nights	14.138551
4	availability_365	9.038644
0	price	4.251806
...	...	...
118	neighbourhood_Melrose	0.000000
197	neighbourhood_Woodhaven	0.000000
193	neighbourhood_Whitestone	0.000000
121	neighbourhood_Midwood	0.000000
131	neighbourhood_New Springville	0.000000

263 rows × 2 columns

In [30]:

```
import eli5

pipe_cat = make_pipeline(preprocessor, CatBoostRegressor(verbose=False,
                                                         max_depth=9,
                                                         learning_rate=0.03,
                                                         random_state=573))

pipe_cat.fit(X_train, y_train)
eli5.explain_weights(pipe_cat.named_steps["catboostregressor"], feature_name
```

```
/opt/miniconda3/lib/python3.10/site-packages/sklearn/preprocessing/_encoder
s.py:808: FutureWarning: `sparse` was renamed to `sparse_output` in version
1.2 and will be removed in 1.4. `sparse_output` is ignored unless you leave
`sparse` to its default value.
  warnings.warn(
```

Out[30]:

Weight	Feature
0.2925	days_from_last_review
0.2184	number_of_reviews
0.1407	minimum_nights
0.0953	availability_365
0.0527	price
0.0309	calculated_host_listings_count
0.0140	room_type_Private room
0.0109	room_type_Entire home/apt
0.0082	neighbourhood_group_Brooklyn
0.0080	neighbourhood_Theater District
0.0075	longitude_5.0
0.0069	neighbourhood_group_Queens
0.0064	longitude_29.0
0.0060	neighbourhood_group_Manhattan
0.0053	latitude_21.0
0.0042	latitude_22.0
0.0041	neighbourhood_East Elmhurst
0.0036	longitude_24.0
0.0034	neighbourhood_Hell's Kitchen
0.0029	latitude_23.0
	... 243 more ...

We examined the feature importances both from the `.feature_importances_` attribute and `eli5` library. Both tools yield roughly the same rank of feature importance and similar weights of the importance.

We can see that the feature engineering of `days_from_last_review` is a successful attempt that contributes to the prediction the most. It is because this engineered feature captures the timespan from the last review. If there has been a long time since an AirBnB receives a review, it is likely the review per month will be lower.

12. Results on the test set

rubric={accuracy,reasoning}

Your tasks:

1. Try your best performing model on the test data and report test scores.
2. Do the test scores agree with the validation scores from before? To what extent do you trust your results? Do you think you've had issues with optimization bias?
3. Take one or two test predictions and explain them with SHAP force plots.

Points: 6

1. The code below shows the best-performing model's score on the test set.
2. Yes, the test score is slightly better than the cross-validation score. The test set is partitioned relatively large; thus the result could be trusted. I do not think the

optimization bias is influencing the results as we are using a reasonably large search range and the large test set's score is better than the validation score.

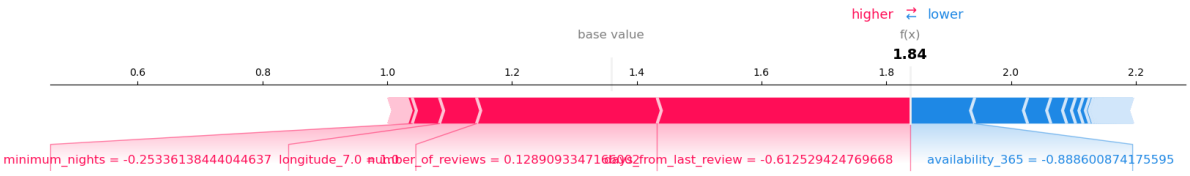
In [31]:
```
grid_search.score(X_test, y_test)
```

Out[31]:
```
0.6516834151491739
```

In [32]:
```
import shap
shap.initjs()
pipe_cat = make_pipeline(preprocessor, CatBoostRegressor(verbose=False,
                                                         max_depth=grid_sear
                                                         learning_rate=grid_
                                                         random_state=573))
pipe_cat.fit(X_train, y_train)

X_test_enc = pd.DataFrame(
    data=preprocessor.transform(X_test),
    columns=feature_names,
    index=X_test.index,
)

cat_explainer = shap.TreeExplainer(pipe_cat["catboostregressor"])
test_cat_shap_values = cat_explainer.shap_values(X_test_enc)
```

```
`sparse` was renamed to `sparse_output` in version 1.2 and will be removed
in 1.4. `sparse_output` is ignored unless you leave `sparse` to its default
value.
iteritems is deprecated and will be removed in a future version. Use .items
instead.
```

In [33]:
```
shap.force_plot(
    cat_explainer.expected_value,
    test_cat_shap_values[1, :],
    X_test_enc.iloc[1, :],
    matplotlib = True
)
```



In [34]:
```
# actual target
y_test.iloc[1]
```

Out[34]:
```
1.22
```

13. Summary of results

rubric={reasoning}

Imagine that you want to present the summary of these results to your boss and co-workers.

Your tasks:

1. Create a table summarizing important results.
2. Write concluding remarks.
3. Discuss other ideas that you did not try but could potentially improve the performance/interpretability .
4. Report your final test score along with the metric you used at the top of this notebook.

Points: 8

1.

Model	Description	Validation Score	Complexity	Important Feature
DummyRegressor	Baseline Model	-0.01	Low	N/A
Ridge	Linear Model with L2 Regularization	0.395	Low	N/A
SVR	Support Vector Machines Regressor	0.606	Medium	N/A
Random Forest Regressor	Tree Based Ensemble Model with Randomness	0.472	Medium	N/A
Catboost Regressor	Tree Based Ensemble Boosting Model	0.614	High	N/A
Catboost Regressor (Tuned)	Tree Based Ensemble Boosting Model with Hyperparameter Optimization	0.628	High	`days_frpm_last_review`
Catboost Regressor RFE	Catboosting Model with Feature Selection	0.595	Medium	N/A

In [35]:
```
pd.DataFrame(cross_val_results)
```

Out[35]:

	Dummy	Ridge	Random Forests	SVR	LGBMR	CatBoost	CatBoost RFE
fit_time	0.014 (+/- 0.002)	0.029 (+/- 0.011)	1.035 (+/- 0.311)	2.254 (+/- 0.417)	0.253 (+/- 0.003)	1.595 (+/- 0.062)	17.229 (+/- 0.431)
score_time	0.004 (+/- 0.000)	0.004 (+/- 0.000)	0.020 (+/- 0.001)	0.829 (+/- 0.111)	0.006 (+/- 0.000)	0.059 (+/- 0.001)	0.012 (+/- 0.009)
test_score	-0.001 (+/- 0.001)	0.395 (+/- 0.038)	0.607 (+/- 0.035)	0.472 (+/- 0.029)	0.614 (+/- 0.044)	0.628 (+/- 0.037)	0.595 (+/- 0.034)
train_score	0.000 (+/- 0.000)	0.439 (+/- 0.010)	0.945 (+/- 0.002)	0.535 (+/- 0.008)	0.811 (+/- 0.009)	0.815 (+/- 0.003)	0.765 (+/- 0.008)

2. We gathered the AirBnB data and aim to construct a predictive model that estimates the number of reviews per month. We preprocessed the data using `OneHotEncoder` and `StandardScaler` on categorical and numerical data. Additionally, we engineered a potentially helpful feature that indicates the timespan since the last review. We achieve the best score of () using the processed data and `CatBoostRegressor` with a learning rate of 0.3 and a maximum tree depth of 8. While the high-performing boosting model is complex to interpret, we can cooperate `SHAP` as the tool to visualize and explain the decision of the model upon new cases.

3. The potential of some simpler models has not been fully explored. It is still possible that `SVR` and `Ridge` can achieve a higher score after hyperparameter optimization. In particular, if simpler models like `Ridge` can be tuned up to similar performance as `CatBoostRegressor`, we should choose `Ridge` over `CatBoostRegressor` for improved interpretability.

4. The report is placed at the top of the notebook

14. Creating a data analysis pipeline (Challenging)

rubric={reasoning}

Your tasks:

- In 522 you learned how build a reproducible data analysis pipeline. Convert this notebook into scripts and create a reproducible data analysis pipeline with appropriate documentation. Submit your project folder in addition to this notebook on GitHub and briefly comment on your organization in the text box below.

Points: 2

The scripts can be found in the same repository of this notebook.

15. Your takeaway from the course (Challenging)

rubric={reasoning}

Your tasks:

What is your biggest takeaway from this course?

Points: 0.25

As a continuation of the DSCI 571 course, we learned how to carry out a supervised machine learning project from EDA, preprocessing, feature engineering and transformation, model building, model selection, feature selection, and presentation and interpretation of results for both classification and regression data sets. I personally am amazed by the number of different models out there and especially the ensemble model that can take the good parts of different models into one.

Restart, run all and export a PDF before submitting

Before submitting, don't forget to run all cells in your notebook to make sure there are no errors and so that the TAs can see your plots on Gradescope. You can do this by clicking the ►► button or going to `Kernel -> Restart Kernel and Run All Cells...` in the menu. This is not only important for MDS, but a good habit you should get into before ever committing a notebook to GitHub, so that your collaborators can run it from top to bottom without issues.

After running all the cells, export a PDF of the notebook (preferably the WebPDF export) and upload this PDF together with the ipynb file to Gradescope (you can select two files when uploading to Gradescope)

Help us improve the labs

The MDS program is continually looking to improve our courses, including lab questions and content. The following optional questions will not affect your grade in any way nor will they be used for anything other than program improvement:

1. Approximately how many hours did you spend working or thinking about this assignment (including lab time)?

Ans:

2. Do you have any feedback on the lab you be willing to share? For example, any part or question that you particularly liked or disliked?

Ans: