Python Projects

Anime ALS recommendation and statistical analysis

NASDAQ missing dates

NYC flights EDA

London houses prices

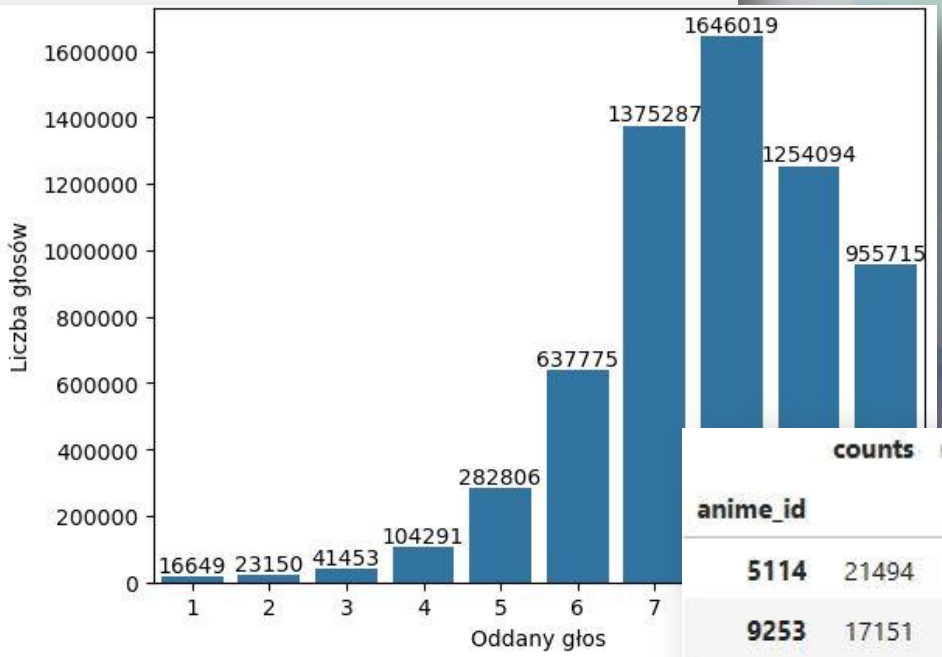
Anime ALS recommendation and statistical analysis

```
[58]: mostVotedAnime = rating[rating['rating']>-1]

[59]: mostVotedAnime.groupby('user_id').count().sort_values('anime_id', ascending=False)
```

[59]:

user_id	anime_id	rating
42635	3747	3747
53698	2905	2905
57620	2689	2689
59643	2632	2632
51693	2621	2621
...	...	...
66747	1	1
35463	1	1
55557	1	1
55563	1	1
22208	1	1

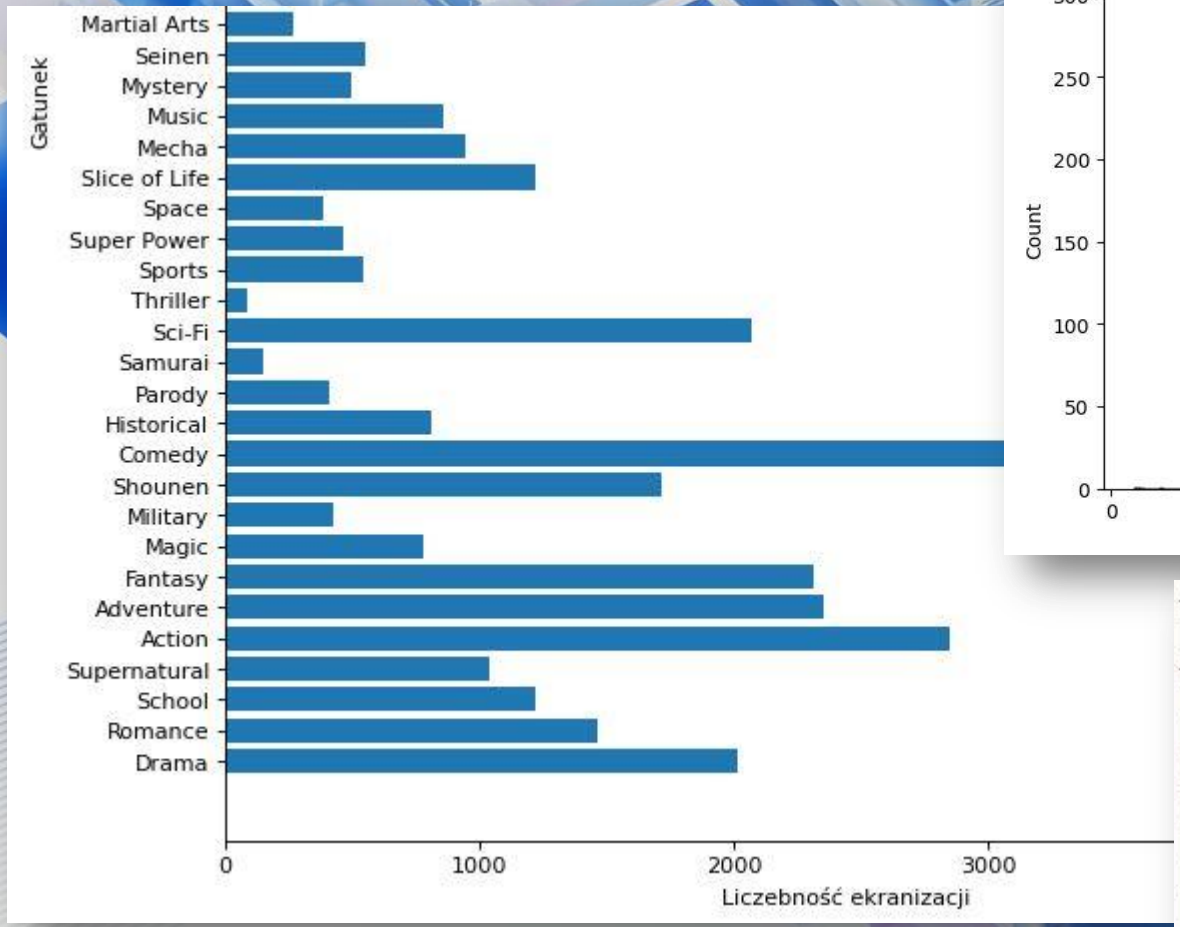


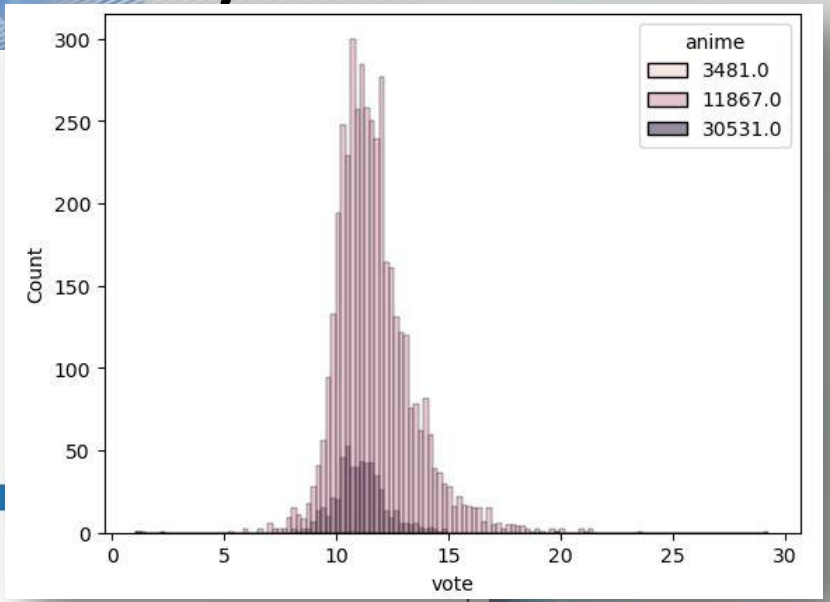

```
[98]: mostVotedAnime['rating'].mean()

[98]: 7.808496570825244
```

anime_id	counts	ratingSum	ratingMean
5114	21494	200383	9.322741
9253	17151	158841	9.261326
4181	15518	142227	9.165292
2904	21124	191380	9.059837
30276	11323	101224	8.939680

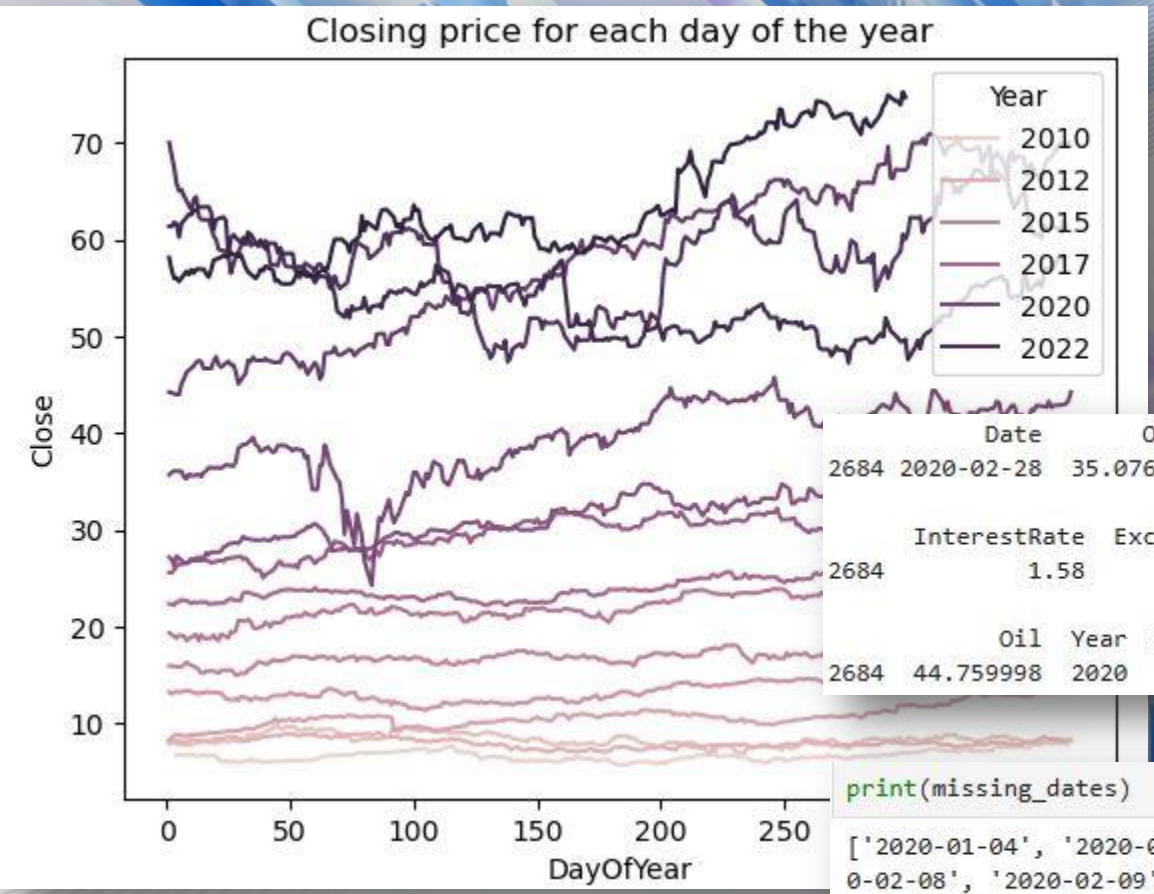
Anime ALS recommendation and statistical analysis





user_id	anime_id	rating	prediction
148	225	7	5.51359
148	265	7	8.145976
148	356	6	5.727167
148	384	8	5.783889
148	1575	8	8.378609
148	2904	9	8.538317
148	4224	8	7.3941135
148	5114	8	8.998776
148	7674	9	8.064965
148	9130	7	7.903865

NASDAQ missing dates



	Date	Open	High	Low	Close	Volume	\
2684	2020-02-28	35.076668	35.150002	33.490002	34.183334	6843000.0	

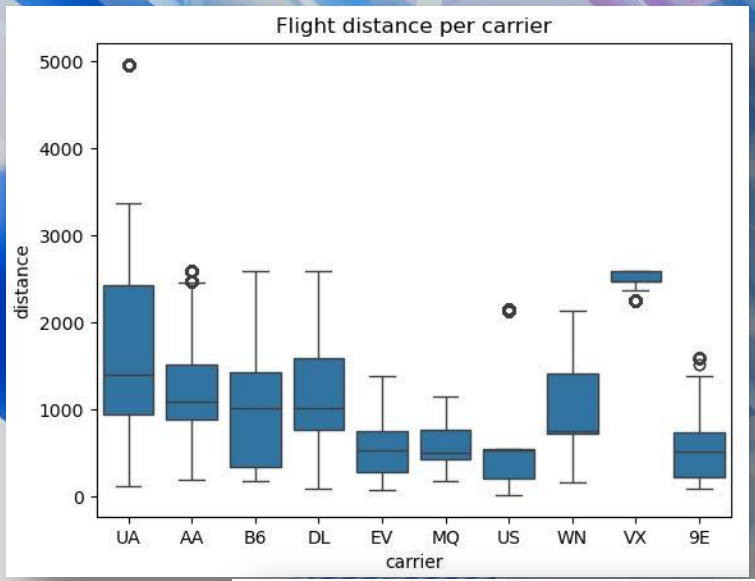
	InterestRate	ExchangeRate	VIX	TEDSpread	EFFR	Gold	\
2684	1.58	1.1001	40.11	0.21	1.58	1564.099976	

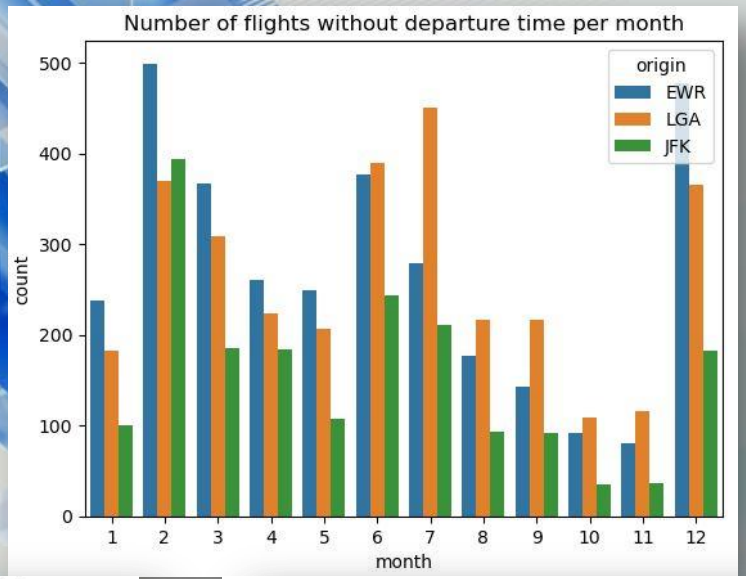
	Oil	Year	DayOfYear
2684	44.759998	2020	59

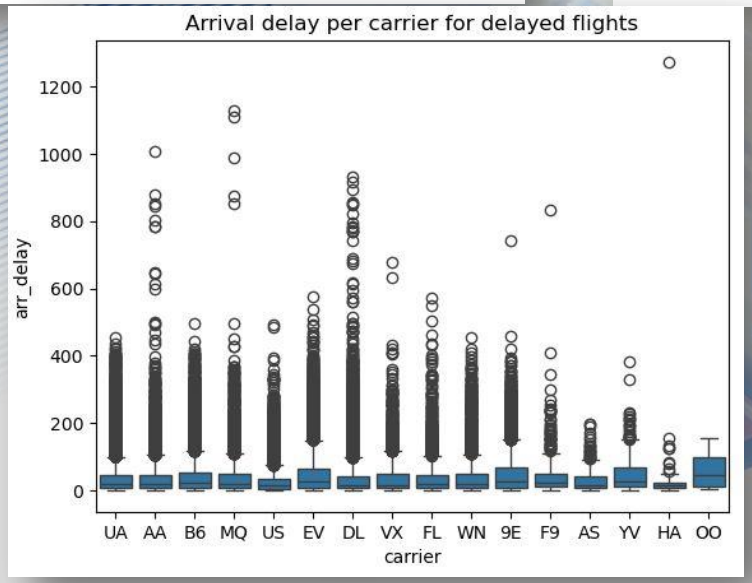
```
print(missing_dates)
```

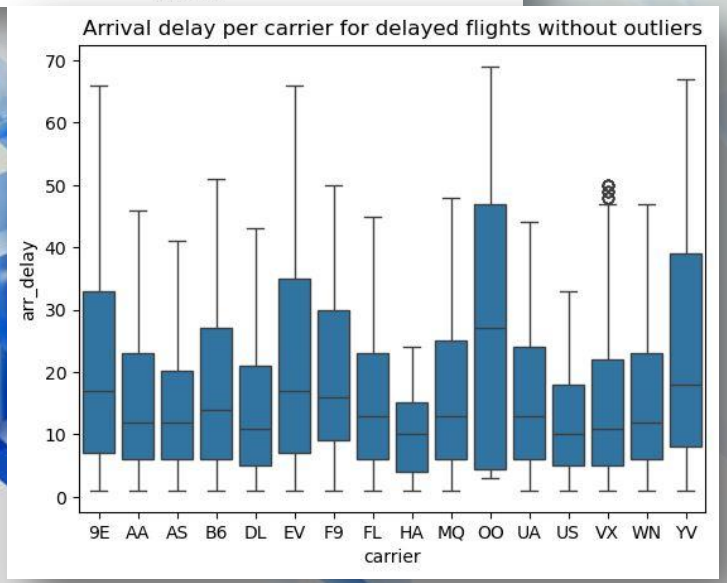
```
['2020-01-04', '2020-01-05', '2020-01-11', '2020-01-12', '2020-01-18', '2
0-02-08', '2020-02-09', '2020-02-15', '2020-02-16', '2020-02-22', '2020-0
14', '2020-03-15', '2020-03-21', '2020-03-22', '2020-03-28', '2020-03-29'
'2020-04-18', '2020-04-19', '2020-04-25', '2020-04-26', '2020-05-02', '20
-05-17', '2020-05-23', '2020-05-24', '2020-05-30', '2020-05-31', '2020-06
0', '2020-06-21', '2020-06-27', '2020-06-28', '2020-07-04', '2020-07-05',
'2020-07-25', '2020-07-26', '2020-08-02', '2020-08-08', '2020-08-09', '20
-08-29', '2020-08-30', '2020-09-05', '2020-09-06', '2020-09-12', '2020-09
```

NYC flights EDA

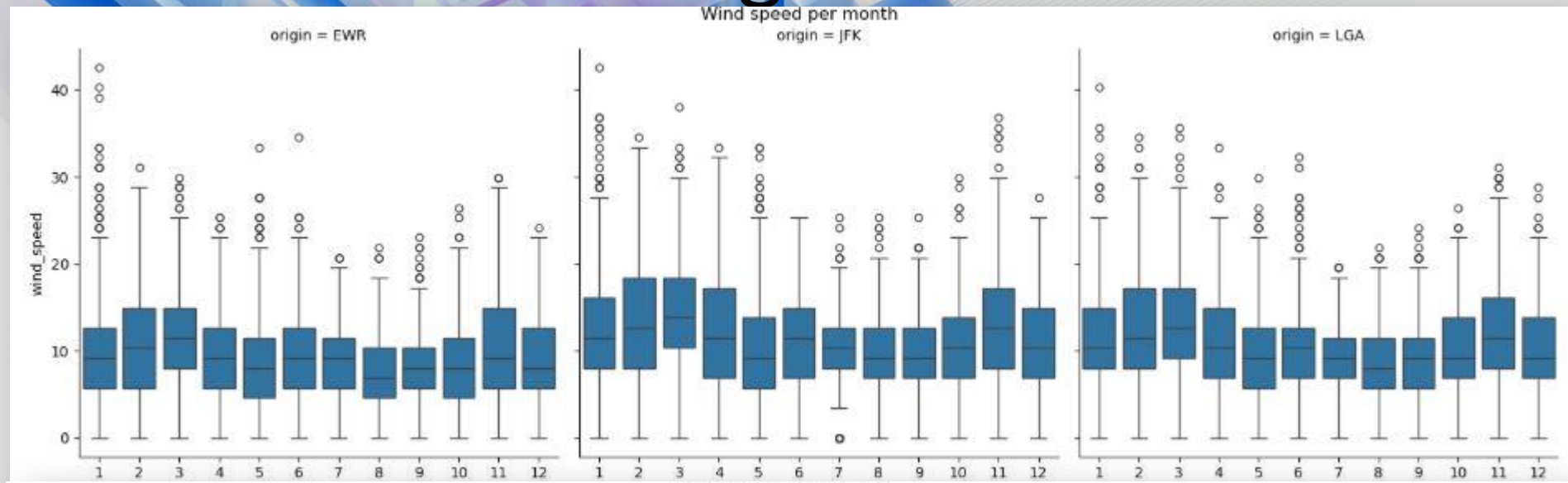


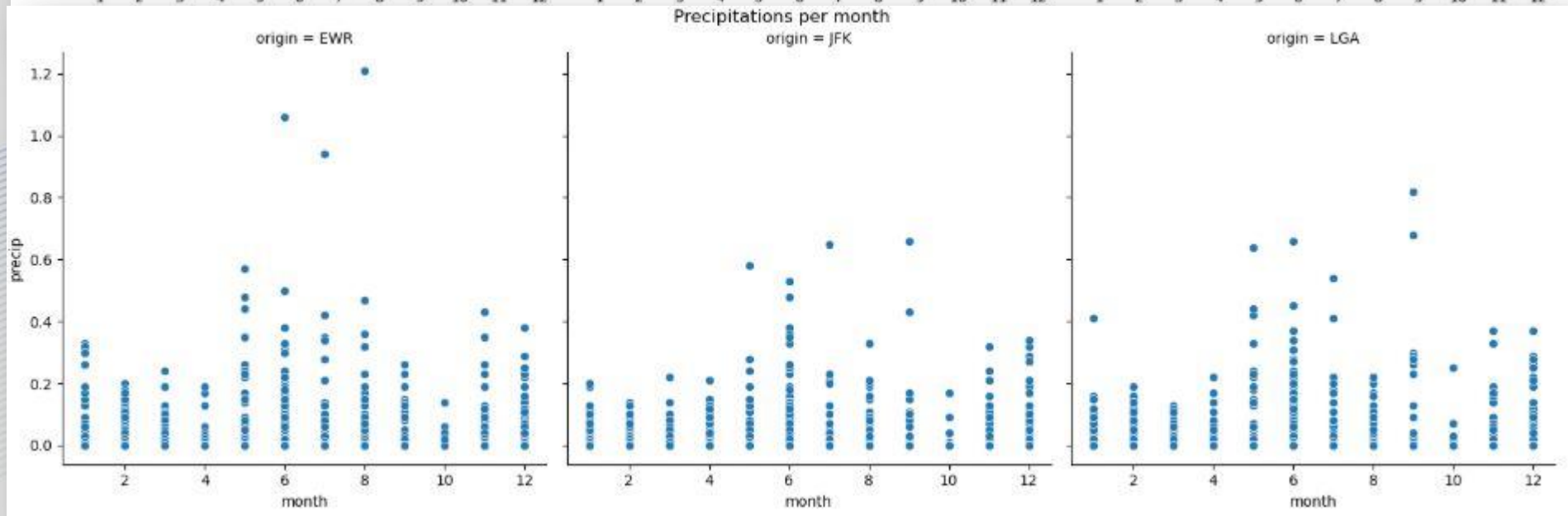




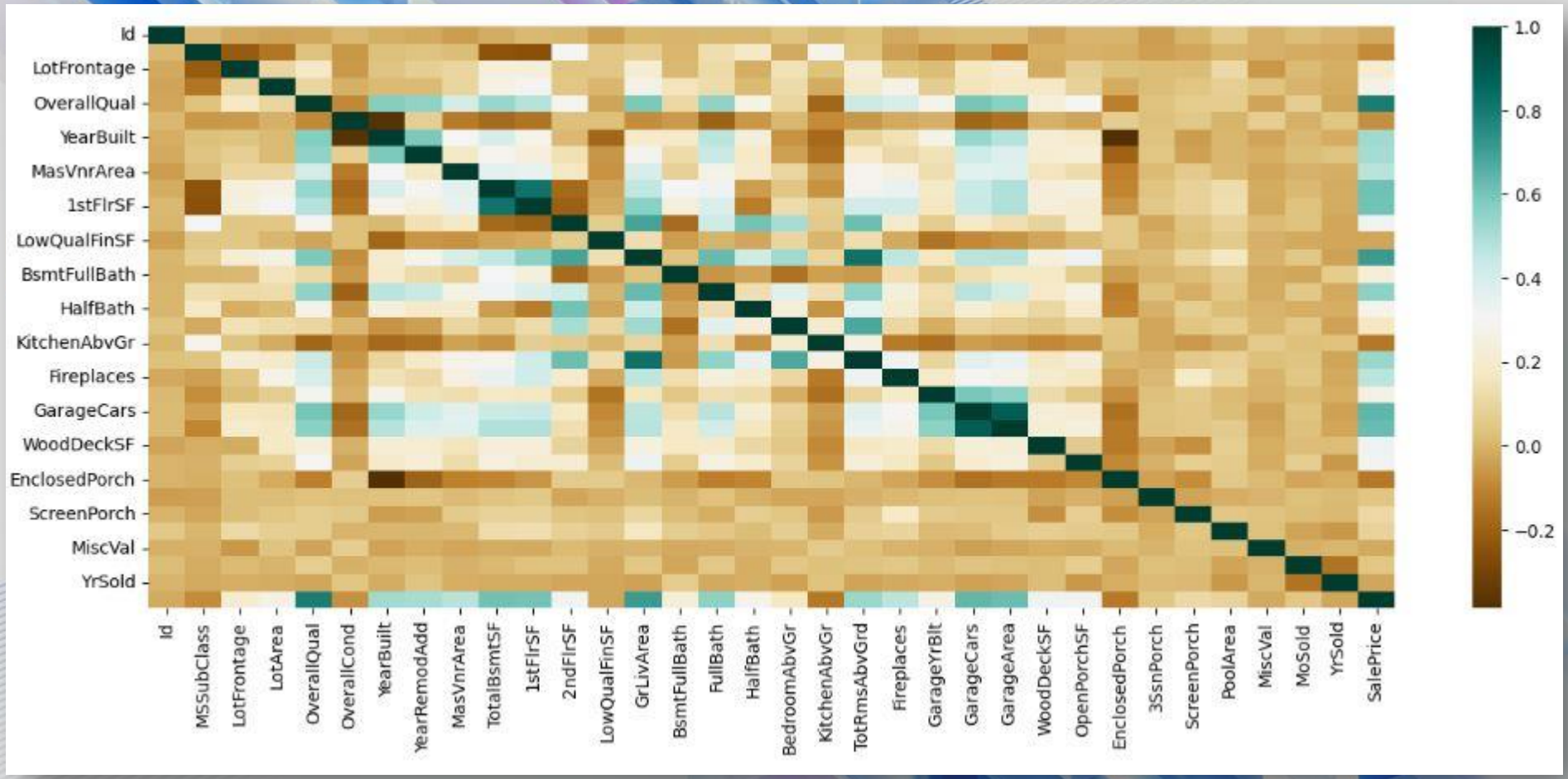


NYC flights EDA





London houses prices



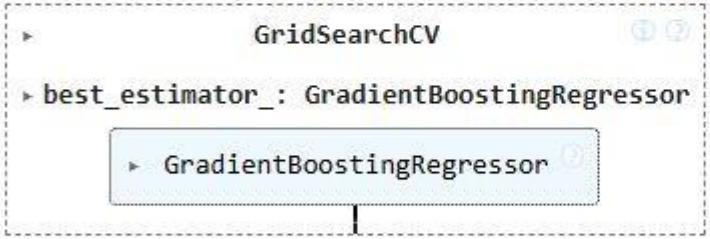
London houses prices

```
param_grid_gbr={
    "learning_rate":[0.05, 0.1, 0.2],
    "n_estimators" : [40, 100, 250],
    "max_depth" : [3, 5, 8]
}
```

```
gbr2 = GradientBoostingRegressor(criterion='squared_error', random_stat
```

```
CV_gbr = GridSearchCV(estimator=gbr2, param_grid=param_grid_gbr, cv=5)
```

```
CV_gbr.fit(X_train_best_num, y_train)
```




```
CV_gbr.best_params_
```

```
{'learning_rate': 0.1, 'max_depth': 5, 'n_estimators': 40}
```

```
CV_gbr.best_score_
```

```
0.8250241125102503
```

```
lasso_pred = lasso.predict(X_valid_num)
```

```
print(root_mean_squared_error(lasso_pred, y_valid))
```

```
39254.88822704823
```

```
importance_df = pd.DataFrame({
    'Feature': X_train_num.columns,
    'Importance': np.abs(lasso.coef_)
}).sort_values(by='Importance', ascending=False)
```

```
importance_df.head(10)
```

	Feature	Importance
18	KitchenAbvGr	17878.939810
22	GarageCars	16658.577801
4	OverallQual	16086.699326
14	BsmtFullBath	14969.486666
17	BedroomAbvGr	11215.729890
19	TotRmsAbvGrd	5526.306931
5	OverallCond	5370.636825