

Вычисления для домашней работы по Финансовой математике

Питун-З Деменчук Георгий

```
In [1]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import datetime
```

```
In [2]: custom_date_parser = lambda x: datetime.datetime.strptime(x, "%d-%m-%Y")
data = pd.read_csv("./data.csv", delimiter=";", parse_dates=["DATE"], date_parser=custom_date_parser)
data
```

Out[2]:

	DATE	Aligent	Centene	DEXcom	Carmax	Amazon	Voniter	Equinix	Laboratory corporation of american holding	Baxter International	Lowe's companies	S&P
0	2021-09-24	172.390	62.82	561.17	144.45	3426.98	34.890	834.62	296.295	80.855	207.90	4455.48
1	2021-09-27	166.530	63.82	548.54	145.24	3406.22	34.860	807.66	290.990	80.570	210.63	4443.11
2	2021-09-28	159.820	63.54	536.92	143.01	3317.28	34.450	799.48	283.040	80.720	205.43	4352.63
3	2021-09-29	160.090	63.26	544.83	146.50	3301.40	34.420	801.38	286.940	81.810	207.13	4359.46
4	2021-09-30	157.470	62.30	546.94	127.96	3285.40	33.620	790.12	281.450	80.420	202.71	4307.54
5	2021-10-01	155.940	63.05	549.61	127.17	3282.15	34.040	789.18	279.010	81.290	203.69	4357.04
6	2021-10-04	152.360	62.53	526.29	122.50	3191.12	33.720	785.00	275.000	80.370	204.53	4300.46
7	2021-10-05	152.995	62.72	540.49	126.26	3219.03	33.820	760.73	276.100	80.850	204.22	4345.72
8	2021-10-06	153.670	63.04	541.81	129.31	3263.74	33.250	761.86	275.440	80.850	203.96	4363.55
9	2021-10-07	155.250	63.98	539.05	131.33	3300.49	33.675	766.16	279.230	80.580	207.90	4399.76
10	2021-10-08	152.470	64.33	529.61	130.53	3288.83	33.780	750.51	278.220	80.210	206.09	4391.34
11	2021-10-11	150.010	63.09	528.26	129.29	3245.89	32.550	750.34	274.230	78.910	208.81	4361.19
12	2021-10-12	147.940	63.06	538.54	131.52	3247.65	32.795	759.48	273.940	78.710	209.82	4350.65
13	2021-10-13	150.120	61.93	548.35	133.36	3283.69	33.280	763.87	273.380	78.780	212.06	4363.80
14	2021-10-14	152.580	63.23	554.45	135.68	3299.23	33.950	769.91	275.700	79.880	216.44	4438.26
15	2021-10-15	153.350	65.36	543.76	136.71	3409.56	34.120	781.23	271.660	79.700	219.11	4471.37
16	2021-10-18	154.770	64.01	540.95	137.55	3446.70	34.050	766.95	274.260	78.180	221.73	4486.46
17	2021-10-19	155.570	64.73	549.79	138.45	3443.25	34.220	779.37	276.090	79.690	222.90	4519.63
18	2021-10-20	157.560	68.10	556.15	138.56	3411.61	34.320	794.17	278.710	81.245	224.96	4536.19
19	2021-10-21	156.950	69.01	568.31	141.92	3435.99	34.440	810.37	279.840	81.470	227.46	4549.78
20	2021-10-22	158.030	69.21	566.57	143.42	3335.25	34.335	821.48	282.150	82.190	228.72	4544.90
21	2021-10-25	158.300	67.89	568.11	140.00	3322.64	34.195	823.08	281.090	81.370	231.73	4566.48

In [3]: data.dtypes

Out[3]:

DATE	datetime64[ns]
Aligent	float64
Centene	float64
DEXcom	float64
Carmax	float64
Amazon	float64
Voniter	float64
Equinix	float64
Laboratory corporation of american holding	float64
Baxter International	float64
Lowe's companies	float64
S&P	float64
dtype:	object

In [4]: all_items_list = ("Aligent", "Centene", "DEXcom", "Carmax", "Amazon", "Voniter", "Equinix", "Laboratory corpor

In [5]:

```
def show_plot(df, plot_name):
    plt.figure(figsize=(15,15))
    dates = df["DATE"]
    for item in all_items_list:
        plt.plot(dates, df[item], label=item)

    plt.legend()
    plt.savefig(f'./img/{plot_name}.png')
    plt.show()
```

График по-умолчанию

In [6]: show_plot(data, "result_standard")

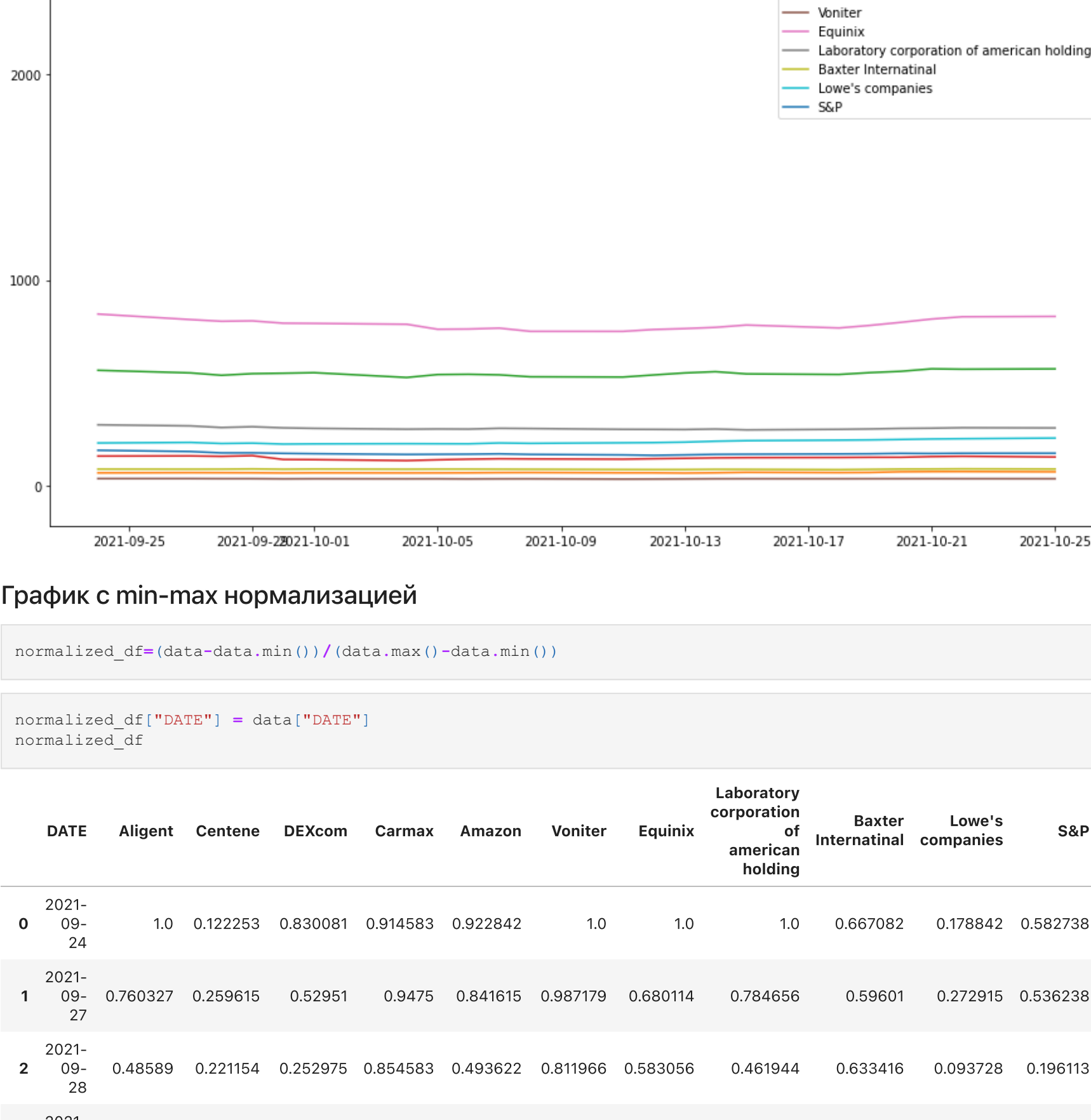


График с min-max нормализацией

In [7]: normalized_df=(data-data.min())/(data.max()-data.min())

In [8]:

```
normalized_df["DATE"] = data["DATE"]
normalized_df
```

Out[8]:

	DATE	Aligent	Centene	DEXcom	Carmax	Amazon	Voniter	Equinix	Laboratory corporation of american holding	Baxter International	Lowe's companies	S&P
0	2021-09-24	1.0	0.122253	0.830081	0.914583	0.922842	1.0	1.0	1.0	0.667082	0.178842	0.582738
1	2021-09-27	0.760327	0.259615	0.52951	0.9475	0.841615	0.987179	0.680114	0.784656	0.59601	0.272915	0.536238
2	2021-09-28	0.48589	0.221154	0.252975	0.854583	0.493622	0.811966	0.583056	0.461944	0.633416	0.093728	0.196113
3	2021-09-29	0.496933	0.182692	0.441218	1.0	0.431489	0.799145	0.6056	0.620256	0.905237	0.152309	0.221788
4	2021-09-30	0.389775	0.050824	0.491433	0.2275	0.368886	0.457265	0.471998	0.397402	0.558603	0.0	0.026615
5	2021-10-01	0.327198	0.153846	0.554974	0.194583	0.35617	0.636752	0.460845	0.298356	0.775561	0.03377	0.212691
6	2021-10-04	0.180777	0.082418	0.0	0.0	0.0	0.5	0.411248	0.135579	0.546135	0.062715	0.0
7	2021-10-05	0.206748	0.108516	0.337934	0.156667	0.109203	0.542735	0.12328	0.180231	0.665835	0.052033	0.170138
8	2021-10-06	0.234356	0.152473	0.369348	0.28375	0.284138	0.299145	0.136687	0.15344	0.665835	0.043074	0.237163
9	2021-10-07	0.298978	0.281593	0.303665	0.367917	0.427929	0.480769	0.187708	0.307286	0.598504	0.178842	0.37328
10	2021-10-08	0.185276	0.32967	0.07901	0.334583	0.382307	0.525641	0.002017	0.266288	0.506234	0.116471	0.341628
11	2021-10-11	0.084663	0.159341	0.046882	0.282917	0.214297	0.0	0.0	0.104323	0.182045	0.2102	0.228291
12	2021-10-12	0.0	0.15522	0.291528	0.375833	0.221183	0.104701	0.108448	0.092551	0.13217	0.245003	0.18867
13	2021-10-13	0.089162	0.0	0.524988	0.4525	0.362196	0.311966	0.160536	0.069819	0.149626	0.322192	0.238102
14	2021-10-14	0.189775	0.178571	0.670157	0.549167	0.422999	0.598291	0.232202	0.163994	0.42394	0.473122	0.518006
15	2021-10-15	0.221268	0.471154	0.415754	0.592083	0.854683	0.67094	0.366516	0.0	0.379052	0.565127	0.64247
16	2021-10-18	0.279346	0.285714	0.348881	0.627083	1.0	0.641026	0.197081	0.105541	0.0	0.65541	0.699196
17	2021-10-19	0.312065	0.384615	0.559257	0.664583	0.986501	0.713675	0.344447	0.179825	0.376559	0.695727	0.823885
18	2021-10-20	0.393456	0.847527	0.710614	0.669167	0.862704	0.75641	0.520052	0.286178	0.764339	0.766713	0.886136
19	2021-10-21	0.368507	0.972527	1.0	0.809167	0.958095	0.807692	0.712269	0.332048	0.820449	0.85286	0.937223
20	2021-10-22	0.412679	1.0	0.958591	0.871667	0.563933	0.762821	0.844091	0.425817	1.0	0.896278	0.918878
21	2021-10-25	0.423722	0.818681	0.99524	0.729167	0.514594	0.702991	0.863075	0.382789	0.795511	1.0	1.0

In [9]: show_plot(normalized_df, "normalized_df")

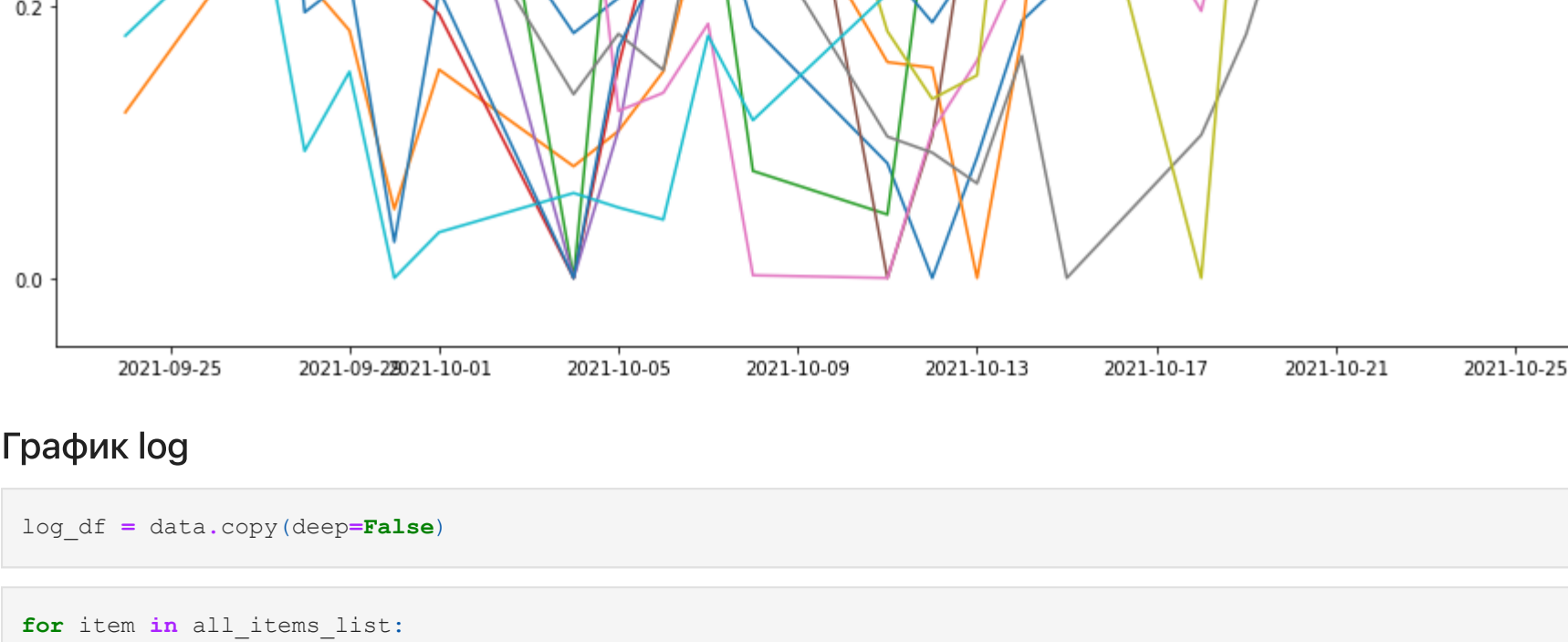


График log

In [10]: log_df = data.copy(deep=False)

In [11]:

```
for item in all_items_list:
    log_df[item] = np.log(log_df[item])
log_df
```

2021-09-27

5.115175

4.156067

6.307260

4.978388

8.133358

3.551340

6.694141

5.673289

4.389126

5.350103

8.399110

2021-09-28

5.074048

4.151670

6.285849

4.962915

8.106900

3.539599

6.683962

5.645588

4.390986

5.325105

8.378536

2021-09-29

5.075736

4.147253

6.300474

4.987025

8.102102

3.538638

6.686335

5.659273

4.404399

5.333347

8.380103

2021-09-30

5.059235

4.131961

6.304339

4.851718

8.097244

3.515121

6.672185

5.639955

4.387263

5.311776

8.368122

2021-10-01

5.049471

4.143928

6.309209

4.845525

8.096254

3.527536

6.670994

5.631248

4.398023

5.316599

8.379548

2021-10-04

5.026246

4.135646

6.265852

4.808111

8.068127

3.518091

6.665684

5.616771

4.386641

5.320715

8.366477

2021-10-05

5.030405

4.138680

6.292476

4.838343

8.076835

3.521052

6.634278

5.620763

4.392596

5.319198

8.376947

2021-10-06

5.034807

4.143769

6.294915

4.862213

8.090629

3.504055

6.635763

5.618370

4.392596

5.317924

8.381041

2021-10-07

5.045037

4.158571

6.289808

4.877713

8.101826

3.516756

6.641391

5.632036

4.389250

5.337057

8.389305

2021-10-08

5.026968

4.164026

6.272141

4.871603

8.098287

3.519869

6.620753

5.628412

4.384648

5.328313

8.387390

2021-10-11

5.010702

4.144562

6.265898

4.862058

8.085145

3.482777

6.620526

5.613967

4.368308

5.341425

8.380500

2021-10-12

4.996807

4.144087

6.288862

4.879159

8.085687

3.490276

6.632634

5.612909

4.365770

5.346250

8.378081

2021-10-13

5.011435

4.126005

6.306914

4.893052

8.096723

3.504957

6.638398

5.610863

4.366659

5.356869

8.381099

2021-10-14

5.027689

4.146779

6.317977

4.910299

8.101444

3.524889

6.646274

5.619313

4.380526

5.377313

8.398018

2021-10-15

5.032723

4.179910

6.298508

4.917862

8.134339

3.529884

6.660870

5.604551

4.378270

5.389574

8.405450

2021-10-18

5.041940

4.159039

6.293327

4.923987

8.145173

3.527830

6.642422

5.614077

4.359014

5.401460

8.408819

2021-10-19

5.047906

4.170225

6.309536

4.930509

8.144171

3.532810

6.658486

5.620727

4.378144

5.406723

8.416185

2021-10-20

5.059806

4.220977

6.321038

4.931303

8.134940

3.535728

6.677298

5.630172

4.397469

5.415923

8.419843

2021-10-21

5.055927

4.234251

6.342667

4.955264

8.142060

3.539219

6.697491

5.634218

4.400235

5.426974

8.422834

2021-10-22

5.062785

4.237145

6.339601

4.965777

8.112303

3.536165

6.711108

5.642439

4.409034

5.432499

8.421761

2021-10-25

5.064492

4.217889

6.342315

4.941642

8.108515

3.532079

6.713053

5.638675

4.399007

5.445573

8.426498

show_plot(log_df, "result_log")

Aligent

Centene

DEXcom

Carmax

Amazon

Voniter

Equinix

Laboratory corporation of american holding

Baxter International

Lowe's companies

S&P

8

7

