In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
stock = pd.read_csv('adani.csv')
```

In [3]:
```
stock.head(10)
```

Out[3]:

	timestamp	symbol	company	open	high	low	close	volume	divide
0	1025461800000000000	ACC	ACC Limited	107.789	109.810	107.115	108.968	659631	
1	1025461800000000000	AMBUJACEM	Ambuja Cements Limited	17.278	17.660	17.202	17.573	630442	
2	1025461800000000000	ADANIENT	Adani Enterprises Limited	-0.010	-0.011	-0.010	-0.010	1080397	
3	1025548200000000000	ACC	ACC Limited	108.496	110.046	107.789	108.059	282660	
4	1025548200000000000	AMBUJACEM	Ambuja Cements Limited	17.573	17.748	17.409	17.560	1007265	
5	1025548200000000000	ADANIENT	Adani Enterprises Limited	-0.011	-0.011	-0.010	-0.011	1016147	
6	1025634600000000000	ACC	ACC Limited	108.463	109.136	107.014	107.654	500553	
7	1025634600000000000	AMBUJACEM	Ambuja Cements Limited	17.434	17.566	17.202	17.303	178065	
8	1025634600000000000	ADANIENT	Adani Enterprises Limited	-0.011	-0.011	-0.010	-0.011	980394	
9	1025721000000000000	AMBUJACEM	Ambuja Cements Limited	17.284	17.409	17.246	17.296	199282	

In [4]:
```
stock.shape
```

Out[4]:
```
(31396, 10)
```

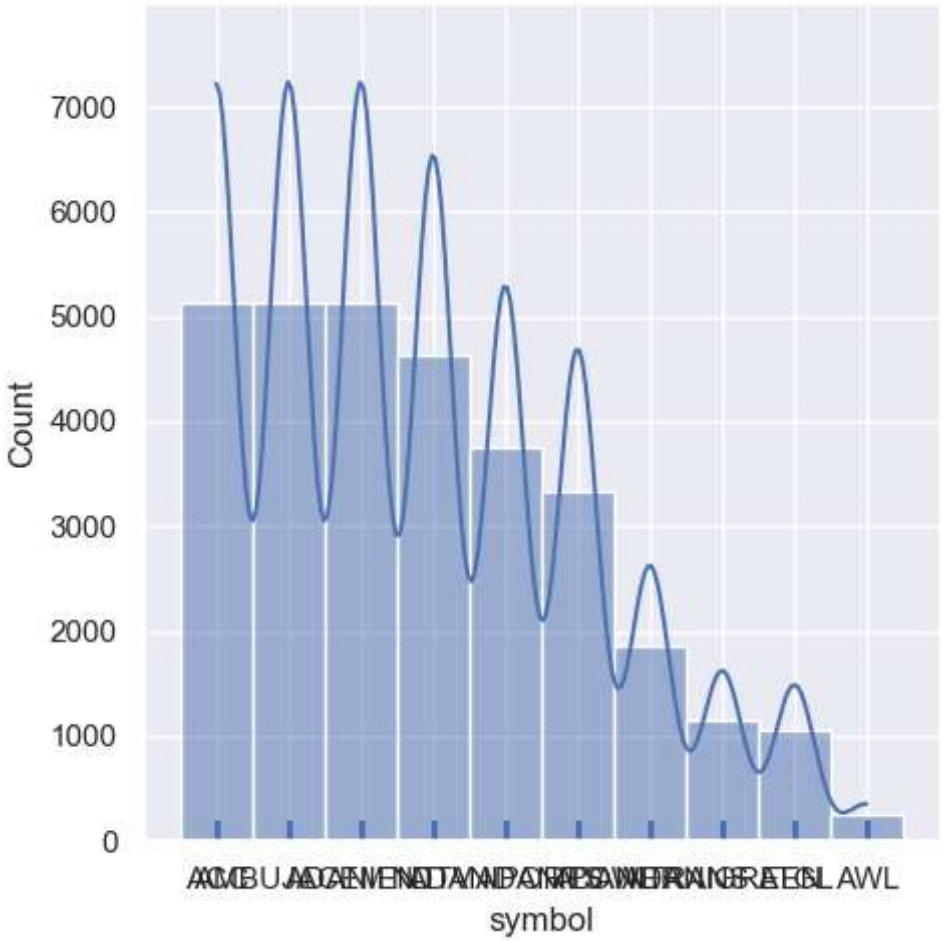
In [5]:
```
pd.isnull(stock).sum()
```

Out[5]:
```
timestamp       0
symbol          0
company         0
open            0
high            0
low             0
close           0
volume          0
dividends       0
stock_splits    0
dtype: int64
```

In [6]:
```
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(color_codes=True)
%matplotlib inline
```

In [7]:
```
sns.displot(stock['symbol'],rug=True,kde=True)
```

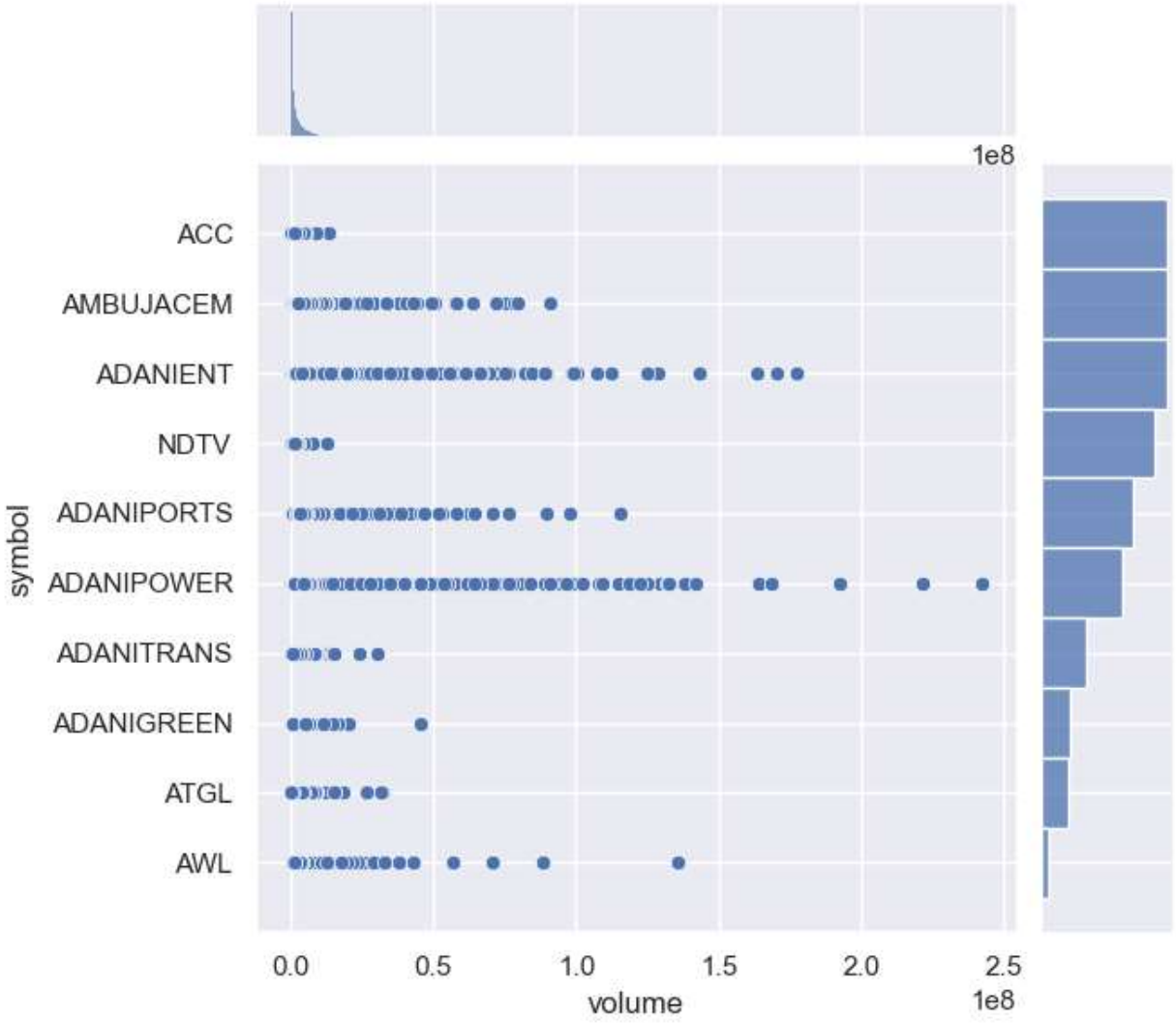
Out[7]:
```
<seaborn.axisgrid.FacetGrid at 0x1a096a28af0>
```



In [8]:
```
sns.jointplot(stock['volume'],stock['symbol'],)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

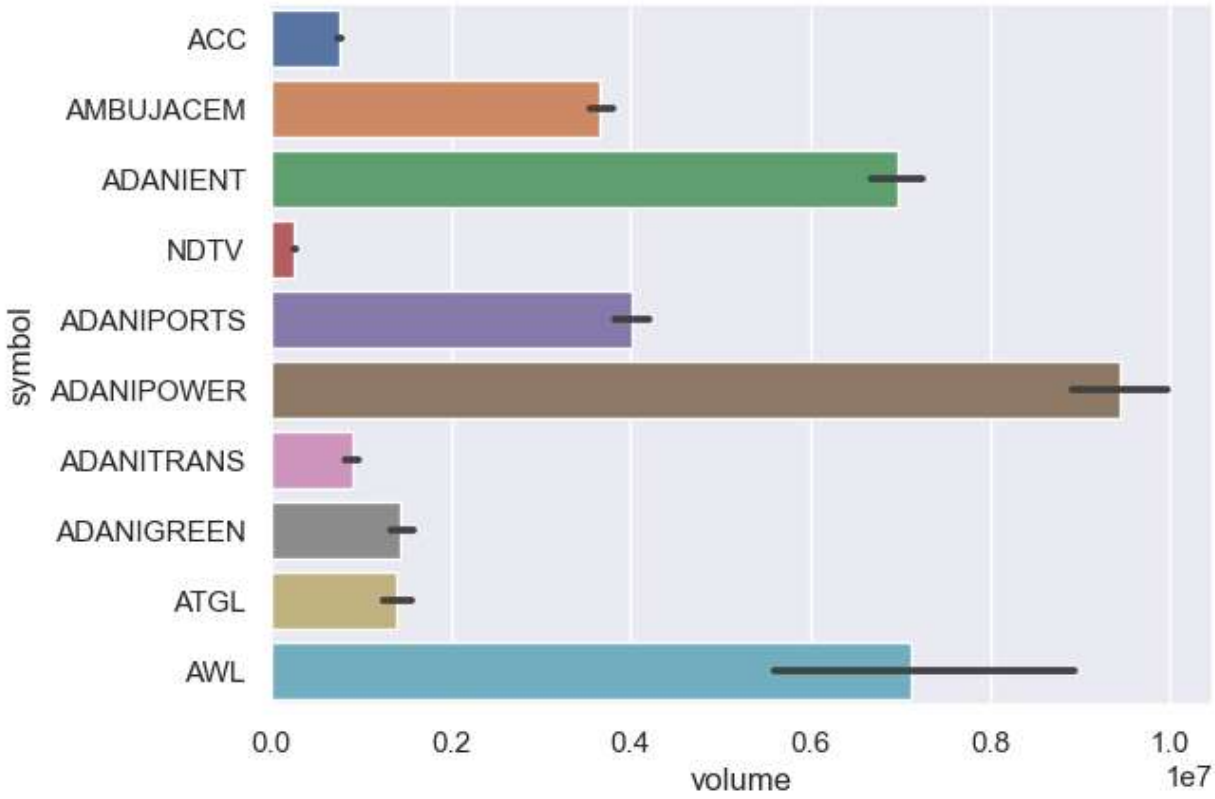
Out[8]: <seaborn.axisgrid.JointGrid at 0x1a099e838b0>



In [9]:
```
sns.barplot(stock['volume'],stock['symbol'])
```

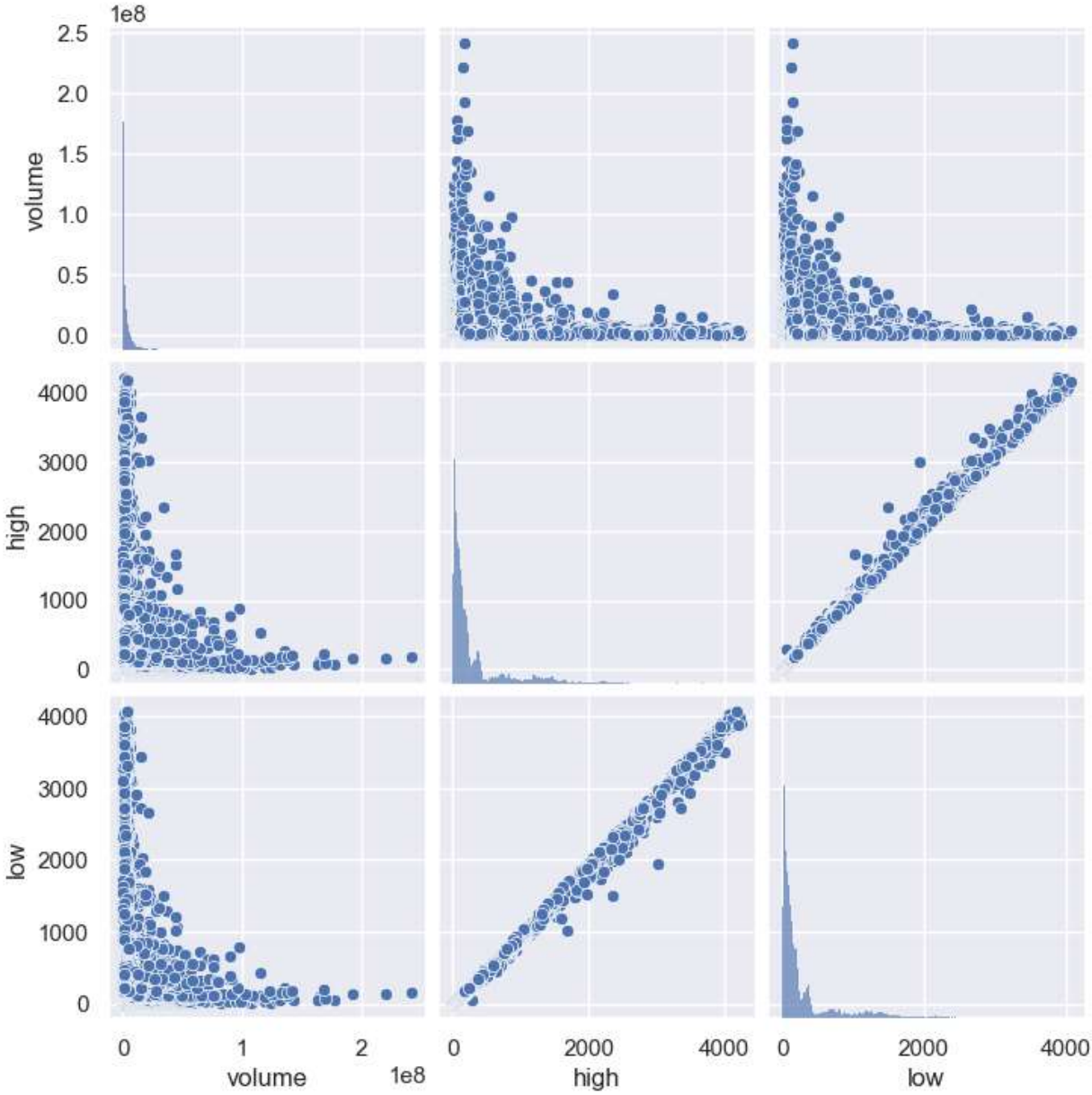
```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[9]: <AxesSubplot:xlabel='volume', ylabel='symbol'>



In [12]:
```
sns.pairplot(stock[['volume','symbol','high','low']])
```

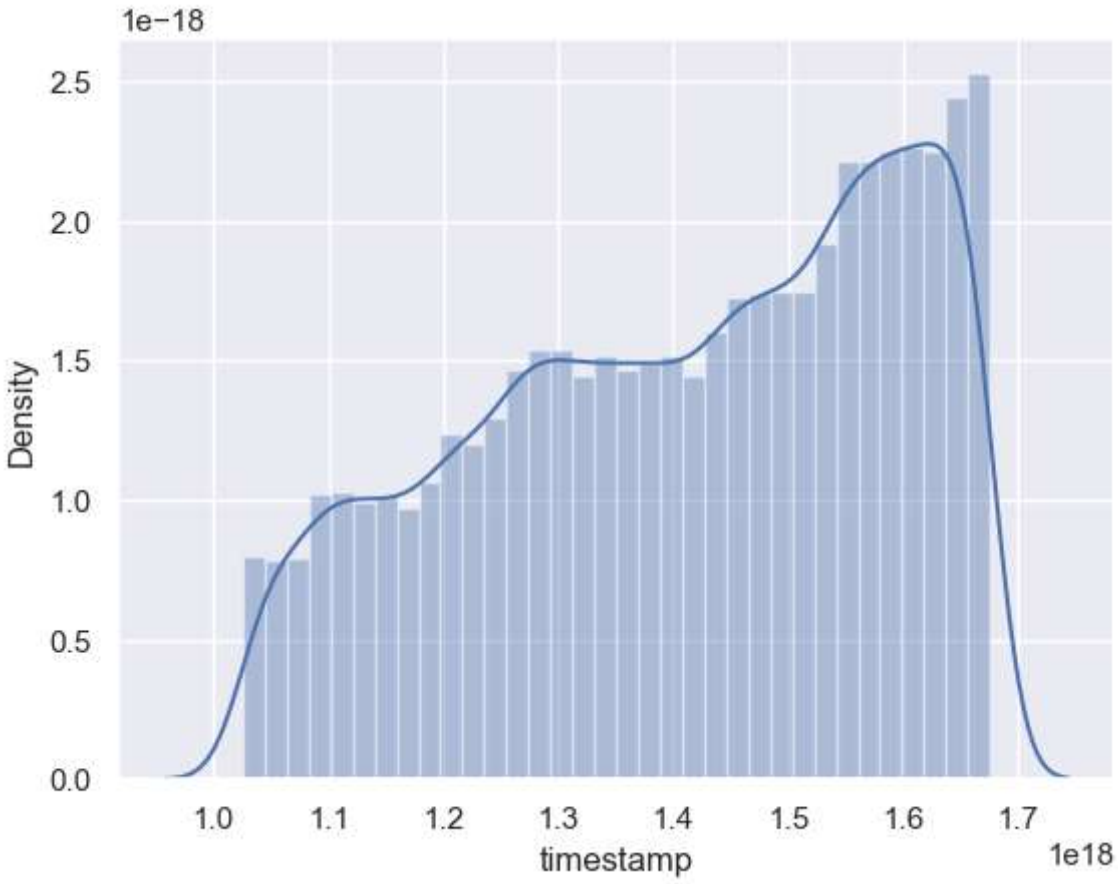
Out[12]:
```
<seaborn.axisgrid.PairGrid at 0x1d99d97c070>
```



In [13]:
```
sns.distplot(stock['timestamp'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

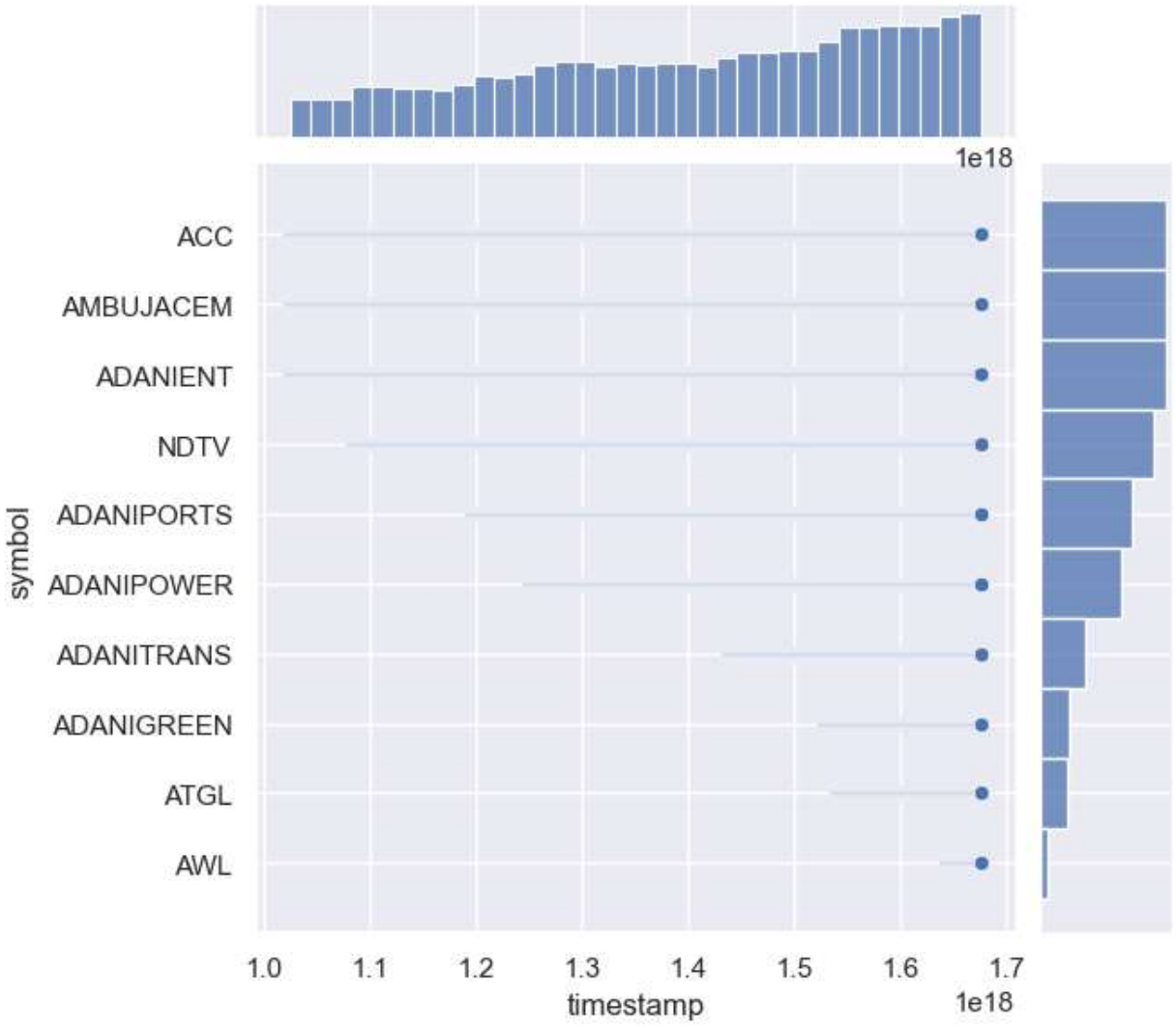
Out[13]:
```
<AxesSubplot:xlabel='timestamp', ylabel='Density'>
```



In [14]:
```
sns.jointplot(stock['timestamp'],stock['symbol'],)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

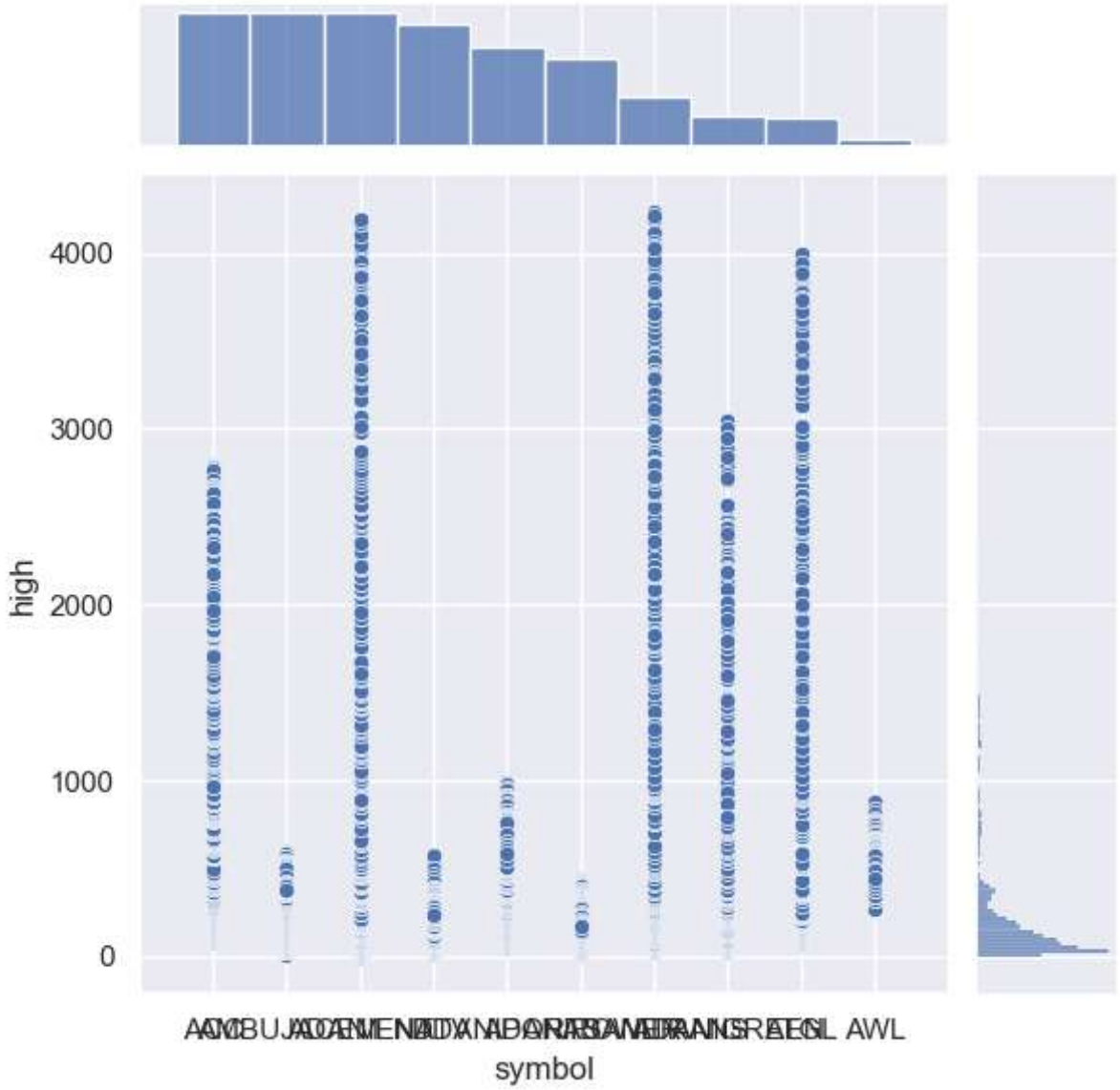
Out[14]:
```
<seaborn.axisgrid.JointGrid at 0x1d9a3558d90>
```



In [15]:
```
sns.jointplot(stock['symbol'],stock['high'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

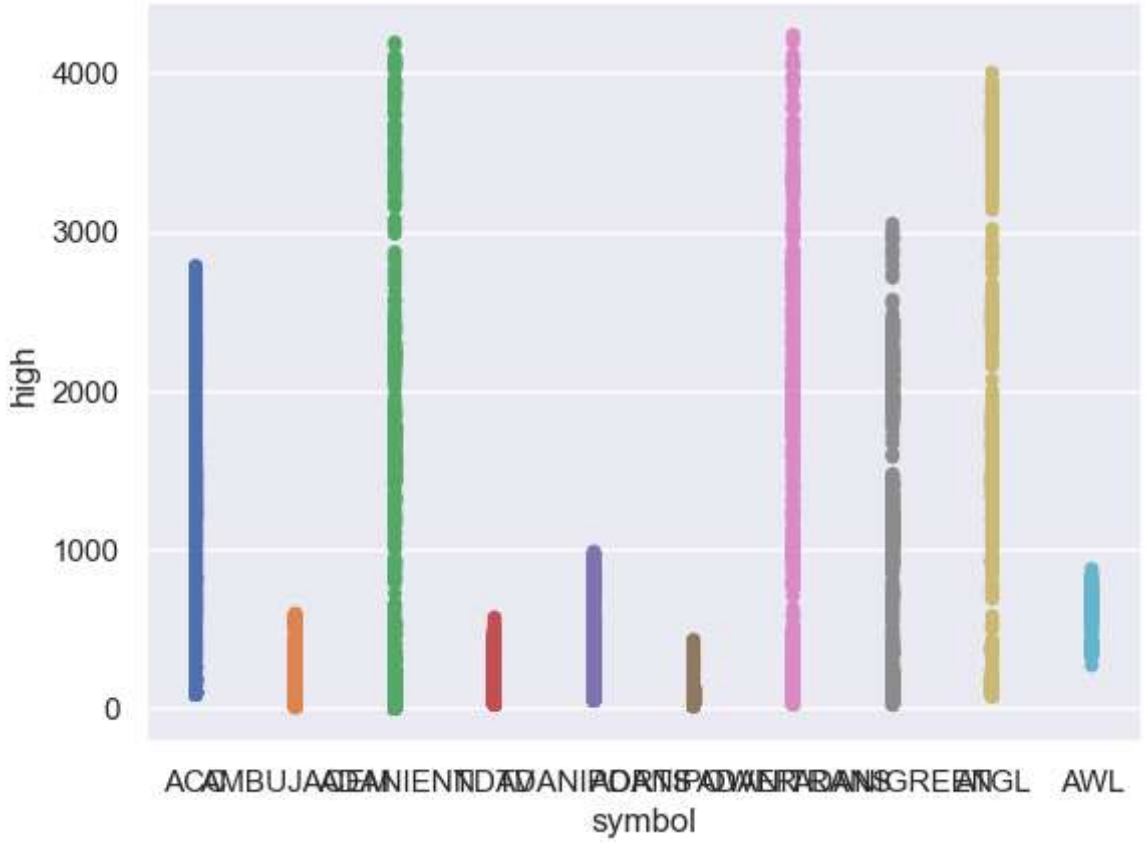
Out[15]:
```
<seaborn.axisgrid.JointGrid at 0x1d99fe69220>
```



In [10]:
```
sns.stripplot(stock['symbol'],stock['high'],jitter=False)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

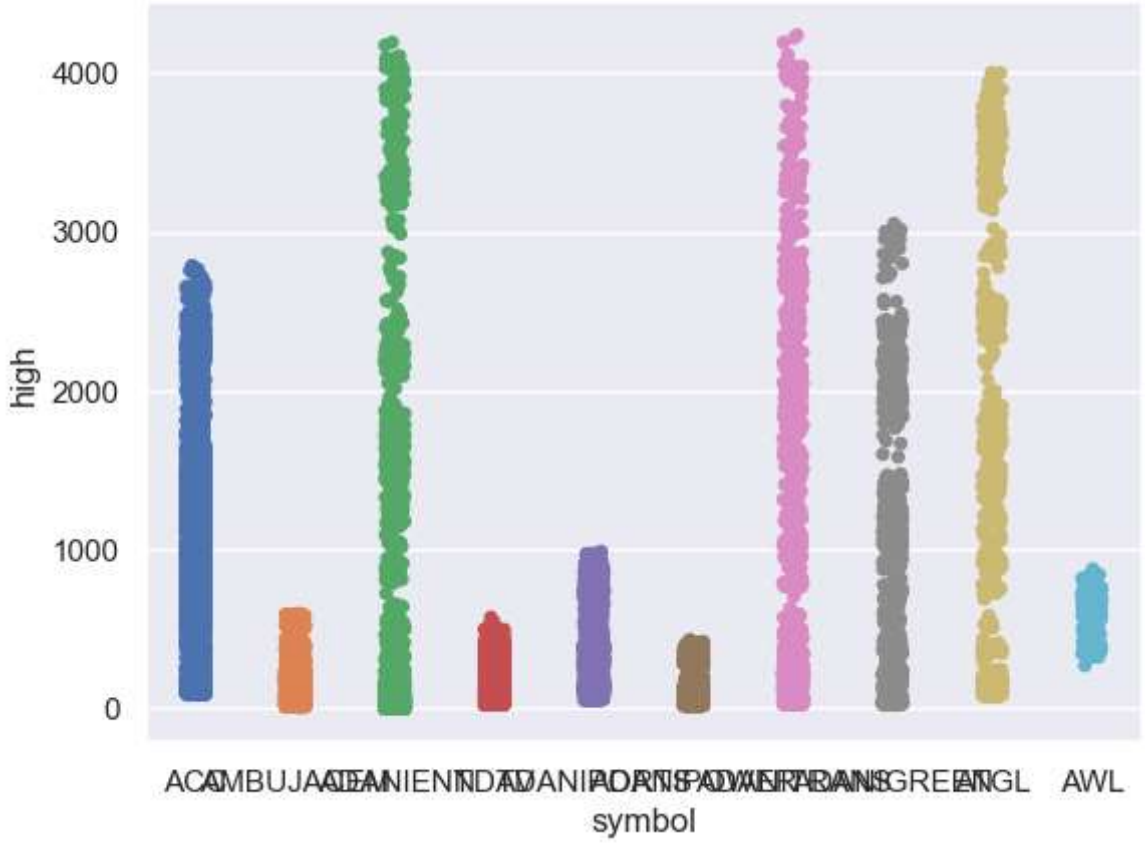
Out[10]:
```
<AxesSubplot:xlabel='symbol', ylabel='high'>
```



In [11]:
```
sns.stripplot(stock['symbol'],stock['high'],jitter=True)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

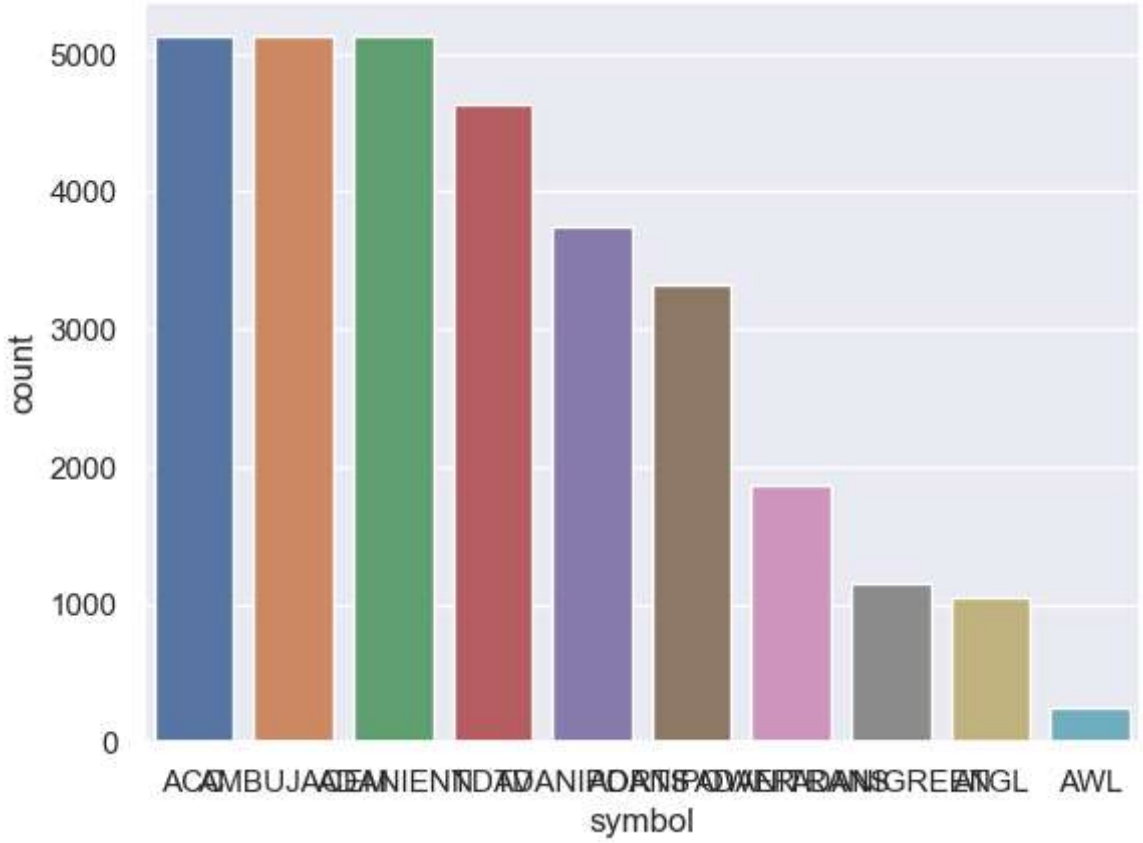
Out[11]:
```
<AxesSubplot:xlabel='symbol', ylabel='high'>
```



In [12]:
```
sns.countplot(stock['symbol'])
```

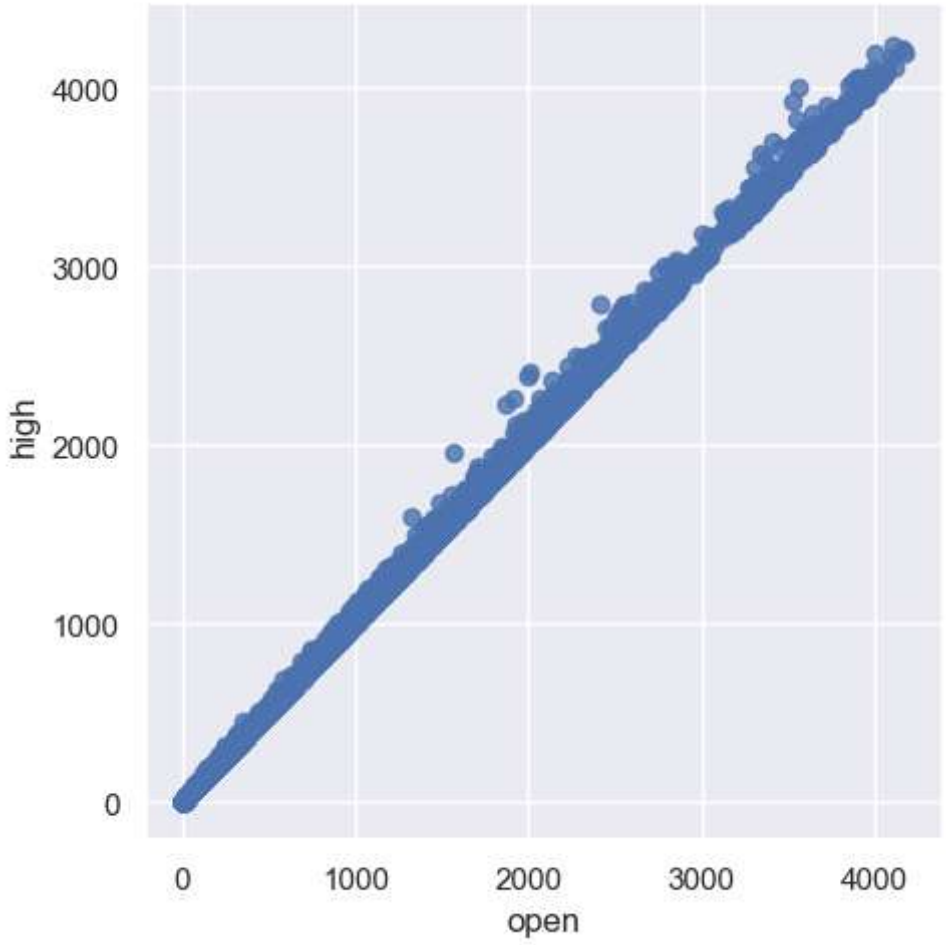
```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variable as a keyword arg: x. From version 0.12, the only valid po
sitional argument will be `data`, and passing other arguments without an explicit key
word will result in an error or misinterpretation.
  warnings.warn(
```

Out[12]:
```
<AxesSubplot:xlabel='symbol', ylabel='count'>
```



In [13]:
```
sns.lmplot(y="high",x="open",data=stock)
```

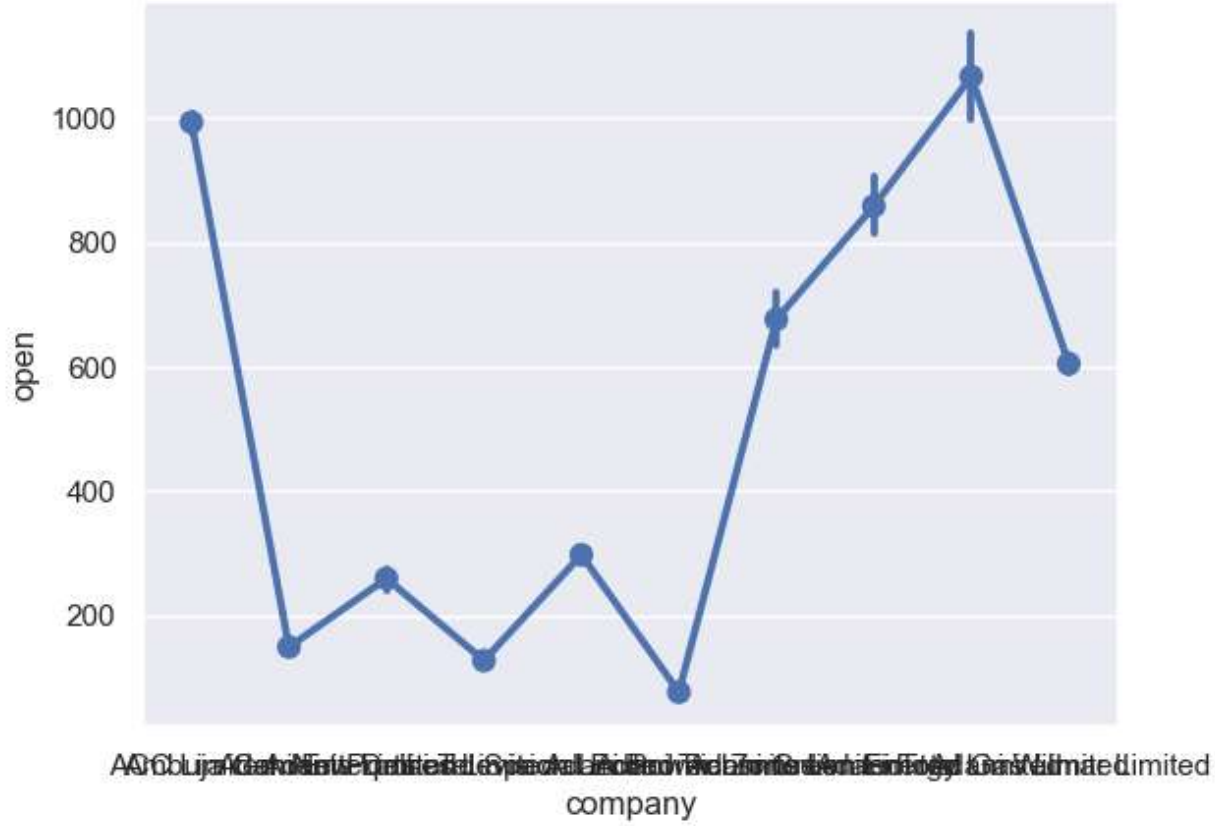
Out[13]:
```
<seaborn.axisgrid.FacetGrid at 0x1a09b81d5e0>
```



In [14]:
```
sns.pointplot(stock['company'],stock['open'])
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning:
Pass the following variables as keyword args: x, y. From version 0.12, the only valid
positional argument will be `data`, and passing other arguments without an explicit k
eyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[14]:
```
<AxesSubplot:xlabel='company', ylabel='open'>
```



In [15]:
```
from pandas_profiling import ProfileReport
```

```
C:\Users\shivesh\AppData\Local\Temp\ipykernel_9700\2274191625.py:1: DeprecationWarnin
g: `import pandas_profiling` is going to be deprecated by April 1st. Please use `impo
rt ydata_profiling` instead.
  from pandas_profiling import ProfileReport
```

In [16]:
```
profile = ProfileReport(stock,minimal = True)
```

In [17]:
```
profile
```

```
Summarize dataset:   0%|          | 0/5 [00:00<?, ?it/s]
Generate report structure:   0%|          | 0/1 [00:00<?, ?it/s]
Render HTML:   0%|          | 0/1 [00:00<?, ?it/s]
```

Overview

Dataset statistics

Number of variables	10
Number of observations	31396
Missing cells	0
Missing cells (%)	0.0%
Total size in memory	2.4 MiB
Average record size in memory	80.0 B

Variable types

Numeric	8
Categorical	2

Alerts

`dividends` is highly skewed (γ1 = 57.489116)	**Skewed**
`stock_splits` is highly skewed (γ1 = 115.1961792)	**Skewed**
`dividends` has 31275 (99.6%) zeros	**Zeros**
`stock_splits` has 31393 (> 99.9%) zeros	**Zeros**

Reproduction

Out[17]:

In [18]:
```
from autoviz.AutoViz_Class import AutoViz_Class
```

```
Imported v0.1.58. After importing, execute '%matplotlib inline' to display charts in
Jupyter.
    AV = AutoViz_Class()
    dfte = AV.AutoViz(filename, sep=',', depVar='', dfte=None, header=0, verbose=1, l
owess=False,
                  chart_format='svg',max_rows_analyzed=150000,max_cols_analyzed=30, save
_plot_dir=None)
Update: verbose=0 displays charts in your local Jupyter notebook.
        verbose=1 additionally provides EDA data cleaning suggestions. It also displa
ys charts.
        verbose=2 does not display charts but saves them in AutoViz_Plots folder in l
ocal machine.
        chart_format='bokeh' displays charts in your local Jupyter notebook.
        chart_format='server' displays charts in your browser: one tab for each chart
type
        chart_format='html' silently saves interactive HTML files in your local machi
ne
```

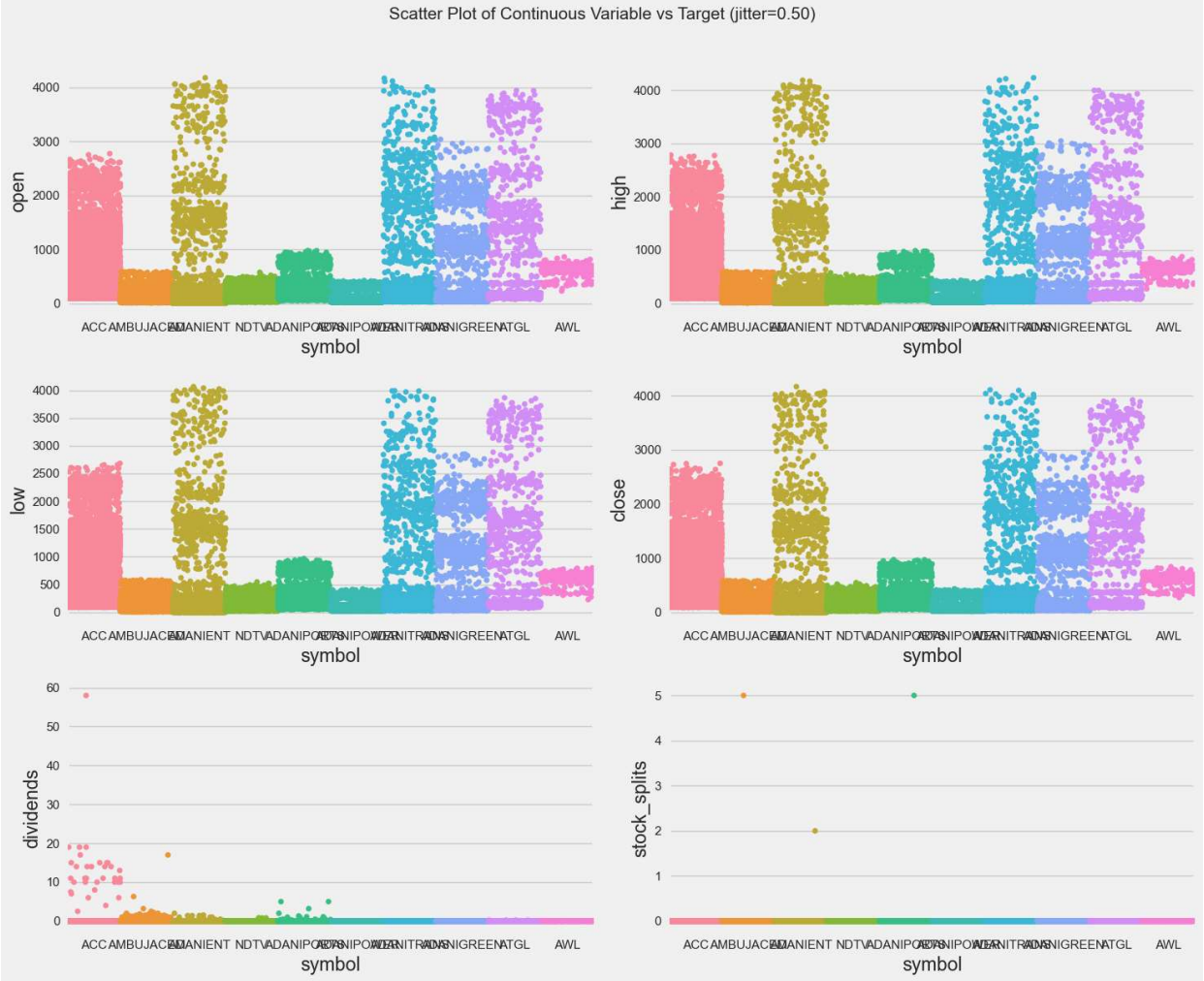
In [19]:
```
AV = AutoViz_Class()
```

In [21]:
```
%matplotlib inline
stock = AV.AutoViz('adani.csv', depVar='symbol')
```

```
Shape of your Data Set loaded: (31396, 10)
############################################################################################
##
######################## C L A S S I F Y I N G  V A R I A B L E S  ##################
##
############################################################################################
##
Classifying variables in data set...
Data cleaning improvement suggestions. Complete them before proceeding to ML modelin
g.
```

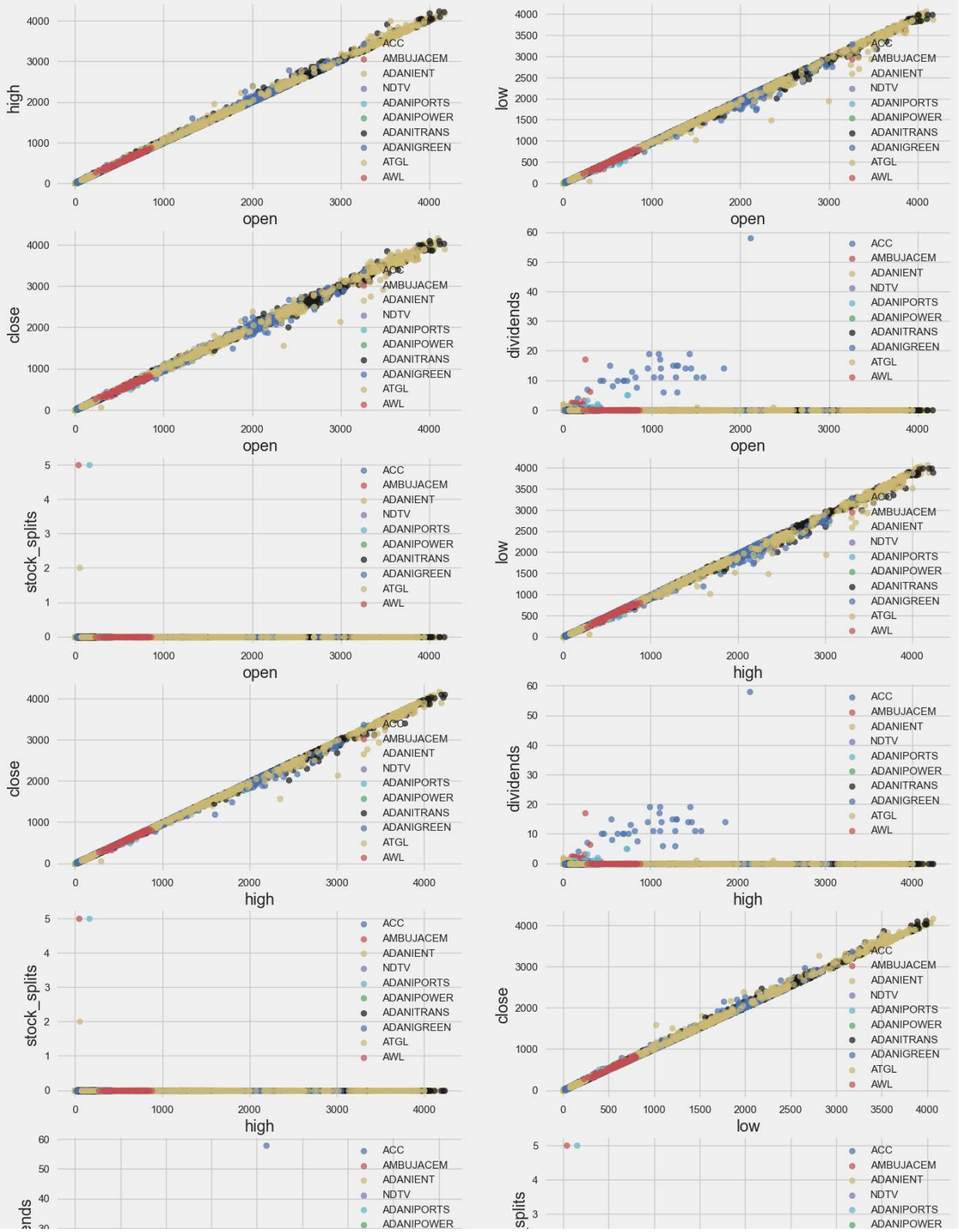
	Nuniques	dtype	Nulls	Nullpercent	NuniquePercent	Value counts Min	Data cleaning improvement suggestions
volume	31121	int64	0	0.000000	99.124092	0	
close	23875	float64	0	0.000000	76.044719	0	skewed: cap or drop outliers
high	22771	float64	0	0.000000	72.528348	0	skewed: cap or drop outliers
low	22729	float64	0	0.000000	72.394573	0	skewed: cap or drop outliers
open	21682	float64	0	0.000000	69.059753	0	skewed: cap or drop outliers
timestamp	5126	int64	0	0.000000	16.326921	0	
dividends	40	float64	0	0.000000	0.127405	0	highly skewed: drop outliers or do box-cox transform
company	10	object	0	0.000000	0.031851	251	
stock_splits	3	float64	0	0.000000	0.009555	0	highly skewed: drop outliers or do box-cox transform

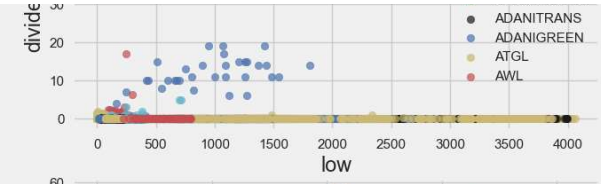
```
    9 Predictors classified...
        No variables removed since no ID or low-information variables found in data s
et

################## Multi_Classification problem ######################
```

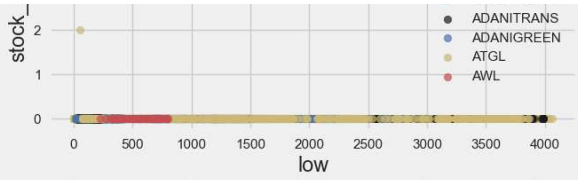


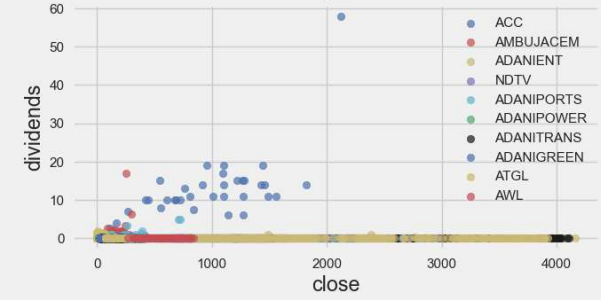
Total Number of Scatter Plots = 21

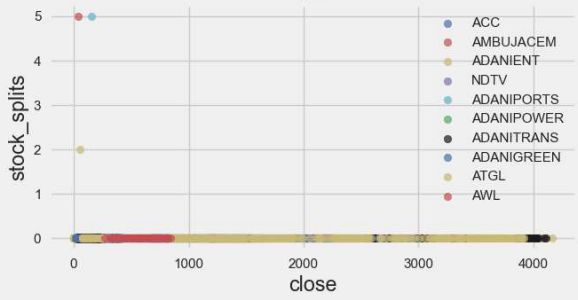
Pair-wise Scatter Plot of all Continuous Variables

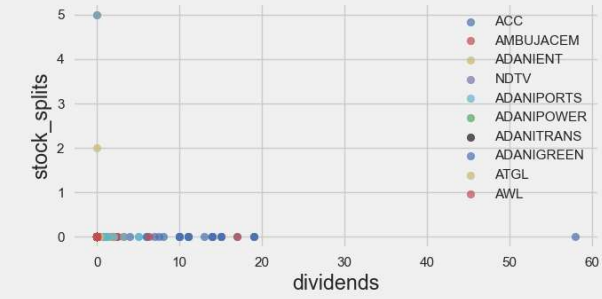


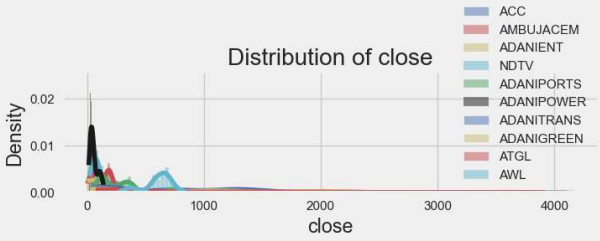

low
ADANITRANS
ADANIGREEN
ATGL
AWL

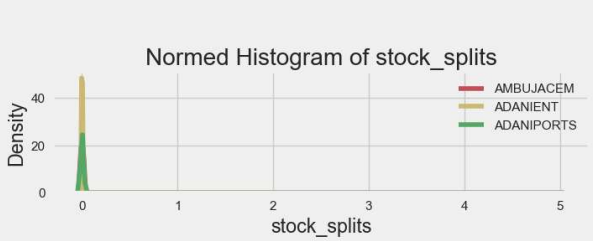

low
ADANITRANS
ADANIGREEN
ATGL
AWL

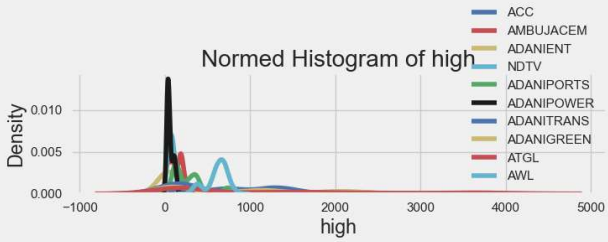

dividends
close
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL

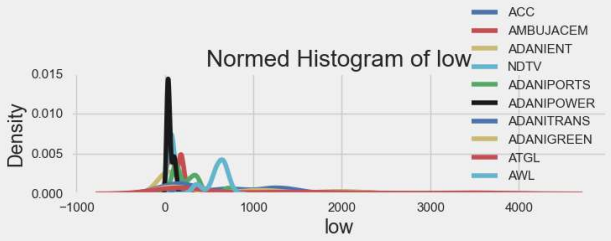

stock_splits
close
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL

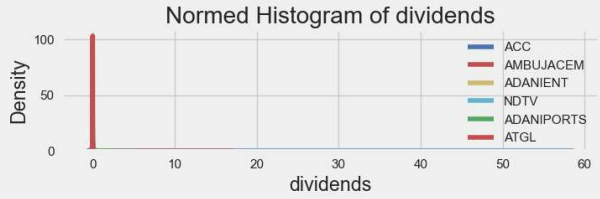

stock_splits
dividends
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL

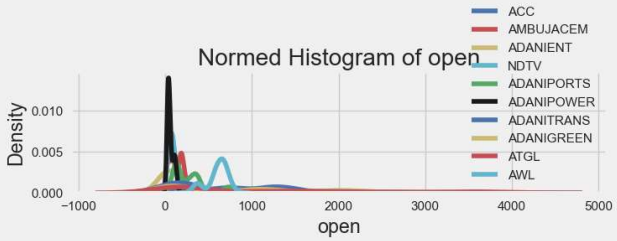

Distribution of close
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
Density
close

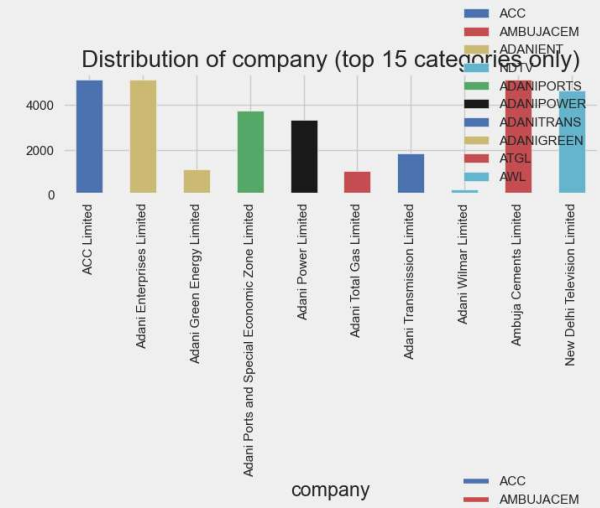

Normed Histogram of stock_splits
AMBUJACEM
ADANIENT
ADANIPORTS
Density
stock_splits

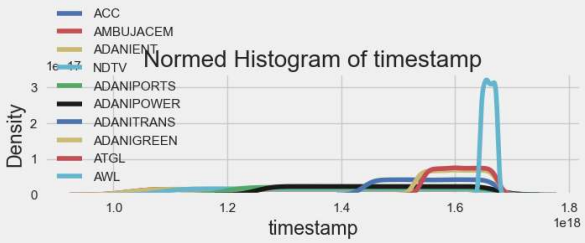

Normed Histogram of high
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
Density
high

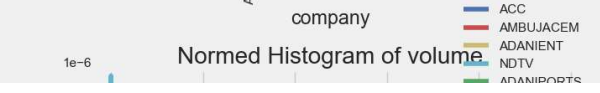

Normed Histogram of low
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
Density
low

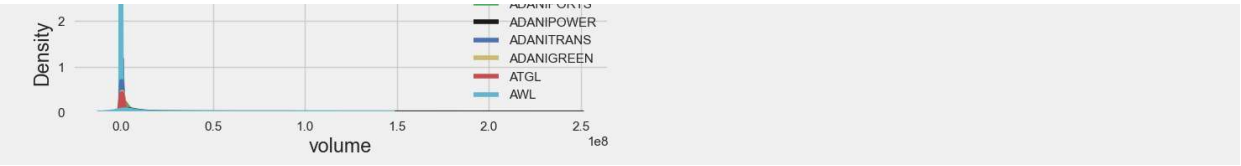

Normed Histogram of dividends
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ATGL
Density
dividends

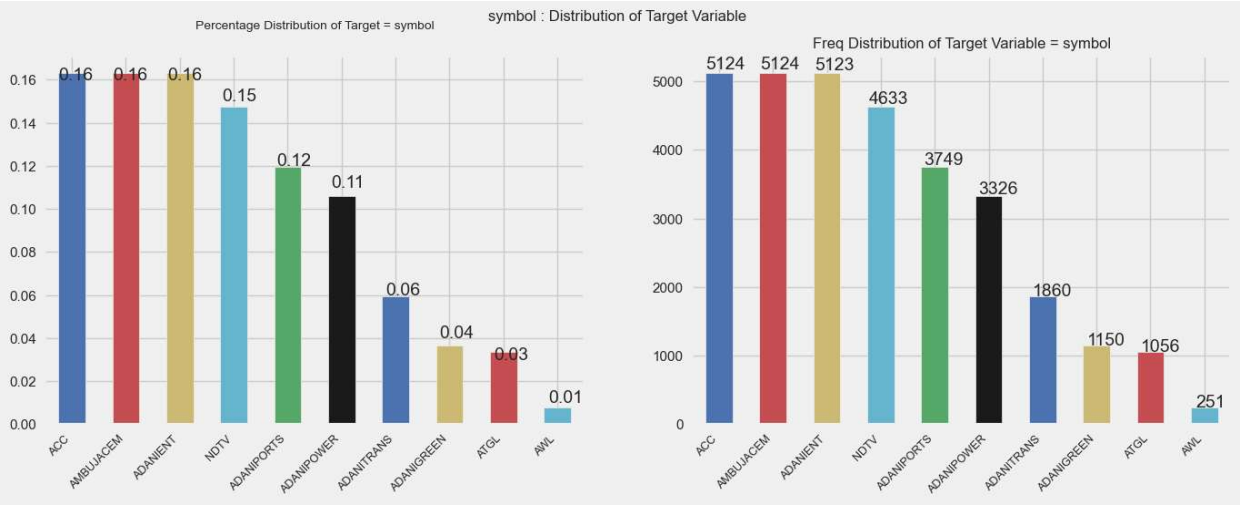

Normed Histogram of open
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
Density
open

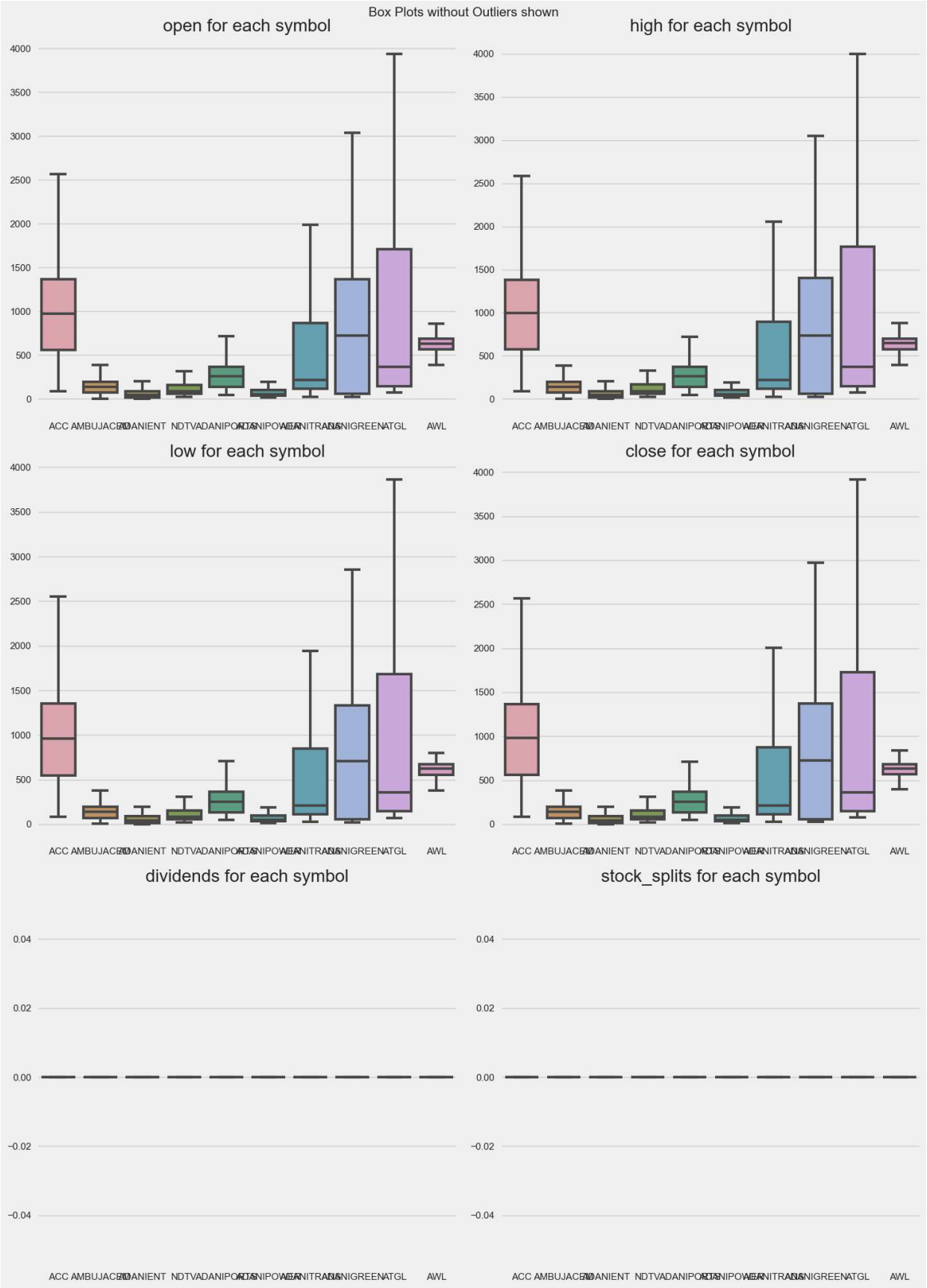

Distribution of company (top 15 categories only)
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
ACC Limited
Adani Enterprises Limited
Adani Green Energy Limited
Adani Ports and Special Economic Zone Limited
Adani Power Limited
Adani Total Gas Limited
Adani Transmission Limited
Adani Wilmar Limited
Ambuja Cements Limited
New Delhi Television Limited
company

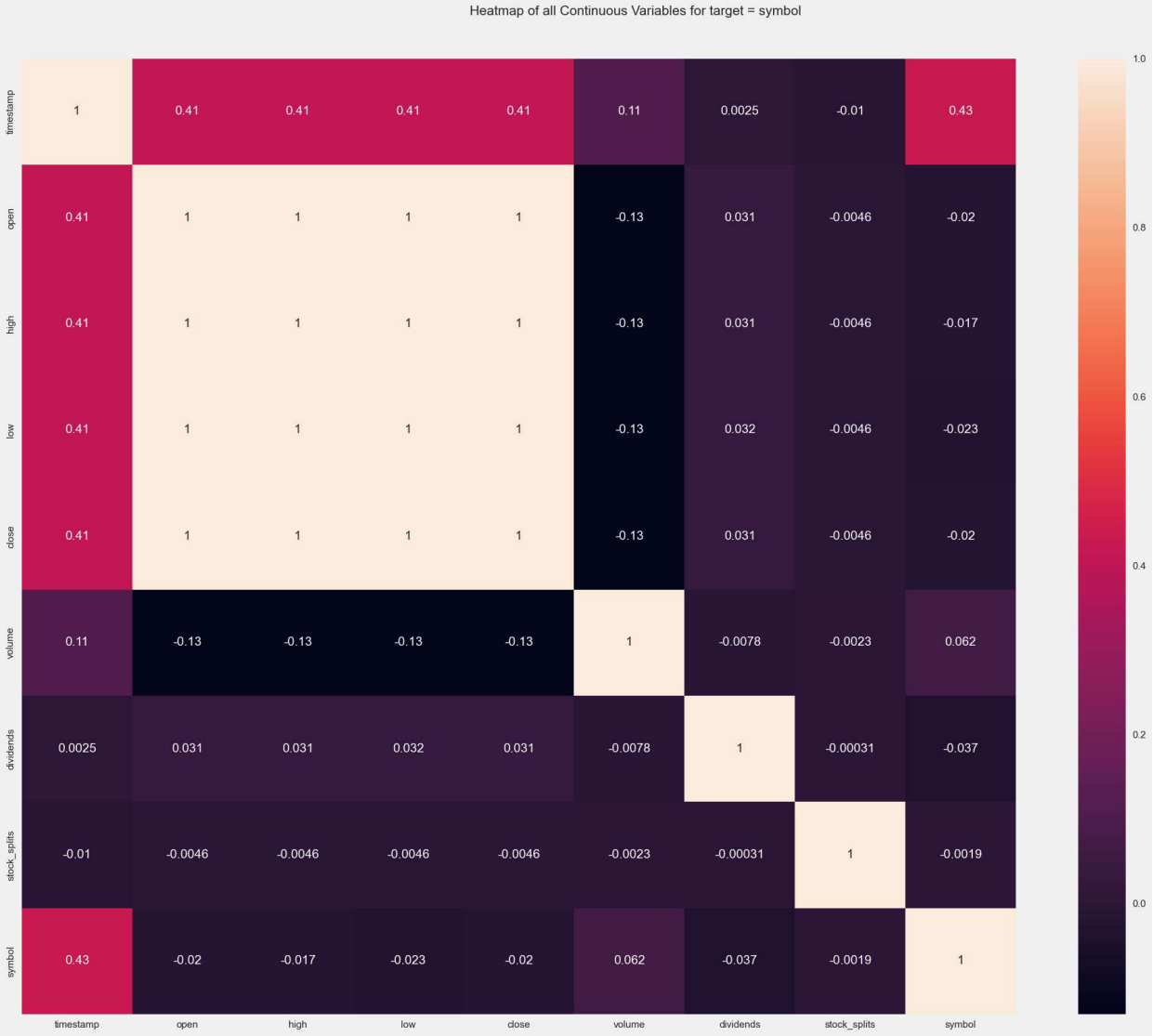

Normed Histogram of timestamp
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
Density
timestamp

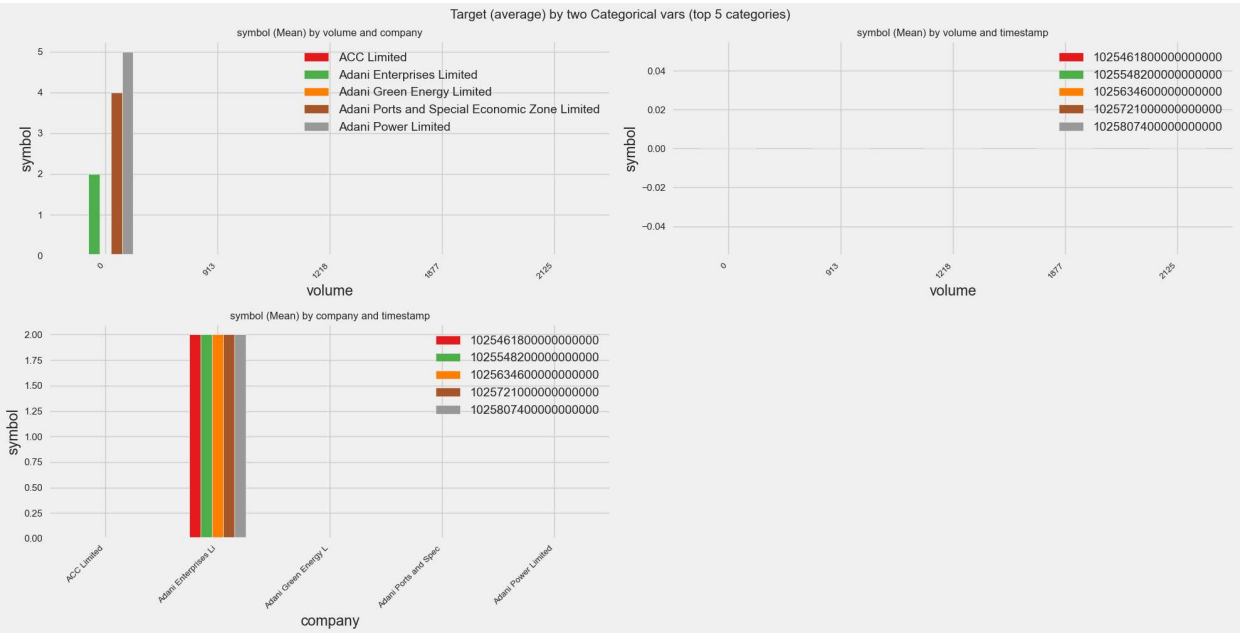

Normed Histogram of volume
ACC
AMBUJACEM
ADANIENT
NDTV


Density
volume
1e8
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL

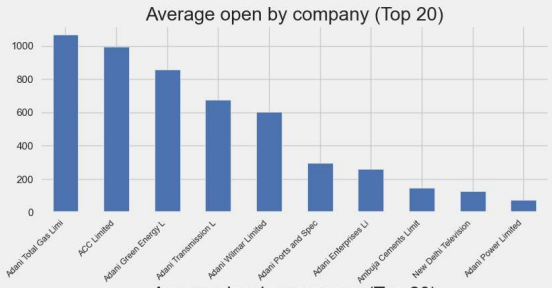

Percentage Distribution of Target = symbol
0.16
0.16
0.16
0.15
0.12
0.11
0.06
0.04
0.03
0.01
ACC
AMBUJACEM
ADANIENT
NDTV
ADANIPORTS
ADANIPOWER
ADANITRANS
ADANIGREEN
ATGL
AWL
symbol : Distribution of Target Variable
Freq Distribution of Target Variable = symbol
5124
5124
5123
4633
3749
3326
1860
1150
1056
251

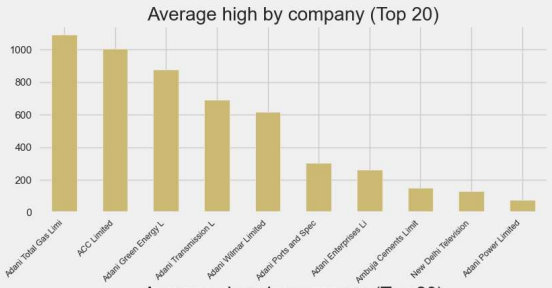

Box Plots without Outliers shown
open for each symbol
high for each symbol
low for each symbol
close for each symbol
dividends for each symbol
stock_splits for each symbol
ACC AMBUJACEM ADANIENT NDTV ADANIPORTS ADANIPOWER ABFRL ADANITRANS ADANIGREEN ATGL AWL

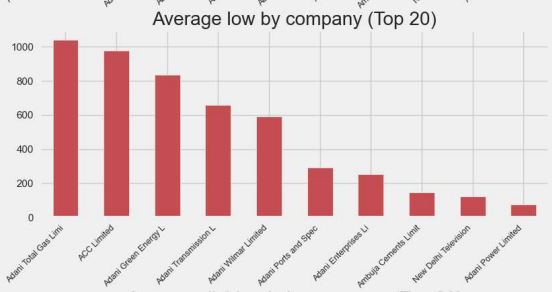

Heatmap of all Continuous Variables for target = symbol
timestamp
open
high
low
close
volume
dividends
stock_splits
symbol
1 0.41 0.41 0.41 0.41 0.11 0.0025 -0.01 0.43
0.41 1 1 1 1 -0.13 0.031 -0.0046 -0.02
0.41 1 1 1 1 -0.13 0.031 -0.0046 -0.017
0.41 1 1 1 1 -0.13 0.032 -0.0046 -0.023
0.41 1 1 1 1 -0.13 0.031 -0.0046 -0.02
0.11 -0.13 -0.13 -0.13 -0.13 1 -0.0078 -0.0023 0.062
0.0025 0.031 0.031 0.032 0.031 -0.0078 1 -0.00031 -0.037
-0.01 -0.0046 -0.0046 -0.0046 -0.0046 -0.0023 -0.00031 1 -0.0019
0.43 -0.02 -0.017 -0.023 -0.02 0.062 -0.037 -0.0019 1
1.0
0.8
0.6
0.4
0.2
0.0

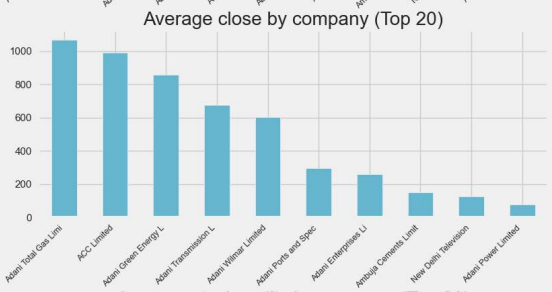

Target (average) by two Categorical vars (top 5 categories)
symbol (Mean) by volume and company
ACC Limited
Adani Enterprises Limited
Adani Green Energy Limited
Adani Ports and Special Economic Zone Limited
Adani Power Limited
symbol
volume
symbol (Mean) by volume and timestamp
1025461800000000000
1025548200000000000
1025634600000000000
1025721000000000000
1025807400000000000
symbol
volume
symbol (Mean) by company and timestamp
1025461800000000000
1025548200000000000
1025634600000000000
1025721000000000000
1025807400000000000
symbol
ACC Limited
Adani Enterprises Li
Adani Green Energy L
Adani Ports and Spec
Adani Power Limited
company

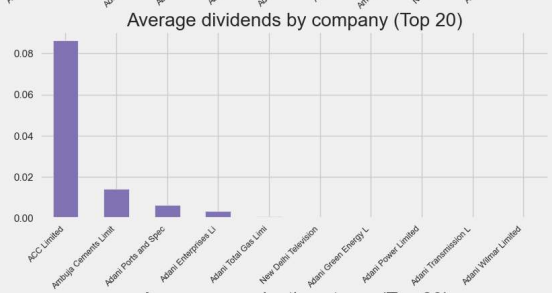
Bar plots for each Continuous by each Categorical variable

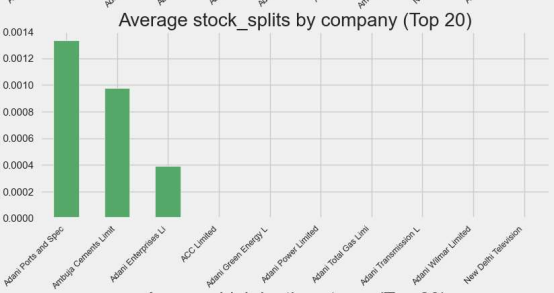


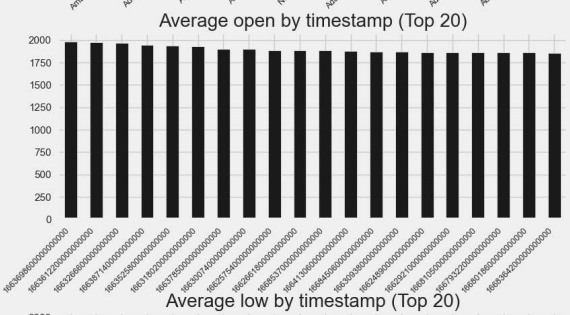


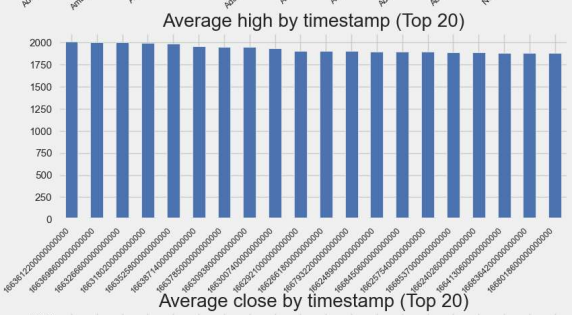


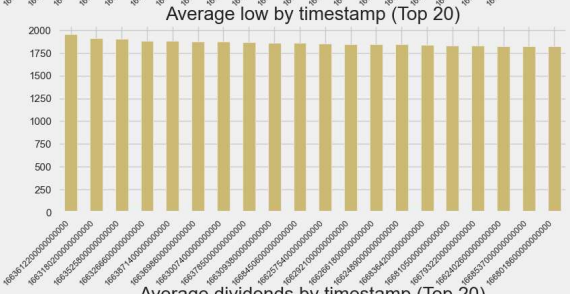


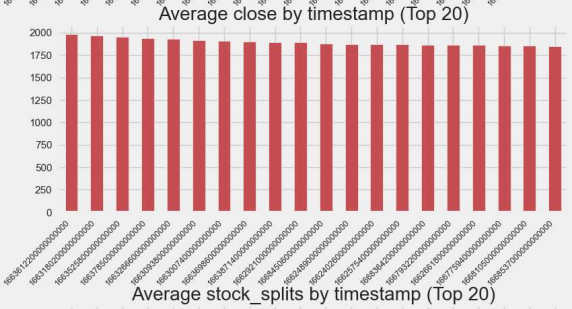


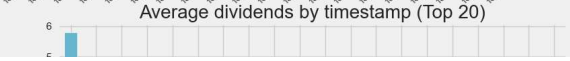


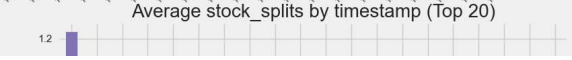


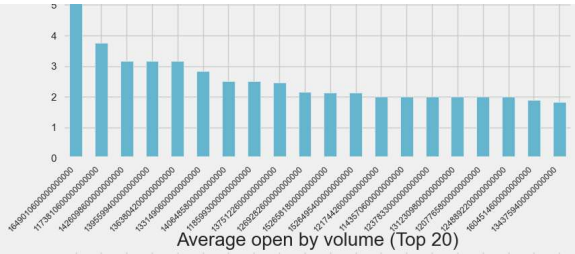


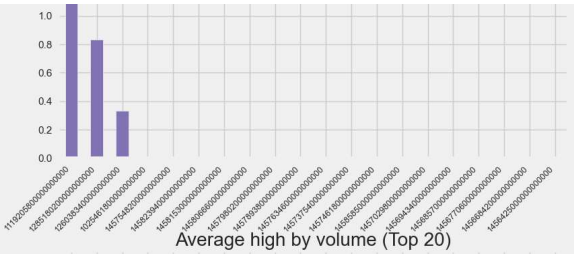




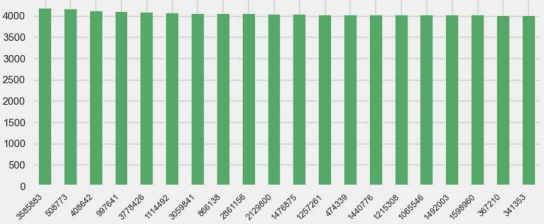




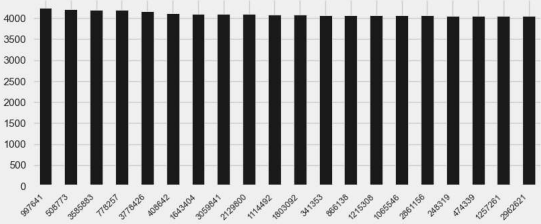




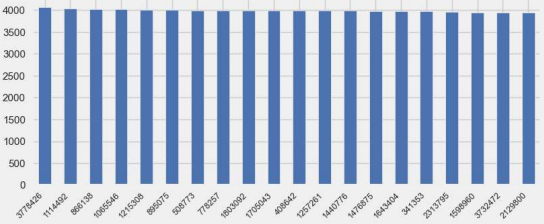
Average open by volume (Top 20)



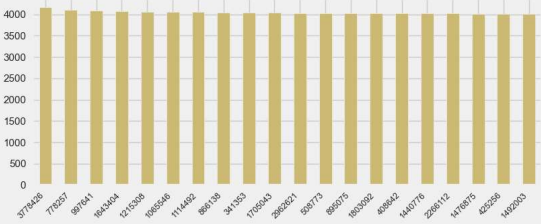
Average high by volume (Top 20)



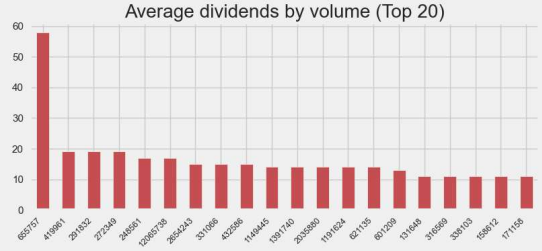
Average low by volume (Top 20)



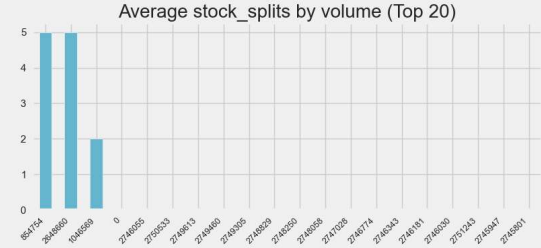
Average close by volume (Top 20)



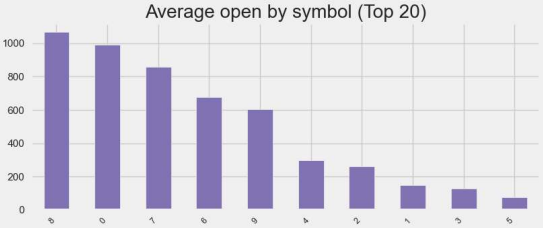
Average dividends by volume (Top 20)



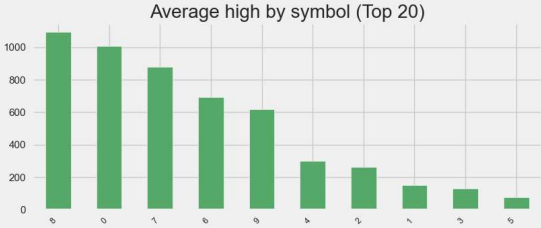
Average stock_splits by volume (Top 20)



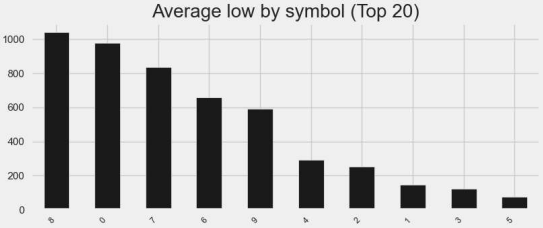
Average open by symbol (Top 20)



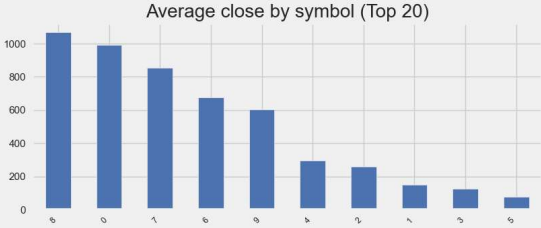
Average high by symbol (Top 20)



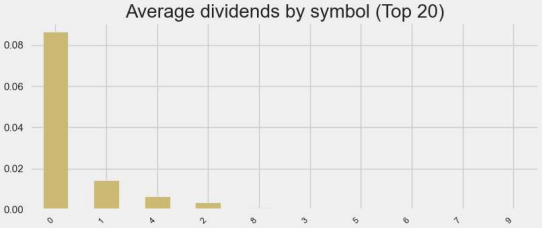
Average low by symbol (Top 20)



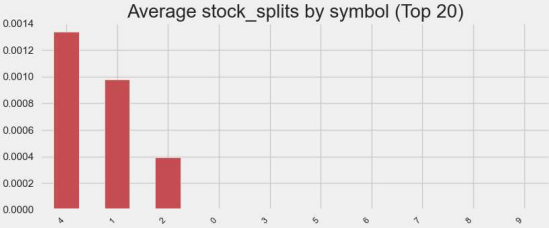
Average close by symbol (Top 20)



Average dividends by symbol (Top 20)



Average stock_splits by symbol (Top 20)



```
All Plots done
Time to run AutoViz = 39 seconds

 ###################### AUTO VISUALIZATION Completed ########################
```

In [ ]: