Use case

Task of the work – make neural network to predict stock prices as far as it’s possible.

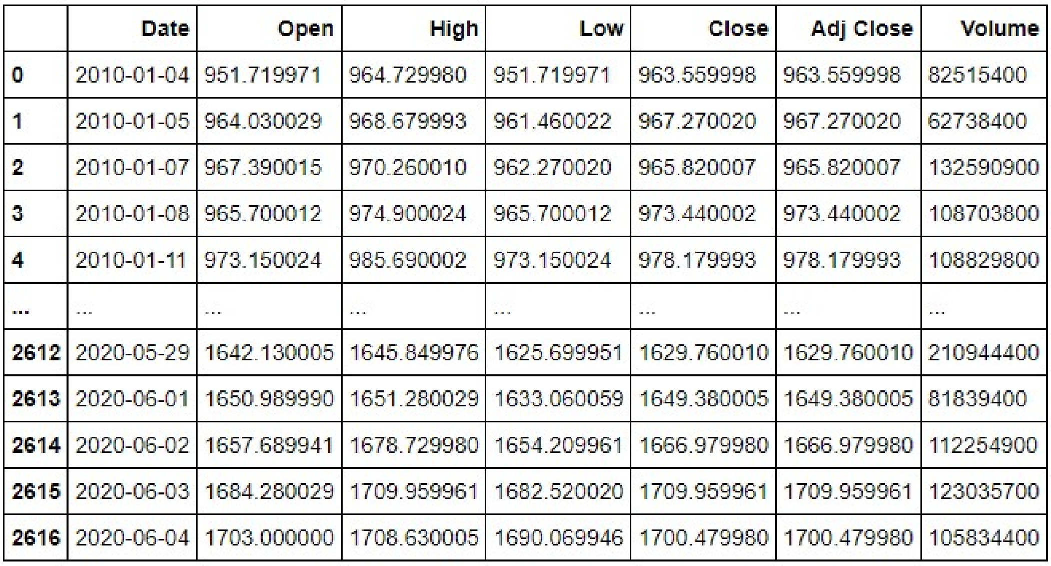
Data set

As a dataset I am using daily prices from several years grabbed from Yahoo Finance web site. For this task I am using ready-made python library, but it is only BeautifulSoup inside.

As a result, I am getting data set with bunch of prices for selected ticker.

Data exploration, visualization and Quality assessment.

We have a data set with Date, Open, High, Low and Close prices, Adjusted close price and daily volume of trade



	Date	Open	High	Low	Close	Adj Close	Volume
0	2010-01-04	951.719971	964.729980	951.719971	963.559998	963.559998	82515400
1	2010-01-05	964.030029	968.679993	961.460022	967.270020	967.270020	62738400
2	2010-01-07	967.390015	970.260010	962.270020	965.820007	965.820007	132590900
3	2010-01-08	965.700012	974.900024	965.700012	973.440002	973.440002	108703800
4	2010-01-11	973.150024	985.690002	973.150024	978.179993	978.179993	108829800
...	...	...	...	...	...	...	...
2612	2020-05-29	1642.130005	1645.849976	1625.699951	1629.760010	1629.760010	210944400
2613	2020-06-01	1650.989990	1651.280029	1633.060059	1649.380005	1649.380005	81839400
2614	2020-06-02	1657.689941	1678.729980	1654.209961	1666.979980	1666.979980	112254900
2615	2020-06-03	1684.280029	1709.959961	1682.520020	1709.959961	1709.959961	123035700
2616	2020-06-04	1703.000000	1708.630005	1690.069946	1700.479980	1700.479980	105834400

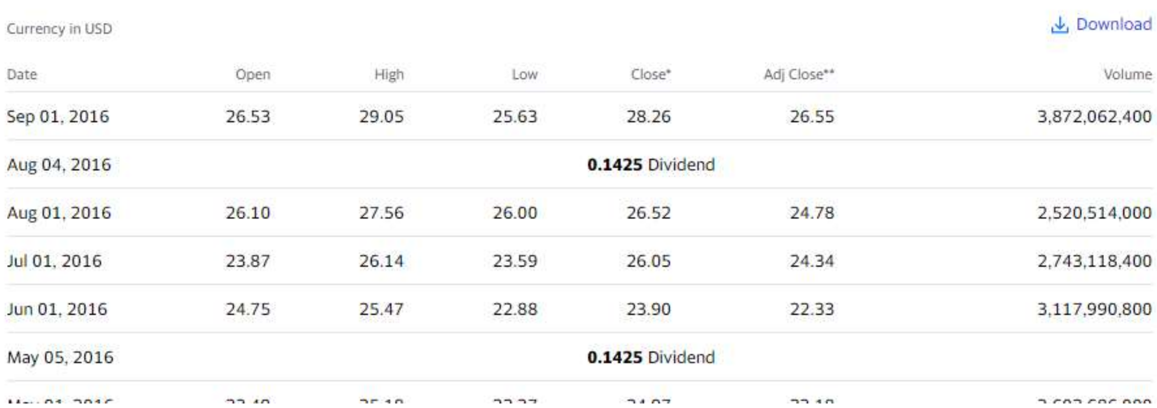
We can plot a graph.

We need only date and close price. Rest of prices can be dropped.

There are two possible issues in that dataframe.

1)Sometimes we have “null” value (spelled exactly like this). I am just dropping such rows, because it is only few cases it will not affect training or testing neural network.

2)Sometimes Yahoo adds dividend data like here



Currency in USD

Date	Open	High	Low	Close*	Adj Close**	Volume
Sep 01, 2016	26.53	29.05	25.63	28.26	26.55	3,872,062,400
Aug 04, 2016				**0.1425** Dividend		
Aug 01, 2016	26.10	27.56	26.00	26.52	24.78	2,520,514,000
Jul 01, 2016	23.87	26.14	23.59	26.05	24.34	2,743,118,400
Jun 01, 2016	24.75	25.47	22.88	23.90	22.33	3,117,990,800
May 05, 2016				**0.1425** Dividend		

On the screenshot we have monthly data, but issue with daily is exactly the same. Such rows shall be deleted.

All columns are strings, so data needs to be converted to datetime and float64 formats

Data is normalized and divided between training and test set (approx.. 90% and 10%)

Model and metrics selection.

Stock price is a process developed in time, but do not really depending on time. There is no direct correlation between time and price, so most of simple Machine Learning techniques won’t work.

To solve the task, I will use an LSTM neural network.

After few experiments I came to an architecture:

Sequence model, 3 LSTM layers, 3 Dropout layers between them and Dense layer at the end.

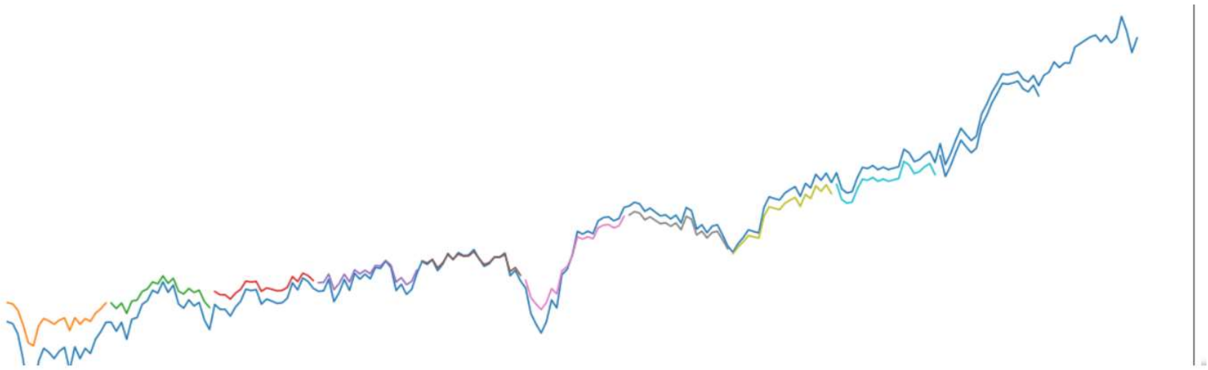
MSE loss function will be use.

To measure prediction performance, we can use any “distance” metrics, but it is better to just build a plot.

Testing

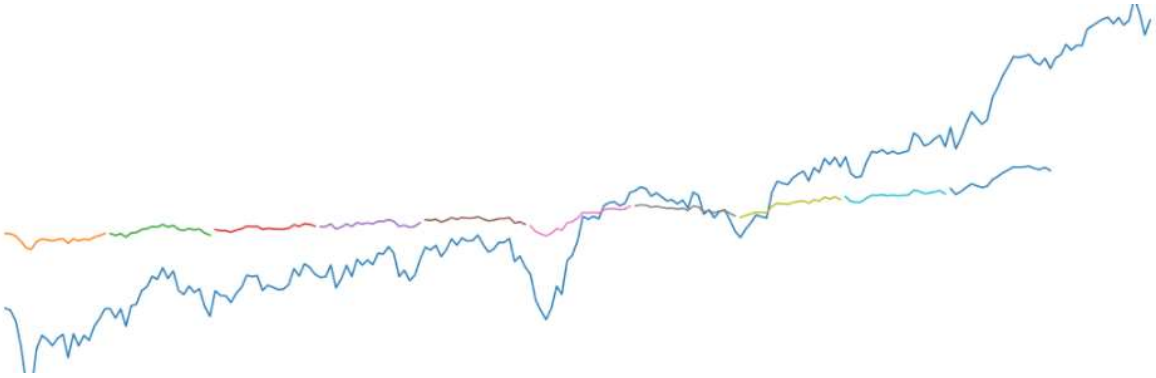
To test model I am using completely new set of data (1 year) which directly follows after training set

Predicted prices are presented as short color lines on a picture. Experiments showed, that we have a good prediction up to 25-30 steps forward. After that, we have much worse result

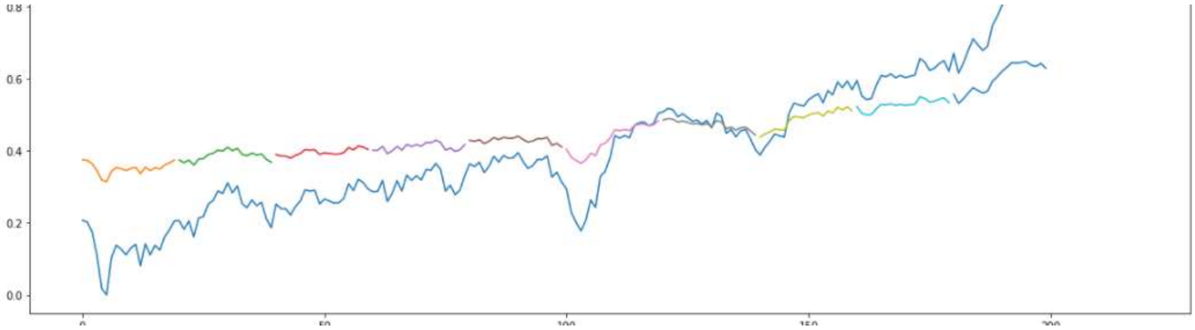


This graph is a result of long feature engineering. For example SGD optimizer gives us very bad performance after model training.

If you will reduce training set two times result will be like on the picture below.



Reducing dropout rate to 0.05 improves performace, but model is still underperforming


0.8
0.6
0.4
0.2
0.0
0
50
100
150
200