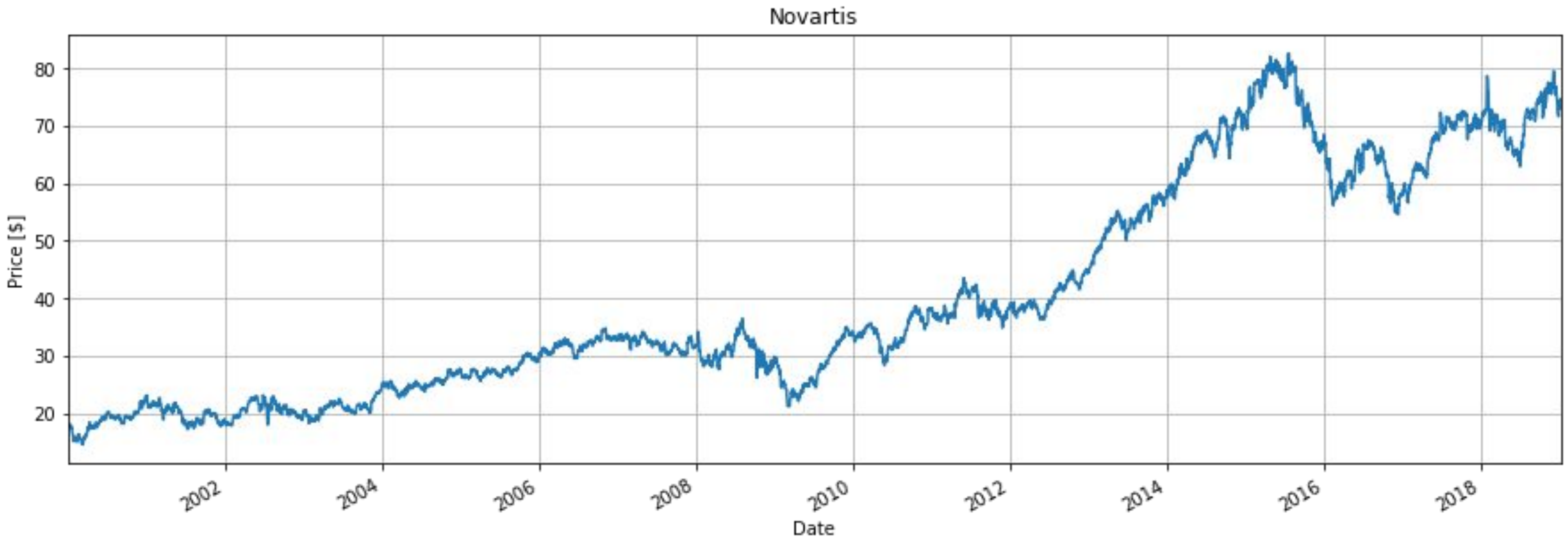
Stock Market Prediction

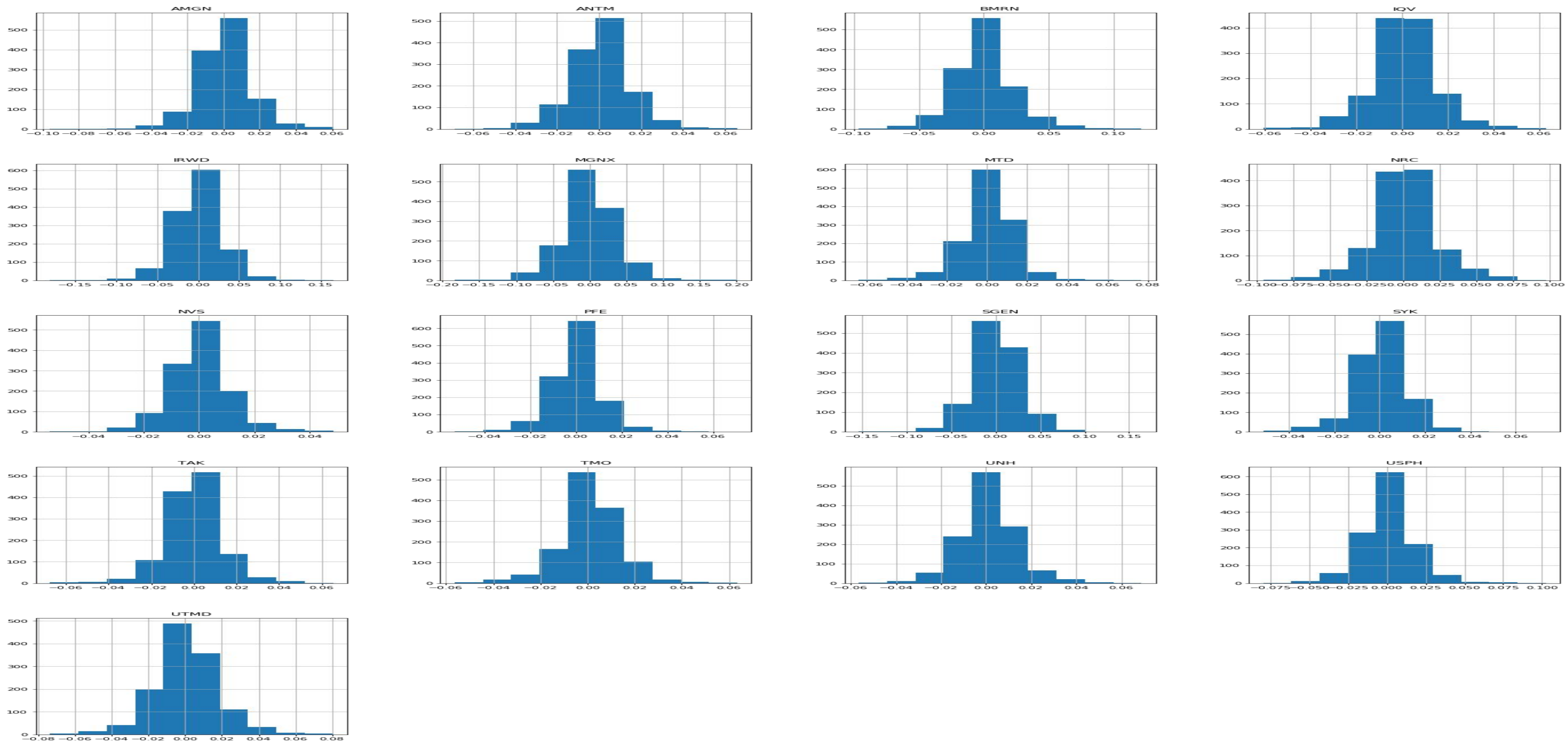


Biotech Industry

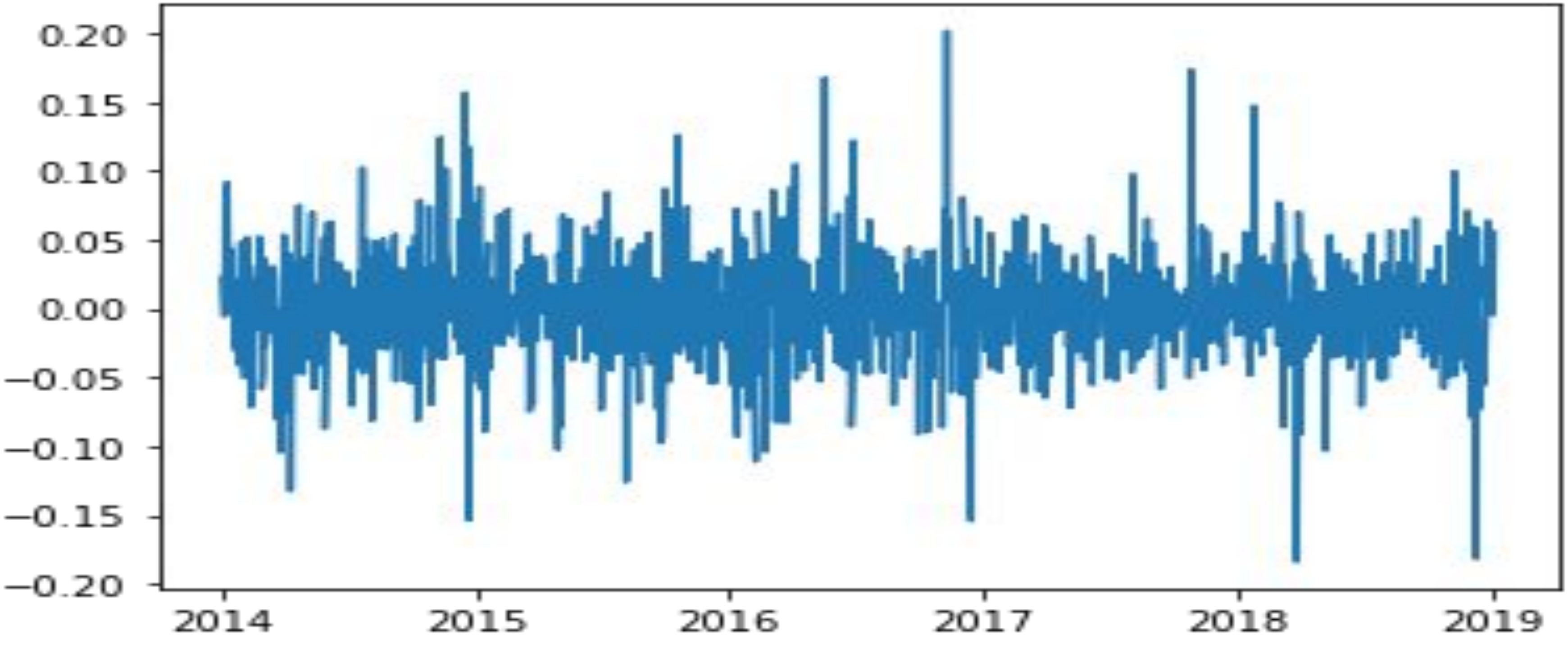
Portfolio Optimization for Stock Selection

- Choose a frequency at which to calculate returns, daily, monthly.
- Test tickers for normality using Kurtosis and Skewness thresholds.
- Skew: -0.5 < price < 0.5
- Kurtosis: price <= 3
- Finally, calculate Sortino Ratio by setting your Target Rate to 0 and your Risk Free Rate to the current RFR.
- Collect all the negative returns from your returns dataframe.
- Next calculate the expected returns on your returns dataframe.
- Lastly, calculate the Standard Deviation of the Negative Returns.
- Sort the values in descending order, choose top 5 stocks and you're done.

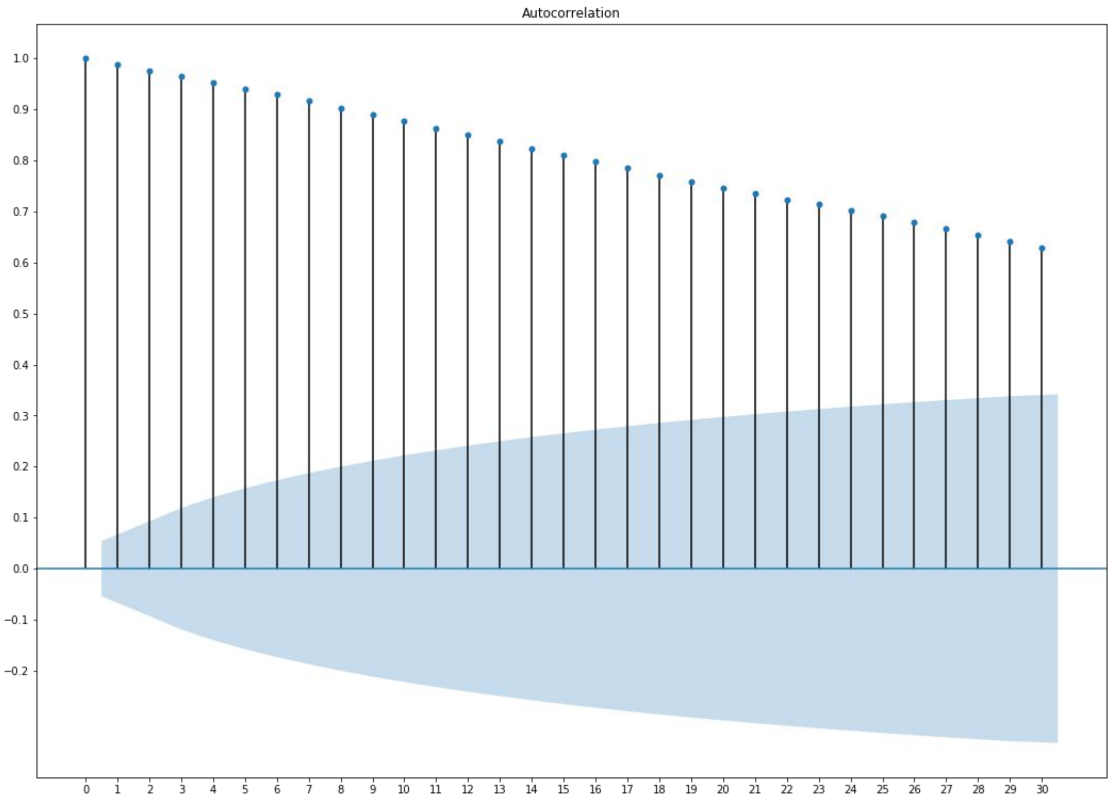
From 196 Tickers to 17 Normal Tickers

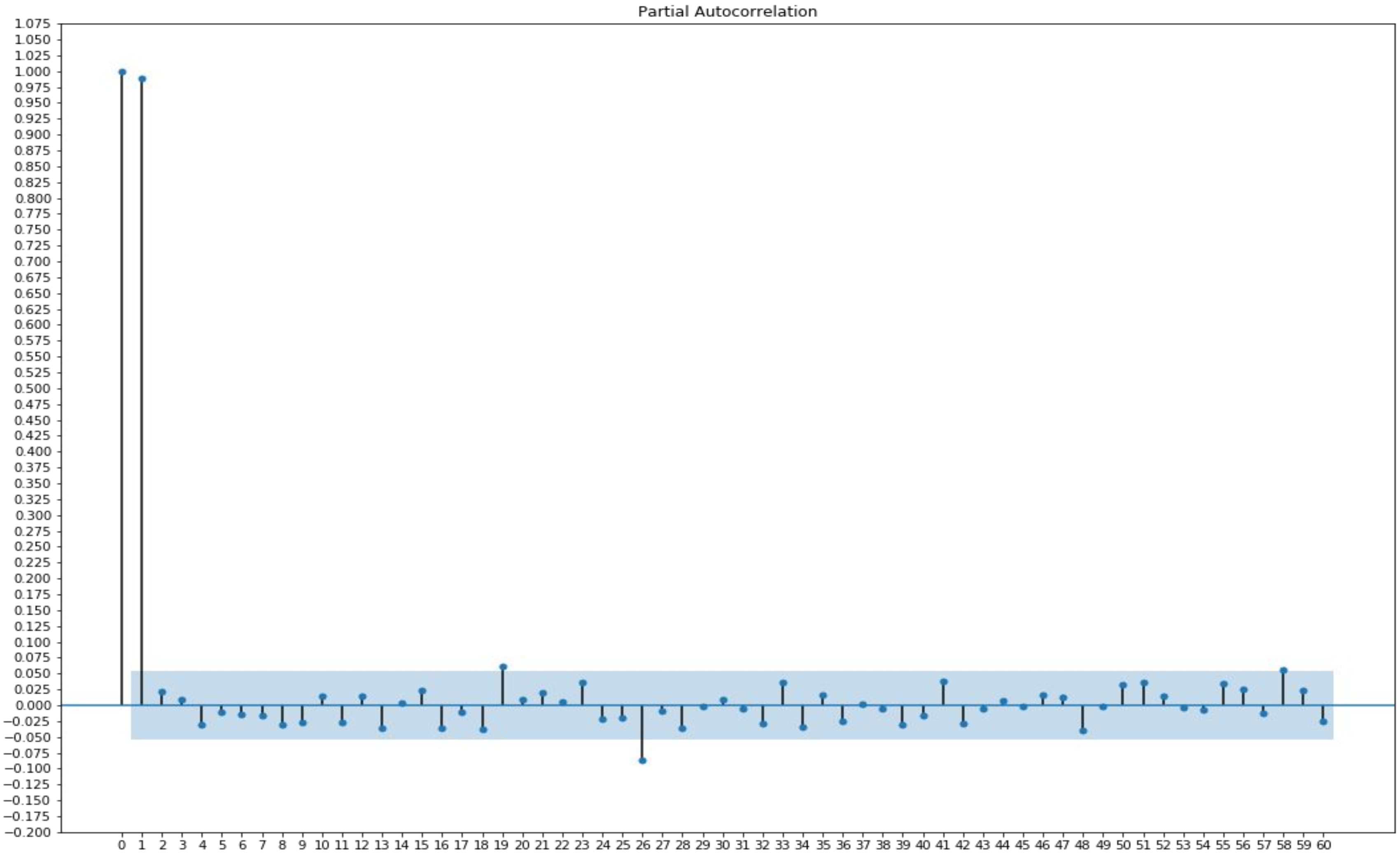


MGNX Returns for Best Sortino Ratio



Autocorrelation of MGNX with 30 Lags




Partial Autocorrelation
1.075
1.050
1.025
1.000
0.975
0.950
0.925
0.900
0.875
0.850
0.825
0.800
0.775
0.750
0.725
0.700
0.675
0.650
0.625
0.600
0.575
0.550
0.525
0.500
0.475
0.450
0.425
0.400
0.375
0.350
0.325
0.300
0.275
0.250
0.225
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
−0.025
−0.050
−0.075
−0.100
−0.125
−0.150
−0.175
−0.200
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60

Autocorrelation and Partial Autocorrelation

- AC is a times series technique used specifically for autoregression models. Given some number of lag variables originating from a univariate time series, you can find the correlation between the lag variable and the original time series. If correlation exists on the lag variable, it may be considered statistically significant.
- PAC is a similar technique that is able to overlook intermediate time steps in the past and give an estimate of correlation between distant lags and the original time series thereby being useful in deciding the order of parameter p in an arima model. P is the number of lags.

Autoregressive Model

Autoregressive Moving Average Model: *ARMA(p,q)*

$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \ldots + \beta_p Y_{t-p}$$

$$+ \varepsilon_t + \theta_1 \varepsilon_{t-1} + \ldots + \theta_q \varepsilon_{t-q}$$

MAPE Scores for all Models

- MAPE Scores:

  LSTM: 0.099

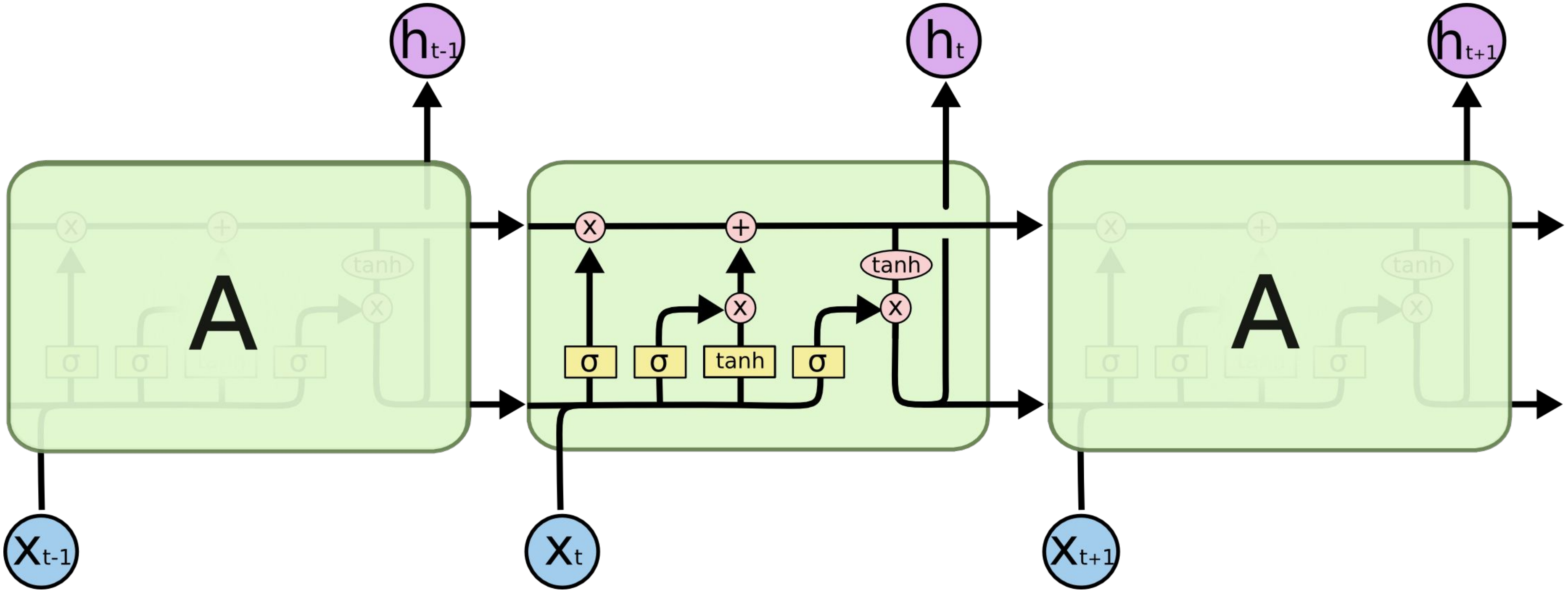
  Auto Arima: 2.38

  Simple Moving Average: 4.37

  FB Prophet: 29.7

  TBATS: 5.97

LSTM Architecture (1)



LSTM Architecture (2)

- The architecture of an LSTM is special.
- It is an answer to the vanishing gradient problem.
- It contains a chain of cells, each with three gates regulating inputs based on weight outcomes. These gates are called the input, forget and output gates. These gates control the flow of information and activation functions such as the sigmoid and tanh function ensure that the scale of information is manageable so that the gradients do not explode.
- These are the basic ingredients governing LSTM model capacity to generate weights that generalize well for prediction.
- This may explain why LTSM outperforms autoregressive models in a short term time series.