Machine Learning Model for Stock Price Prediction

2020. 03. 19.

Seonkyu Kim, Gimok Kim,
Sol-yi Park, Sangjeong Lee, Wonjin Joo

Content

- Research Purpose
- Methods
- Related Indicators
- Stock Price Prediction
- Results

Researh Purpose

- To predict the current market price based on past data in the stock market.

- To utilize machine learning for actual stock investment through more objective analysis.

Methods

① Stock Data Collection

② Prediction Model Building

③ Model Comparison and Evaluation

Time	2010.01.04~2018.12.28
Training Data	2010.01.04~2017.03.06
Test Data	2017.03.07~2018.12.28

Related Indicators

Reference: Zhou, Xingyu, et al. "Stock market prediction on high-frequency data using generative adversarial nets." *Mathematical Problems in Engineering* 2018 (2018).

Related Indicators

- **Bollinger Bands** (HBB(High), MBB(Middle), LBB(Low)):
  - An indicator that shows where the stock price is based on the 20-day moving average line.
- **DMI**: Directional Movement Index,
  (PDI(+DI: Stock Price Increase), MDI(-DI: Stock Price Decrease), ADX)
- **Indicator Moving Average**
- **Stochastic Index**(KDJ_K, KDJ_D, KDJ_J)
  - %K : An indicator that shows where today's closing price is located within the 12-day high and low price range.
  - %D(Slow %K): 5-day moving average of %K
  - Slow %D: 5-day moving average of %D
- **MACD**: Auxiliary indicator that identifies stock price trends through the difference between the short-term and the long-term moving average.
- **RSI**: An indicator that shows the relative strength of upward and downward price pressure.

출처: Zhou, Xingyu, et al. "Stock market prediction on high-frequency data using generative adversarial nets." *Mathematical Problems in Engineering* 2018 (2018).

Stock Price Prediction

| Data |

2010.01.04~2018.12.28

Training Data 80%
Testing Data 20%

| Classification & Prediction Models |

SVM

Logistic Regression

Gradient Boosting

Naïve Bayes

Decision Tree

KNN

LSTM

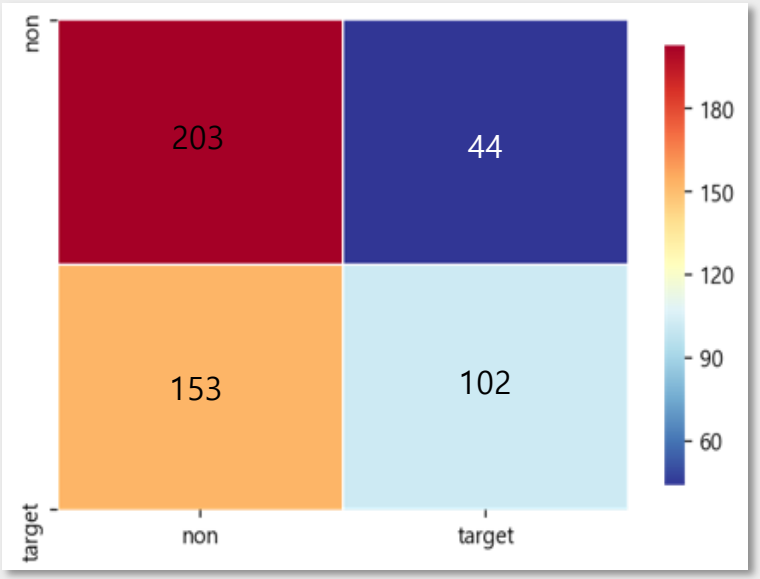
ARIMA

Stock Price Rise or Fall Prediction

Logistic Regression

Logistic	precision	recall	f1-score	support
0 [Stock Price Fall]	0.57	0.82	0.67	247
1 [Stock Price Rise]	0.7	0.4	0.51	255
accuracy			0.61	502
macro avg	0.63	0.61	0.59	502
weighted avg	0.64	0.61	0.59	502

Confusion Matrix

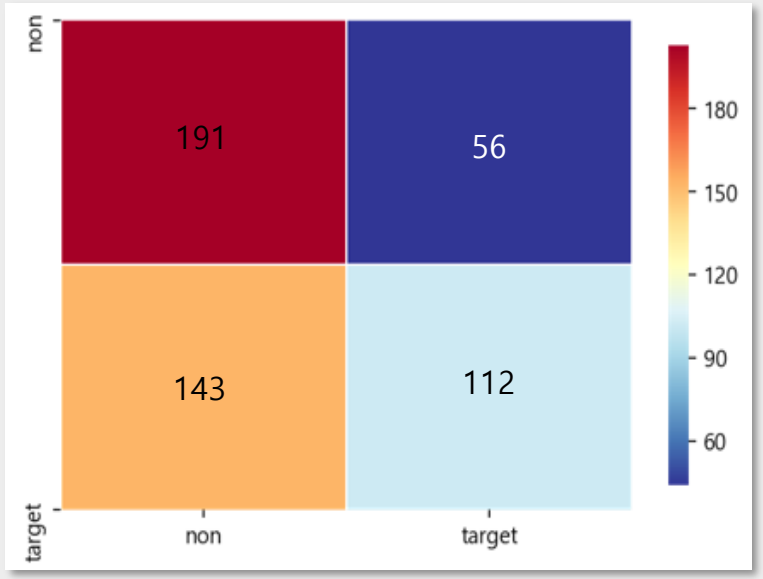


- macro : average
- weighted : weighted average by the number of samples belonging to each class
- accuracy : the ratio of the number of times

Gradient Boosting

Gradient	precision	recall	f1-score	support
0 [Stock Price Fall]	0.57	0.77	0.66	247
1 [Stock Price Rise]	0.67	0.44	0.53	255
accuracy			0.60	502
macro avg	0.62	0.61	0.59	502
weighted avg	0.62	0.60	0.59	502

Confusion Matrix

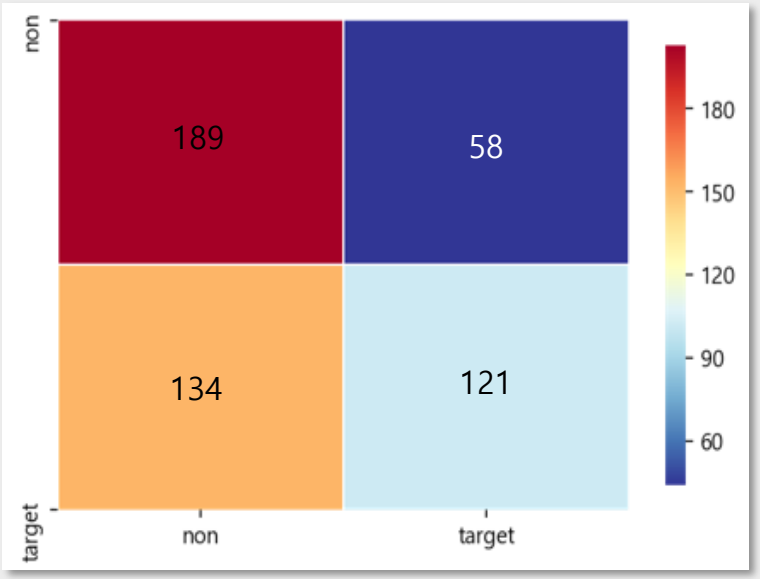


- macro : average
- weighted : weighted average by the number of samples belonging to each class
- accuracy : the ratio of the number of times

Support Vector Machine (SVM)

SVC	precision	recall	f1-score	support
0 [Stock Price Fall]	0.59	0.77	0.66	247
1 [Stock Price Rise]	0.68	0.47	0.56	255
accuracy			0.62	502
macro avg	0.63	0.62	0.61	502
weighted avg	0.63	0.62	0.61	502

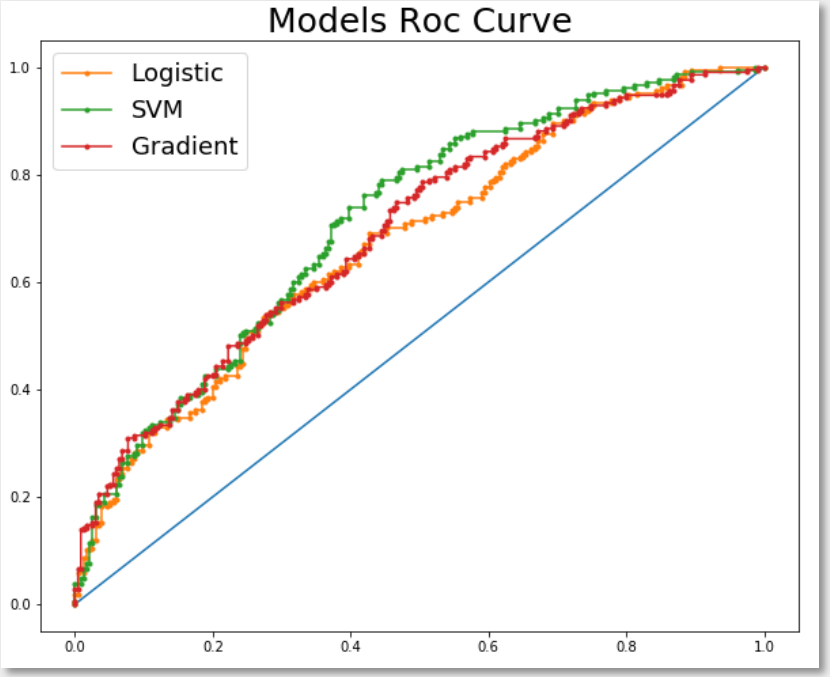
Confusion Matrix



- macro : average
- weighted : weighted average by the number of samples belonging to each class
- accuracy : the ratio of the number of times

Results of All Models

	accuracy	precision	recall	f1-score	AUC sroce
SVM	0.62	0.68	0.47	0.56	0.62
Logistic	0.61	0.7	0.4	0.509	0.61
Gradient	0.6	0.67	0.44	0.53	0.61
NaiveBayes	0.59	0.7	0.36	0.48	0.6
DecisionTree	0.57	0.61	0.47	0.53	0.58
RandomForest	0.55	0.61	0.35	0.44	0.56
KNN	0.54	0.57	0.41	0.48	0.54

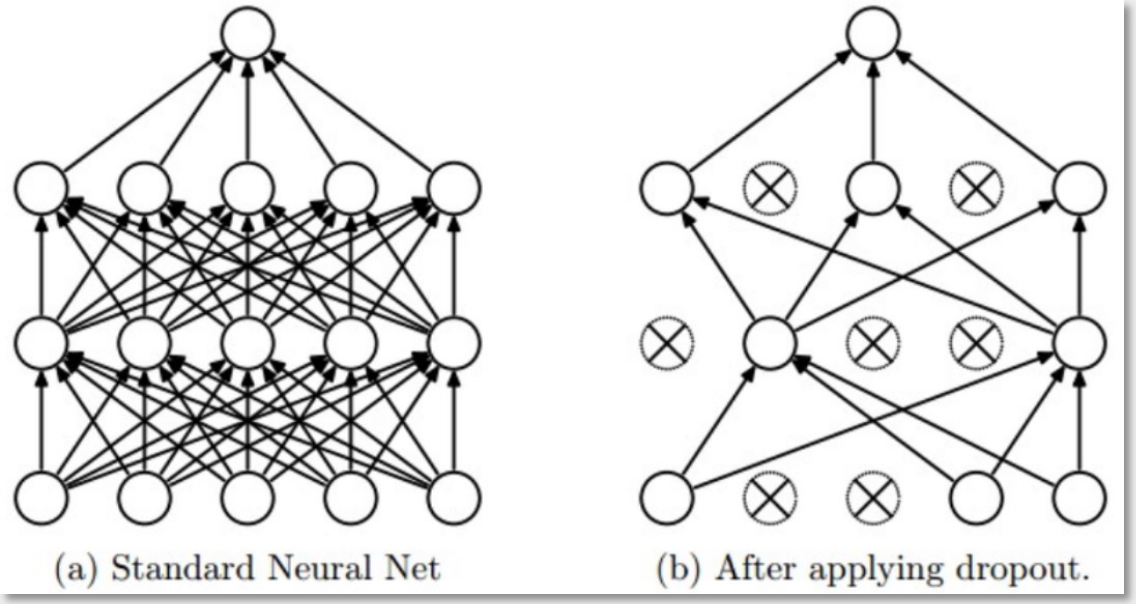


SVM > Logistic > Gradient

Stock Price Prediction (LSTM)

Long Short Term Memory (LSTM)

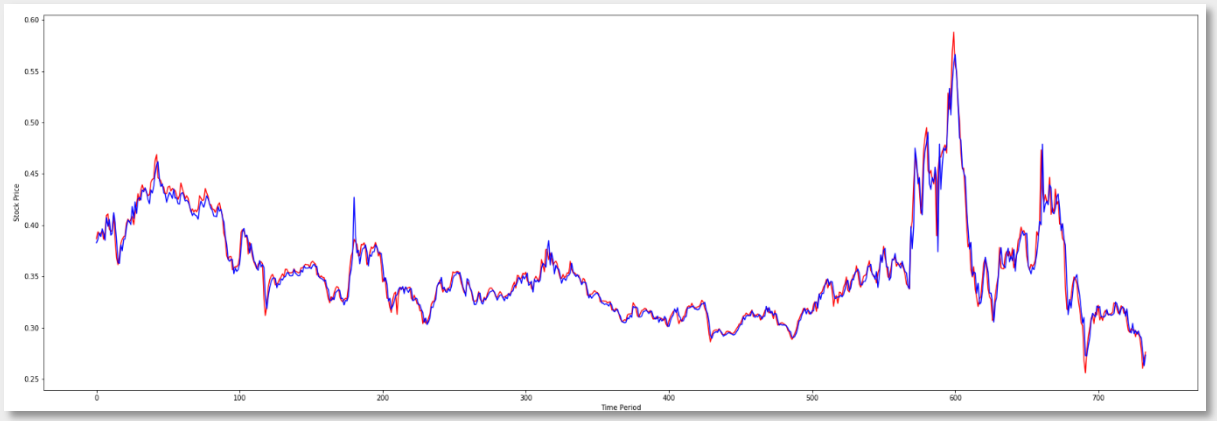
- ✓ LSTM model is used because the stock price is time series data.
- ✓ In general, as the number of hidden layers in a Neural Network increases, the learning ability improves, but the possibility of overfitting increases.



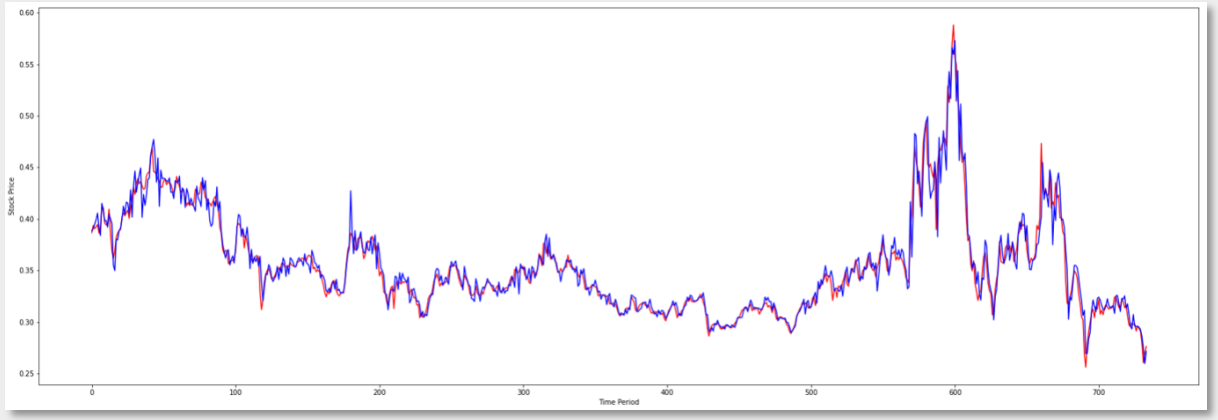
- ✓ Dropout: Regularization technology to reduce overfitting.
- ✓ Temporarily excludes a unit from the network and disconnects all excluded units.
- ✓ keep_prob: Probability of maintaining a given unit. In other words, the probability of not dropping.

keep_prob Comparison

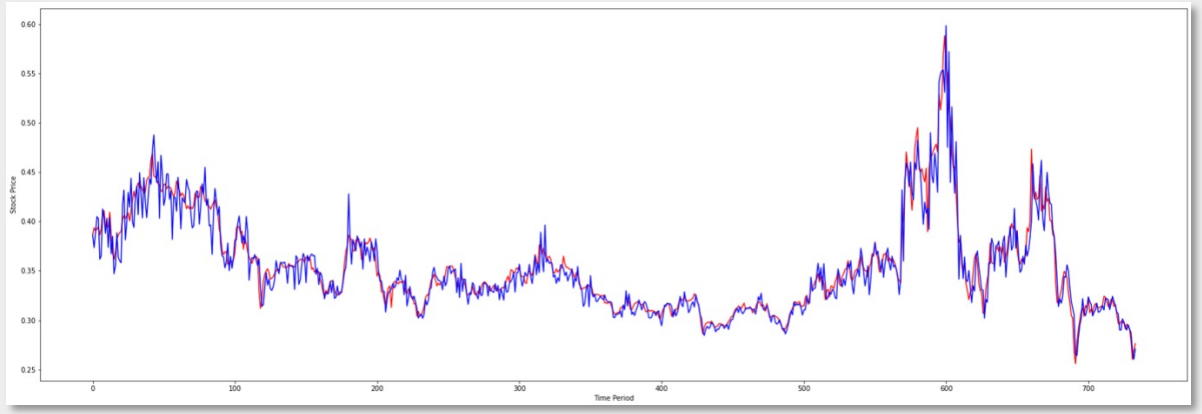
- keep_prob = 1.0, softsign



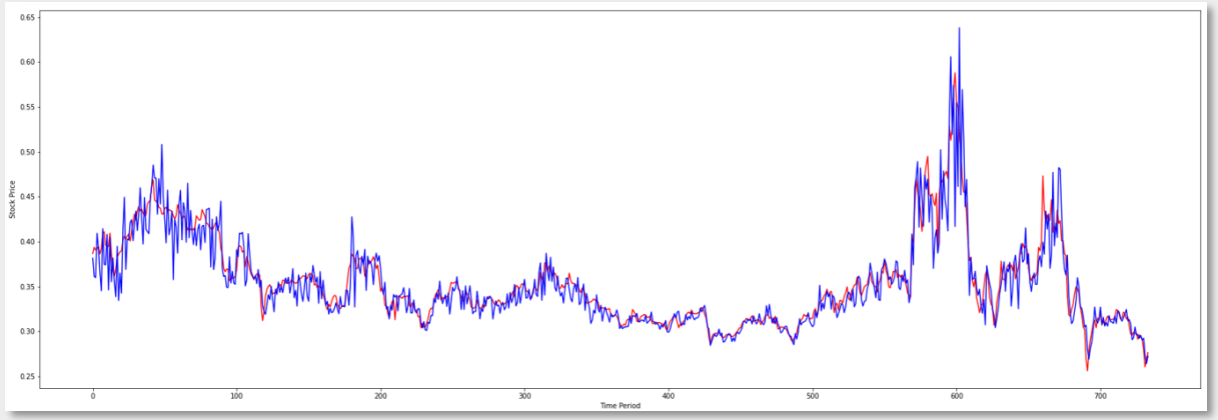
- keep_prob = 0.9, softsign



- keep_prob = 0.7, softsign

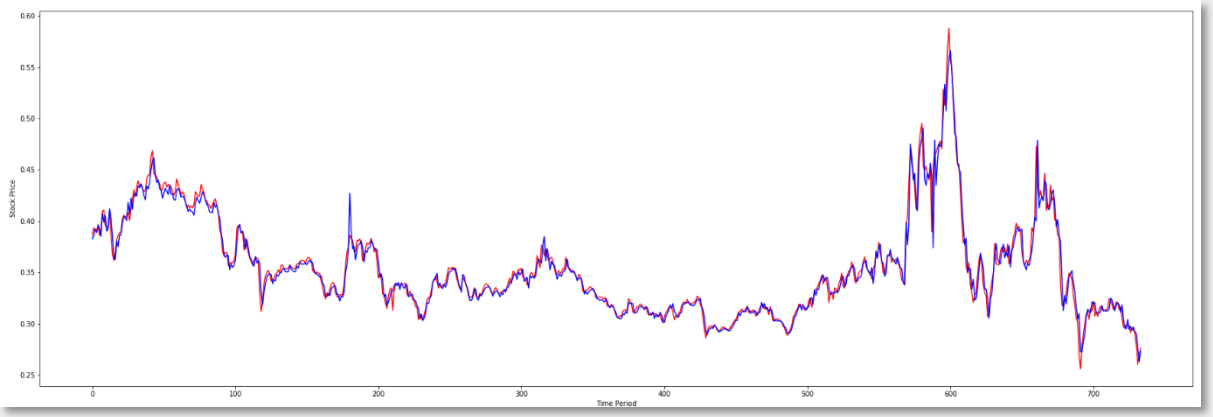


- keep_prob = 0.5, softsign

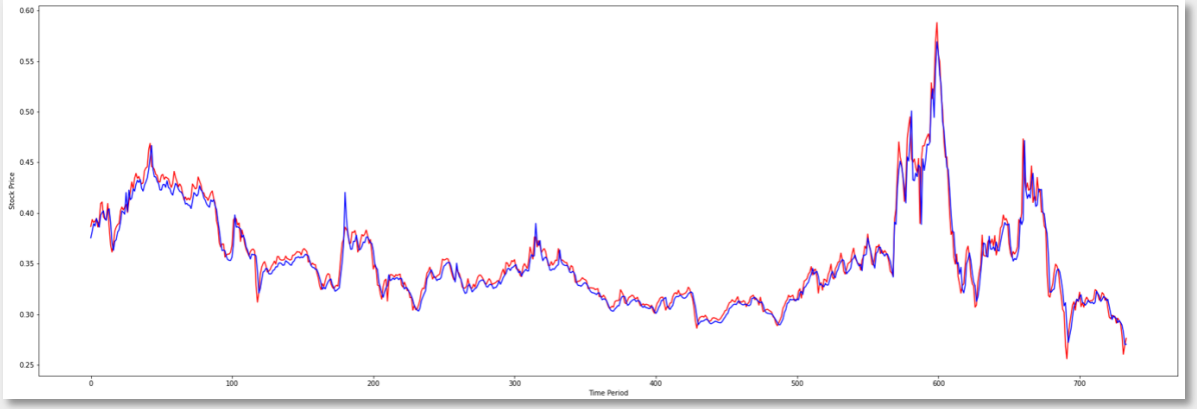


Activation Function Comparison

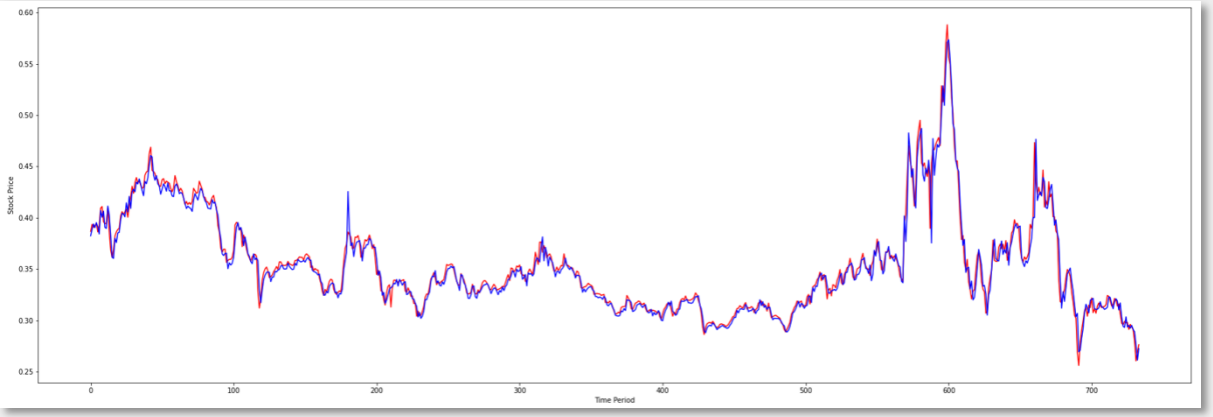
- keep_prob = 1.0, softsign



- keep_prob = 1.0, relu

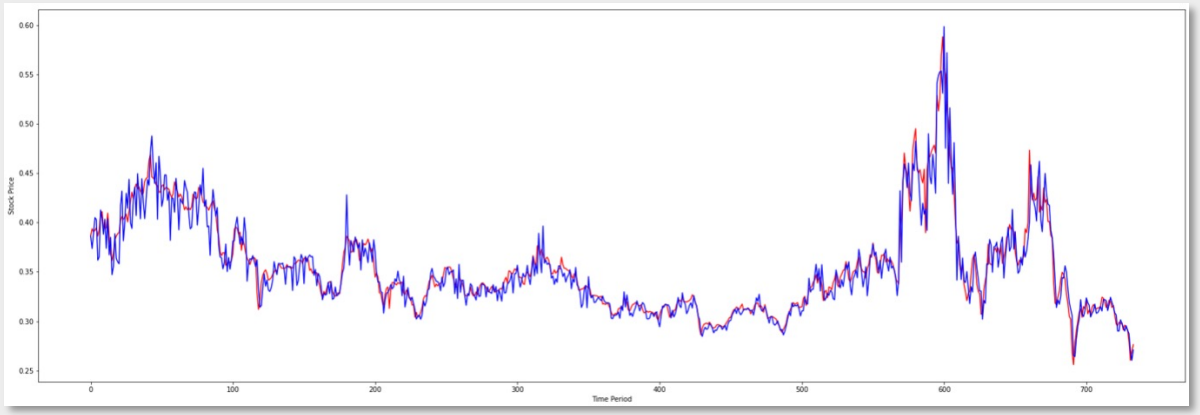


- keep_prob = 1.0, tanh

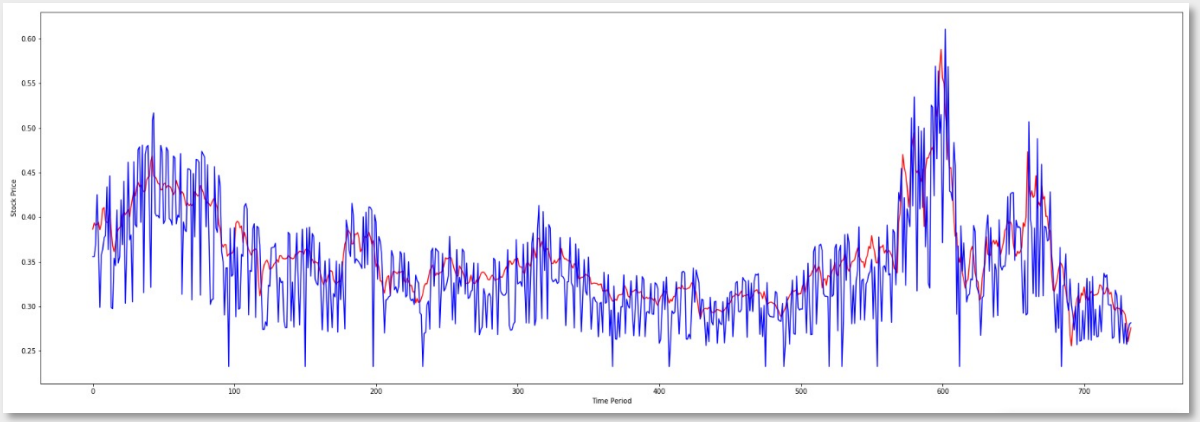


Activation Function Comparison

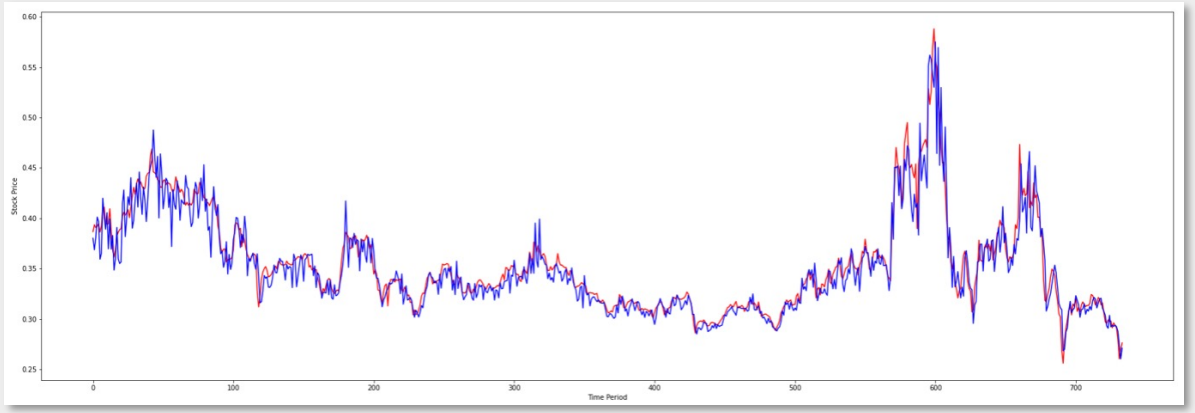
- keep_prob = 0.7, softsign



- keep_prob = 0.7, relu

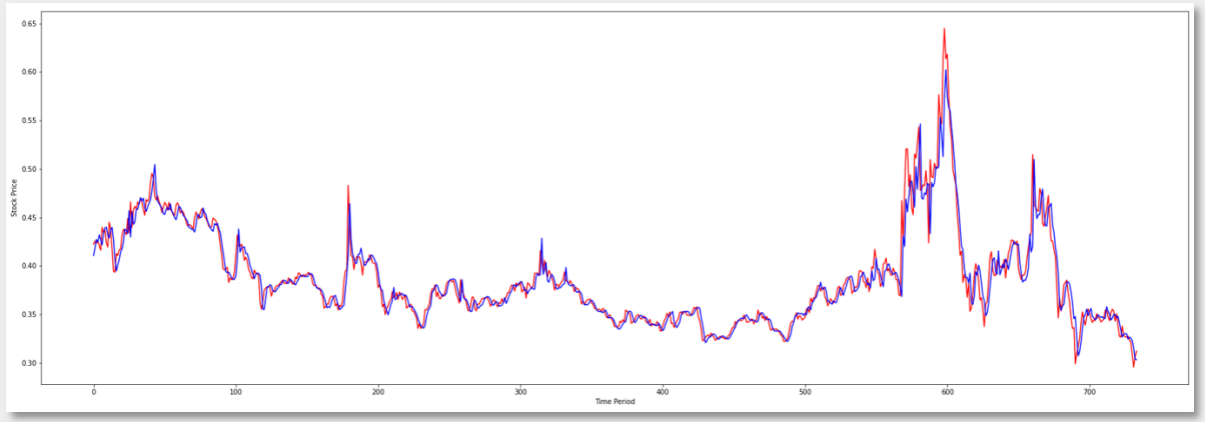


- keep_prob = 0.7, tanh

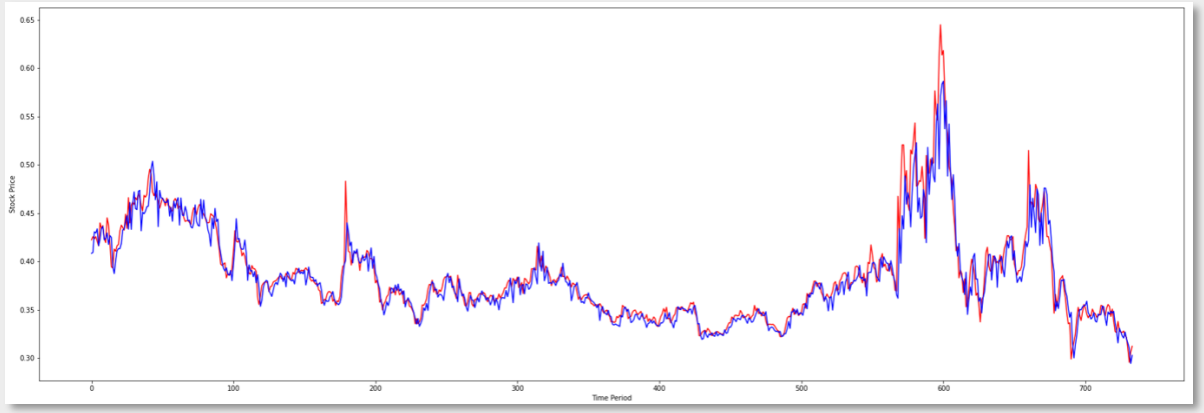


keep_prob Comparison (Add Variables)

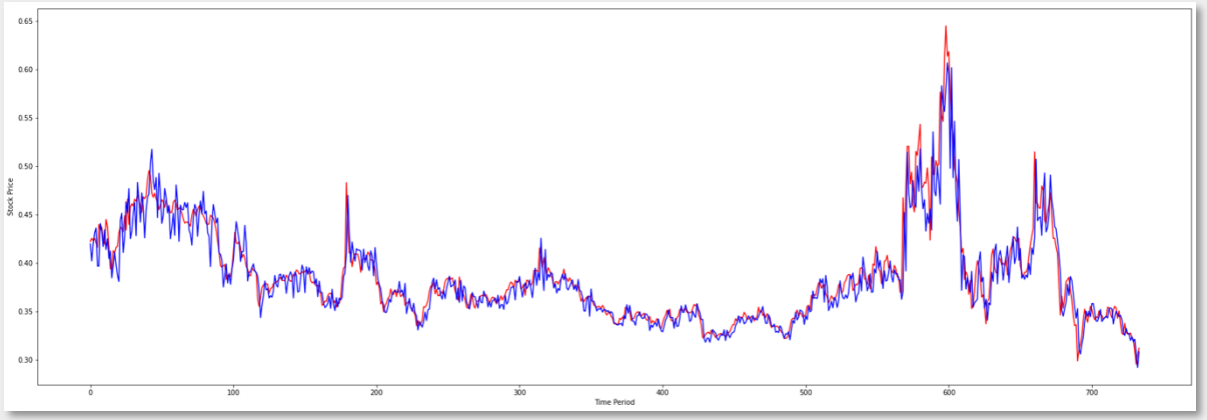
- keep_prob = 1.0, softsign



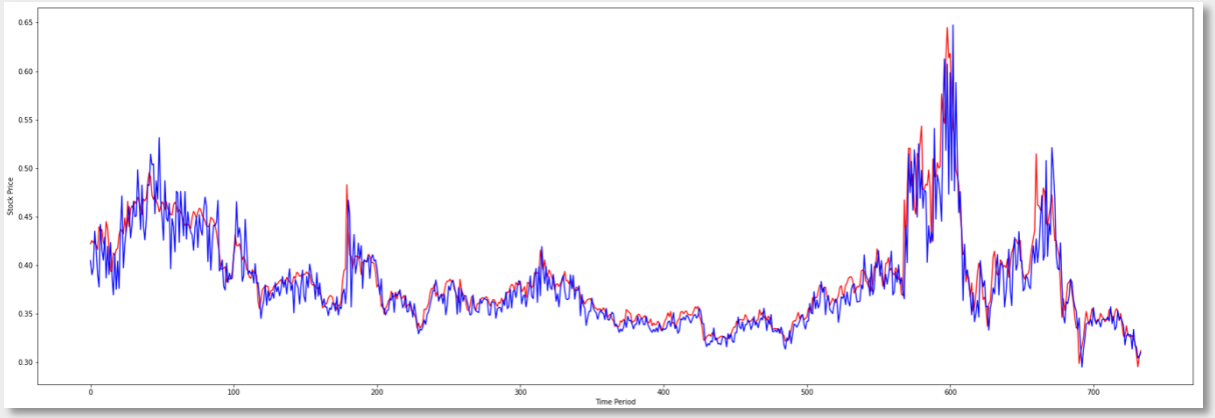
- keep_prob = 0.9, softsign



- keep_prob = 0.7, softsign



- keep_prob = 0.5, softsign



Results

Before adding variables

keep_prob	activation function	rmse	predicted value	real value
1	softsign	0.009	2787	2840
1	relu	0.01	2760	
1	tanh	0.008	2799	
0.9	softsign	0.011	2785	
0.7	softsign	0.014	2850	
0.7	relu	0.044	2711	
0.7	tanh	0.0151	2769	
0.5	softsign	0.0199	2827	

After adding variables

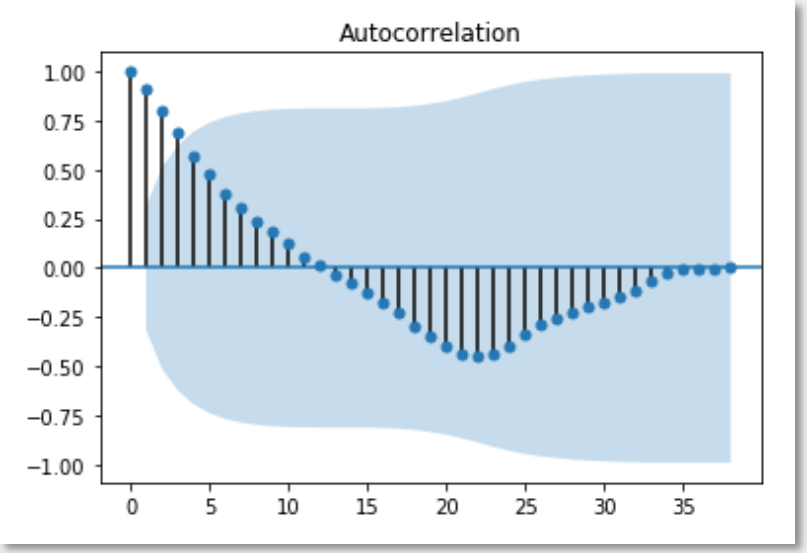
keep_prob	activation function	rmse	predicted value	real value
1	softsign	0.014	2803	2840
1	relu	0.013	2821	
1	tanh	0.014	2817	
0.9	softsign	0.0163	2826	
0.7	softsign	0.0174	2825	
0.7	relu	0.04	2706	
0.7	tanh	0.027	2866	
0.5	softsign	0.022	2893	

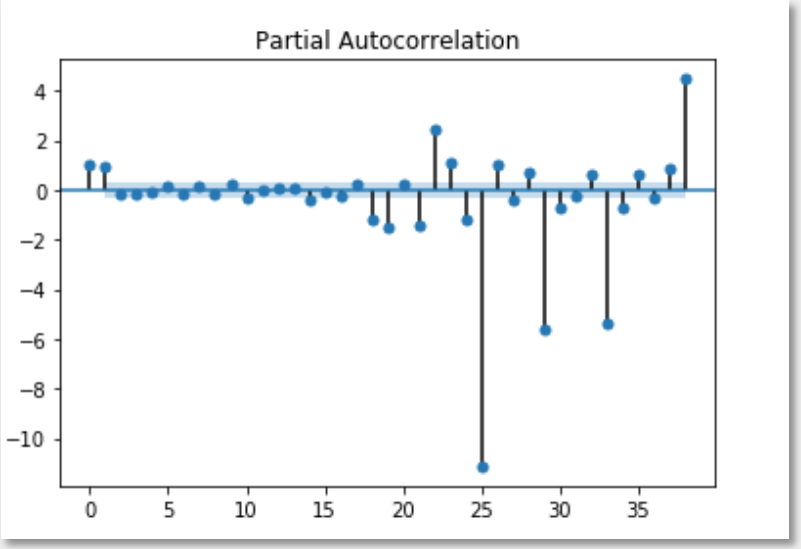
Stock Price Prediction (ARIMA)

ARIMA

Prediction based only on closing price

- **AR** : AutoRegression, a model in which the error term of previous observations affects subsequent observations
- **I** : Intgrated, expression given to time series models that use differences
- **MA** : Moving Average, a model in which observations are influenced by previous continuous error terms





	Date	Adj Close
0	2017-12-01	3230.0
1	2017-12-04	3185.0
2	2017-12-05	3230.0
3	2017-12-06	3205.0
4	2017-12-07	3130.0
5	2017-12-08	3080.0
6	2017-12-11	3095.0
7	2017-12-12	3095.0
8	2017-12-13	3090.0
9	2017-12-14	3095.0
10	2017-12-15	3080.0
11	2017-12-18	3055.0
12	2017-12-19	3030.0
13	2017-12-21	2960.0
14	2017-12-22	2960.0

ARIMA_RMSE

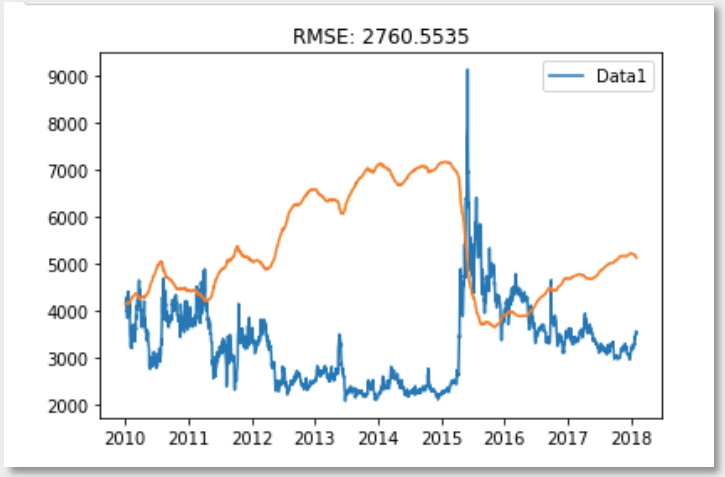
10.01.01. ~ 18.01.31. → 2760

16.01.01. ~ 18.01.31. → 423

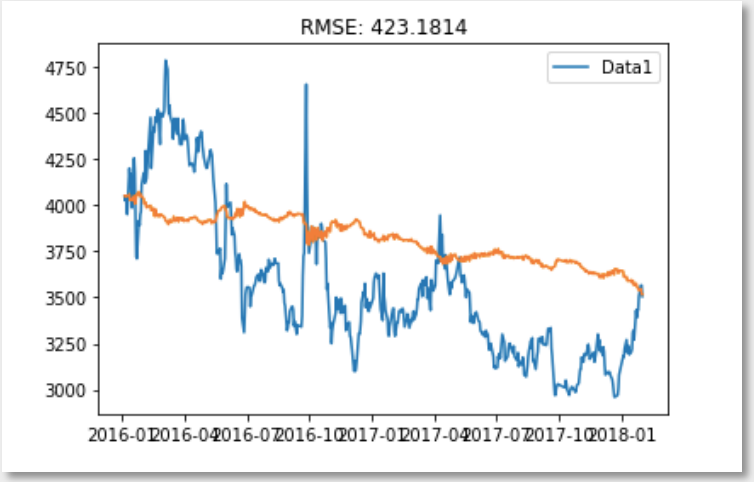
17.01.01. ~ 18.01.31. → 340

17.12.01. ~ 18.01.31. → 142

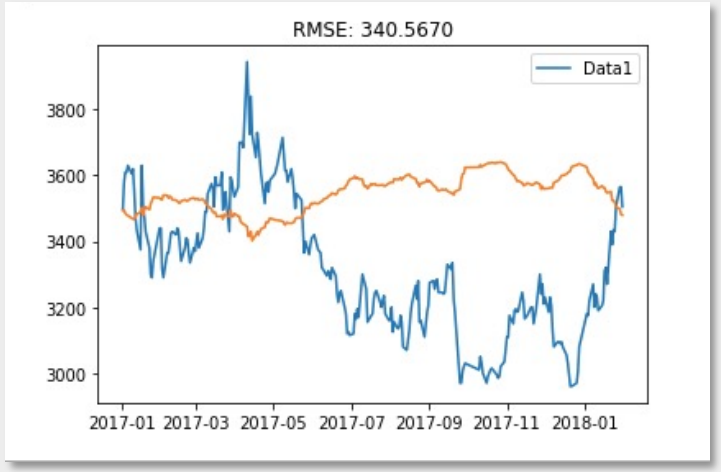
: As the period decreases, RMSE gets smaller.



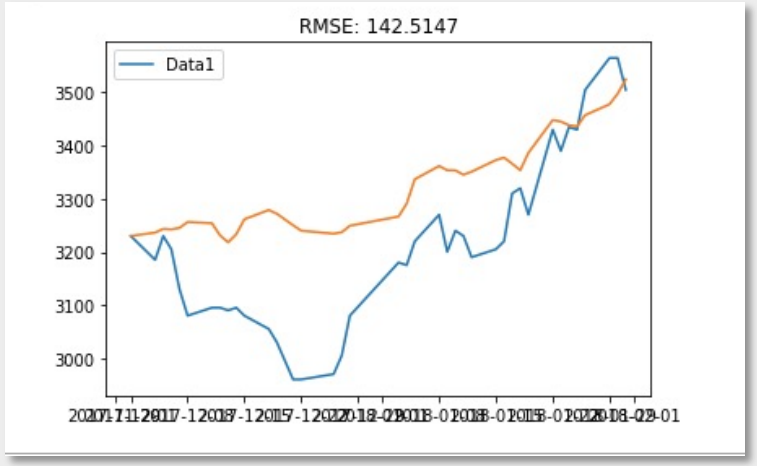
RMSE: 2760.5535



RMSE: 423.1814

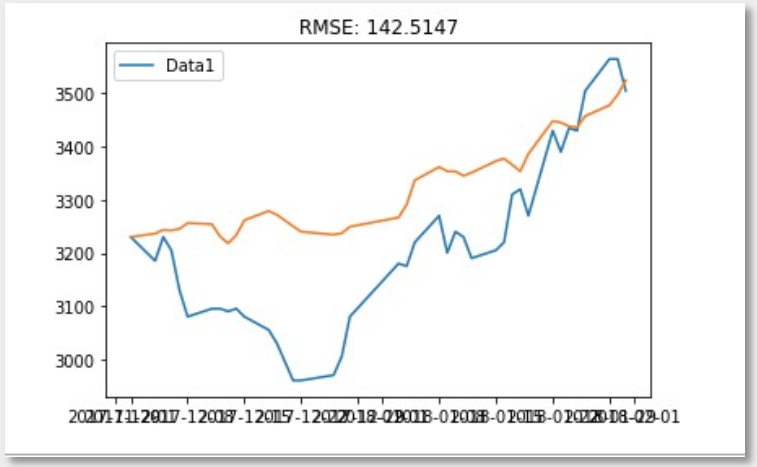


RMSE: 340.5670



RMSE: 142.5147

ARIMA_RMSE



RMSE: 142.5147

→ 20171201 ~ 20180131 기간의 종가(Close)를 학습시킴

In [39]:
```
# ARIMA(3,1,2)
model = ARIMA(series, order = (3,1,2))
model_fit = model.fit(trend='nc', full_output = True, disp = 1)
print(model_fit.summary())
```

ARIMA Model Results

Dep. Variable:	D.Adj Close	No. Observations:	38
Model:	ARIMA(3, 1, 2)	Log Likelihood	-201.288
Method:	css-mle	S.D. of innovations	45.934
Date:	Thu, 19 Mar 2020	AIC	414.577
Time:	02:17:26	BIC	424.402
Sample:	1	HQIC	418.072

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1.D.Adj Close	0.7457	0.279	2.675	0.012	0.199	1.292
ar.L2.D.Adj Close	-0.6473	0.243	-2.666	0.012	-1.123	-0.171
ar.L3.D.Adj Close	0.1136	0.231	0.491	0.627	-0.340	0.567
ma.L1.D.Adj Close	-0.8788	0.538	-1.635	0.112	-1.932	0.175
ma.L2.D.Adj Close	1.0000	1.055	0.948	0.350	-1.067	3.067

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	0.4991	-1.2743j	1.3686	-0.1906
AR.2	0.4991	+1.2743j	1.3686	0.1906
AR.3	4.7009	-0.0000j	4.7009	-0.0000
MA.1	0.4394	-0.8983j	1.0000	-0.1776
MA.2	0.4394	+0.8983j	1.0000	0.1776

ARIMA_Results

Predict Closing Price of

2018.02.01.

```
In [19]: fore = model_fit.forecast(steps=1)
         print(fore)
         # 예측값, stderr, upper bound, lower bound

(array([3528.58285722]), array([48.94399406]), array([[3432.6543916 , 3624.51132284]]))
```

Date	Closing Price
2018.02.01	3,540

Predict Closing Price of

2020.03.19.

```
In [99]: fore = model_fit.forecast(steps=1)
         print(fore)
         # 예측값, stderr, upper bound, lower bound

(array([5928.18713066]), array([733.87445142]), array([[4489.8196367 , 7366.55462462]]))
```

(Reference : NAVER Finance)