Stock Prices Prediction using traditional and machine learning methods

ARIMA, PCA+DNN

Grzegorz Pielot Damian Burczyk

Koło Naukowe Finansów Obliczeniowych MIMUW, 2019

Table of Contents

ARIMA

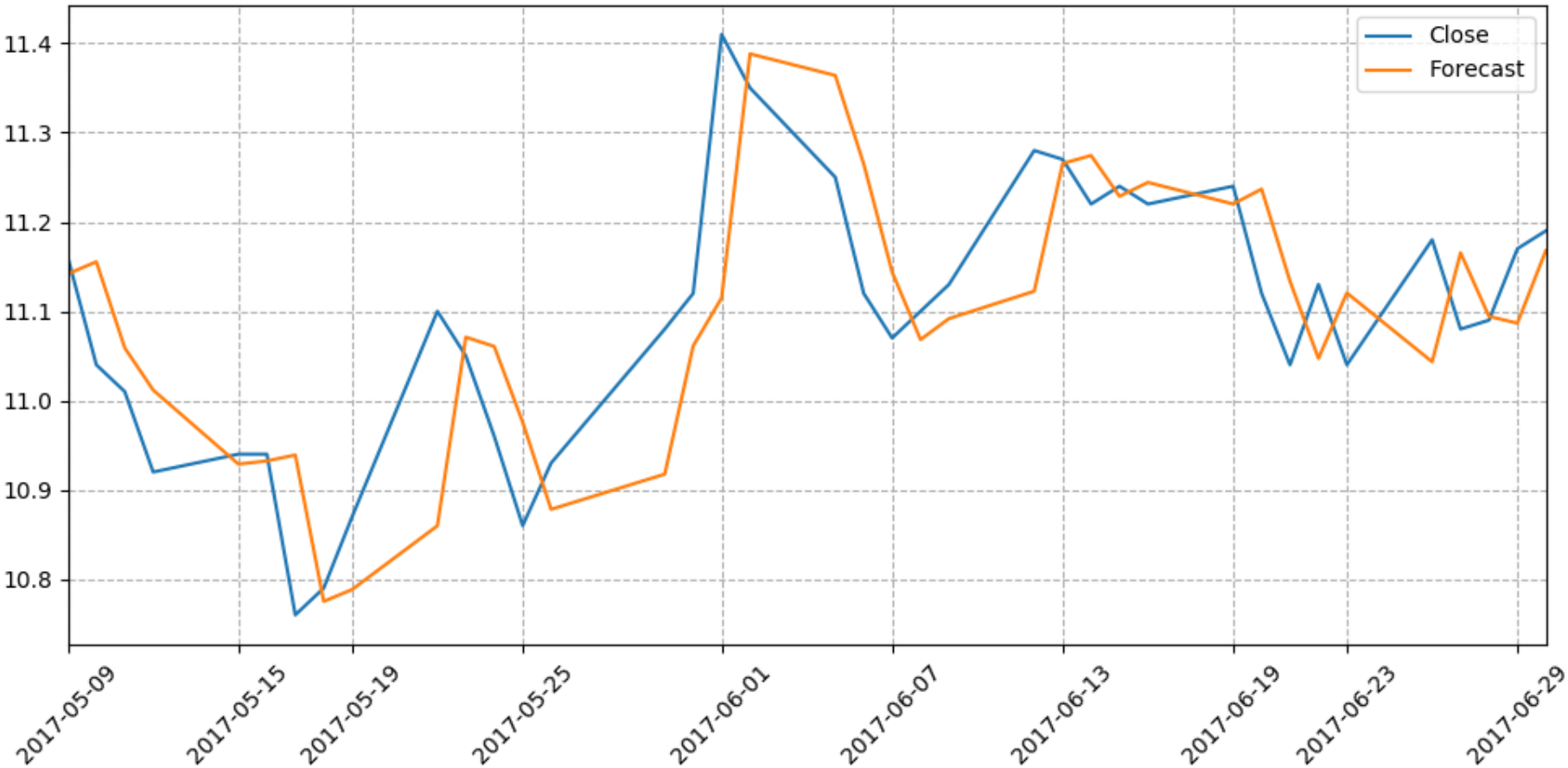
Avaliable packages:

- statsmodels
  statsmodels.tsa.arima_model.ARIMA(endog, order, exog=None, dates=None, freq=None, missing='none')
  - p - order (number of time lags) of the autoregressive model
  - d - degree of differencing
  - q - order of the moving-average model

ARIMA(5,1,0), test ratio = 20%, predicting from Close

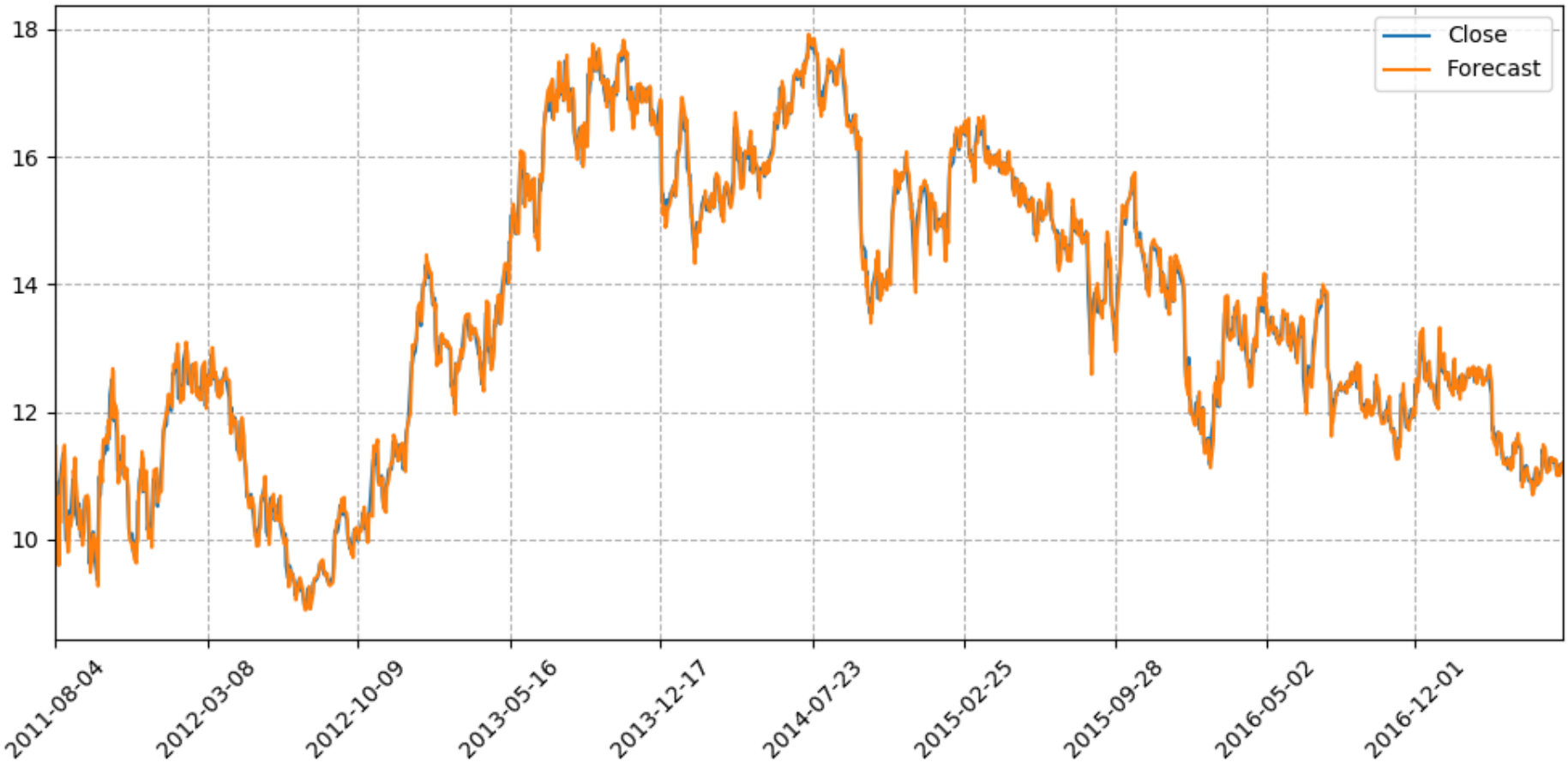
Mean Square Error = 0.0465

ARIMA(5,1,0), test ratio = 0.5%, predicting from Close



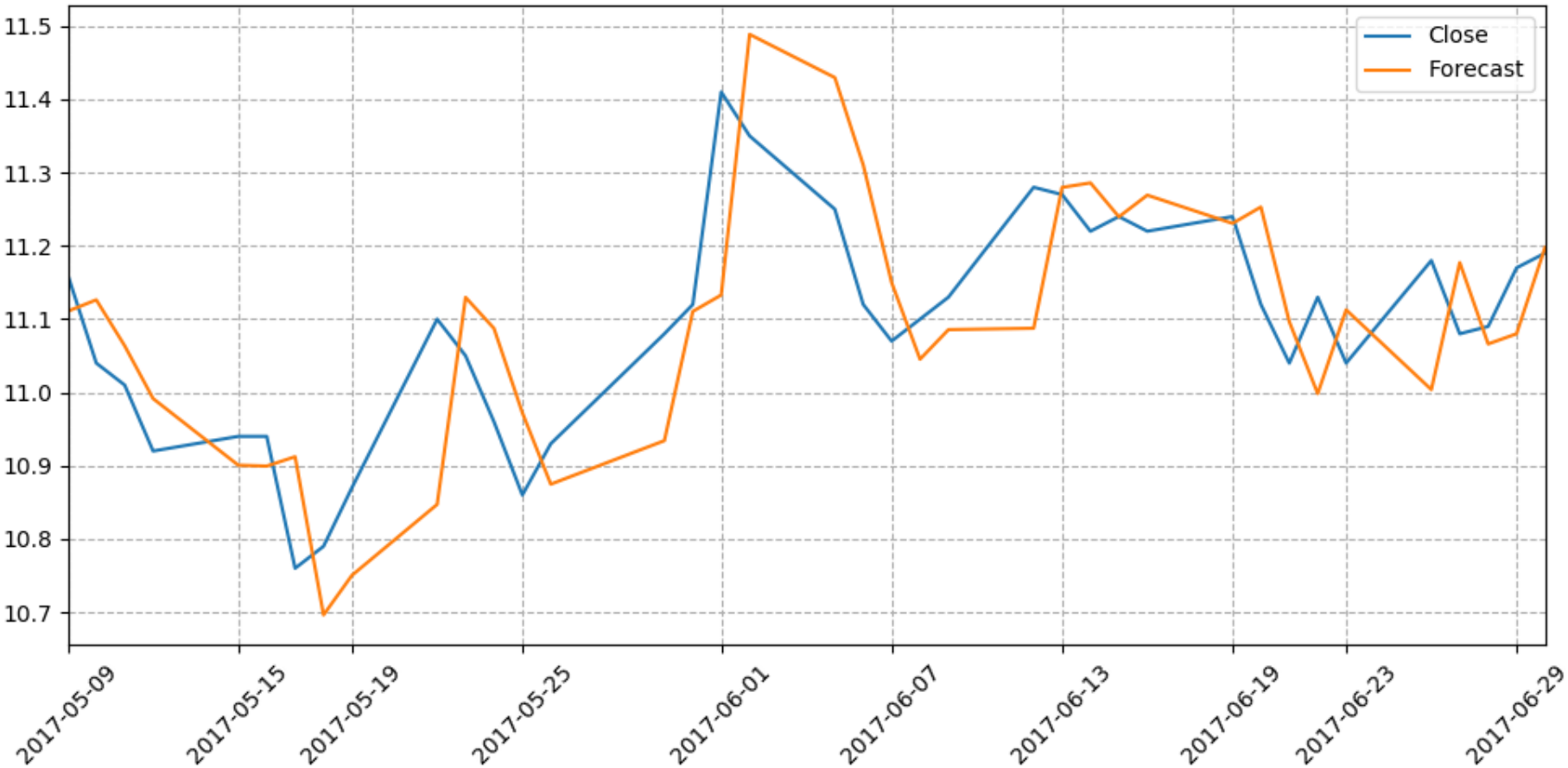
Mean Square Error = 0.0107

ARIMA(5,1,0), test ratio = 20%, predicting from ROI



Mean Square Error = 0.0532

ARIMA(5,1,0), test ratio = 0.5%, predicting from ROI

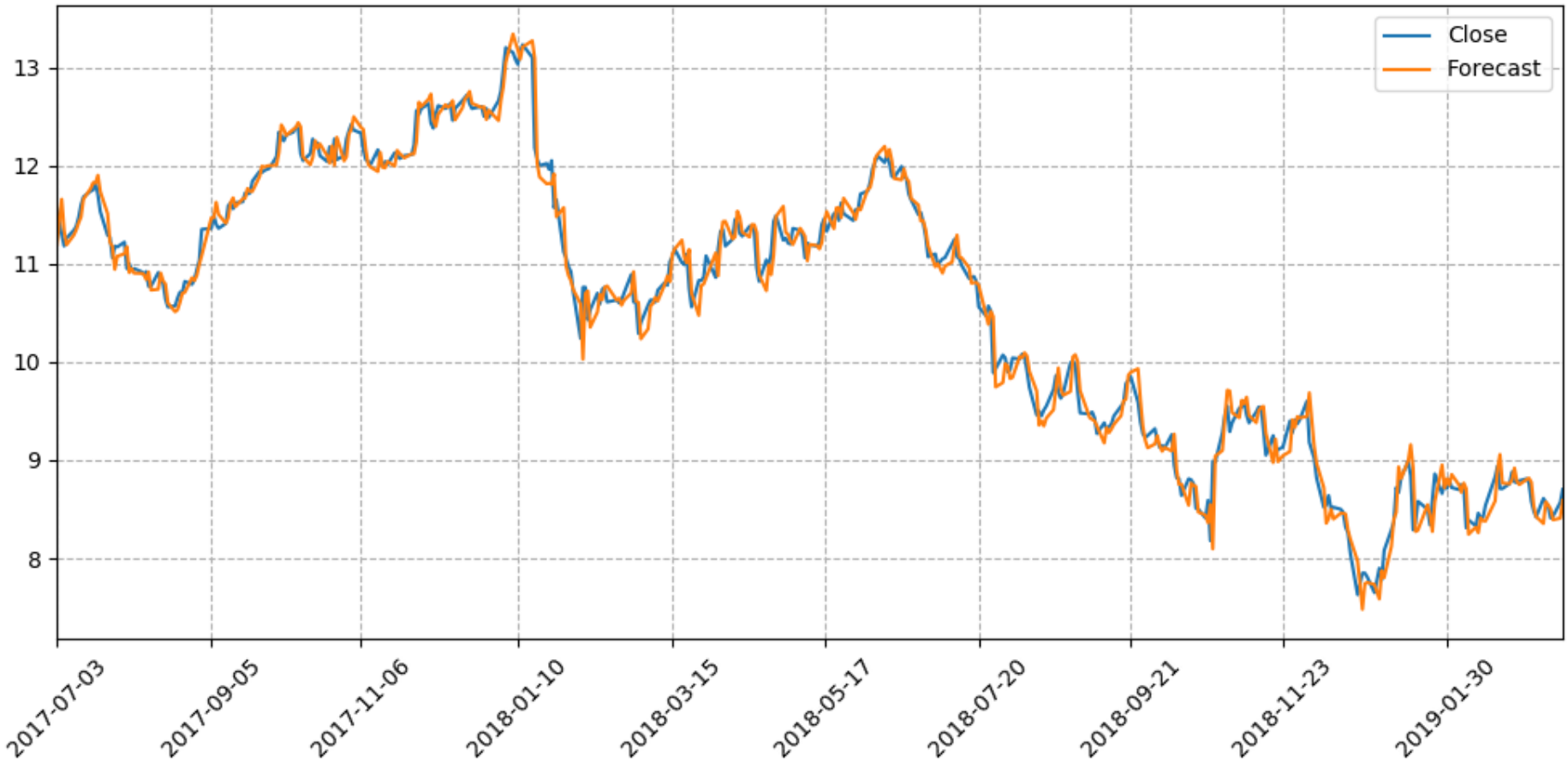


Mean Square Error = 0.0132

ARIMA(5,1,0), official tests, predicting from Close

Mean Square Error = 0.0260

ARIMA(5,1,0), official tests, predicting from ROI



Mean Square Error = 0.0296

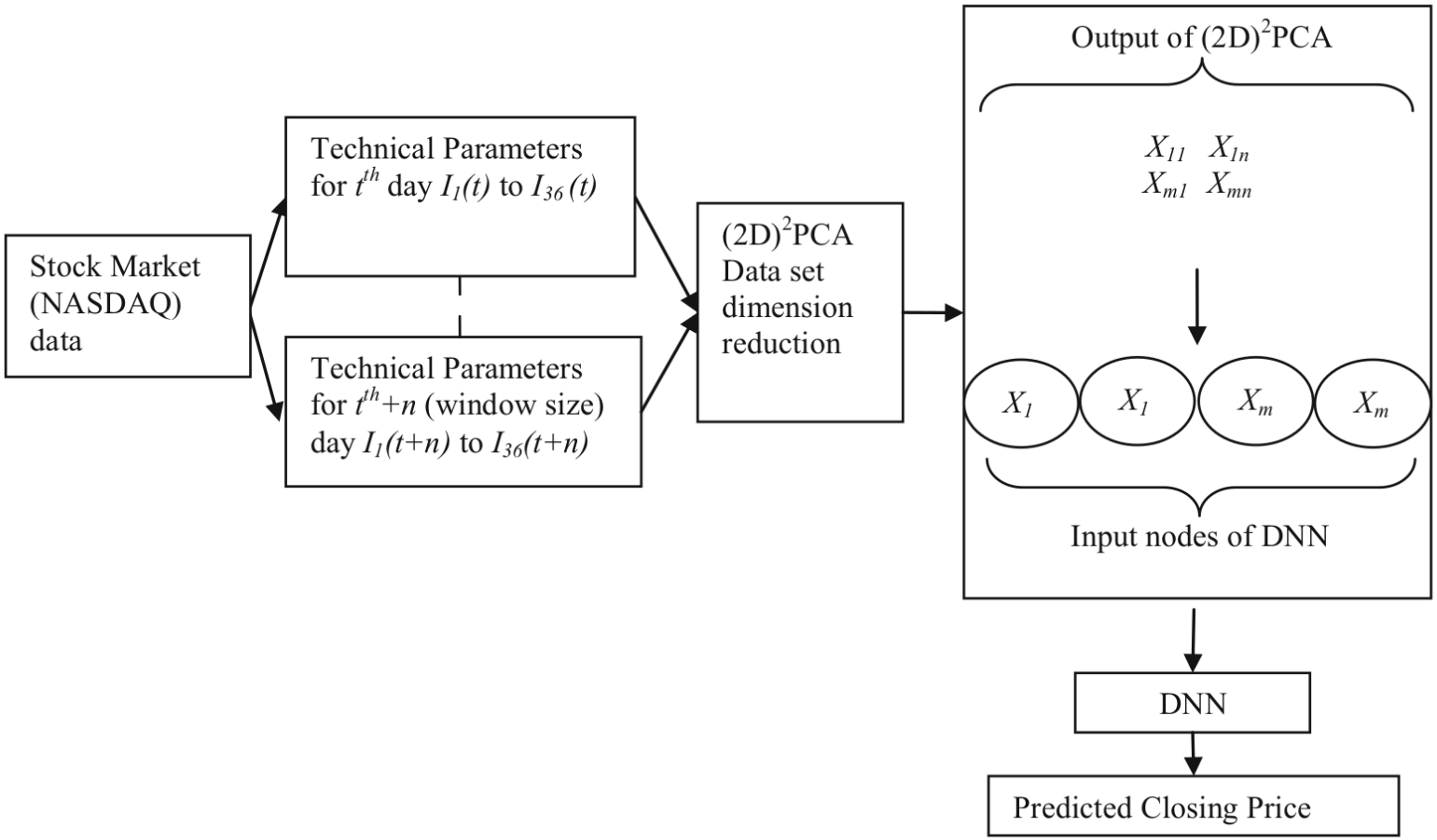
ARIMA other orders

Mean square error with (1, 0, 0) is 0.0181
Mean square error with (1, 1, 0) is 0.0185
Mean square error with (0, 1, 1) is 0.0185
Mean square error with (1, 2, 1) is 0.0185
Mean square error with (3, 0, 0) is 0.0185
Mean square error with (3, 1, 1) is 0.0187
Mean square error with (1, 2, 3) is 0.0185
Mean square error with (4, 0, 0) is 0.0186
Mean square error with (5, 1, 0) is 0.0187
Mean square error with (5, 1, 1) is 0.0187

Machine Learning

Based on "Stock prediction using deep learning" by Ritika Singh and Shashi Srivastava

Machine Learning

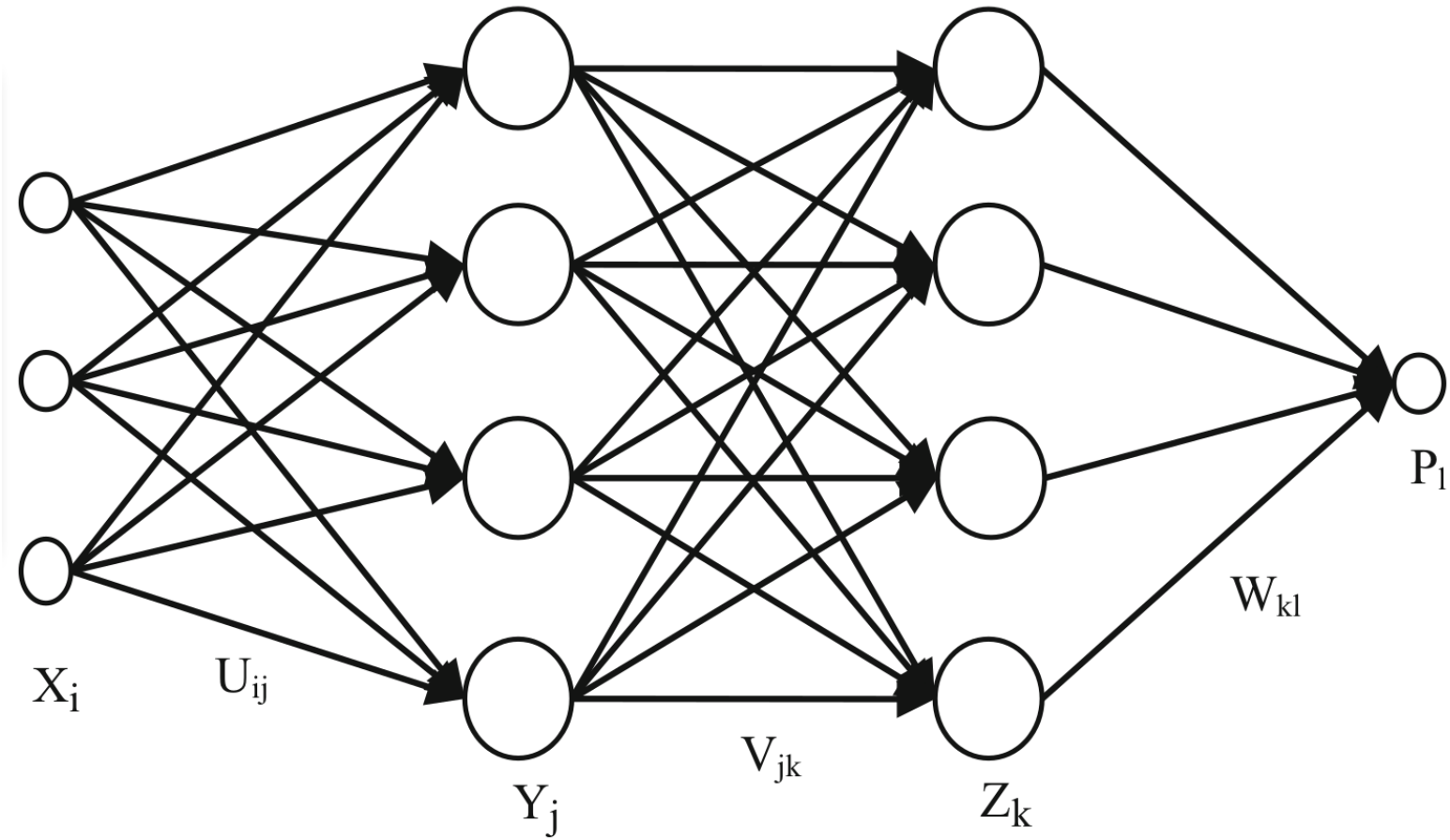


Machine Learning

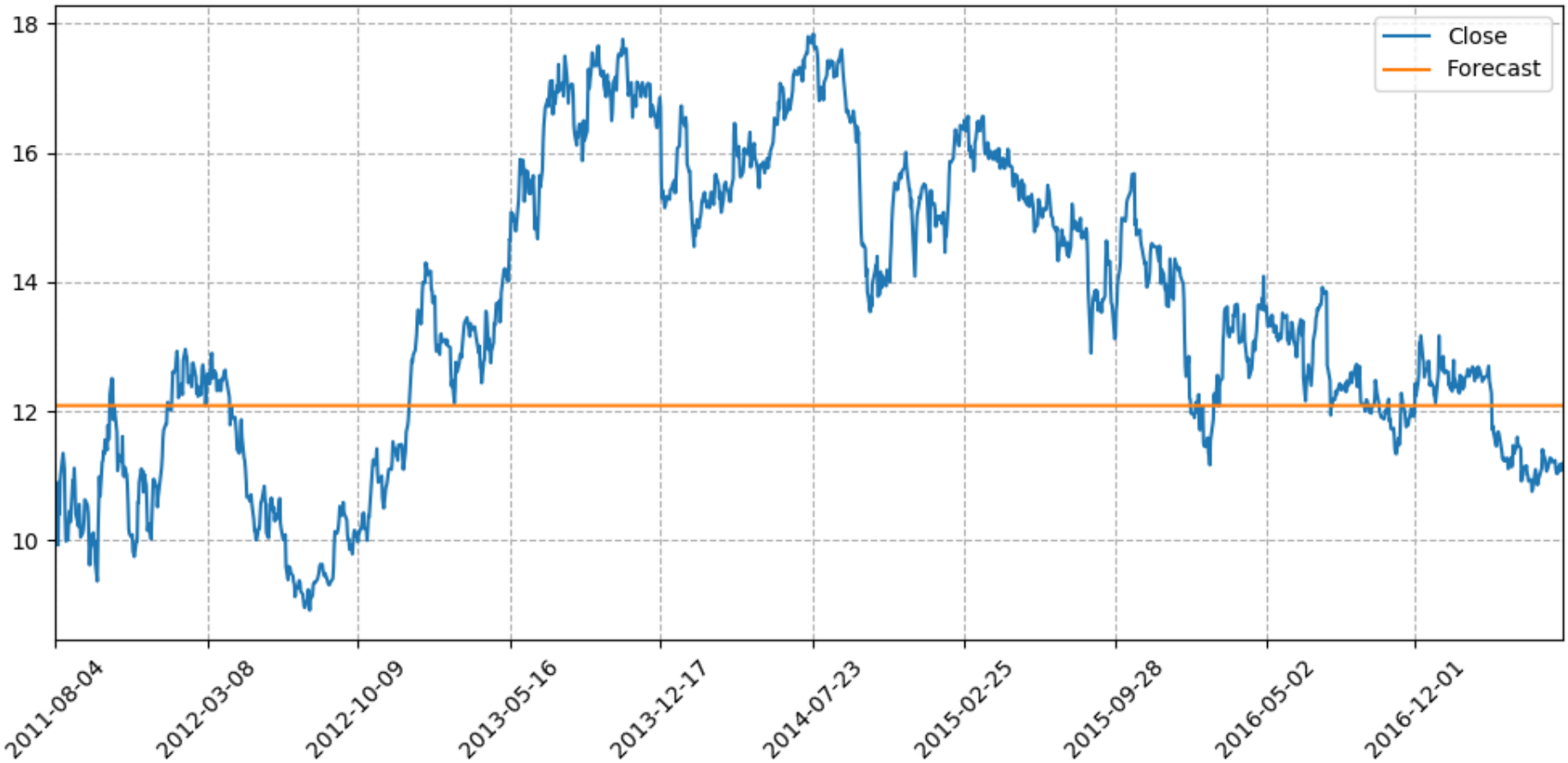
Table 1 Input variables for the stock market data set

Name of the Variable	Description and Formula
$I_1 = x_o(t)$	Open Price
$I_2 = x_h(t)$	High Price
$I_3 = x_l(t)$	Low Price
$I_4 = x(t)$	Close Price
$I_5 = MA5$, $I_6 = MA10$, $I_7 = MA20$	Moving Average
$I_8 = BIAS5$, $I_9 = BIAS10$	BIAS
$I_{10} = DIFF$	$EMA12\text{-}EMA26$
$I_{11} = ,BU$	$(x(t)\text{-}bollinger_{upper})/bollinger_{upper}$
$I_{12} = BL$	$(x(t)\text{-}bollinger_{lower})/bollinger_{lower}$
$I_{13} = K$, $I_{14} = D$	Stochastic Fast %K ,Fast %D
$I_{15} = ROC$	Price rate of change
$I_{16} = TR$	True range of price movements
$I_{17} = MTM6$, $I_{18} = MTM12$	Momentum
$I_{19} = WR\%10$, $I_{20} = WR\%5$	Williams index
$I_{21} = OSC6$, $I_{22} = OSC12$	Oscillator
$I_{23} = RSI6$, $I_{24} = RSI12$	Relative strength index
$I_{25} = PSY$	Psychological line
I_{26}	$K(t)\text{-}K(t\text{-}1)$
I_{27}	$D(t)\text{-}D(t\text{-}1)$
I_{28}	$(x(t)\text{-}x(t\text{-}1))/x(t\text{-}1)$
I_{29}	$(x(t)\text{-}x_o(t))/x_o(t)$
I_{30}	$(x(t)\text{-}x_l(t))/(x_h(t)\text{-}x_l(t))$
I_{31}	$(MA5(t)\text{-}MA5(t\text{-}1))/MA5(t\text{-}1)$
I_{32}	$(MA20(t)\text{-}MA20(t\text{-}1))/MA20(t\text{-}1)$
I_{33}	$(MA5(t)\text{-}MA20(t\text{-}1))/MA20(t\text{-}1)$
I_{34}	$(x(t)\text{-}MA20(t))/MA20(t)$
I_{35}	$(x(t)\text{-}min(x(t\text{-}1),x(t\text{-}2),\ldots,x(t\text{-}N)))/\ min(x(t),x(t\text{-}1),x(t\text{-}2),\ldots,x(t\text{-}N))$
I_{36}	$(x(t)\text{-}max(x(t\text{-}1),x(t\text{-}2),\ldots,x(t\text{-}N)))/\ max(x(t),x(t\text{-}1),x(t\text{-}2),\ldots,x(t\text{-}N))$

Deep Neural Network

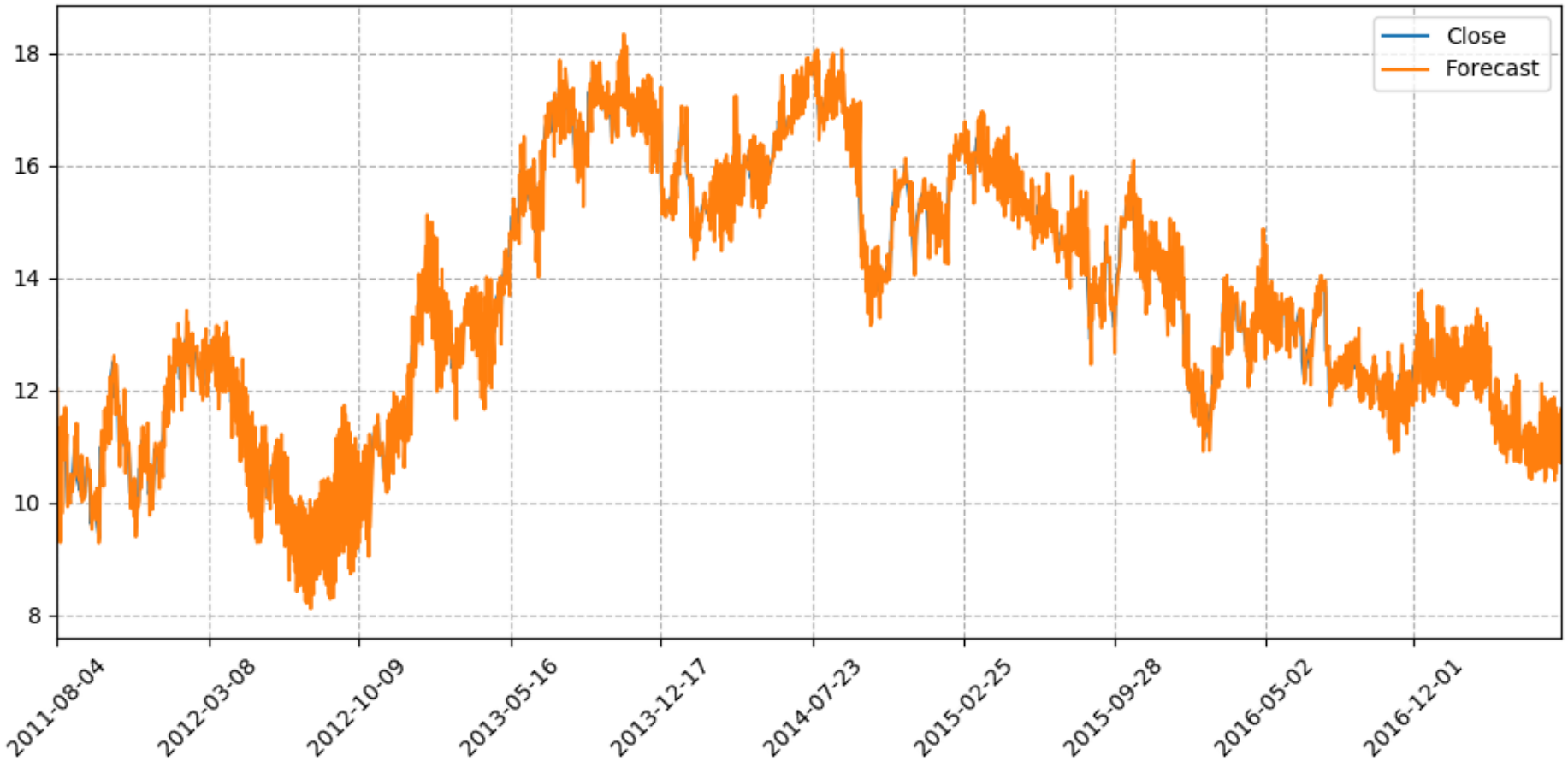


ML, test ratio = 20%, garbage data, Close



MSE: 1E = 7.4694, 5E = 7.3964, 10E = 7.4152, 100E = 7.2695

ML, test ratio = 20%, garbage data, Close, train on tests



MSE: 1E = 0.2584, 5E = 0.3740, 10E = 0.3790, 100E = 0.3819

ML, test ratio = 20%, garbage data, Close, better constants

MSE: 1E = 0.1311, 5E = 0.1286, 10E = 0.1173, 100E = 0.1089

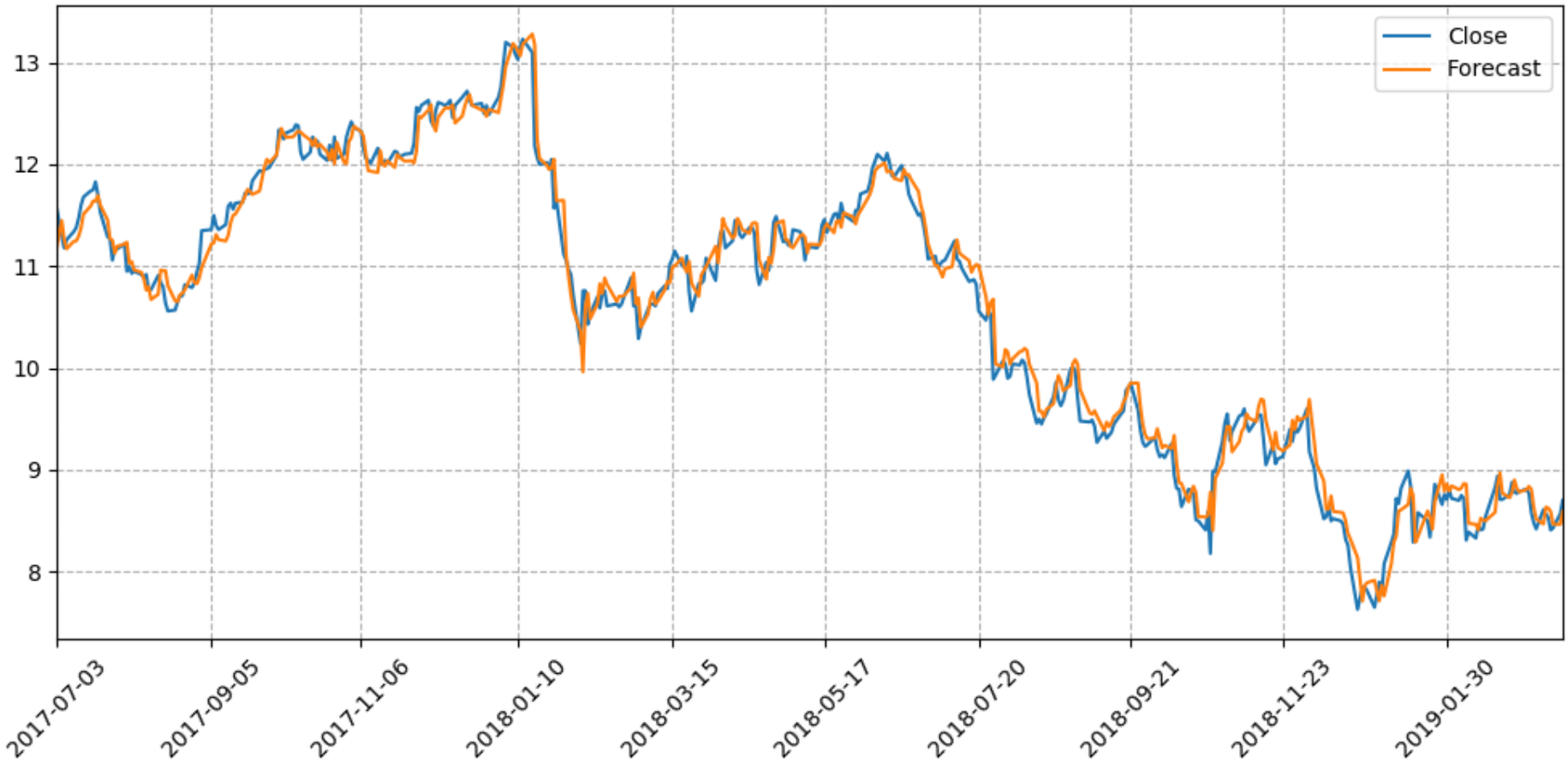
ML, test ratio = 20%, proper data, Close

MSE: 1E = 0.1249, 5E = 0.0972, 10E = 0.0936, 100E = 0.0833

ML, test ratio = 20%, proper data, Close, additional line, index randomization, don't train on tests

1E = 0.3484, 5E = 0.1586, 10E = 0.0707, 100E = 0.0592, 1000E = 0.0579

ML, official tests, proper data, Close



$1E = 0.3603$, $5E = 0.1012$, $10\ E = 0.0712$, $100\ E = 0.1091$, $1000E = 0.0331$

ARIMA(5,1,0), official tests, predicting from Close

Mean Square Error = 0.0260

ML, test ratio = 20%, proper data, ROI, don't train on tests

1E = 0.0400, 5E = 0.0507, 10E = 0.0472, 100E = 0.0400, 1000E = 0.0419

ML, official tests, proper data, ROI, don't train on tests

$1E = 0.0263$, $5E = 0.0258$, $10E = 0.0267$, $100E = 0.0259$, $1000E = 0.0283$

ARIMA(5,1,0), official tests, predicting from Close

Mean Square Error = 0.0260

ML, test ratio = 20%, proper data, ROI, train on tests

1E = 0.0673, 5E = 0.0494, 10E = 0.0423, 100E = 0.0393, 1000E = 0.0388

ML, official tests, proper data, ROI, train on tests

$1E = 0.0444$, $5E = 0.0305$, $10E = 0.0298$, $100E = 0.0265$, $1000E = 0.0267$

ARIMA(5,1,0), official tests, predicting from Close

Mean Square Error = 0.0260

Stock Prices Prediction

The End