2017 Bitcoin Price Prediction Report

Ceren Abay

January 15, 2018

0.1 Experimental Settings

For price prediction of 2017, we have tried all these experimental settings defined in Table 1 and we demonstrate the RMSE values of models.

Explanation	**Values**
window length	3, 5, 7
dimension reduction with PCA	5, 10, 15, 20
horizon	1, 2, 5, 7, 10, 15, 20, 25, 30
training length	25, 50, 100, 200
filtration threshold	0, 10, 20, 30, 40, 50
betti threshold	50, 100, 200, 400

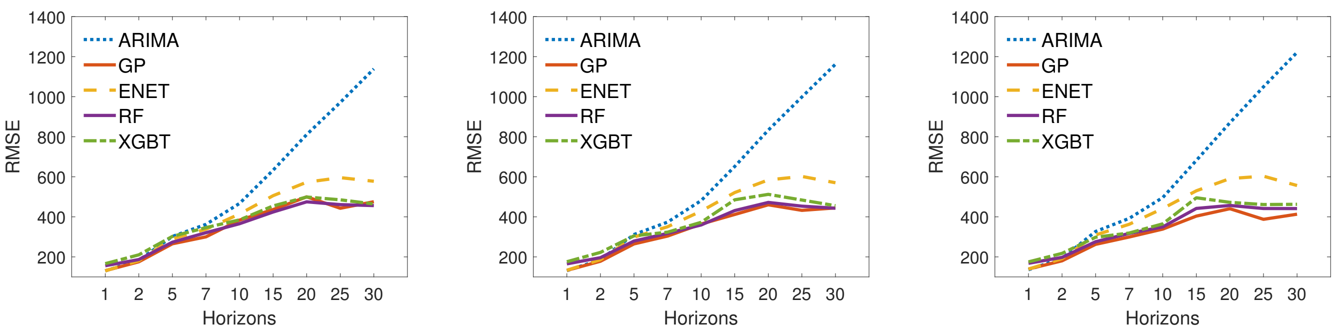
Table 1: Model parameter descriptions and values.

0.2 Baseline Models and RMSE Values

In this experimental settings, we have tried different training lengths in the optimization of the models on bitcoin price prediction. We have chosen *training_length* as 100 and reported the results of the models in this report.

Model	**Input**	**Output**	**Explanation**
arima	Price, TotalTx	Price	ARIMA Model
rf	Price, TotalTx	Price	Random Forest Regressor Model
xgbt	Price, TotalTx	Price	Extreme Gradient Boosting Regressor Model
enet	Price, TotalTx	Price	Elastic Net Regressor Model
gp	Price, TotalTx	Price	Gaussian Process Based Regression Model

Table 2: Baseline Model Descriptions



(a) Window=3 (b) Window=5 (c) Window=7

Figure 1: Line graphs of Figure 1 with different window and horizon values.

0.3 Random Forest Models and Gain Percentage Based on ARIMA

In our analysis, we report the percentage predictive gain, or decrease in $RMSE$ for a specific model m w.r.t. a baseline model m_0 as

$$\Delta_m(h) = 100 \times \left(1 - \frac{RMSE_m(h)}{RMSE_0(h)}\right), \quad (1)$$

where $RMSE_0(h)$ and $RMSE_m(h)$ are delivered by a baseline model m_0 and a competing model m, respectively.

Model	Input	Output
betti50	Price, TotalTx, $B_0(50)$, $B_1(50)$	Price
betti100	Price, TotalTx, $B_0(100)$, $B_1(100)$	Price
betti200	Price, TotalTx, $B_0(200)$, $B_1(200)$	Price
betti400	Price, TotalTx, $B_0(400)$, $B_1(400)$	Price
betti50+betti50'	Price, TotalTx, $B_0(50)$, $B_1(50), B'_0(50)$, $B'_1(50)$	Price
betti100+betti100'	Price, TotalTx, $B_0(100)$, $B_1(100), B'_0(100)$, $B'_1(100)$	Price
betti200+betti200'	Price, TotalTx, $B_0(200)$, $B_1(200), B'_0(200)$, $B'_1(200)$	Price
betti400+betti400'	Price, TotalTx, $B_0(400)$, $B_1(400), B'_0(400)$, $B'_1(400)$	Price
one_step_filtration	Price, TotalTx, 20x20 chainlet filtered with thresholds 0,10,20,30,40,50	Price
deep_rf_filtration	Price, TotalTx, 20x20 chainlet filtered with thresholds 0,10,20,30,40,50	Price

Table 3: Random Forest Model Descriptions

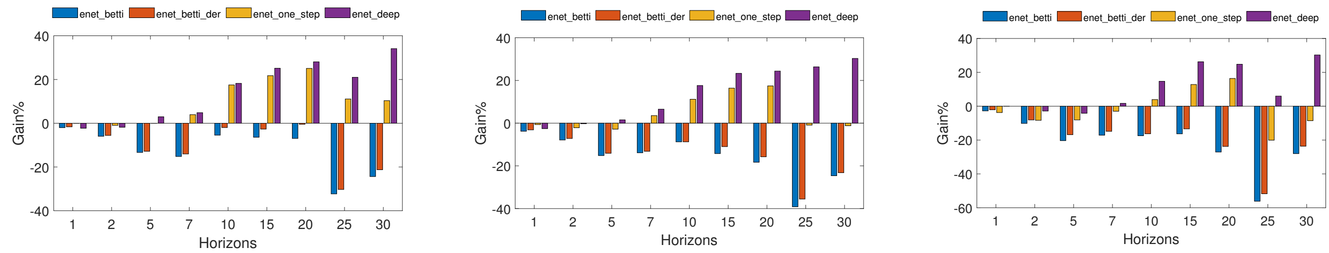
Models	Explanation
one_step_filtration	Vector of chainlets are filtered with thresholds 0,10,20,30,40,50 and random forest model is run on these filtered chainlets.
deep_rf_filtration	Vector of chainlets are filtered with thresholds 0,10,20,30,40,50 and different random forest is run to predict the price. Results of random forest previously trained on each threshold are used on second step random forest to predict price.

Table 4: Filtration Models Explanation

0.4 Observations

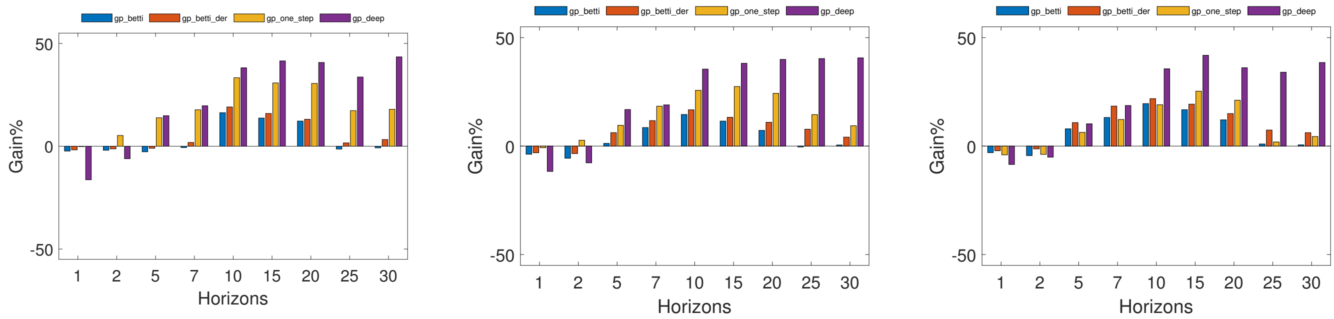
According to percentage of gain presented in Figure 2,

- among seven baseline models, arima has worst rmse for all horizon and window values.
- for window $\in \{3, 5, 7\}$ and horizon < 3, random forest with betti numbers have best price prediction.
- for window $\in \{3, 5, 7\}$ and horizon > 3, deep_filtration has better prediction with the exception of window=7, horizon $in \{11, 12\}$.
- one_step_filtration is not comparable with deep_rf_filtration and random forest with betti.



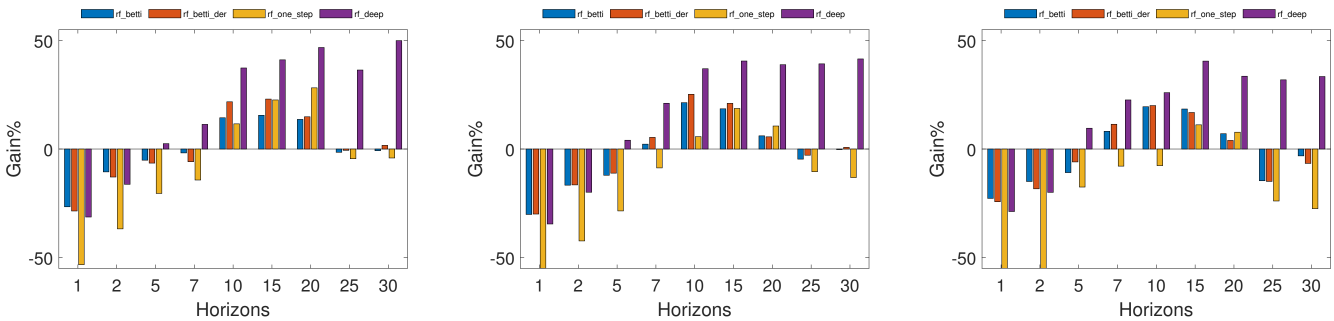
(a) Window=3 (b) Window=5 (c) Window=7

Figure 2: Enet based on gp



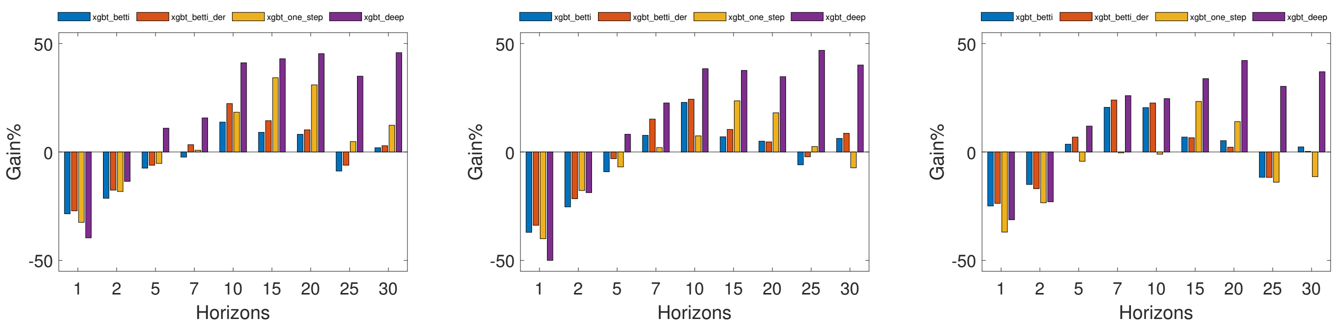
(a) Window=3 (b) Window=5 (c) Window=7

Figure 3: GP based on gp



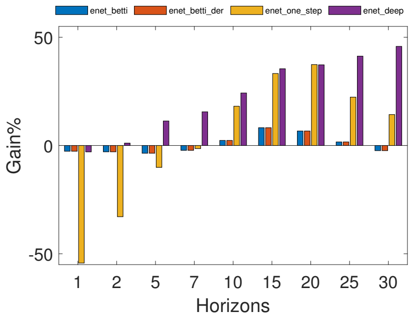
(a) Window=3 (b) Window=5 (c) Window=7

Figure 4: RF based on gp

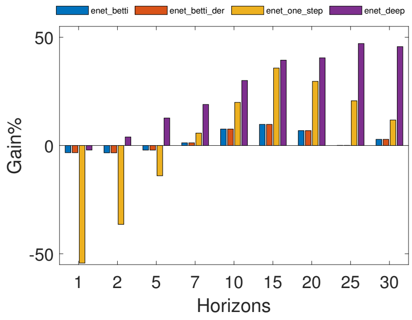


(a) Window=3 (b) Window=5 (c) Window=7

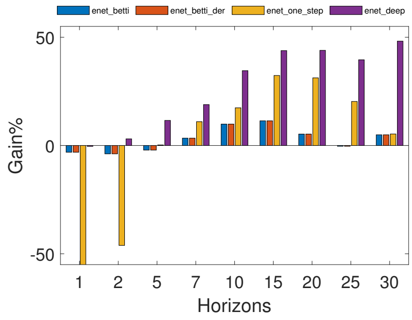
Figure 5: XGBT based on gp



(a) Window=3

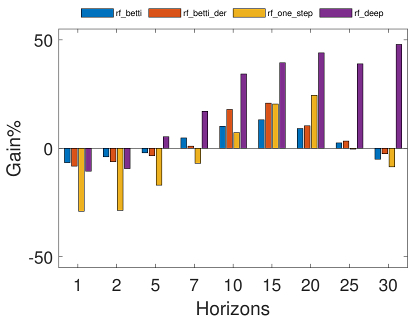


(b) Window=5

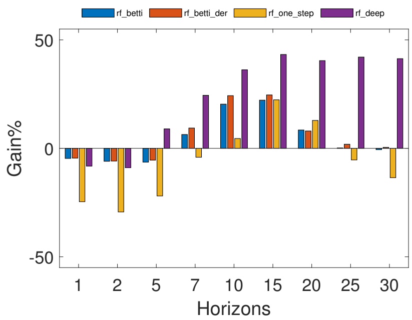


(c) Window=7

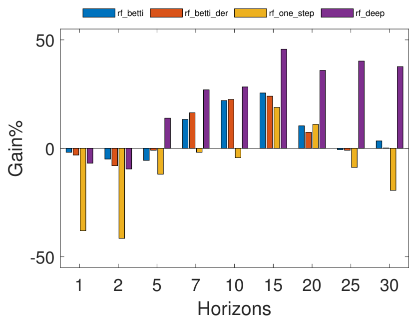
Figure 6: enet based on enet



(a) Window=3

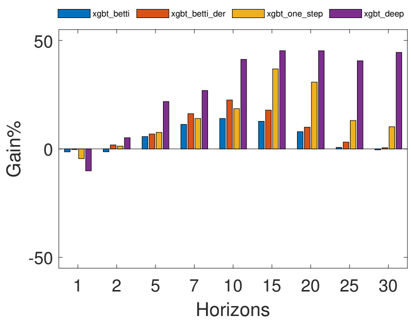


(b) Window=5

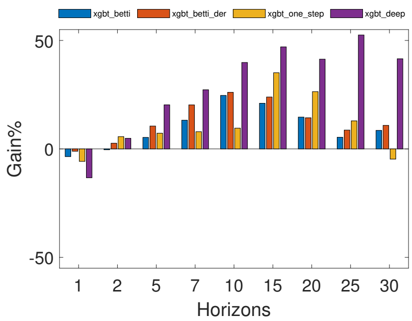


(c) Window=7

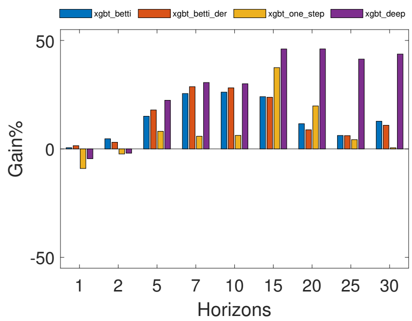
Figure 7: rf based on rf



(a) Window=3



(b) Window=5



(c) Window=7

Figure 8: xgbt based on xgbt