Applying models on On-chain data to predict regimes in Bitcoin.

Yizhao Chen, Like'er Xu, Guhan Chen

Abstract

We want to detect underlying regime switches in cryptocurrency market, so as to support the decision makers.We attempt to conduct researches on a larger time range, and apply hybrid machine learning algorithms as well as time-series analysis to detect regime switches in Bitcoin market and explain them.

Keywords: Machine Learning, Bitcoin, regimes, Time series

1 Introduction

Along with the development of the Internet, Bitcoin has been put into use since 2009. In recent years, cryptocurrencies are used both as media of exchange and as components of investment portfolios. The price of Bitcoin has been increasing since 2021, but it's fluctuating at the same time, which largely influences holders of investment portfolios. Therefore, we want to detect underlying regime switches in cryptocurrency market, so as to support the decision makers.

Figá-Talamanca et al.(2019) found that at most three common states may be considered for Bitcoin through time-series analyses, yet their researches are limited within 2016 to 2019 and methods are not comprehensive. We attempt to conduct researches on a larger time range, and apply hybrid machine learning algorithms as well as time-series analysis to detect regime switches in Bitcoin market and explain them.

2 Hybrid Machine Learning Method

We identify regimes in cryptocurrency market based on publicly available economic data. The data includes daily timestamps, number of transactions, Bitcoin block counts, asset prices, etc.(altogether 37 dimensions that may relate to the Bitcoin price)

2.1 Preprocessing

To process the original data, we first changed them into percentage data. Define our data as a matrix sized $m * n$, where m indicates the number of samples, n is the feature dimensionality. For each entry in this matrix $x_{i,j}$ and its adjacent entry $x_{i+1,j}$, $x_{i+1,j}$ is reassigned as follows.

$$x_{i+1,j} = \frac{x_{i+1,j} - x_{i,j}}{x_{i,j}} (x_{i,j} \neq 0)$$

$$x_{i+1,j} = 1 (x_{i,j} = 0)$$

For economic data, certain data fluctuates largely while some have trivial percentage changes. We then did standardization through MinMaxScaler in Sklearn to normalize the differences.

2.2 Dimensionality Reduction

In the past, Akioyamen et al.(2021) applied Principal Component Analysis to reduce data dimensionality and extract features. Essentially speaking, PCA project data to certain orthogonal axes which generate the largest variance. However, our data includes highly related variables, which may make one of the components take over most of the explained variance ratio. So we use Nonnegative Matrix Factorization(NMF) to reduce the dimensions of data to promote the performance of clustering algorithms.

In NMF, the original data matrix V is factorized into two matrices W and H, and all three matrices have no negative elements.

After initialization, the estimated entry value of V is:

$$\hat{v}_{i,j} = \sum_{k=1}^{k} w_{i,k} h_{k,j}$$

The differentiable loss function of each entry, which demonstrate the reconstruction error between V and $W * H$ in each entry, is:

$$e_{i,j}^2 = (v_{i,j} - \hat{v}_{i,j})^2 = (v_{i,j} - \sum_{k=1}^{k} w_{i,k} h_{k,j})^2$$

The gradient is then derived through taking the derivative of the total loss function on w and h respectively. To prevent overfitting, we can also add regularization into the function.

According to Figá-Talamanca et al.(2019), up to three regimes in Bitcoin market are explainable, so we let k to be 2 and 3 for better interpretability.

3 Clustering

3.1 K-means

k-means is a kind of widely used unsupervised machine learning algorithm, which aims to partition n unlabeled observations into k clusters. Firstly, we need to choose k observations as initial clustering centers. Then, according to the distances between observations and centers we categorize n observations into k clusters and choose k new clustering centers. In this way, we can adjust clusters dynamically until they grind to a halt. And then we get the final clustering result.

3.2 Visualization

We use the first 3 principle components to do k-means clustering. In this figure, we can see the time series of the first 3 components. We choose k=2, and different color denote different clusters,as the figure 1 shows.

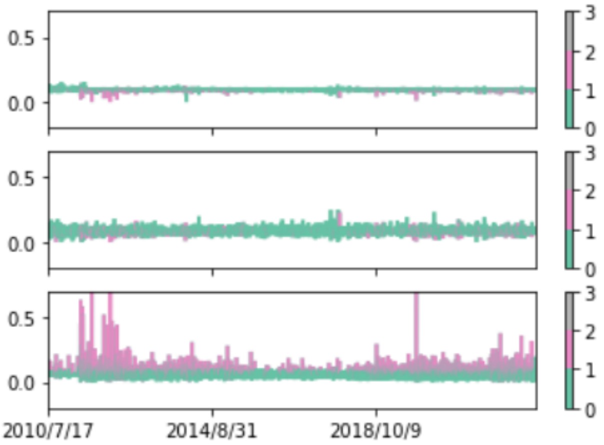

Figure 1: Time Series of Principle Components

3.3 Price of BTC

We apply the clustering result on the BTC time series, and this is the figure. In this figure 2, the red line and green line denote different clusters. we can see that in green region, the price of BTC fluctuates

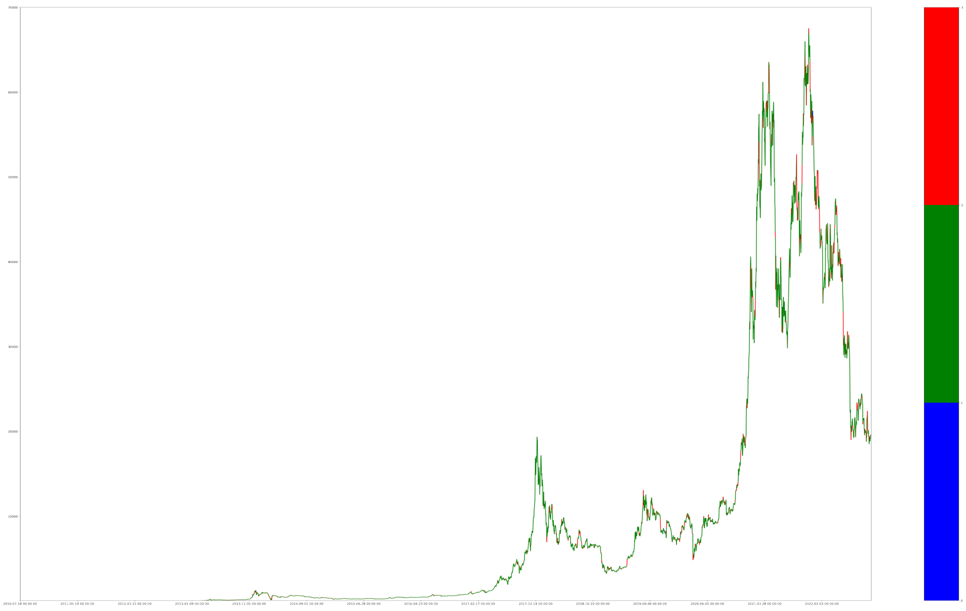

Figure 2: Time Series of BTC Price

fiercely. And in the red region, the price changes mildly. For example, between 2020 and 2021, the price rose sharply, which marked by green. On the contrary, in the calm period of 2022, the curve tends to be red. So we can see the clustering result is reasonable.

4 Classification

4.1 visualization

we use the Axes3D method to visualize our clustering results, coloring them with three different colors. It can be clearly seen that the results of clustering are still effective.

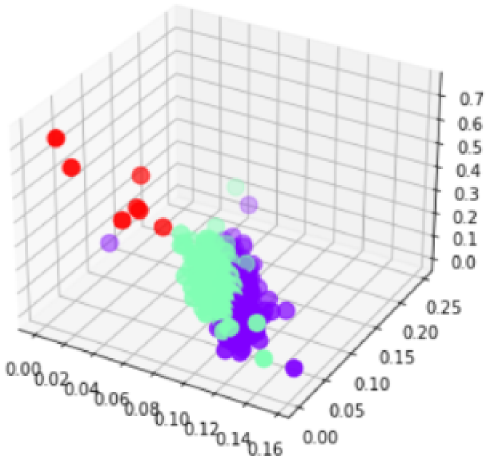

Figure 3: visualization

4.2 Compare classification models

In order to determine the regime of the cryptocurrency market, we use publicly available economic data for analytical processing, including daily timestamps, number of transactions, Bitcoin block count, asset prices, etc. We first perform cluster analysis to obtain its category subordination, and then classify it. For the labeled data after clustering, we apply a variety of machine learning methods to classify and model and predict discrete random variables, so as to compare and obtain good performance.

We use naive-bayes classification, Decision-Tree classification, LDA classification, Random-Forest classification, Adaboost for classification comparison.To find best parameters,we use Grid-search to optimize our classification model. Thus, our correlation model scores the clustered economic data. Then we visualize the classification observed on the data for better understanding on the performance of the model.

First,we show the scores corresponding to the different models with the data.Then We present it in the form of a table,The data for the classification model is as Table 1 shows.

4.3 Curves of different classification methods

We then plot the learning curves of different classification methods and compare their accuracy to find the most appropriate classification method for this economic data.

Those models are used to predict the regime corresponding to each observation in the out-of-sample data.At last,we enter the loop and draw learning curve of different models to find the most appropriate classification method for this economic data,as the figure 4 shows.

5 Time-series Analysis

Using data processed to percentage format, we conducted PCA and time-series analysis to detect regimes, too. Since regime switches had been proposed by Quandt(1958) and Goldfeld and Quandt(1973), James Hamilton introduced Hidden Markov Models, and an estimation procedure based on Bayesian filter to analyze regime swithes in economic market. We deployed MS Regress, a module in Matlab to conduct time-series analyses on Bitcoin market data with Maximum Likelihood method. In each practice, after fitting our dataset to the model, we use Hamilton filter to calculate the smoothed state probability, and considers the time a certain regime if the probability is larger than 0.5.

5.1 Gaussian Markov Model

When thinking in computational terms, a univariate markov swithcing model can be represented in a generalized notation. Consider the following model with eleven explanatory variables $x_{i,t}, i = 1, 2, 3, ..., 11$ where the innovations follow a Gaussian distribution:

$$y_t = \sum_{i=1}^{n} \beta_{n,S_t} x_{n,t} + \epsilon_t$$

Where: y_t: The dependent variable(i.e., Bitcoin price, in this case). S_t: The state at time t, $S_t = 1, 2, 3$. β_{n,S_t}: Beta coefficient for explanatory variable i at state S_t where i goes from 1 to 11. ϵ_t: Residual vector which follows a Gaussian distribution.

This representation can nest a high variety of univariate markov switching specifications. We processed our dataset to Gaussian distribution, to fit this model.

5.2 Autoregressive Model

We also used a simple wrapper function for estimating a general autoregressive markov switching model. Based on the univariate Gaussian markov model, we add the dependent variables from the last time index to independent variables. The Autoregressive Model is given as follows:

$$y_t = \sum_{i=1}^{n} \beta_{n,S_t} x_{n,t} + y_{t-1} + \epsilon_t$$

6 Visualization and Comparison

To compare our results from machine learning and time-series, we plotted the regimes that we found onto Bitcoin price with different colors. The performance of the models could be evaluated by the synchronization between the fluctuation of Bitcoin price and our inferences.

Classification Methods	Naive-Bayes	LDA	Decision Tree	RandomForest	AdaBoost
Training Score	0.978	0.961	1.000	1.000	0.996
Test Score	0.980	0.966	0.998	0.996	0.994

Table 1: classification results

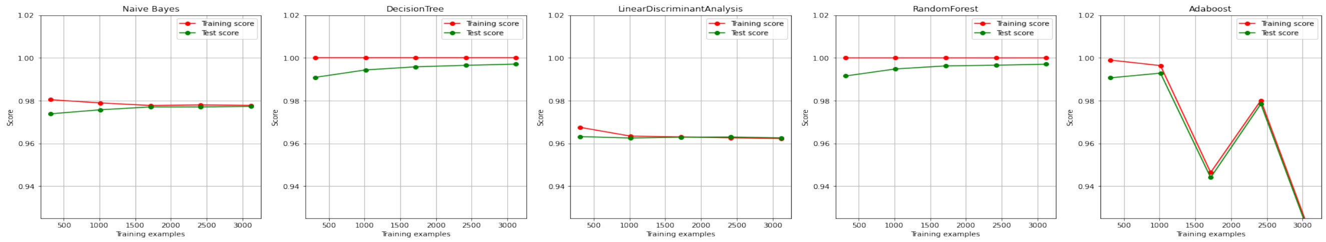

Figure 4: comparison of training scores

As we can see, Gaussian Markov Model and machine learning methods perform the best.

Gaussian Markov Model: As shown in Figure 13, Gaussian Markov Model detected three regimes in our data. Trace 0(denoted as green), shows that the Bitcoin marketing is in recession. Trace 1(denoted as blue), shows a highly volatile condition for Bitcoin price, in which the market is likely to change to other two regimes. Trace 2(denoted as red), demonstrates a booming market. Though highly volatile regime lacks sufficient interpretability, booming and recession are precisely detected, especially for the time periods right after or before volatile regimes.

Autoregressive Model: As shown in Figure 14, autoregressive model has a similar performance with Gaussian Markov Model on detecting volatile regimes, while it performs poorly on the other two. During 2020 epidemic, the model classifies most time ranges as trace 0, which shows that it confuses booming and recession for Bitcoin market.

machine learning model K-means: In Figure 15. we got the best clustering result with the initialization of K-means clusters to be 2. The model performs generally well on clustering booming and recession regimes.

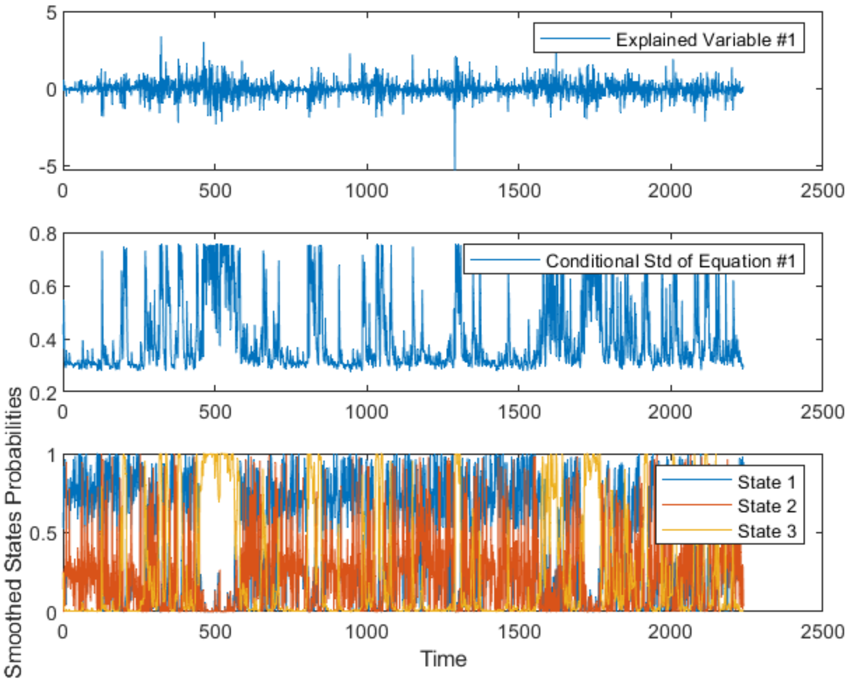

Figure 6: results from Gaussian Markov Model

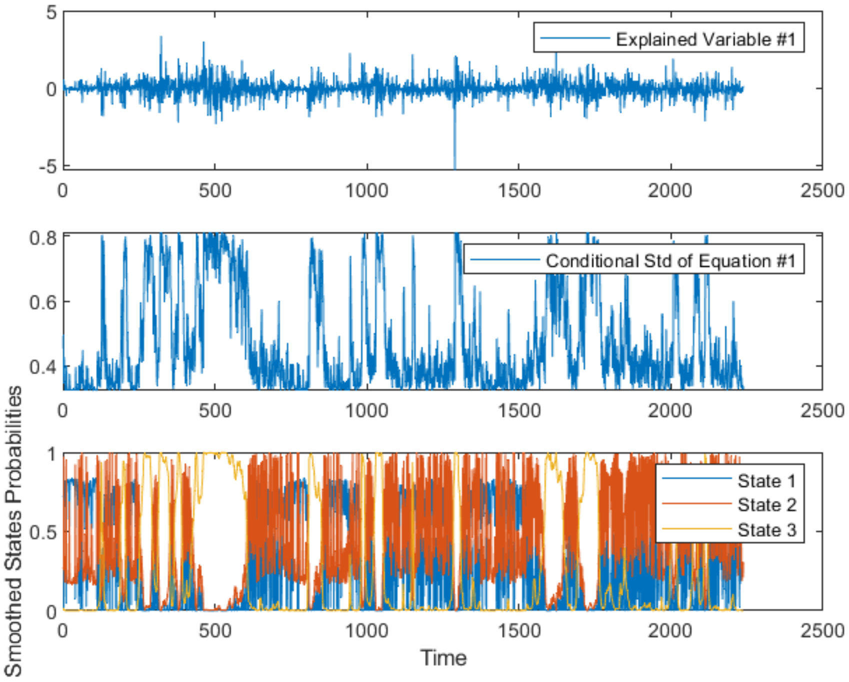

Figure 7: results from Autoregressive Model

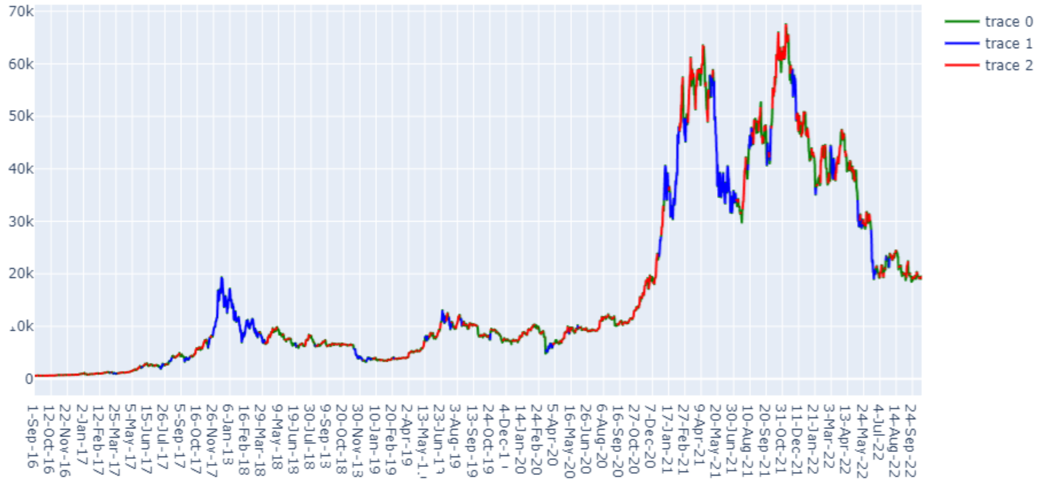

Figure 8: results from Gaussian Markov Model

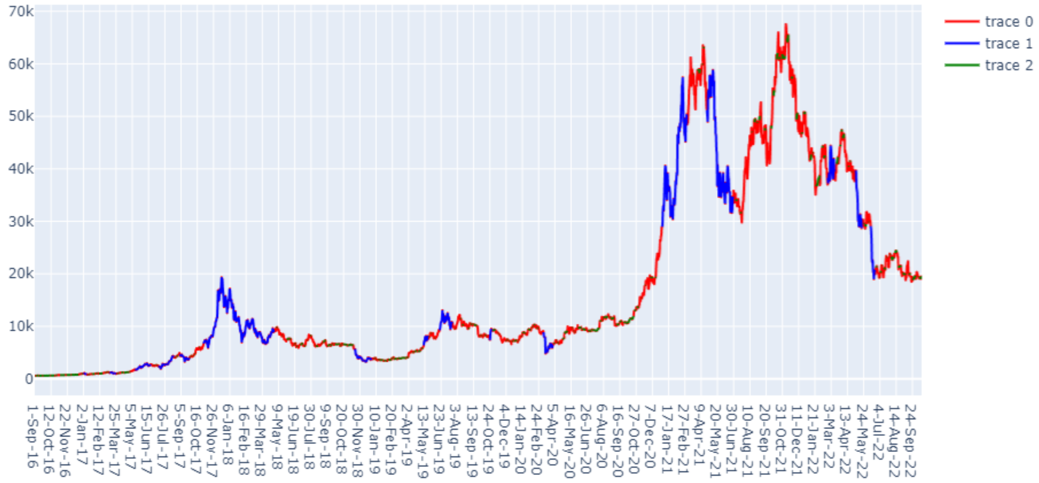

Figure 9: results from Autoregressive Model

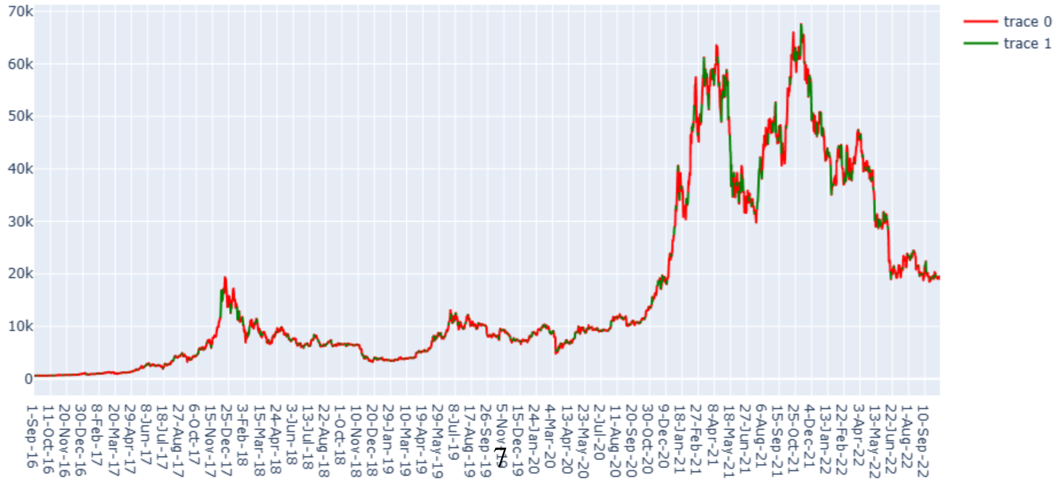

Figure 10: results from K-means