Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής

Πρόγραμμα Μεταπτυχιακών Σπουδών

«Πληροφορική»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Ανίχνευση Γεγονότων Ανόδου/Καθόδου σε Limit-Order-Book Δεδομένα Αγορών με χρήση SVM Ταξινομητών.** **AN SVM-based Classification Approach for the Detection of Upward/Downward Events in Limit-Order-Book Market Data**
Ονοματεπώνυμο Φοιτητή	**ΑΝΑΡΓΥΡΟΣ ΚΑΡΑΜΑΝΤΑΣ**
Πατρώνυμο	**ΠΑΡΑΣΧΟΣ**
Αριθμός Μητρώου	**ΜΠΠΛ/ 15024**
Επιβλέπων	**ΓΕΩΡΓΙΟΣ ΤΣΙΧΡΙΝΤΖΗΣ, ΚΑΘΗΓΗΤΗΣ**

Ημερομηνία Παράδοσης **ΝΟΕΜΒΡΙΟΣ 2018**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή)	(υπογραφή)	(υπογραφή)
ΤΣΙΧΡΙΝΤΖΗΣ ΓΕΩΡΓΙΟΣ ΚΑΘΗΓΗΤΗΣ	ΑΛΕΠΗΣ ΕΥΘΥΜΙΟΣ ΕΠΙΚ. ΚΑΘΗΓΗΤΗΣ	ΣΩΤΗΡΟΠΟΥΛΟΣ ΔΙΟΝΥΣΙΟΣ ΜΕΤΑ-ΔΙΔΑΚΤΟΡΙΚΟΣ ΕΡΕΥΝΗΤΗΣ

Περίληψη

Τα limit order book δεδομένα περιλαμβάνουν πληροφορίες σχετικές με κάποια συναλλαγή που έχει πραγματοποιηθεί στο παρελθόν. Στην προκειμένη περίπτωση οι συναλλαγές αφορούν την τιμή του κρυπτοσυναλλάγματος Bitcoin. Χρησιμοποιώντας μια μηχανή διανυσμάτων επιτήρησης (SVM) και κατά επέκταση μια προσέγγιση κανονικοποίησης, τα δεδομένα αυτά οργανώνονται και μπορούν να προσφέρουν μελλοντικά αποτελέσματα τα οποία βρίσκονται πολύ κοντά στην πραγματική τιμή του Bitcoin. Το πρακτικό μέρος εφαρμόζεται σε γλώσσα προγραμματισμού Matlab. Τα αποτελέσματα, γραφικά και αριθμητικά, δείχνουν τις πιθανότητες να συμβεί ή να μην συμβεί την επόμενη χρονική στιγμή ένα σωστά προβλεπόμενο γεγονός. Σκοπός είναι να βρεθεί η βέλτιστη επιλογή παραμέτρων για να επιτευχθεί το βέλτιστο χρονικό και ποιοτικό αποτέλεσμα.

Abstract

Limit order book data include information about a transaction that have been completed in the past. In this case, the transactions relate to the cryptographic value known as Bitcoin. By using SVM and also by extension, a classification approach, these data are organized and offer future results that are very close to Bitcoin's actual future value. The practical part is applied to Matlab programming language. The results, graphical and numerical, show the possibility of a well-predicted event occurring or not occurring at the next moment. The aim is to find the optimal parameter’s selection and achieve the best time and quality result.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Limit Order Book

Limit Order Book is a terminology used by the market. Every exchange maintains an order book for every transaction. It can also be called a method that stores in a database every buy or sell, all equity or future listed markets. This, immediately brings the main idea of big data, because the number of data to be used exceeds the 'small' usable data in a simple platform.

The data collection must be well - organized, with perfect architecture so afterwards the selection and update to be fast and easy to achieve the maximun profit. Such values that meant to be stored are: the sign of order (buy or sell), the price, the quantity, the time of transaction and many other values, extremely useful for the market environment.[1] [2] [3] [4].

Considering the number of transactions ecery day, it becomes even more clear that the big data that are mentioned before is the very meaning for the subject. After many studies into Limit Order Books, it is being concluded to three types of order:

- **Limit Order**: It is an order that is being fulfiled if the stock reaches a specified price. Then the certain number of shares that the user have been determined, are bought or sold, depending on the situation each time. So this method has no certain time of fulfilment.
- **Market Order**: This method fulfils an immidiate buy or sell of a certain number of shares, to the best available transaction in every case.
- **Cancellation**: As the name disseminates, it is a method to exit the Order Book.

1.2 Upward - Downward Spread Crossing

In a Limit Order Book, there are limit buys, also called bid and limit sells, also called ask. The first are categorized in descending order and the second in ascending order. The best bid is the highest and the best ask, the lowest. This happens because the last two are the closest prices to mid - price.

The difference between a bid and an ask is called Spread. Logically, an upward event represents the prices that are intended for sell and a downward event represents the prices that are intended for buy.

There are many strategies and algorithms, with the help of modern technology and knowledge that 'predict' at a decent percentage those kinds of events. The main purpose of this study is to exploit these strategies and algorithms as tool to find the best case scenario and combinations of predicting the price of the most famous Cryptocurrency, the Bitcoin.

1.3 Financial Data

Financial data are very useful nowadays. The stock market is unpredictable, because of the vast number changes that happen every second. But every change is data. Some of them can be useful and profitable, some cannot. The only certain is that with the progress of technology and the advance of studies, mainly in machine learning, it is possible to 'predict' at a certain percentage, future prices of the closing price of a stock. So, there are two factors that can affect and provide the best profit, power and data.

Firstly, a system with great requirements is needed, to manage so many data. A multiprocessor with high frequency and many graphic cards would be the best option to achive power. This part is very important because if the system can achieve high speeds, it can provide more representive and efficient results.

Secondly, it is needed a vast quantity of data. Some of the elements that are stored in a Limit Order Book are being mentioned above. One of these is the timestamp, also known as the exact time an event happened. It the world of computers time is discrete. For example, one minute can be separated to sixty seconds, which corresponds to sixty timestamps. Also, one minute can be separated 60000 milliseconds which corresponds to 60000 timestamps.

Each timestamp contains other more complicated financial data. So, it becomes obvious that in the case with the 60000 timestamps, storage necessity is increased and administration of data becomes slower and harder. But even though there are disadvantages, more timestamps in one minute provide better and more accurate results for 'predicting' the future closing price of a stock.

2 Chapter 1

2.1 Limit Order Book

2.1.1 Limit - Market Orders

Limit Order Books are used for financial exchanges in the market. Everyone in the marketplace can trade assets either by buying or by selling orders. The most advantageous move is to buy at the lowest price or to sell at the highest price of a certain stock. The opposite moves are the least efficient. It is very common for traders to try to find the best strategy so that they can achieve maximun profit.

Orders are separated into Market and Limit Orders, as its already mentioned. Limit orders represent the orders for buying or selling a specific percentage of an asset for a specified price. In fact, limit orders wait in a condition that is called active, for a match. So, the duration of the executioon is not certain. If the market price keeps fending off from the price that has been specified, the order may last a lot. While being in active mode, the order can be canceled whenever the trader decides, before loosing profit. These kinds of orders are known for providing liquidity.

Market Orders represent the orders for buying or selling a specific percentage of an asset for the best available price. So, in this case the duration is certain and it can be achieved with an immediate execution. These kinds of orders are known for consuming liquidity.

2.1.2 Buy - Sell Orders

The terms 'buy' and 'sell' have mentioned in the above paragraphs, refering to financial terms. To be more exact for the the definitions of the Limit Order Book:

- A Sell Order stands for an order x with size $\omega_x > 0$ and price P_x. The Owner has set the commitment to sell up to $|\omega_x|$ units of the share. The price must exceed the defined P_x. So, the conclusion is $P_{sell} \geq P_x$.
- A Buy Order stands for an order x with size $\omega_x < 0$ and price P_x. The Owner has set the commitment to buy up to $|\omega_x|$ units of the share. The price must not exceed the defined P_x. So, the conclusion is $P_{buy} \leq P_x$.

2.1.3 Long - Short Position

The are two possible strategies, for the user to profit in the stock market:

- The Long Position. In this trategy, there is a buyer who is purchasingshares of stock in relatively low price, and he is waiting for them to rise. Whene they reach the desirable upper limit price, the shares are sold. So, the main idea is to buy low and sell high.

  For example, the buyer purchase 1000 shares at the price of 50$. In a month the price rice to 75$. If he or she decides to to sell, the profit is 25000$, without the brokers fee.

- The Short Position. In this strategy, there is a seller who is selling his shares of stock, when they start falling. The seller anticipates until they reach the desirable lower limit price, and he or she buys them back.

  For example, the seller gives his 1000 shares at the price of 35$. In a month the price has fallen to 25$. If he or she decides to buy them back, where have not been lost 10000$. As before, in the example have not been calculated the broker fee.

The first strategy is more famous and obvious method. The second is less known but still trustworthy. Many stockholders use it continuously to prevent risky falls of a stock. Both methods take time, considaration and anticipation. The is a large system of financial data that suggests the falling or rising of a stock.[5] [6] [7]

2.2 Upward - Downward Spread Crossing

The phrase ”cross the spread” means that there is an event that happens if a trader buys at the offer price or sells athe the bid price. This can also be handled by a high - sopfisticated algorithm, that buys or sell at the specified price. But for this to succeed, it needs a large database of history records. Thes records are also known as limit order books.

In the previous chapter, it has been analyzed the subject of the long and short potition strategies. Such startegies are the key of this algorithm while the limit order books contain the data to edit.

One startegy refers to bid and the other to ask. An upward spread crossing event happens if the best bid price, which is the highest, surpass the best ask price, at time $t + \Delta t$. So:

$$P^{bid}_{t+\Delta t} > P^{ask}_{t} \quad (1)$$

Respectively, a downward spread crossing event happens if the best ask price, at $t+\Delta t$, becomes inferior to the best bid price at t time. So:

$$P^{ask}_{t+\Delta t} < P^{bid}_{t} \quad (2)$$

The Δt is a variable that defines a very short time difference, and its price choice depends on the circumstances of the problem.[8]

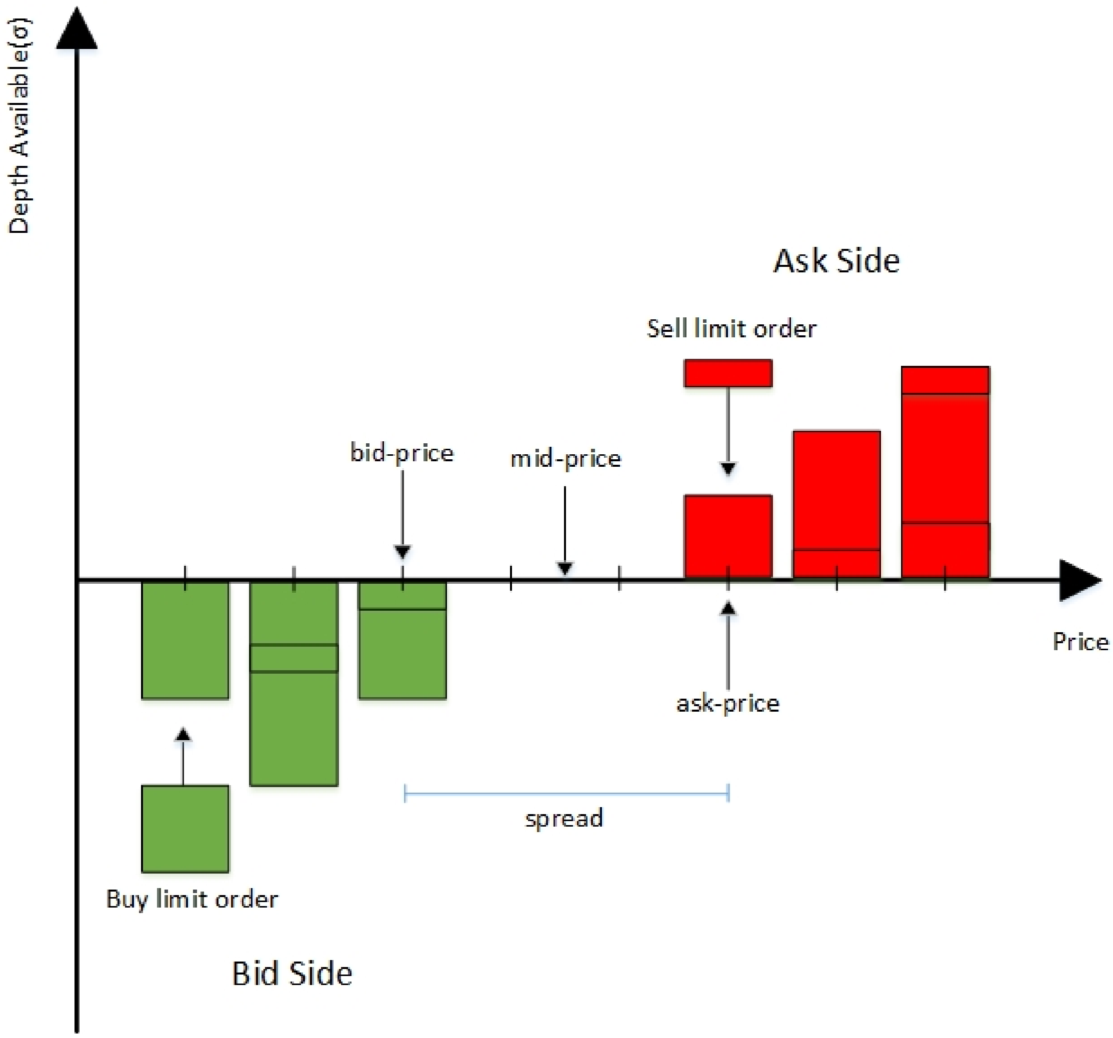

Figure 1: **Schematic of an LOB**

In figure 1, the bid side is set in a decreacing order and the ask side in an increasing order as it is already refered. Also, in the bid side, the bid price b_t is the highest active buy order at time t. Respectively, the ask price a_t, is the lowest active sell order at time t. The last definitions are called best quotes.

The bid side contains the area with prices, that are smaller than the bid best quote, and the ask side, the area with prices greater than the best ask quote. The area within the best quotes is defined with definition "inside the spread".

The bid - ask spread can be set by the difference:

$$P_t = a_t - b_t \tag{3}$$

and the mid - price:

$$m_t = \frac{a_t - b_t}{2} \tag{4}$$

If there is a buy or sell order x that arrives right after time t, then there are three cases that may happen:

- $P_x \leq a_t (P_x \geq a_t)$, then x represents a limit order which concludes to an active condition at arrival, without the prices a_t and b_t to change.
- $b_t < P_x < a_t$, then x represents a limit order which concludes to active condition at arrival, and causes the bid price to rise or the ask price to fall.
- $P_x \geq a_t (P_x \leq b_x)$, then x represents a market order that concludes to matching one or more sell (buy) orders at arrival.

3 Chapter 2

3.1 SVM - Support Vector Machines

Support Vector Machines are supervised learning methods used mainly by machine learning systems. A supervised learning method uses available data and creates forms of input - output pairs. It works as a function. This means that there are available data from a database or history real data that are the parameters of the function, also known as ”input”.

The SVMs edit their data with well - sophisticated algorithms and create some results, also known as ”output”. The unique attribute of supervised learning is the fact that the result values are closely assosiated with their ”input” values, which either are integer or double data types.

Particulary, SVMs use unspervised methods for solving classification problems, which are defined as pattern recognition problems. The first pattern recognition problem defined in the late 1950s. A supervisor observes situations and sets which of k classes each one belong. The object of the matter is to create an algorithm, managed by a machine that realizes approximate classification as the supervisor did. SVMs solve those kinds of problems, or to be more exact reach very close to the solution

The main function of support vector machines is to find a linear or non - linear hyperplane which separates two or more group data points, generalized best to future data. This also called maximun margin hyperplane, and the closest points of each class have the longest distance available.[9]

To encounter pattern recognition problems like classification, as it is already refered, one can start by mapping the input vectors into a set of features. The features are created by the supervisor, in a way to be matching inputs for the SVM. The type of features are selected, depending on the complexity the system can reach. For example, a two - dimentional problem is not too complex. A high - dimentional problem needs complex convertion to simpler problems for it to be solved. The convertion comes with linear discriminant functions.

3.1.1 Hard Margin Support Vector machines

With $x_i \in R^n$ and $y_i \in \{-1, 1\}$, it can be determined the summary $S = \{(x_1, y_1), ..., (x_l, y_l)\}$ as a set of training patterns. Every training point is part of a class, labeled as $y_i = +1$ and $y_i = -1$. So, in this case there two classes.

Assuming that these data points are separatable, there is a linear decision function:

$$g_{(x)} = w^T x + b =< w, x > +b \tag{5}$$

where $w \in R^n$ as the weight factor in n - dimentions and $b \in R$.

The $g_{(x)}$ function defines the hyperplane that is already refered at the start of this chapter, and it can be set as following:

$$< w, x > +b \begin{Bmatrix} \geq +1, & \text{if} & y_i = +1 \\ \leq -1, & \text{if} & y_i = -1 \end{Bmatrix} \tag{6}$$

If variable $c =< w, x > +b$, then the summary available price is $-1 \leq c \leq 1$. The region $\{x : -1 \leq g_{(x)} \leq 1\}$ is called generalization region of the decision function.[10][11][12]

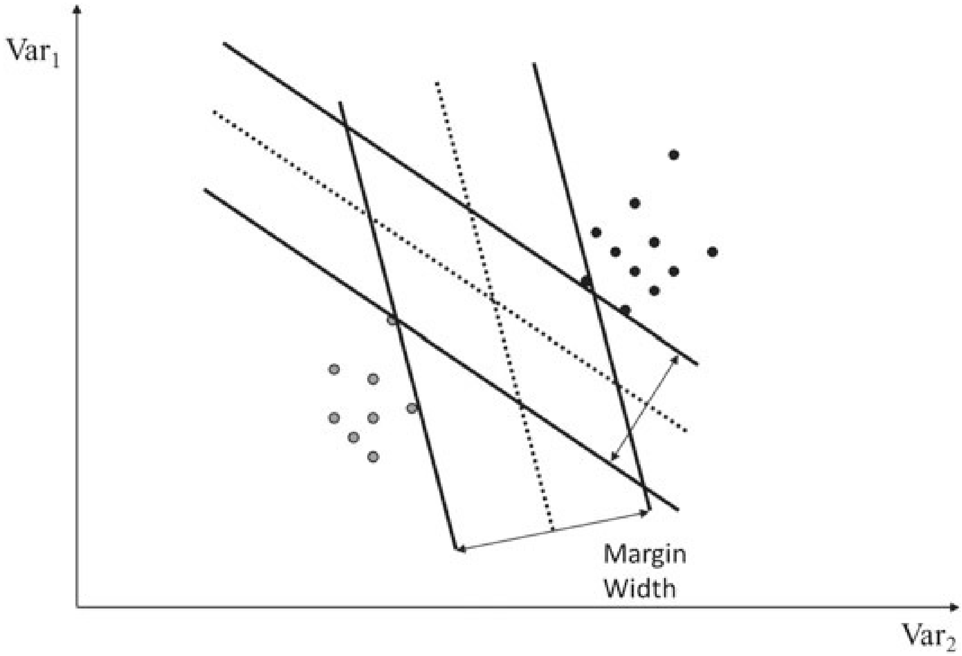

Figure 2: **Decision Functions**

The Distance between the data point x and the hyperplane parameterized by (w, b), is determined by the equation

$$R_{(x;w,b)} = \frac{|g_{(x)}|}{||w||} = \frac{< w, x > +b}{||w||} \tag{7}$$

The line $l(x; w)$ goes through x and is orthological to the hyperplane:

$$l_{(x;w)} = \frac{a}{|w|} w + x \tag{8}$$

To define the variable a, consider that the line crosses the hyperplane, then

$$g_{(l(x;w))} = 0 \Longleftrightarrow \tag{9}$$

$$w^T l(x; w) + b = 0 \Longleftrightarrow \tag{10}$$

$$w^T (\frac{a}{||w||} w + x) + b = 0 \Longleftrightarrow \tag{11}$$

$$\frac{a}{||w||} ||w||^2 = -w^T x - b \Longleftrightarrow \tag{12}$$

$$a = -\frac{g_{(x)}}{||w||} \Longleftrightarrow \tag{13}$$

$$|a| = -\frac{g_{(x)}}{||w||} \Longleftrightarrow \tag{14}$$

If x^+ and x^- are two data points from two different hyperplanes, then $g_{(x)} = +1$ and $g_{(x)} = -1$, respectively.

The quantity γ is also known as geometric margin and defines the mean distance of x^+ and x^- points

$$\gamma = \frac{1}{2}\{R_{(x^+;w,b)} + R_{(w^-;w,b)}\} = \frac{1}{||w||} \tag{15}$$

The primal problem for the hard margin SVM to solve is:

$$\min_{b,w} \frac{1}{2} ||w||^2 \tag{16}$$

subject to $y_i(< w, x_i > +b) \geq 1, \forall i \in [l]$

According to the Langrange function, its primal form may be:

$$L_{(w,b,a)} = \frac{1}{2} < w, w > - \sum_{i=1}^{l} a_i \{y_i(< w, x_i > +b)\} \tag{17}$$

with $a_i \geq 0, r_i \geq 0$

Refering to the KT theorem[13], there is a normal point (w^*, b^*), that is optimum, if the a^* variable applies to the necessery and sufficient conditions that:

$$\frac{\partial L_{w^*,b^*,a^*}}{\partial w} = 0 \tag{18}$$

$$\frac{\partial L_{w^*,b^*,a^*}}{\partial b} = 0 \tag{19}$$

$$a_i^* \{y_i(< w^*, x_i > +b^*) - 1\} = 0, \forall i \in [l] \tag{20}$$

$$y_i(< w^*, x_i > +b^* - 1) \geq 0, \forall i \in [l] \tag{21}$$

$$y_i(< w^*, x_i > +b^* - 1) \geq 0, \forall i \in [l] \tag{22}$$

$$a_i^* \geq 0, \forall i \in [l] \tag{23}$$

The equation (20) contains the variable a_i^*, which may be:

- Active constraints. If $a_i^* = 0$, then $y_i(< w_i^*, x_i > +b^*) - 1 > 0$.
- Inactive constraints. If $a_i^* > 0$, then $y_i(< w_i^*, x_i > +b^*) - 1 = 0$.

The second case refers to the Supprot Vectors of the hyperplane with $g_x = -1$ and $g_x = +1$. The optimal value for the w in the equation (18), is:

$$w^* = \sum_{i=0}^{l} a_i^* y_i x_i \tag{24}$$

From (19):

$$\sum_{i=0}^{l} a_i^* y_i = 0 \tag{25}$$

Combining the last two equations with the Langrange function[14], resulting:

$$L_{(w,b,a)} = \sum_{i=0}^{l} a_i - \frac{1}{2} \sum_{i=0}^{l} \sum_{j=0}^{l} a_i a_j y_i y_j < x_i, x_j > \tag{26}$$

According to Karush - Kuhn - Tucker conditions, always for $a_i^* \neq 0$, it applies:

$$y_i\{(< w^*, x_i > +b^*) - 1\} = 0, \forall i \in [SV] \tag{27}$$

For SV^+ as the support vectors of $y_i = +1$

$$< w^*, x_i > +b^* = +1, \forall i \in [SV^+] \tag{28}$$

and SV^- as the support vectors of $y_i = -1$

$$< w^*, x_i > +b^* = -1, \forall i \in [SV^-] \tag{29}$$

Summing the last equations, it results to:

$$\sum_{i \in SV^+} < w^*, x_i > +b^* + \sum_{i \in SV^-} < w^*, x_i > +b^* = n^+(+1) + n^-(-1) \Longrightarrow \tag{30}$$

$$b^* = \frac{1}{n^+ + n^-}\{(n^+ - n^-) - \sum_{i \in SV} < w^*, x_i >\} \tag{31}$$

So, the separating hyperplane can be represented by the final equation:

$$g_{(x)} = \sum_{i=1}^{l} a_i^* y_i < x_i, x > +b^* = \sum_{i \in SV} a_i^* y_i < x_i, x > +b^* \tag{32}$$

3.1.2 Soft Margin Support Vector machines

The Hard Margin Support Vector Machines are used for solutions that separate the train data linearly. There are cases though, that the Hard Margin is not capable to solve. The Soft Margin works as an extend of the predecessor. The main difference lies in the event that the new vector margin allows some error, which is the ξ_i, $(\xi_i \geq 0)$.[15][16][17][18][19]

So, because the error is permittable, the subject of the equation (16) can be reformed to:

$$y_i(< w, x_i > +b) \geq 1 - \xi_i, \forall i \in [l] \tag{33}$$

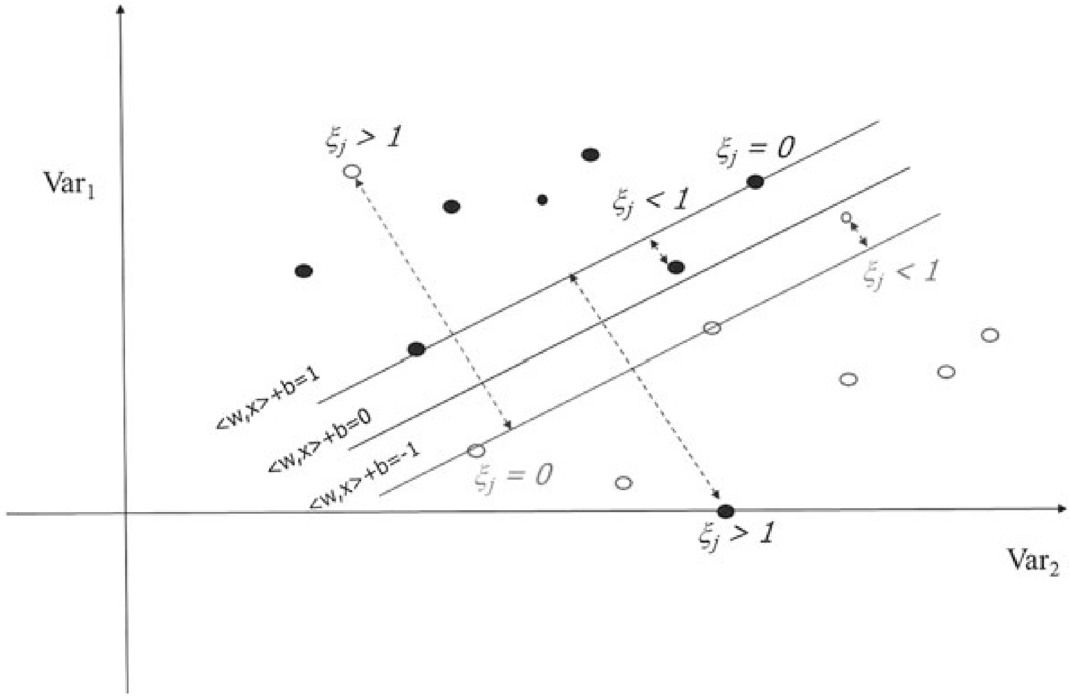

Figure 3: **Soft Vector Margin Hyperplane**

In figure 3, there are three linear equations:

- $< w, x > +b = 1$
- $< w, x > +b = 0$
- $< w, x > +b = -1$

The cases that the data points are on the first and last linear equations, the ξ_i, if it is between those lines, the $\xi_i \leq 0$ and of course the case that the data points are outside are aoutside the boundaries of those two equations, where $\xi_i \geq 0$.

This vector machine, includes one more variable, which represents the trade off parameter C, of the minimization and maximization of ξ_i parameters. So, the reform of the extended primal optimization problem is:

$$\min_{w,b,\xi} \frac{1}{2}||w||^2 + C\sum_{i=1}^{l} \xi_i \tag{34}$$

subject to $y_i(< w, x_i > +b) \geq 1 - \xi_i, \forall i \in [l]$
and $\xi_i \geq 0, \forall i \in [l]$

Also, the lagrangian function of the primal problem of the equation (34), will be:

$$L_{(w,b,\xi,a,\beta)} = \frac{1}{2} < w, w > +C\sum_{i=1}^{l} \xi_i - \sum_{i=1}^{l} a_i\{y_i(< w, x_i > +b) - 1 + \xi_i\} - \sum_{i=1}^{l} \beta_i\xi_i \tag{35}$$

$$L_{(w,b,a)} = \frac{1}{2} < w, w > +\sum_{i=1}^{l} a_i y_i(< w, \xi_i >) - b\sum_{i=1}^{l} a_i y_i - \sum_{i=1}^{l}(C - a_i - \beta_i)\xi_i \tag{36}$$

where $a = [a_i, ..., a_l]^T$ and $\beta = [\beta_i, ..., \beta_l]^T$, represent the matrices of the lagragian, non-negative multipliers.

So, as the KT theorem defines [13] the equations (18) through (23), reforms so that the normal point (w^*, b^*, ξ^*) to be optimun, if there is an existance of (a^*, b^*):

$$\frac{\partial L(w^*, b^*, \xi^*, a^*, \beta^*)}{\partial w} = 0 \tag{37}$$

$$\frac{\partial L(w^*, b^*, \xi^*, a^*, \beta^*)}{\partial \xi} = 0 \tag{38}$$

$$\frac{\partial L(w^*, b^*, \xi^*, a^*, \beta^*)}{\partial b} = 0 \tag{39}$$

$$a^*\{y_i(< w^*, x_i > +b^*) - 1 + \xi_i\} = 0, \forall i \in [l] \tag{40}$$

$$\beta_i\xi_i = 0, \forall i \in [l] \tag{41}$$

$$y_i(< w^*, x_i > +b^*) - 1 + \xi_i = 0, \forall i \in [l] \tag{42}$$

$$a_i^* \geq 0, \forall i \in [l] \tag{43}$$

$$\beta_i^* \geq 0, \forall i \in [l] \tag{44}$$

For the equations (37), (38), (39), yields that:

$$w^* = \sum_{i=1}^{l} a_i^* y_i x_i \tag{45}$$

$$C - a_i^* - \beta_i^* = 0, \forall i \in [l] \tag{46}$$

$$\sum_{i=1}^{l} a_i^* y_i = 0 \tag{47}$$

respectively. It also applies that:

$$0 \leq a_i^* \leq C \tag{48}$$

So, combining the equations (45), (46), (47), the Lagrangian equation takes the form:

$$L_{(w,b,a)} = \sum_{i=1}^{l} a_i - \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} a_i a_j y_i y_j < x_i, x_j > \tag{49}$$

The equation (45), contains the a_i^* which can be:

- $a_i^* \Longrightarrow \beta_i = C \neq 0 \Longrightarrow \xi_i = 0$, the x_i is correctly classified.
- $0 \leq a_i^* \leq C \Longrightarrow \beta_i \neq 0 \Longrightarrow \xi_i = 0 \Longrightarrow y_i(< w^*, x_i > +b^*) = 1$, where x_i is a support vector and is called unbound support vector.
- $a_i^* = C \Longrightarrow \beta_i = 0 \Longrightarrow \xi_i \neq 0 \Longrightarrow y_i(< w^*, x_i > +b^*) - 1 + \xi_i = 0$, where x_i is a support vector and is called bounded support vector.

The equation (32) in soft vector margin can be set as:

$$g_{(x)} = \sum_{i=1}^{l} a_i^* y_i < x_i, x > +b^* = \sum_{i \in SV} a_i^* y_i < x_i, x > +b^* \tag{50}$$

If $X_u = \{x_i \in S : 0 < a_i^* < C\}$ and $X_b = \{x_i \in S : a_i^* = C\}$ are the unbounded and bounded vectors and SV_u and SV_b are the sets of indices to the X_u and X_b ,respectively. Then, it applies:

$$y_i(< w, x_i^* > +b^*) = 1, \forall x_i \in X_u \tag{51}$$

For SV_u^+ as the support vectors of the $|X_u^+| = n_u^+$ where $X_u^+ = \{x_i \in X_u : y_i = +1\}$:

$$< w^*, x_i > +b^* = +1, \forall i \in SV_u^+ \tag{52}$$

For SV_u^- as the support vectors of the $|X_u^-| = n_u^-$ where $X_u^- = \{x_i \in X_u : y_i = -1\}$:

$$< w^*, x_i > +b^* = +1, \forall i \in SV_u^- \tag{53}$$

Adding the last two equations:

$$\sum_{i \in SV_u^+} < w^*, x_i > +b^* + \sum_{i \in SV_u^-} < w^*, x_i > +b^* = n_u^*(+1) + n_u^*(-1) \tag{54}$$

The ξ_i^* variable is set to provide the ξ_i parameters for the primal optimization problem

$$\xi_i^* = 1 - y_i(< w^*, x_i >) + b^*, \forall i \in SV_b \tag{55}$$

The optimal weight vector that is already mentioned, applies for the optimal separating hyperplane with the following equation:

$$\gamma^* = \frac{1}{||w^*||} \tag{56}$$

where γ^* is the maximun geometry margin.

The following two equations are the common choises of the kernel function [20][21][22][23] that is used in [13]:

- Polynomials of degree up to d

$$K(U,V) = (UV + 1)^d \tag{57}$$

- Gaussian Kernels

$$K(U,V) = exp(-\frac{||U - V||^2}{2\sigma^2}) \tag{58}$$

4 Chapter 3

4.1 Bitcoin

Bitcoin is the most important Cryptocurrency. This means that it is not controled by banks or economic systems. Although, it allows to bitcoin traders to complete exchanges over the internet.[24][25]

Accualy, the bitcoin currency become known after its price value got an expodential rise. At January 2017 it was almost 1.000 US dollars per bitcoin. At the end of same year it reached its maximun of almost 17.500 US dollars per bitcoin. So, if someone bought a bitcoin at the start of the year 2017 and sell it at the end of the same year, he or she would have 16.500 US dollars profit, without the brokers fee.[26][27][28]

This caught the attention of stock traders, tech experts and eventually the world. Not only bitcoins but also other cryptocurrencies start rising their price vaues, like etherium. It became obvious that buying and selling bitcoins is very close to stock exchanges. The price of bitcoin is not stable. There are methods and strategies that claim to predict its future price.

Of course for this to be accompliched, it is needed a large group of history data. Specifically, the data are every second historical prices and bitcoin limit order book features. For the need of this paper, the data that are going to be used, are from the website CS224W. They are available to public in a large text file. Every row contains the id of transaction, the sender, the recipient, the value (BTC) and the timestamp. If there are more than one sender or recipient, the database adds more lines with the same transaction identification.[29][30][31][32][33][34]

4.2 Data Separation

4.2.1 Bitcoin partitions

The file that is going to be used for this project has the name AllOrderBooks and its file extention is CSV. These data start at the date 18/05/2015 and ends at the date 29/07/2015. Every row of data for this CSV file is separated through timestamps every one second. Each data is vary important for the experiment.[35]

But, there are so many data to manage that the program that performs the machine learning experiment, is certain that is going to slow the results. Because of the small computer technical abilities, the table that contains all those rows of data is separated into smaller tables with the same columns.

These smaller tables are called partitions, contain only 40.000 rows data and are being called separately. This number is not the same in every partition, but it varies with average the pre - mentioned number. For the necessity of this experiment, there are 34 different smaller tables. The whole tables is almost 5 milion row data. In this point, it has to be mentioned that even 40.000 data rows of data are hard to manage with a mediocre computer system. It take a lot of time for the system to run all of these data.

4.2.2 Separation

Each partition has data that can be managed separately. Every row has one second time distance and it can be previewed in the column timestamp.

The rows are ordered by the data type DateTime, from the oldest to the newest. To begin the structure of the solution for the problem, we need to define two time windows:

- δt : Past Window.
- Δt : Future Window.

There are also going to be used two tables of data:

- The table with the actual, real values(past or present values).
- The table with the predicted, imaginary values(future values).

For the needs of this experiment and based on the computer technical abilities, the past windows are set to 50, 100, 200 rows. The future window is always set to 600 rows. So, if there are 200 rows for the past window and 600 rows for the future window, in a partition of 40.000 rows in average, there are left 39.200 in between.

It is valiable to mention at this point that the largest the windows are set, the hardest it becomes for the program to solve the problem. It requires more time. So, to find a better solution of course more profitable, it is needed the best case scenario in choosing past and future windows. Taking as example the previous settings, the values 600 and 200 take a lot of time. So, this is not the best choice, in addition the result is the best from the available scenarios.

Figure 4: **Partition Separation**

As it is already mentioned in the previous chapter, there are two posible spread crossing events:

- **Positive Events:** Upward or Downward. They are represented as (+1) and the positive value.
- **Negative Events:** Non - Upward or Non - Downward. They are represented as (-1) and the negative value.

For the case of upwardspread crossing, each value may predict the next value in the second table with the use of Support Vector Machine. This prediction would be either +1 if it is an Upward / Downward event(P) or -1 if it is Non - Upward / Non - Downward event(N). But there are two more cases, which come from the prediction if it is true(T) or false(F).

So, in summary there are four scenarios:

- **TP:** Predicted Upward / Downward and fulfilled.
- **FN:** Predicted Non - Upward / Non - Downward and not fulfilled.
- **FP:** Predicted Upward / Downward and not fulfilled.
- **TN:** Predicted Non - Upward / Non - Downward and fulfilled.

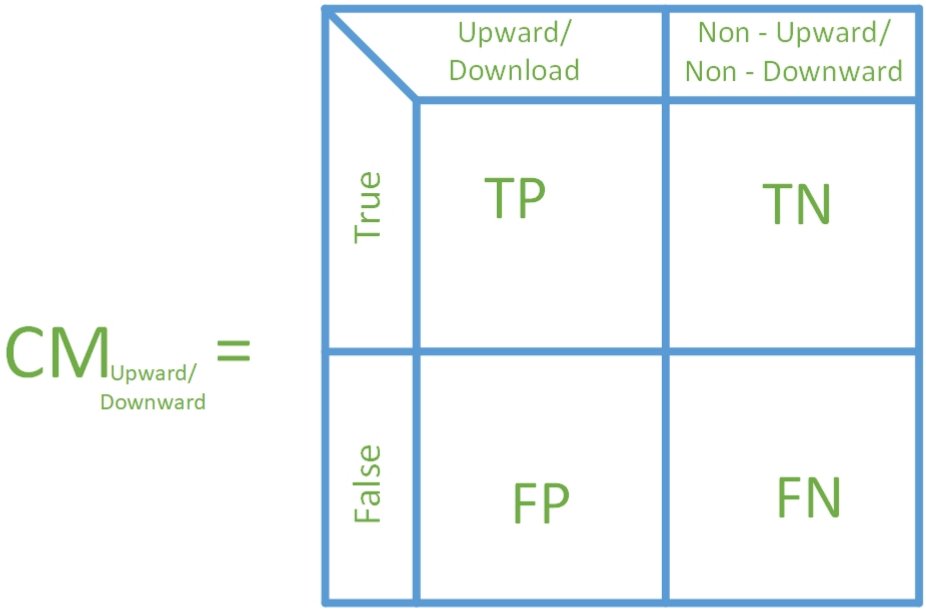

Figure 5: **Table Upward - Downward cases**

The case TP is the best case scenario and in addition the most profitable. The table $CM_{Upward/Downward}$ should contain $TP + FN + FP + TN$ values, which are the average partition minus δt(past window), minus Δt(Future Window).

If the value TP is greater than any of the other values indivitualy, the results are representive and the experiment predicted the most profitable value correctly at good percentage.

To be more exact, the complete percentage of $TP + TN + FN + FP$ is in summary 100%. The most profitable cases are the TP and the TN.

- If $TP + TN \gg FN + FP$, then this case is the best and most porfitable. The predicted values can provide reliable data for the buyer either to buy or to sell. These data can also prevent possible loss if a stock price fell abruptly.
- If $TP + TN \geq FN + FP$, then this case provide possible profit but not as certain as the previous case.
- If $TP + TN < FN + FP$, then this case is the worst scenario. The false future data are less and that provides only uncertainty for the buyer or seller to decide.

4.2.3 Features

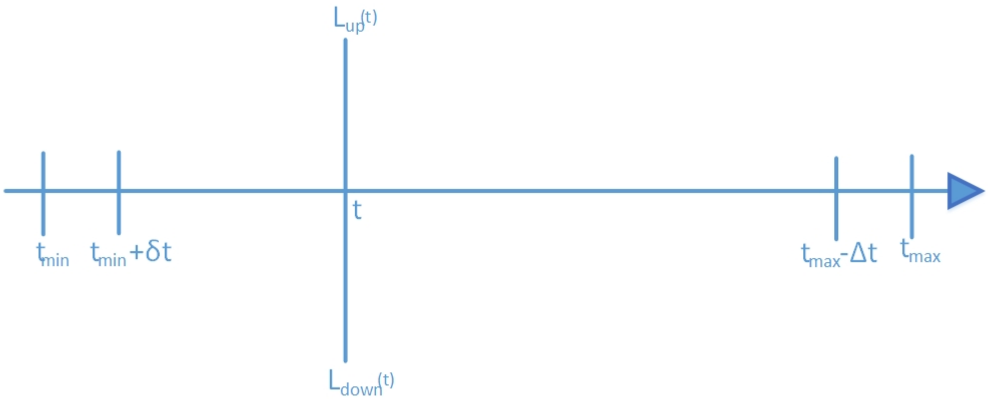

Figure 6: **Random Time In Timesacale**

The available values of timestamps and also primary keys in the table in use, belong in the summation $t_{total} = t_{max} - t_{min}$. But as it is already mentioned in the previous chapter, there are two more partitions, the past window (δt) and the future window (Δt). So, for a random time t, it applies: $t \in [t_{min} + \delta t, t_{max} - \Delta t]$.

In the array of prediction, we import the values $L_{up}(t)$ and $L_{down}(t)$, for each case.

$$L_{up}(t) \left\{ \begin{array}{ll} +1, & V_{up}(t) > 0 \\ -1, & \text{otherwise} \end{array} \right\} \tag{59}$$

$$L_{down}(t) \left\{ \begin{array}{ll} +1, & V_{down}(t) > 0 \\ -1, & \text{otherwise} \end{array} \right\} \tag{60}$$

The $L_{up}(t)$ or $L_{down}(t)$ is +1 if the time t is an upward event or downward event, respectively. The $V_{up}(t) \in [0, 1]$ and $V_{down}(t) \in [0, 1]$ represent the percentage of upward or downward event in the next Δt movements. So, the program calculates at which percent is possible to have a positive event. The longer it occurs an upward / downward event, the more posible it is to happen again.

4.2.4 $V_{threshold}$

There is also one more parameter that is called $V_{threshold}(V_{thres})$. If $V_{thres} = \theta$, where $\theta \in [0, 1]$, then:

$$L_{up}(t;\theta) \begin{Bmatrix} +1, & V_{up}(t) > \theta \\ -1, & \text{otherwise} \end{Bmatrix} \tag{61}$$

$$L_{down}(t;\theta) \begin{Bmatrix} +1, & V_{down}(t) > \theta \\ -1, & \text{otherwise} \end{Bmatrix} \tag{62}$$

The variable θ can be set by the user. It is a very reliable and useful parameter, because it can show the percentage of the successful events and when these are happening. It can also be considered as a filter that lets only the events with V_{up} or V_{down} greater than θ.

- If $\theta = 0$, then $count(L_{up}(t)) = count(L_{up}(t;\theta))$ or $count(L_{down}(t)) = count(L_{down}(t;\theta))$. There is no filter to prevent positive events from happening, so all of the events are numbered.
- If $\theta \in (0, 1)$, then $count(L_{up}(t)) \geq count(L_{up}(t;\theta))$ or $count(L_{down}(t)) \geq count(L_{down}(t;\theta))$. In this case, the paramenter θ filters the $L_{up}(t)$ or $L_{down}(t)$ values, so it previews only those with possibility to succeed, higher than the set value. The counts may be equal only in case that all of the L_{up} or L_{down} have V greater than θ.
- If $\theta = 1$, then $count(L_{up}(t)) \geq count(L_{up}(t;\theta))$ or $count(L_{down}(t)) \geq count(L_{down}(t;\theta))$. This scenario is difficult to happen and it needs high successful ratio.

For this experiment, the θ variable will be set to $10\%, 50\%$ and 80%. So, in summary we will manage 9 cases, which are presented in figure 7.

V_{thres} / Past Window	10%	50%	80%
50	Future Window = 600 Past Window = 50 V_{thres} = 10 %	Future Window = 600 Past Window = 50 V_{thres} = 50 %	Future Window = 600 Past Window = 50 V_{thres} = 80 %
100	Future Window = 600 Past Window = 100 V_{thres} = 10 %	Future Window = 600 Past Window = 100 V_{thres} = 50 %	Future Window = 600 Past Window = 100 V_{thres} = 80 %
200	Future Window = 600 Past Window = 200 V_{thres} = 10 %	Future Window = 600 Past Window = 200 V_{thres} = 50 %	Future Window = 600 Past Window = 200 V_{thres} = 80 %

Figure 7: **Experiment Cases**

5 Experimental Results

The whole timespan available is separated into partitions. Firstly, we will deal with the first partition which contains the 40000 rows of data, that corresponds to different timestamps. Firstly, for every partition the algorithm labels if it is an Upward or Downward event with +1 for the positive case and -1 for the negative case. So, there are two arrays with the same length (40.000 minus the future window) and available values -1 and +1. One array is for the upward events $L_{Up}(t)$ and the other is for the downward events $L_Down(t)$.

There are also two more arrays that stack the volume of each event. Those are the $V_{Up}(t)$ and the $V_{Down}(t)$ arrays. The volume is being calculated by running the whole 40.000 values minus the future window and check if there is an Upward or Downward spread crossing within the future window at each timestamp. In fact, it counts the length of the events at each case.

5.1 Partition 1

5.1.1 Scenario 1.1 (Future window = 600, Past Window = 50, $V_{thres} = 10\%$)

In figure 8, it is presented the volume of Downward events that are in the next future time window. The future window is set to 600 and the past window to 50 timestamps. The figure 10 presents the volume of the Upward event in the same partition. The Upward events contain asterisks at the peak

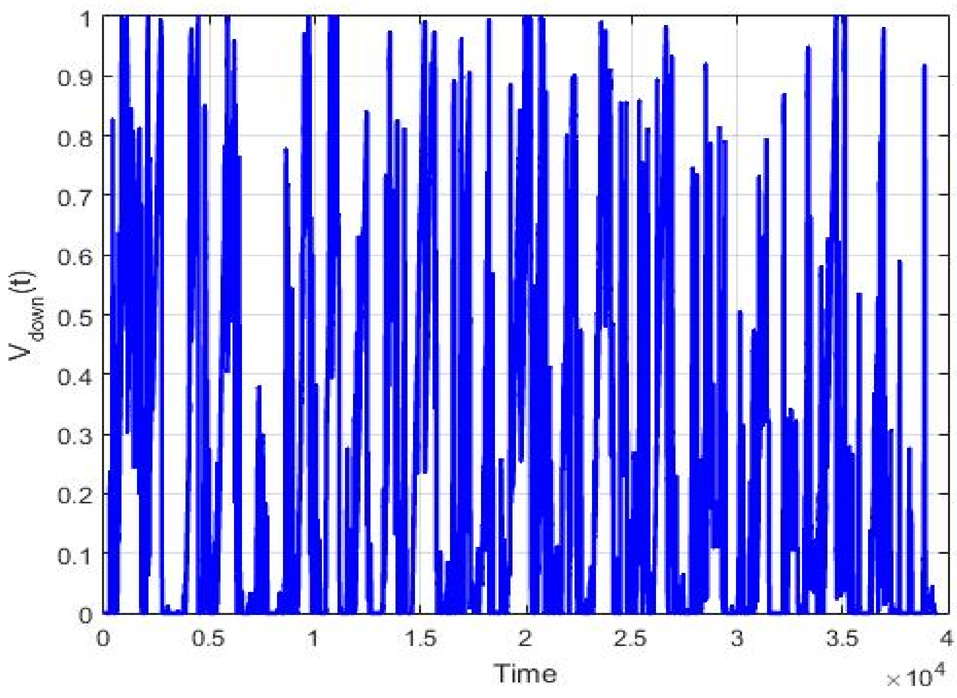

Figure 8: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50

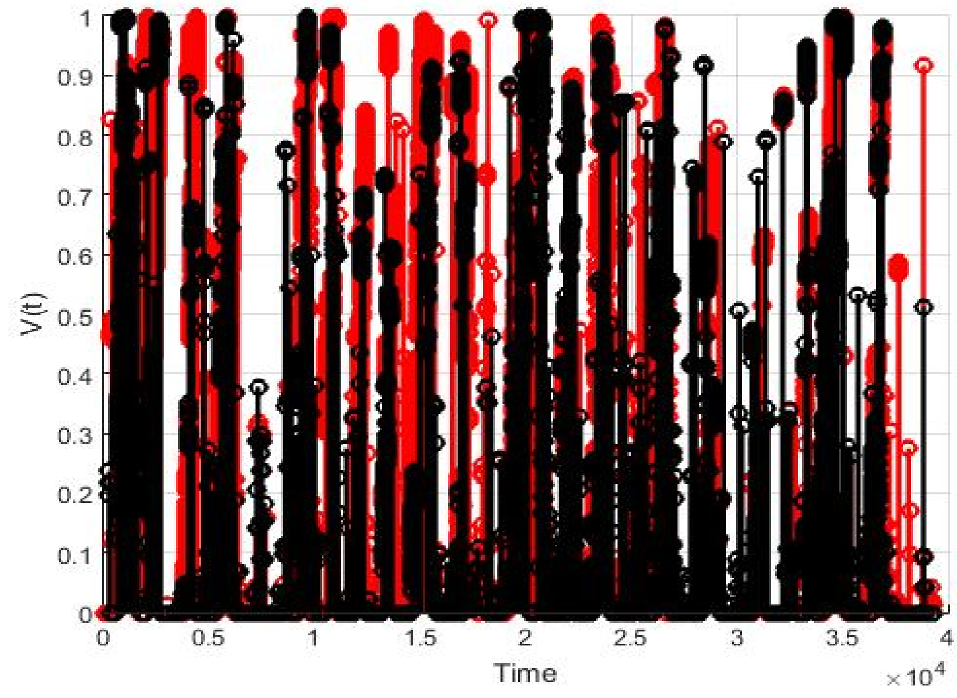

Figure 9: **Plot volume of Downward spread-crossing events after Classification within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50

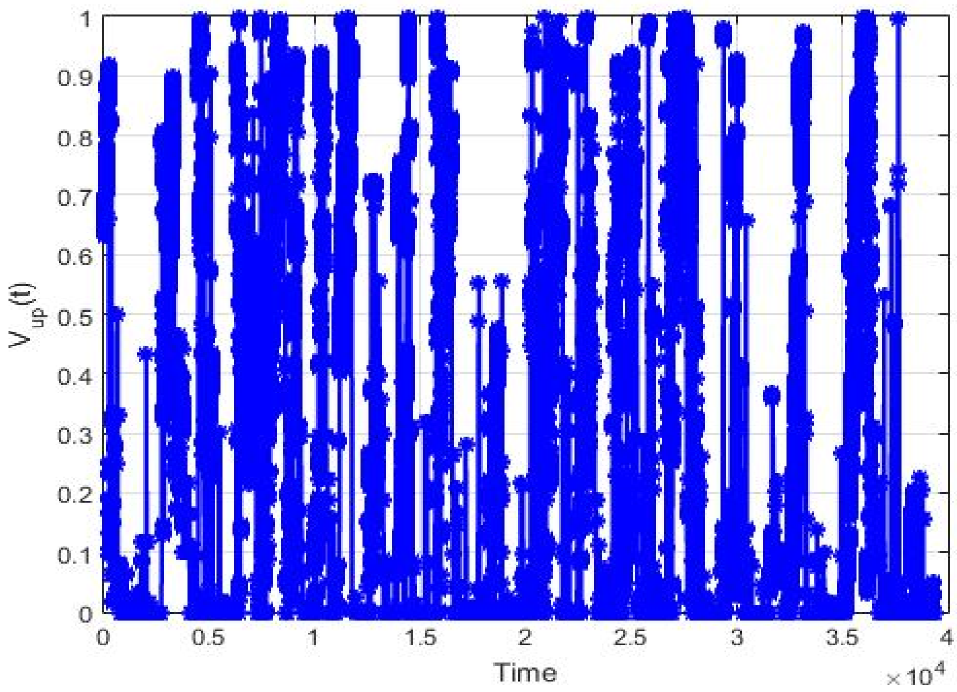

Figure 10: **Plot volume of Upward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50

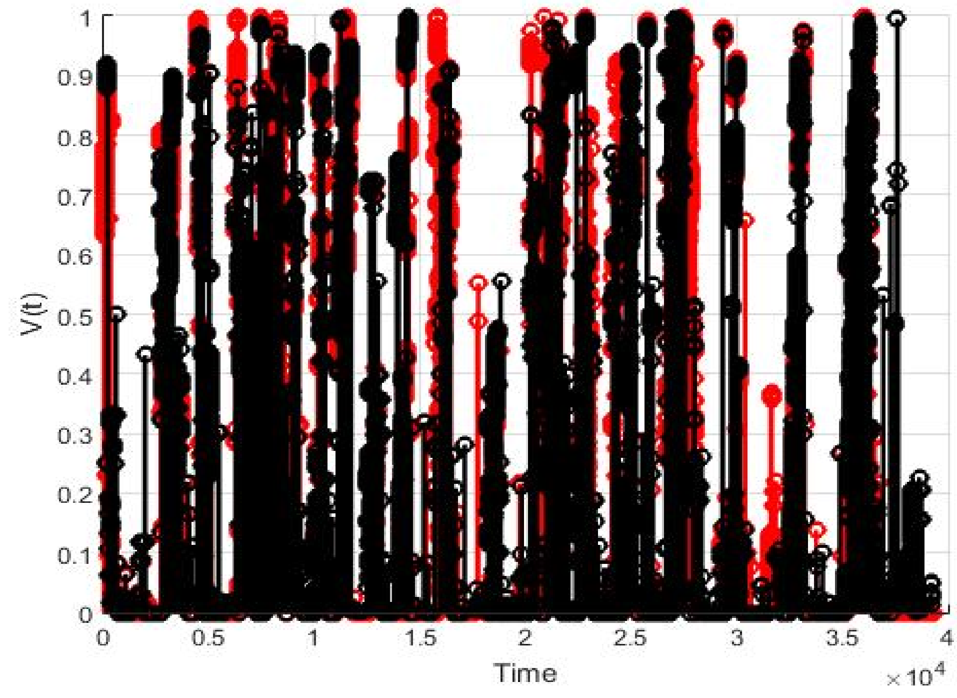

Figure 11: **Plot volume of Upward spread-crossing after Classification events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50

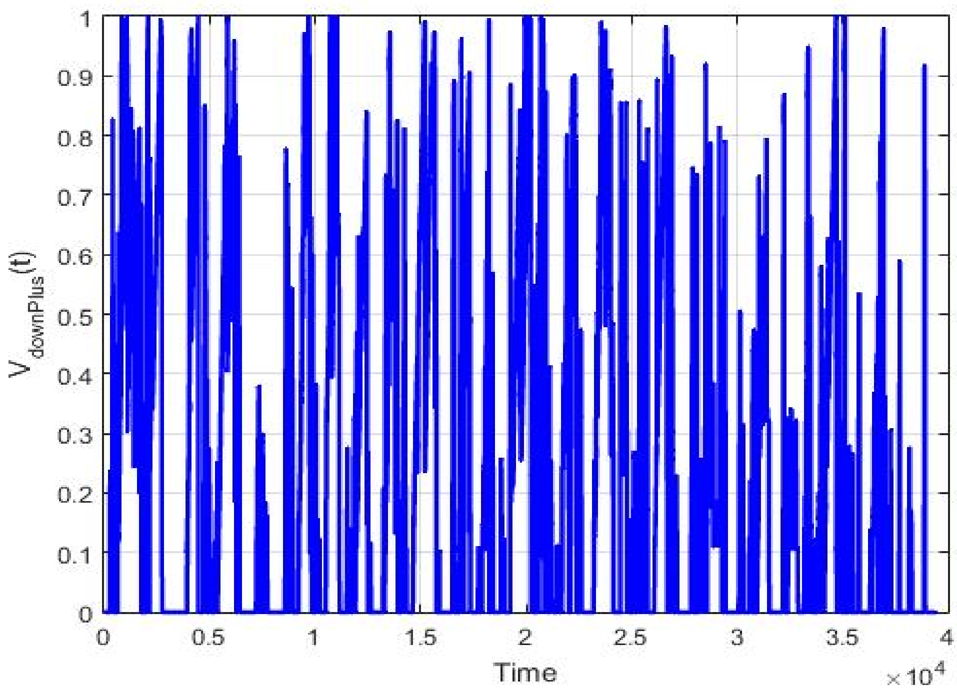

Figure 12: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

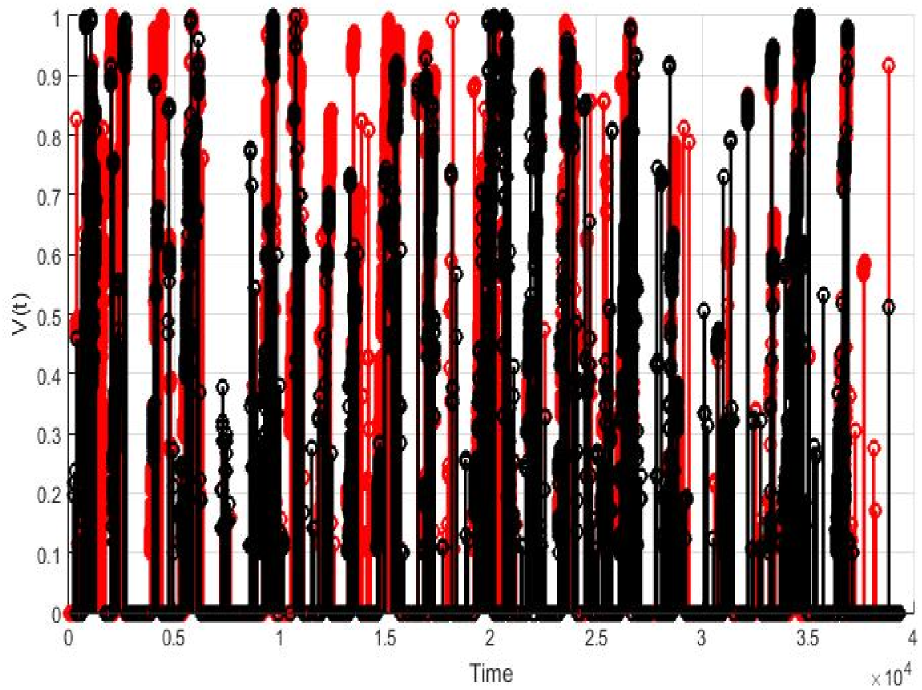

Figure 13: **Plot volume of Downward spread-crossing after Classification events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

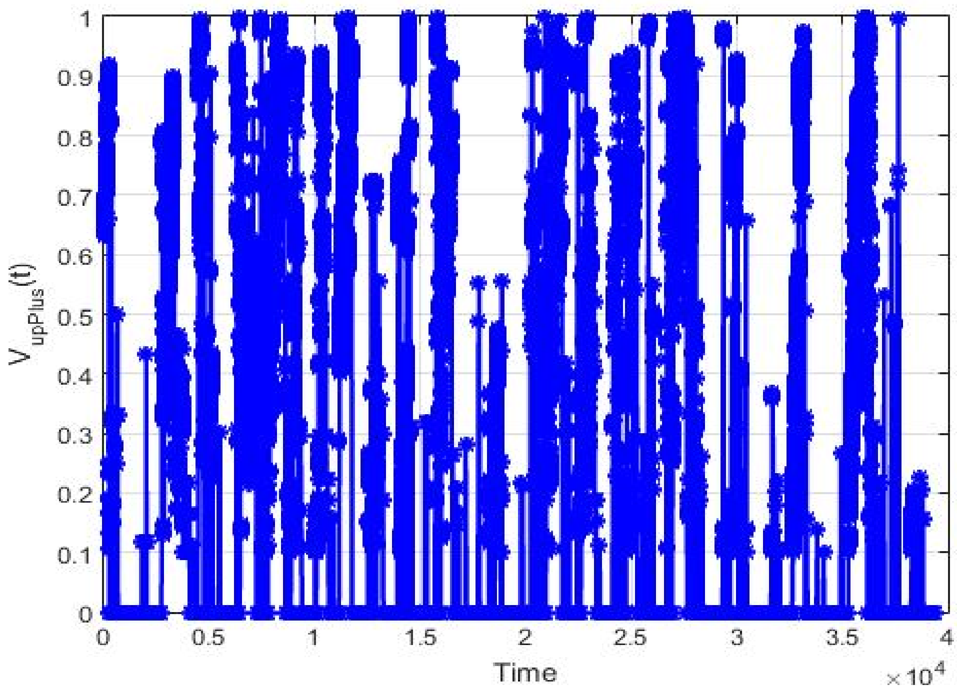

Figure 14: **Plot volume of Upward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

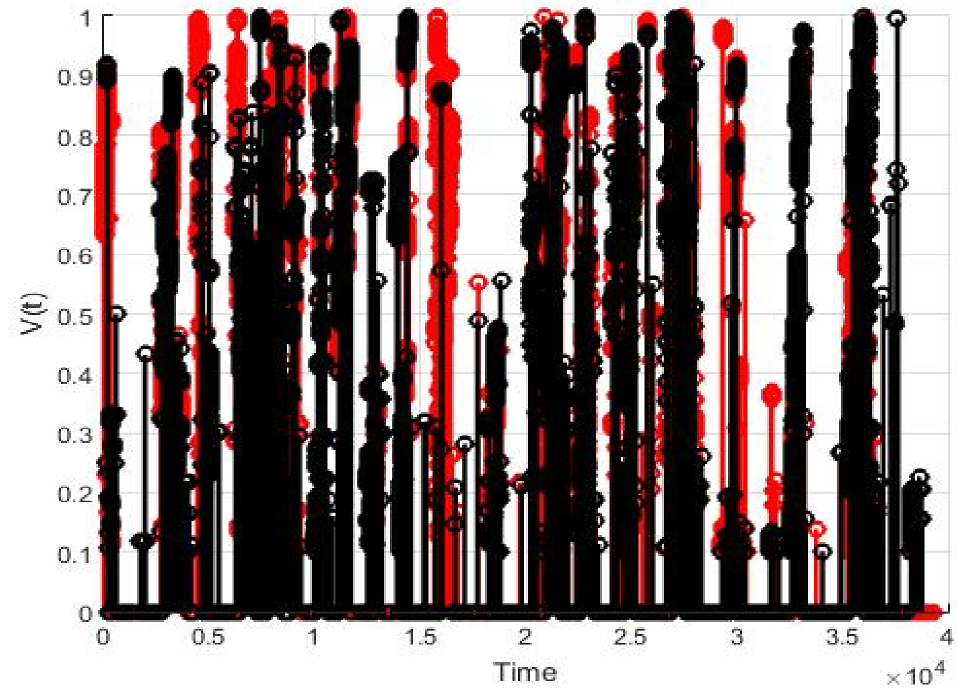

Figure 15: **Plot volume of Upward spread-crossing after Classification events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

This figure has in X-Axis the time dimension and in Y-Axis the volumes. These results can be expoited in various ways. For example if the volume keeps its value over 0.7(70%) for more than 100 timestamps, it is possible to have an Downward event in the next timestamp. On the contrary, if the valuem keeps under 0.2 for a lot of time, it becomes obvious that a Downward event is less possible.

The figure 9, is a plot that presents the Downloading Spread - Crossing Volume $V_{Down}(t)$ as a function of time within the time interval by colouring the data points ccording to the validity of the classification process. Correctly classified points will have the red colour and incorrectly classified points will have the black colour.

The function SVMPredict is used for estimating the training / testing accuracy of the Gaussian SVM classifier. With the assistance from the [36][37][38][39][40], it is possible to use the SVMPredict function. After this function, the classified results are presented in figure 9, for the Downward spread crossing events classification and in **??**, for the Upward spread Crossing events classification. The training percentage that we are using for classification is $0.5(50\%)$.

In the subsection 4.2.4, it has been mentioned a new parameter($\theta \in [0, 1]$). This works as a filter that presents only the volumes greater than the defined parameter. For this example the V_{thres} is set to 10% or 0.1, equivalent. In figure 12 are presented the Downward events with volume over 0.1 and in figure 14, the Upward events, respectively. The figures 13 and 15 are the relative classifications.

Its obvious that with the V_{thres} set to 0.1, the output events are almost the same with the volumes without θ ($\theta = 0$). Although, there are some some timestamps that the volume is under 0.1, so in this case the volume equals to zero. This immediately means that most of the events are record over 0.1. The θ is going to be set to higher values in next subsection to present more distinct results.

After building and running the program, the results of algorith show that the percentage of Downward events is: $24329/39350 \simeq 0.618$ and Upward events is $23646/39350 \simeq 0.601$.

Partition 1(Future window = 600, Past window 50, $V_{thres} = 10\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39350	61.8%
Percantage of upward spread-crossing events	23646/39350	60.1%
Percantage of downward spread-crossing events($V_{thres} = 10\%$)	17743/39350	45.1%
Percantage of upward spread-crossing events($V_{thres} = 10\%$)	15812/39350	40.2%

Table 1: Partition 1(Future window = 600, Past window 50, $V_{thres} = 10\%$)

The classification is separated into two parts. One that is used for training and one for the testing in SVM classification. With the training percentage set to 50%, the results are also separated. The timestamps for Partition 1 where 39.350, after the subtraction of the used windows. Now, in table 2, the timestamps are 19.675 after the division into testing / training, for the needs of SVM.

Although, the V_{thres} allows only volumes with value over 10%, which means that the counter of timestamps would be smaller, it is still the same. This happens for the reason that volumes under 10% are set to zero. So, it is almost certain that in training and testing, for those values the prediction will be true. But, by removing those values the training/testing would not be accurate. Without edit, the result data with V_{thres}, provide an approximate solution to the problem. The data with the highest volume are more possible to be predicted. That is why, percentages with V_{thres} are more accurate.

Partition 1(Future window = 600, Past window 50, $V_{thres} = 10$%, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	13337/19675	67.8%
Percantage of downward spread-crossing events(testing)	11900/19675	60.5%
Percantage of upward spread-crossing events(training)	12937/19675	65.8%
Percantage of upward spread-crossing events(testing)	13742/19675	69.8%
Percantage of downward spread-crossing events($V_{thres} = 10$%,training)	10296/19675	52.3%
Percantage of downward spread-crossing events($V_{thres} = 10$%,testing)	10905/19675	55.4%
Percantage of upward spread-crossing events($V_{thres} = 10$%,training)	9947/19675	50.6%
Percantage of upward spread-crossing events($V_{thres} = 10$%,testing)	11406/19675	58.0%

Table 2: Partition 1(Future window = 600, Past window 50, $V_{thres} = 10$%, Training / Testing)

The CM matrix contains data from the figure 5, is calculated in the following table. They are separated to train and test data. Also, there are the V_{thres} data that let only the data points with volume over 10% in partition 1 in each case.

Ctrain Down = $\begin{matrix} 4881 & 2432 \\ 3906 & 8456 \end{matrix}$	Ctest Down = $\begin{matrix} 4100 & 3608 \\ 5771 & 6196 \end{matrix}$
Ctrain Up = $\begin{matrix} 6349 & 1109 \\ 6666 & 5551 \end{matrix}$	Ctest Up = $\begin{matrix} 5836 & 2410 \\ 6360 & 5069 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 6633 & 3568 \\ 3170 & 6304 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 6442 & 4964 \\ 4764 & 3505 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 9337 & 1889 \\ 4044 & 4405 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 7856 & 4456 \\ 3813 & 3550 \end{matrix}$

In the next scenarios, the result data are presented with the same row.

5.1.2 Scenario 1.2 (Future window = 600, Past Window = 100, $V_{thres} = 10\%$)

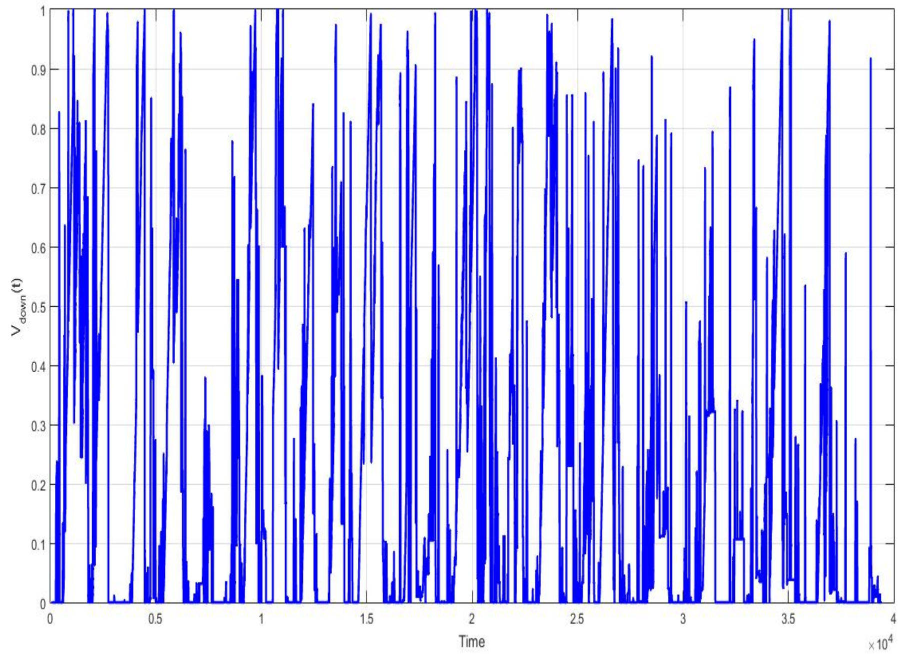

Figure 16: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

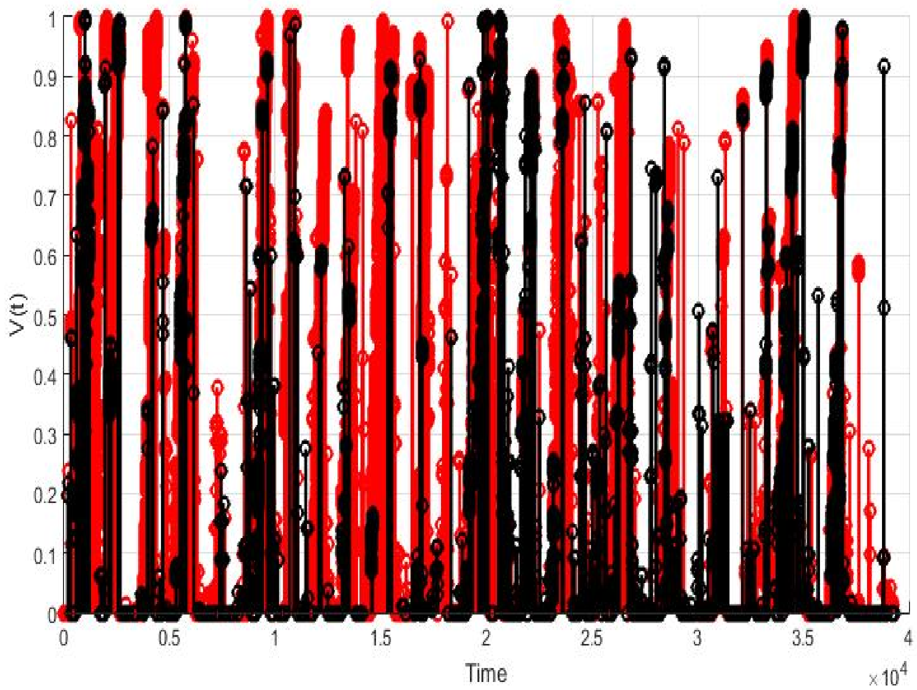

Figure 17: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

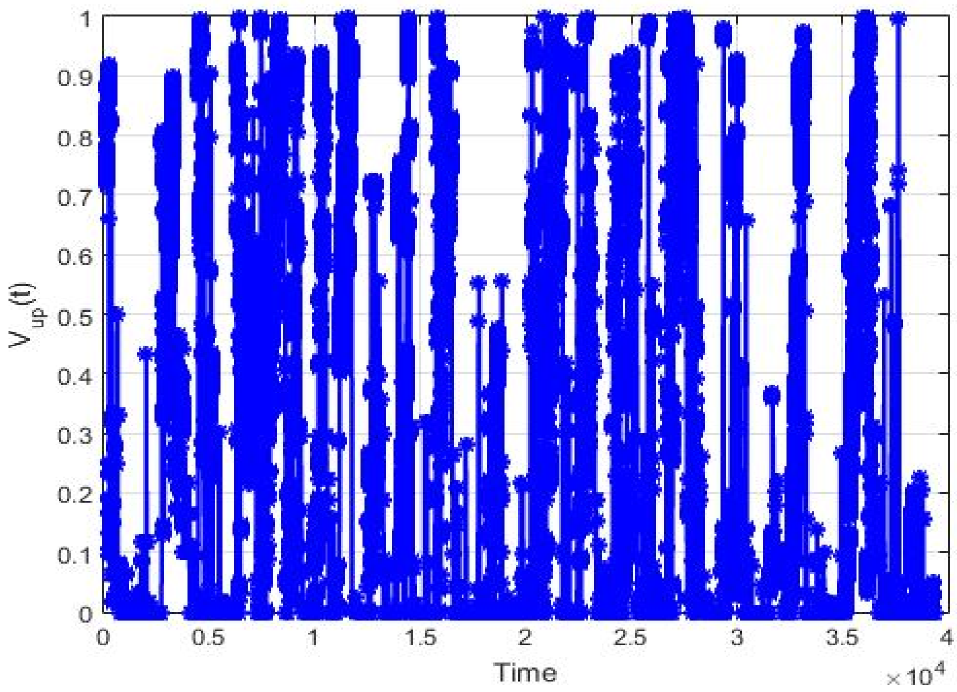

Figure 18: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

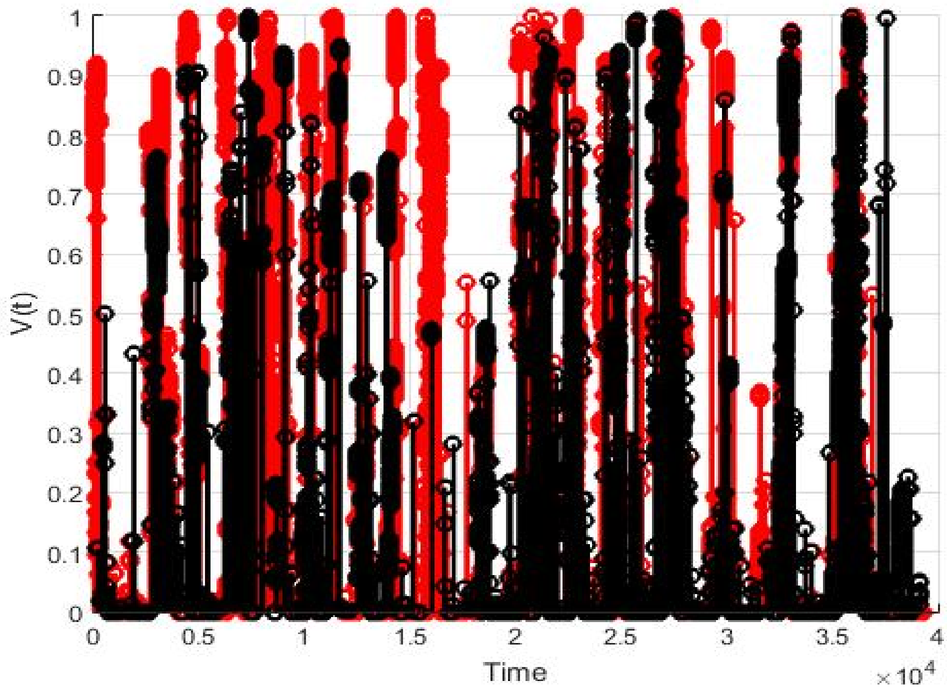

Figure 19: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

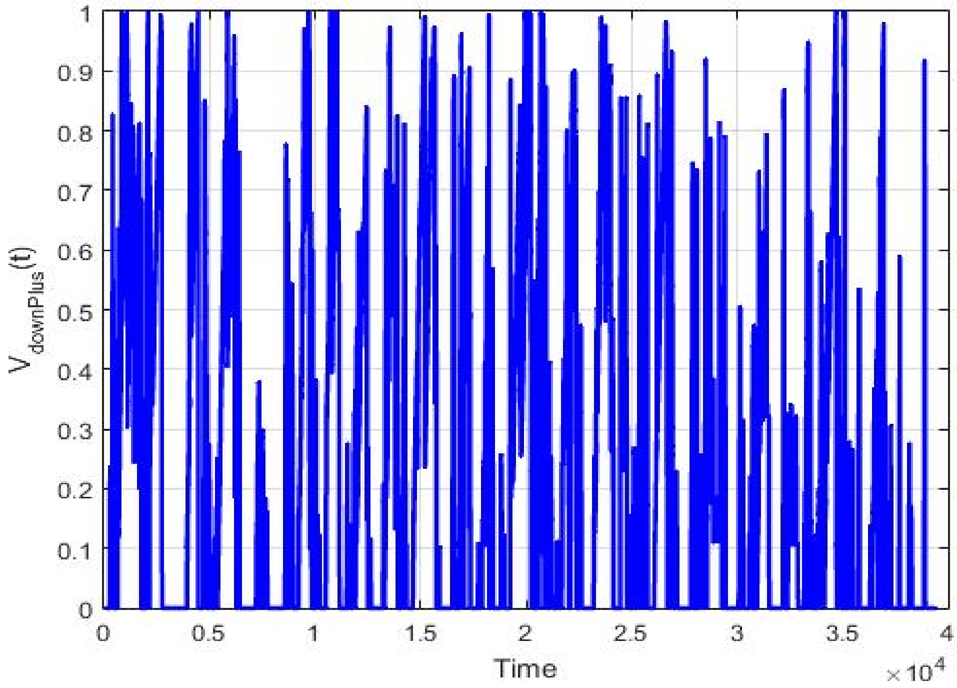

Figure 20: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

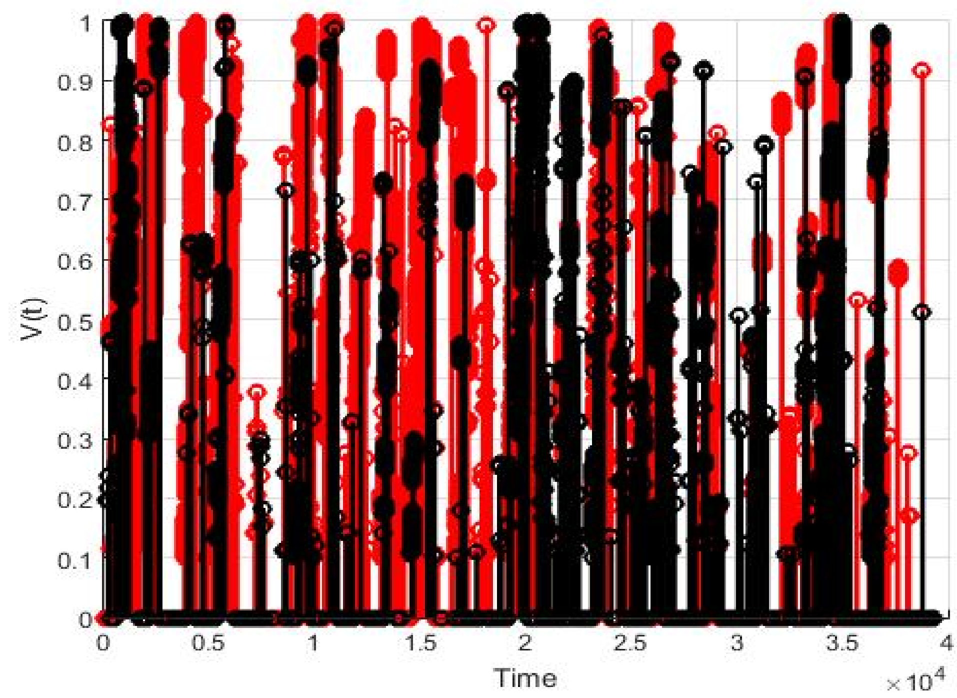

Figure 21: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

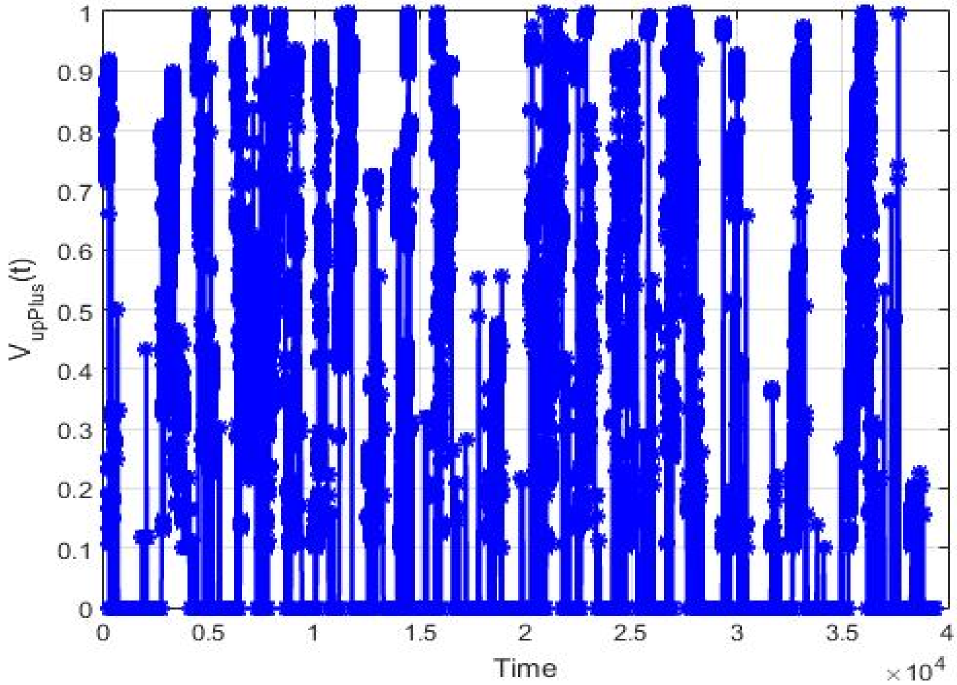

Figure 22: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

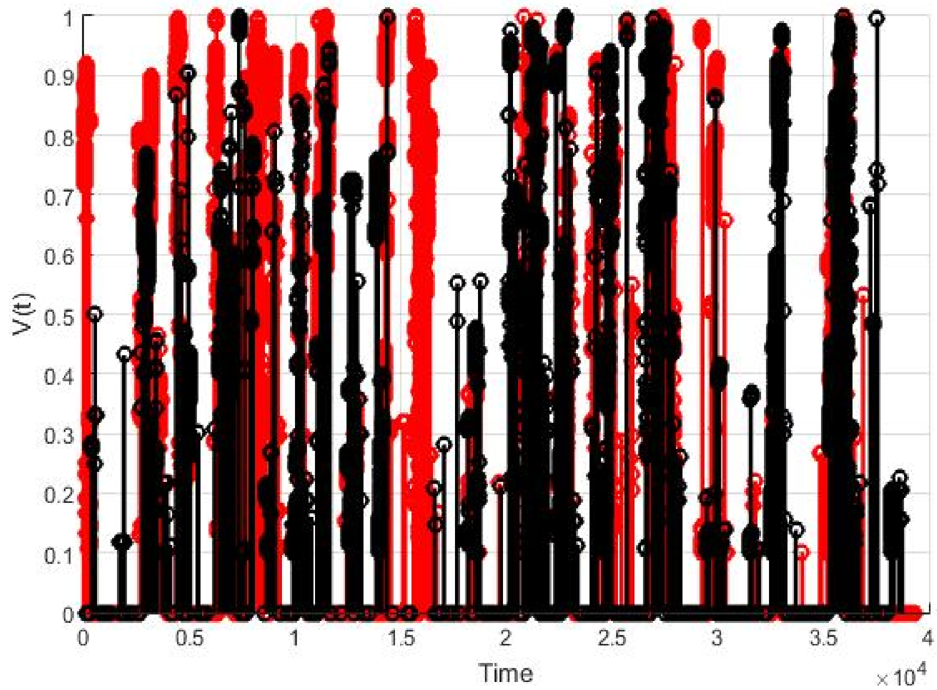

Figure 23: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

Partition 1(Future window = 600, Past window 100, $V_{thres} = 10\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39300	61.9%
Percantage of upward spread-crossing events	23596/39300	60.0%
Percantage of downward spread-crossing events($V_{thres} = 10\%$)	17743/39300	45.1%
Percantage of upward spread-crossing events($V_{thres} = 10\%$)	15762/39300	40.1%

Table 3: Partition 1(Future window = 600, Past window 100, $V_{thres} = 10\%$)

Partition 1(Future window = 600, Past window 100, $V_{thres} = 10\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	15521/19650	79.0%
Percantage of downward spread-crossing events(testing)	14670/19650	74.7%
Percantage of upward spread-crossing events(training)	14692/19650	74.8%
Percantage of upward spread-crossing events(testing)	15931/19650	81.1%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,training)	11037/19650	56.2%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,testing)	11243/19650	57.2%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,training)	9896/19650	50.4%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,testing)	11167/19650	56.8%

Table 4: Partition 1(Future window = 600, Past window 100, $V_{thres} = 10\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 5093 & 2170 \\ 1959 & 10428 \end{matrix}$	Ctest Down = $\begin{matrix} 3370 & 4338 \\ 4275 & 7667 \end{matrix}$
Ctrain Up = $\begin{matrix} 6743 & 735 \\ 4245 & 7927 \end{matrix}$	Ctest Up = $\begin{matrix} 4963 & 3263 \\ 5144 & 6280 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 6924 & 3227 \\ 1731 & 7768 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 5559 & 5847 \\ 3907 & 4337 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 10235 & 1016 \\ 2703 & 5696 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 7744 & 4543 \\ 3940 & 3423 \end{matrix}$

5.1.3 Scenario 1.3 (Future window = 600, Past Window = 200, $V_{thres} = 10\%$)

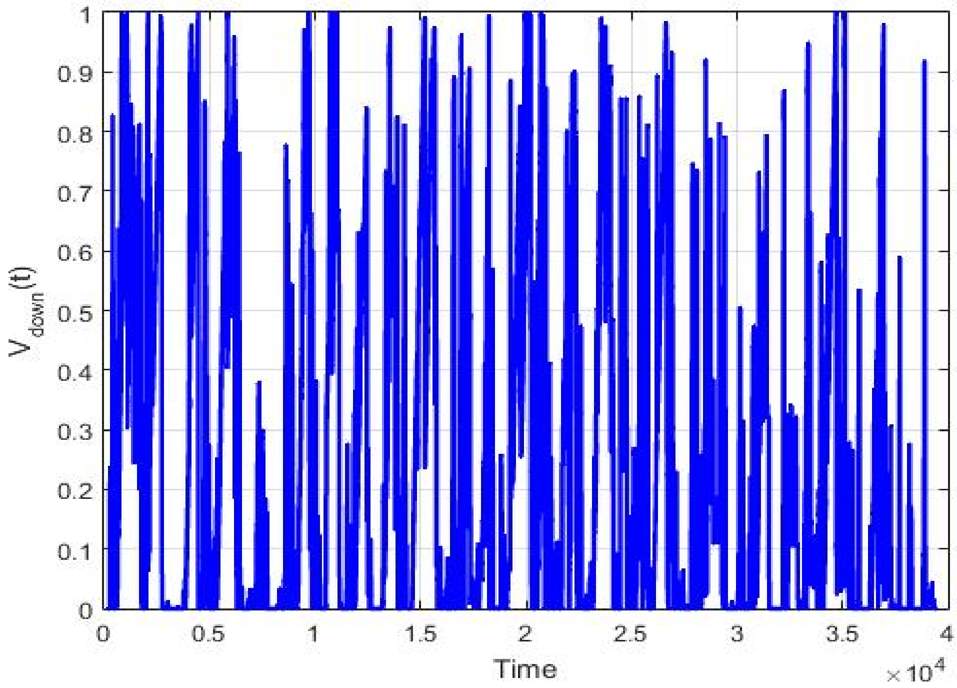

Figure 24: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

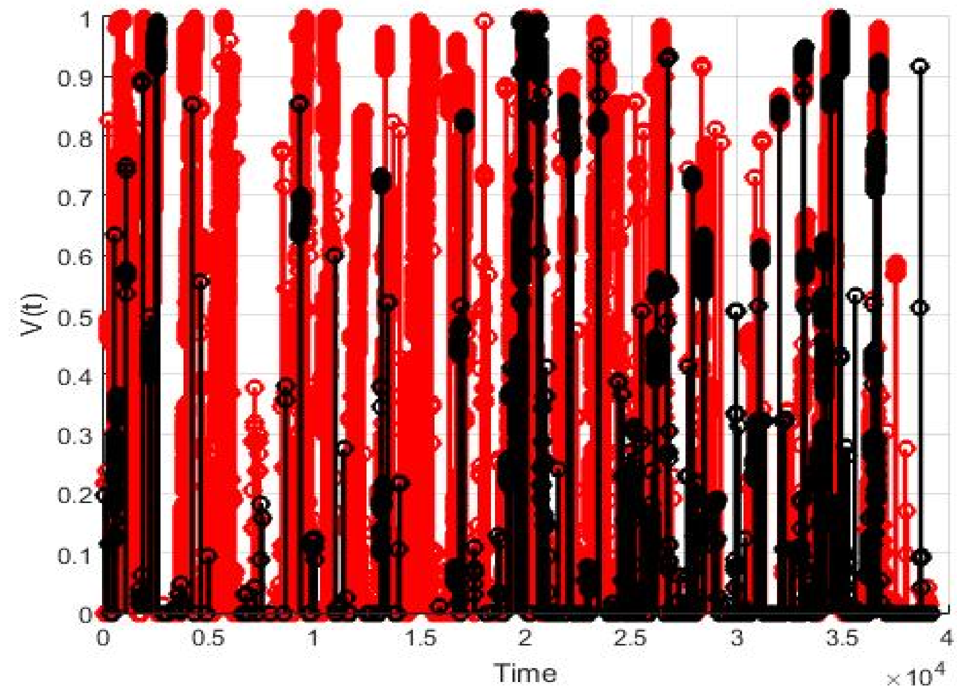

Figure 25: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

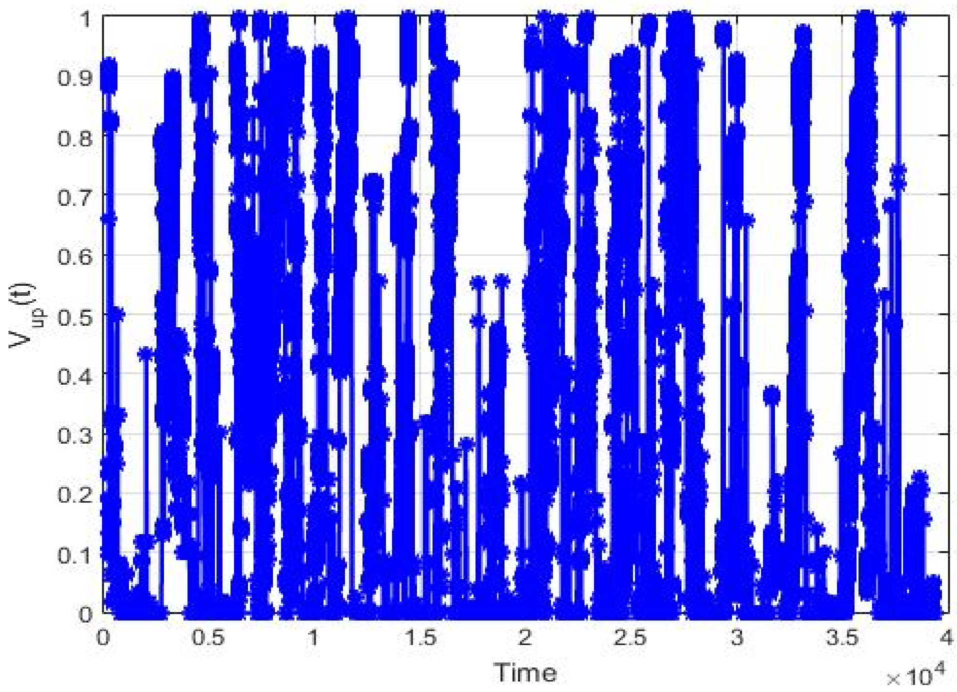

Figure 26: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

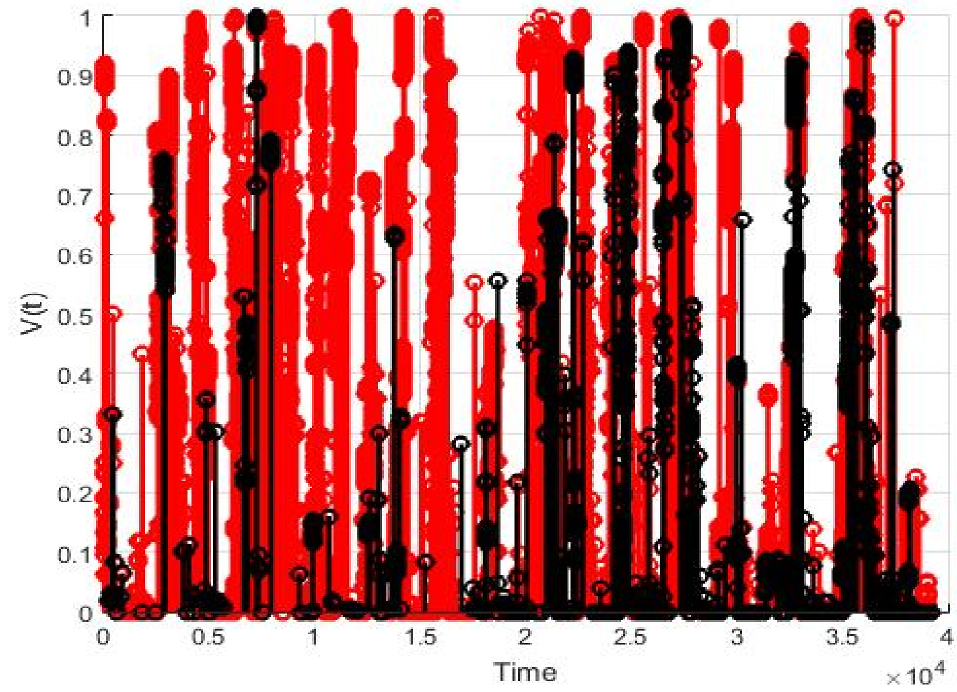

Figure 27: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

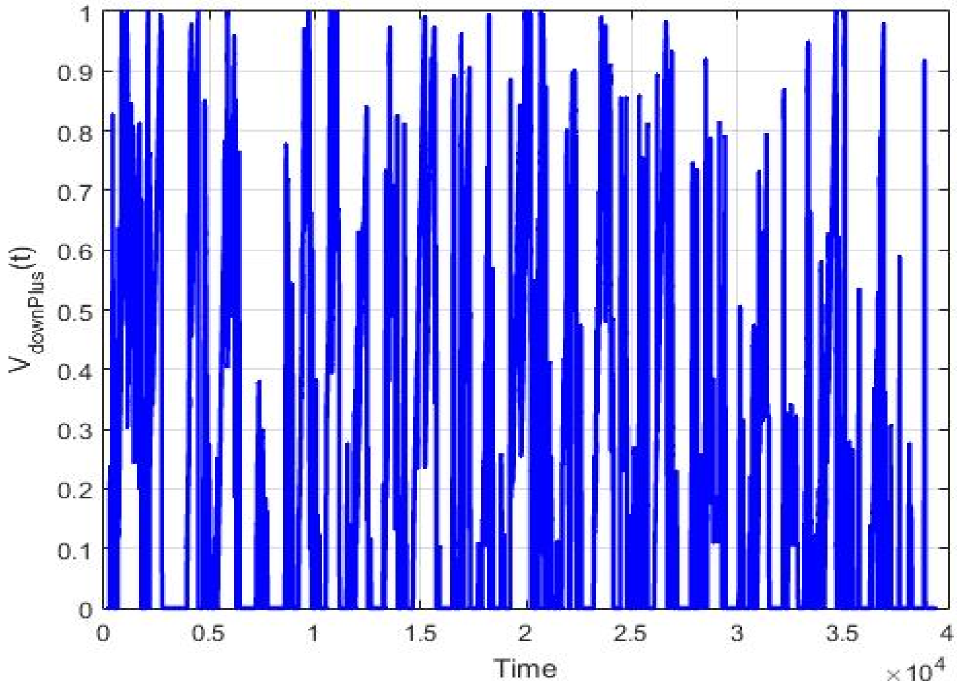

Figure 28: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

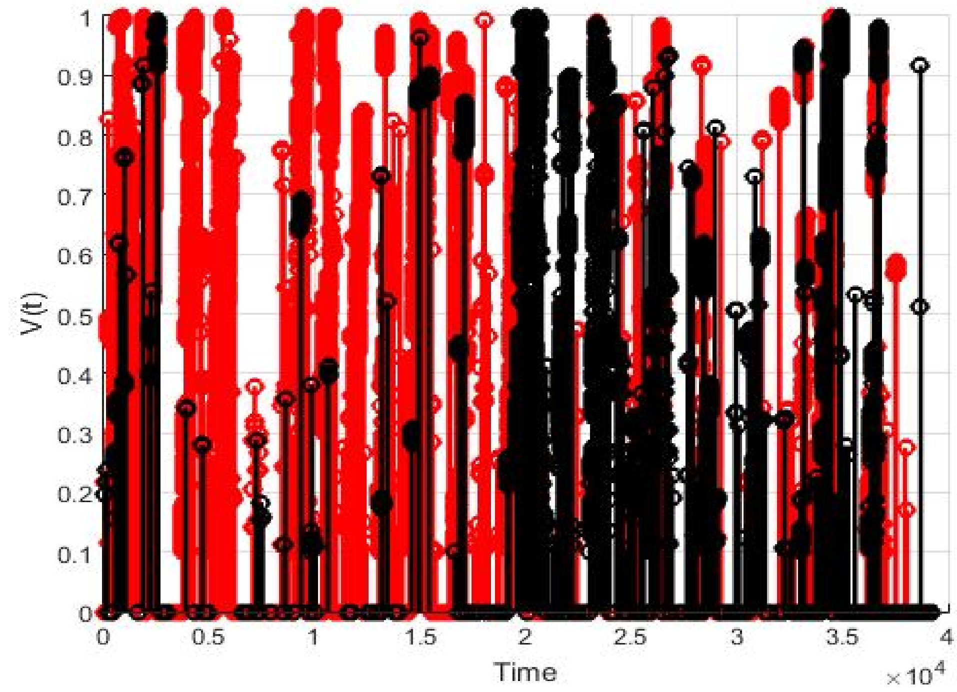

Figure 29: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

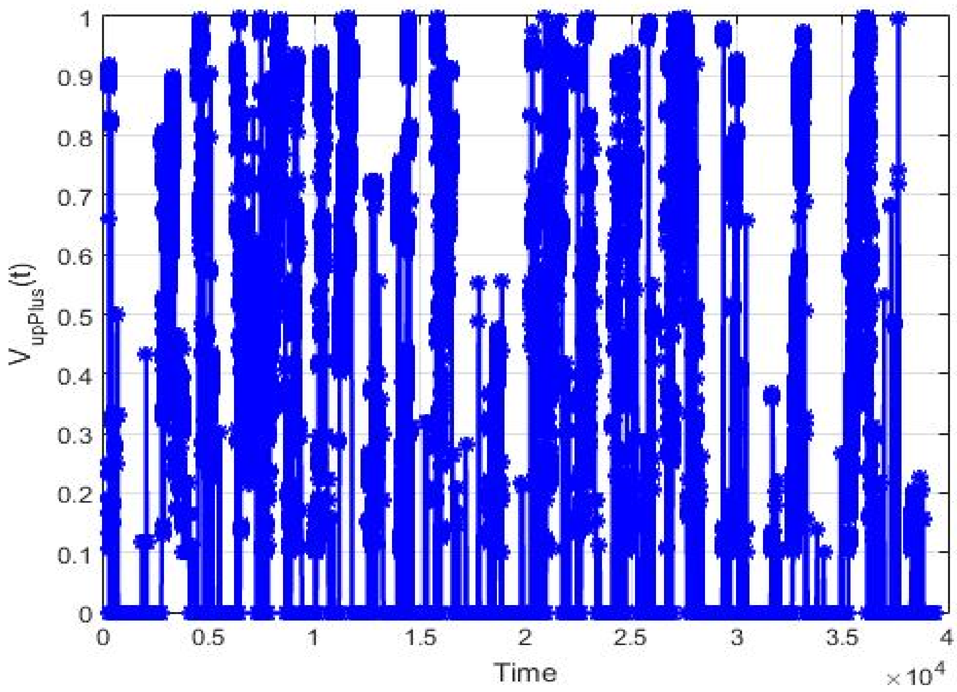

Figure 30: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

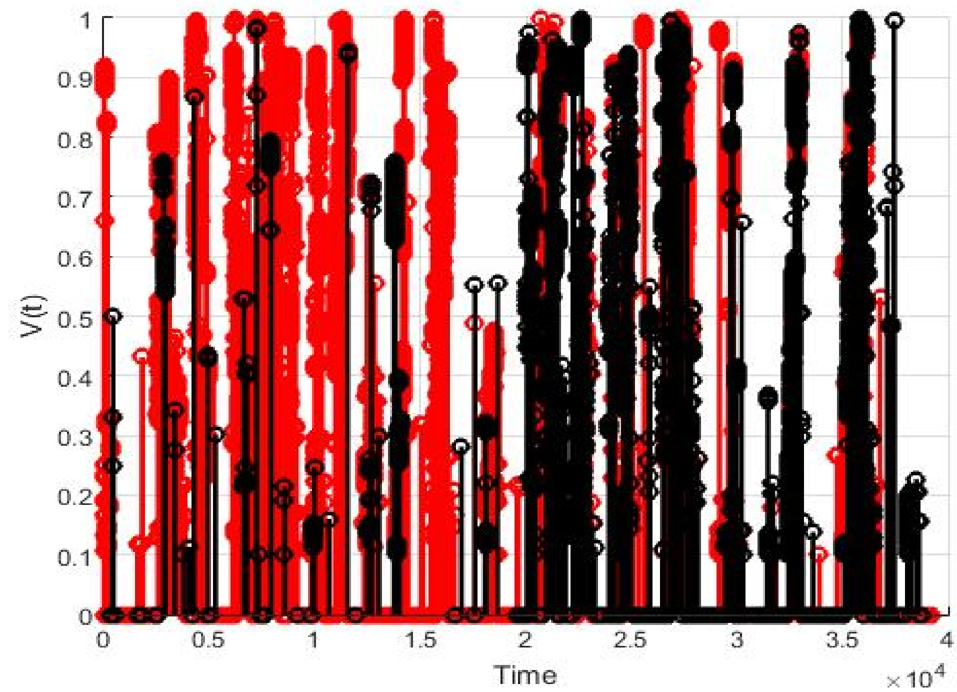

Figure 31: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

Partition 1(Future window = 600, Past window 200, $V_{thres} = 10\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39200	62.1%
Percantage of upward spread-crossing events	23496/39200	59.9%
Percantage of downward spread-crossing events($V_{thres} = 10\%$)	17743/39200	45.3%
Percantage of upward spread-crossing events($V_{thres} = 10\%$)	15662/39200	40.0%

Table 5: Partition 1(Future window = 600, Past window 200, $V_{thres} = 10\%$)

Partition 1(Future window = 600, Past window 200, $V_{thres} = 10\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	18068/19600	92.2%
Percantage of downward spread-crossing events(testing)	18550/19600	94.6%
Percantage of upward spread-crossing events(training)	17579/19600	89.7%
Percantage of upward spread-crossing events(testing)	18811/19600	96.0%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,training)	11061/19600	56.4%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,testing)	11261/19600	57.5%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,training)	9328/19600	47.6%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,testing)	11210/19600	57.2%

Table 6: Partition 1(Future window = 600, Past window 200, $V_{thres} = 10\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 6478 & 685 \\ 847 & 11590 \end{matrix}$	Ctest Down = $\begin{matrix} 2595 & 5113 \\ 3426 & 8466 \end{matrix}$
Ctrain Up = $\begin{matrix} 7212 & 295 \\ 755 & 11338 \end{matrix}$	Ctest Up = $\begin{matrix} 2649 & 5548 \\ 2791 & 8612 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 8522 & 1529 \\ 492 & 9057 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 6335 & 5071 \\ 5201 & 2993 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 11133 & 163 \\ 626 & 7678 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 8187 & 4055 \\ 4335 & 3023 \end{matrix}$

5.1.4 Scenario 1.4 (Future window = 600, Past Window = 50, $V_{thres} = 50\%$)

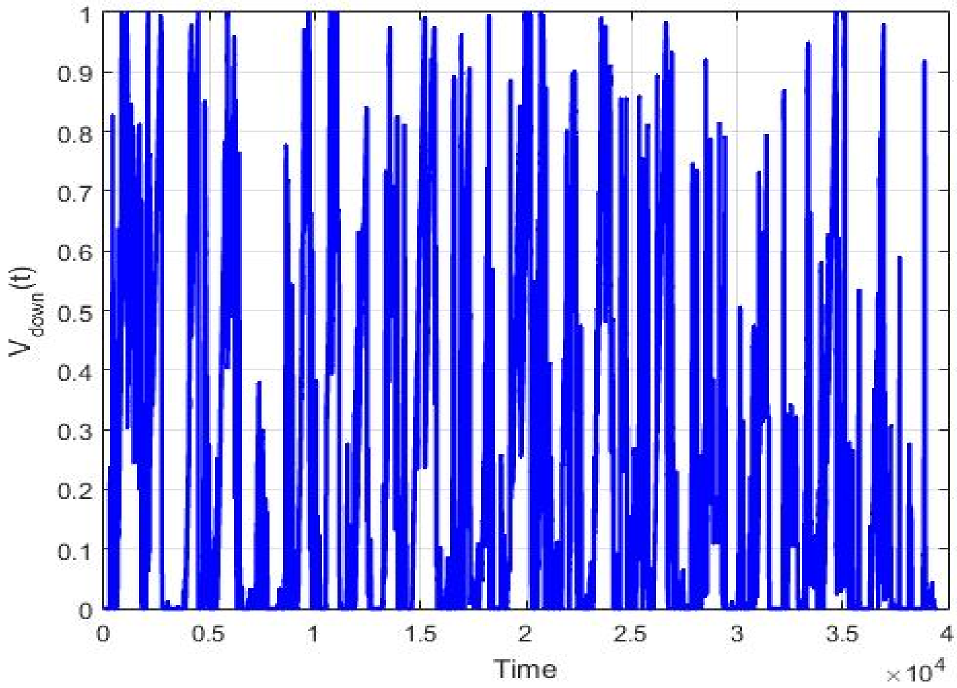

Figure 32: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

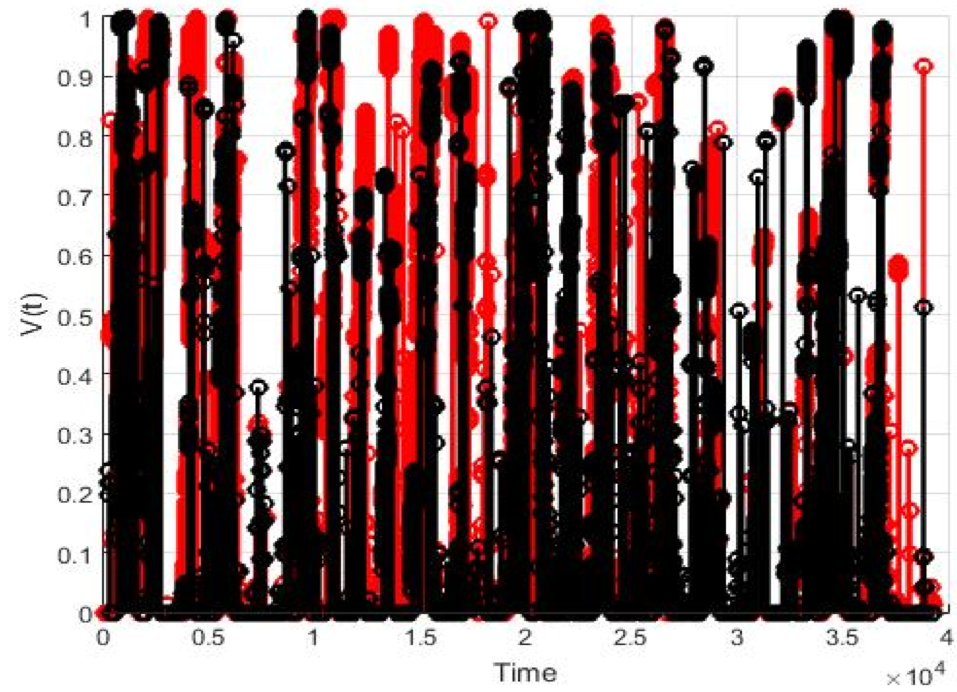

Figure 33: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

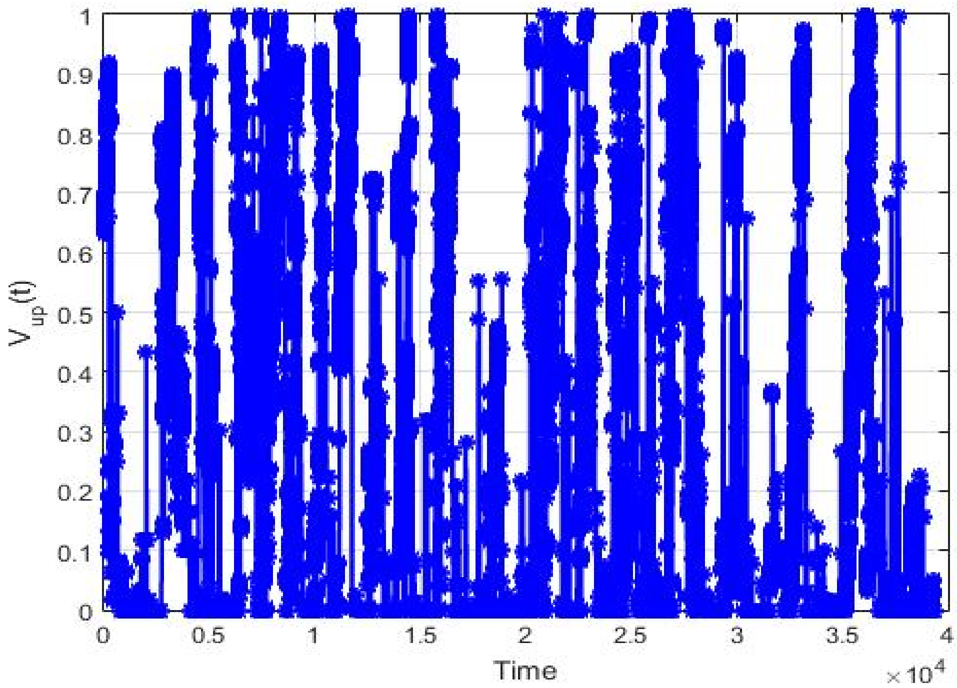

Figure 34: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

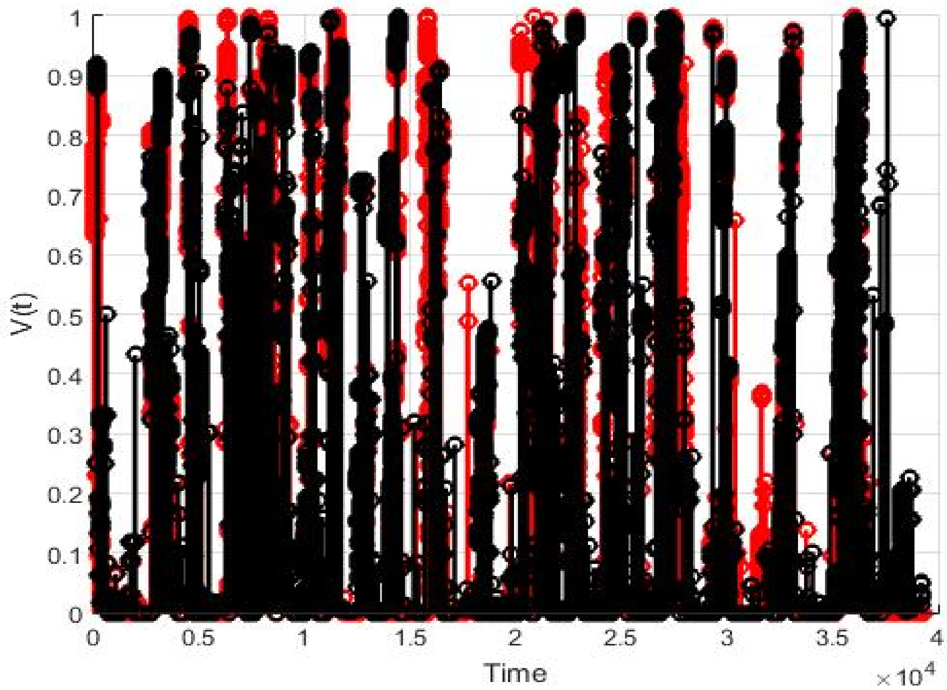

Figure 35: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

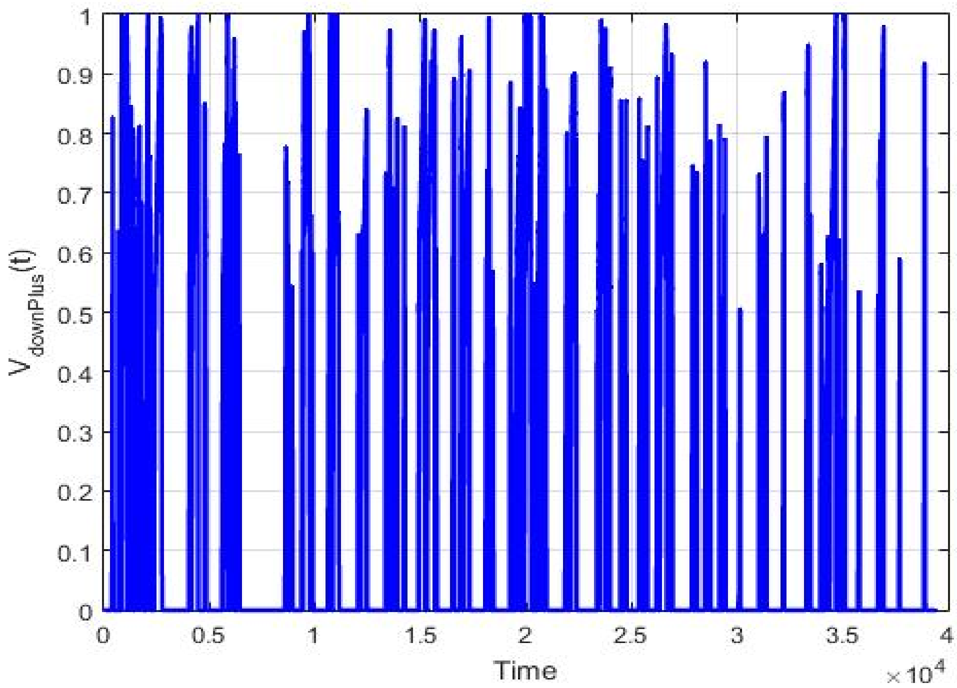

Figure 36: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

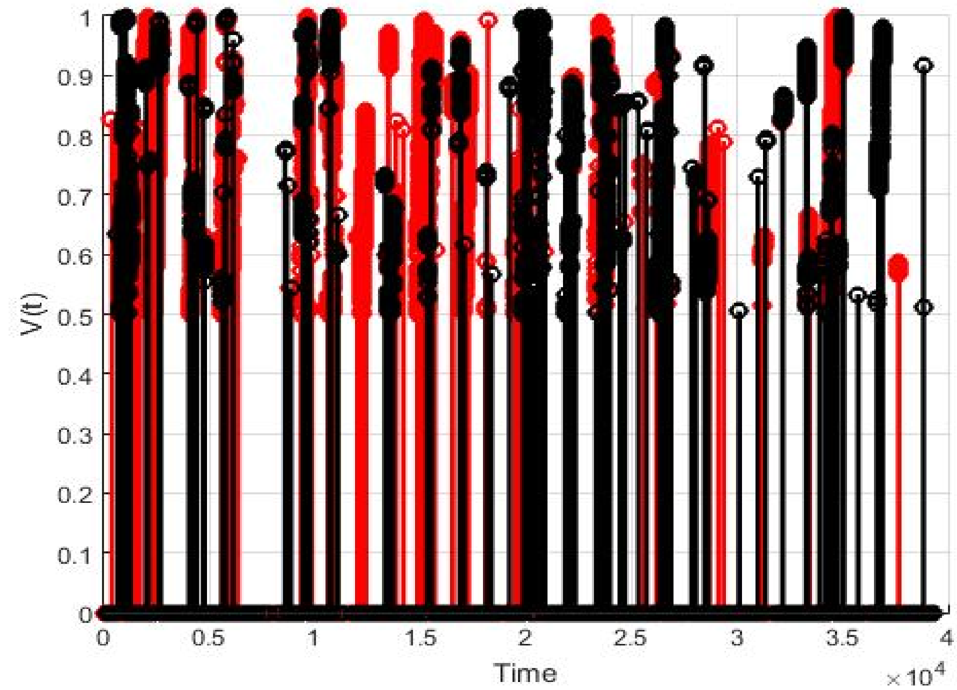

Figure 37: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

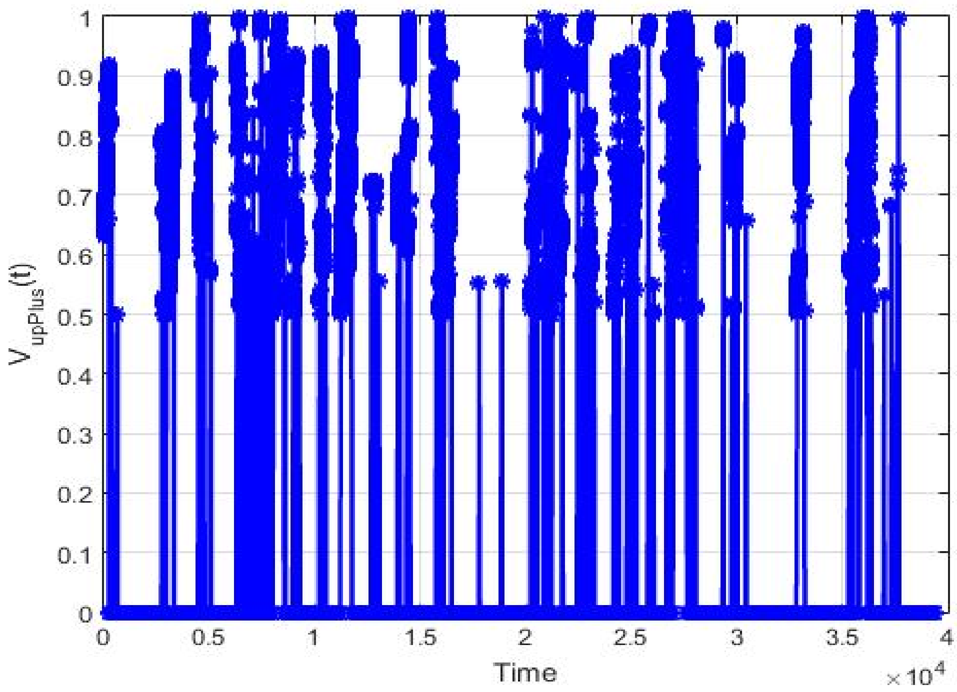

Figure 38: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

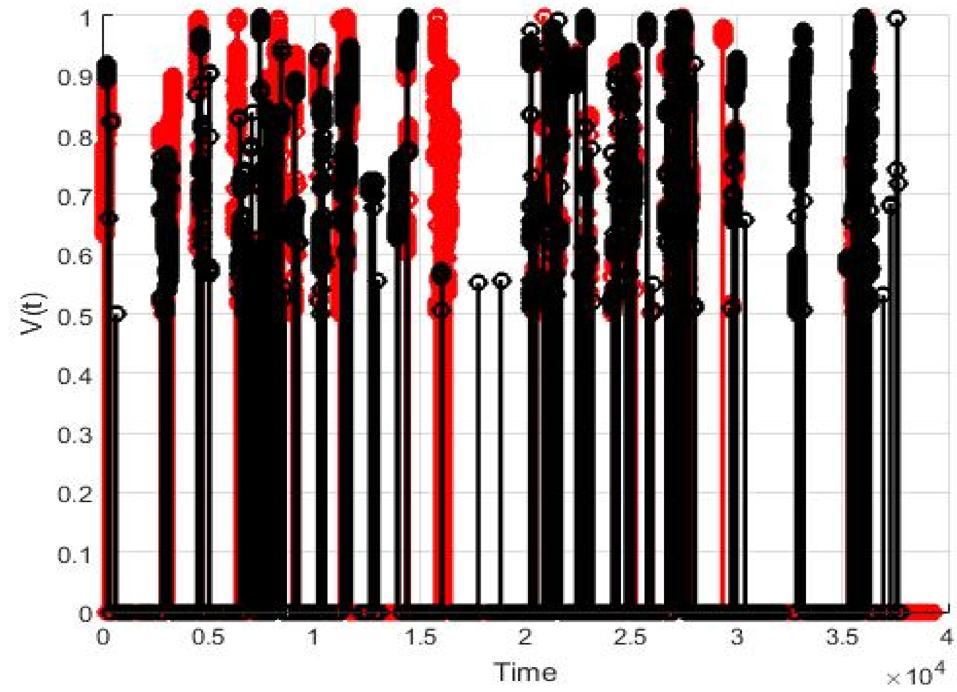

Figure 39: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

Partition 1(Future window = 600, Past window 50, $V_{thres} = 50\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39350	61.8%
Percantage of upward spread-crossing events	23646/39350	60.1%
Percantage of downward spread-crossing events($V_{thres} = 50\%$)	7525/39350	19.1%
Percantage of upward spread-crossing events($V_{thres} = 50\%$)	7690/39350	19.5%

Table 7: Partition 1(Future window = 600, Past window 50, $V_{thres} = 50\%$)

Partition 1(Future window = 600, Past window 50, $V_{thres} = 50\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	13337/19675	67.8%
Percantage of downward spread-crossing events(testing)	11900/19675	60.5%
Percantage of upward spread-crossing events(training)	12969/19675	65.9%
Percantage of upward spread-crossing events(testing)	16140/19675	82.0%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,training)	10296/19675	52.3%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,testing)	10905/19675	55.4%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,training)	10911/19675	55.5%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,testing)	12203/19675	62.0%

Table 8: Partition 1(Future window = 600, Past window 50, $V_{thres} = 50\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 4881 & 2432 \\ 3906 & 8456 \end{matrix}$	Ctest Down = $\begin{matrix} 4100 & 3608 \\ 5771 & 6196 \end{matrix}$
Ctrain Up = $\begin{matrix} 6349 & 1109 \\ 6666 & 5551 \end{matrix}$	Ctest Up = $\begin{matrix} 5836 & 2410 \\ 6360 & 5069 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 9537 & 5546 \\ 1160 & 3432 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 9706 & 7036 \\ 1728 & 1205 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 13564 & 2059 \\ 1476 & 2576 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 10800 & 5237 \\ 2235 & 1403 \end{matrix}$

5.1.5 Scenario 1.5 (Future window = 600, Past Window = 100, $V_{thres} = 50\%$)

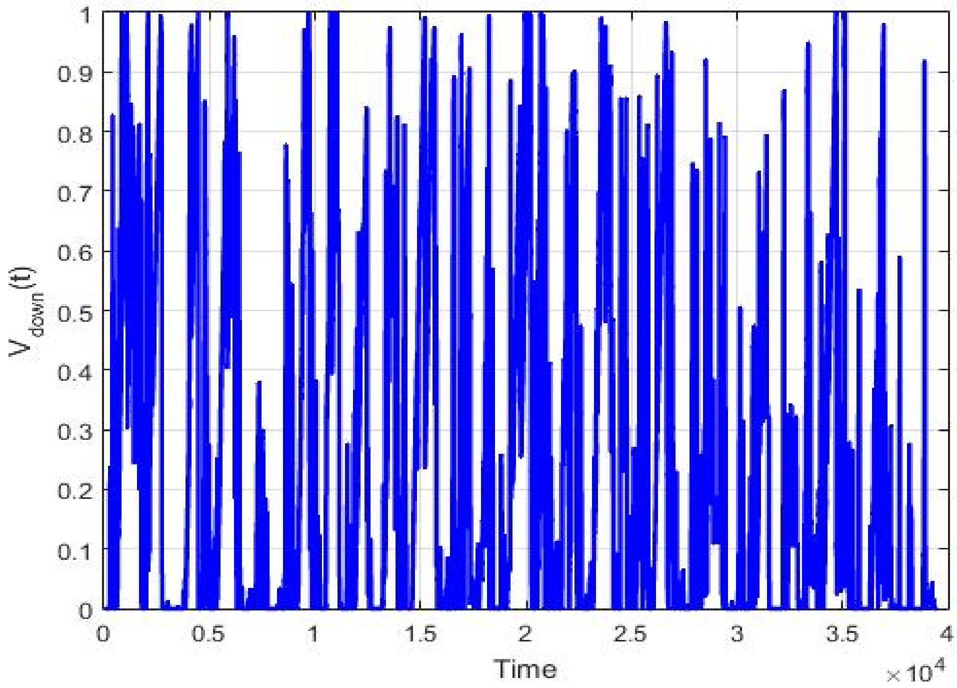

Figure 40: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

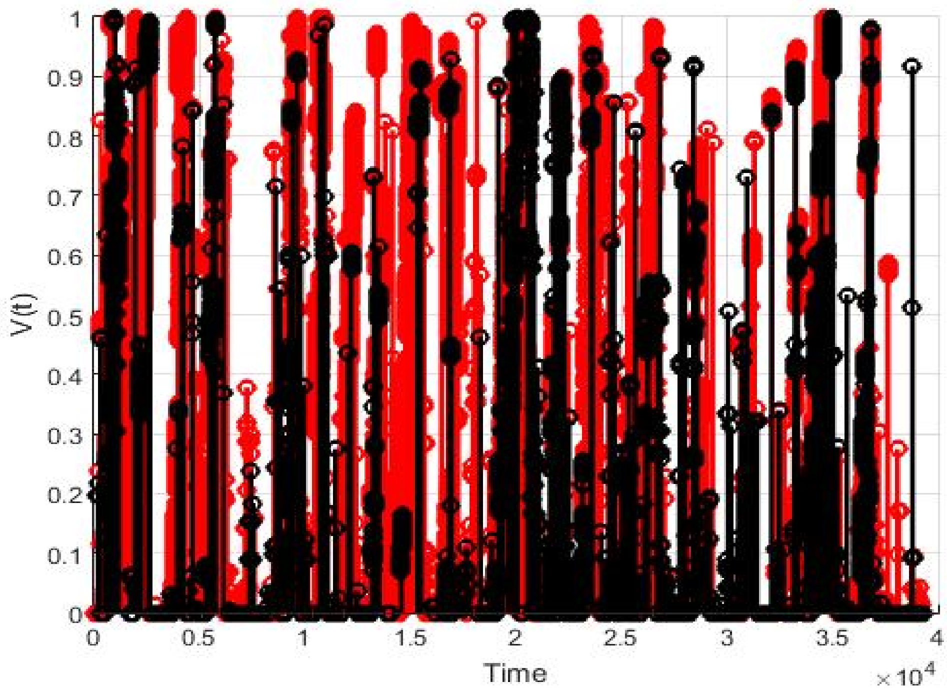

Figure 41: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

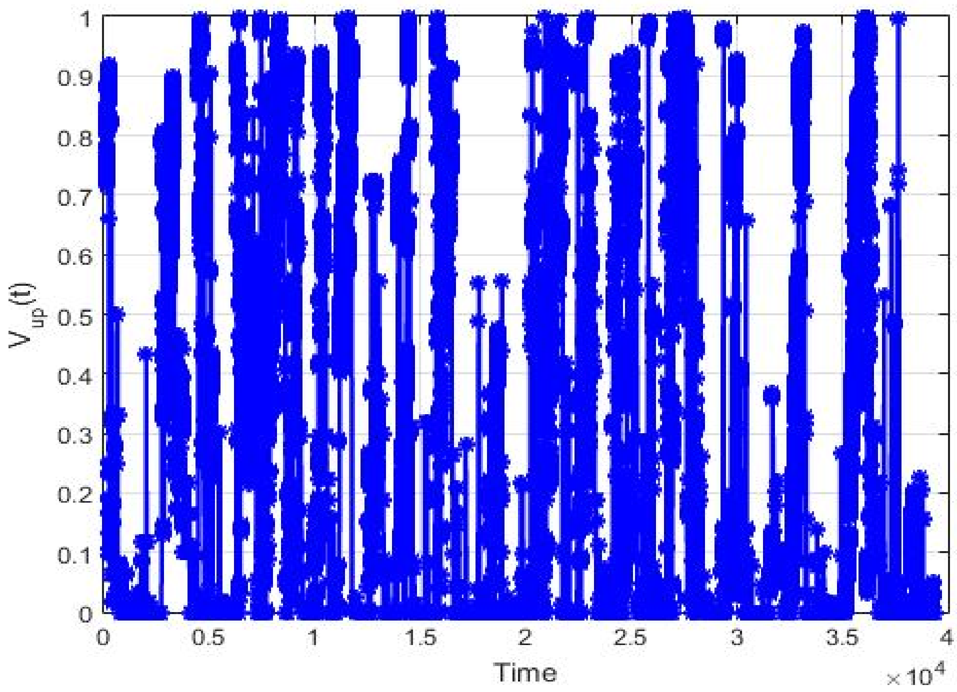

Figure 42: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

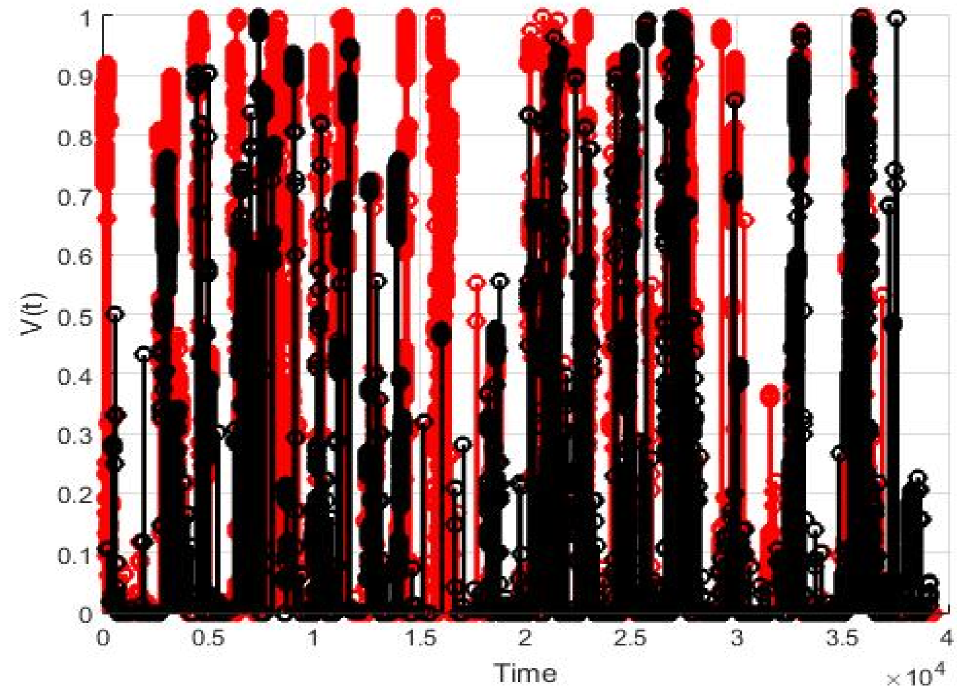

Figure 43: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

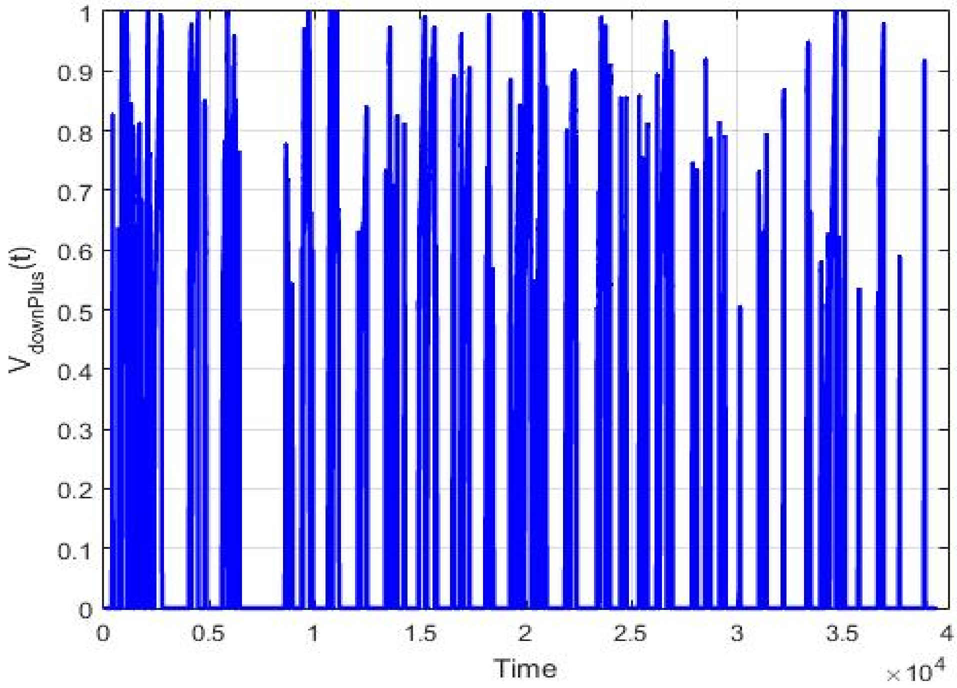

Figure 44: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

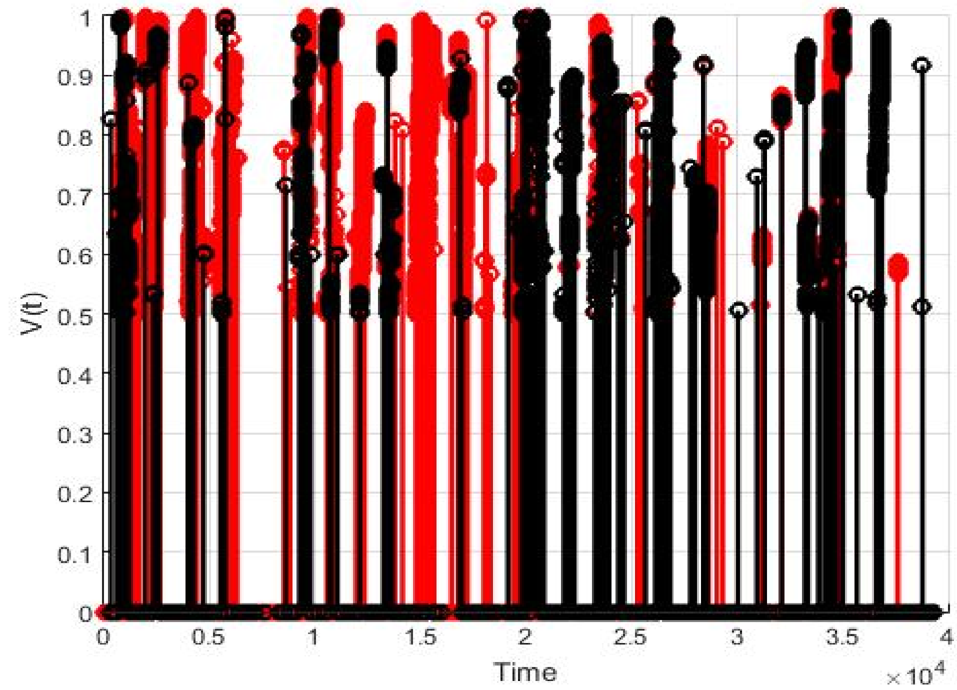

Figure 45: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

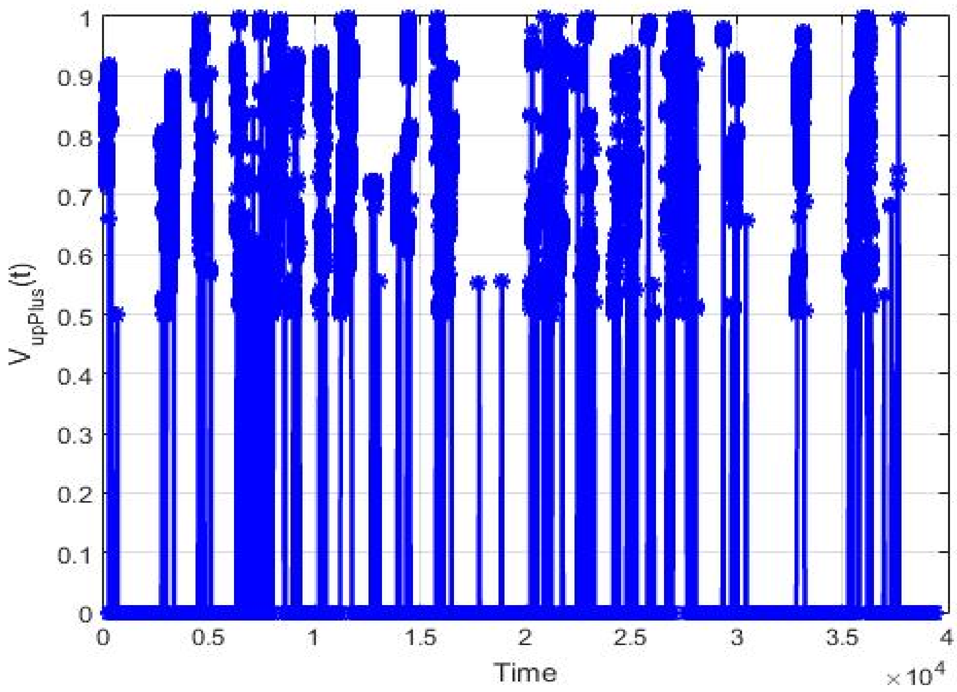

Figure 46: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

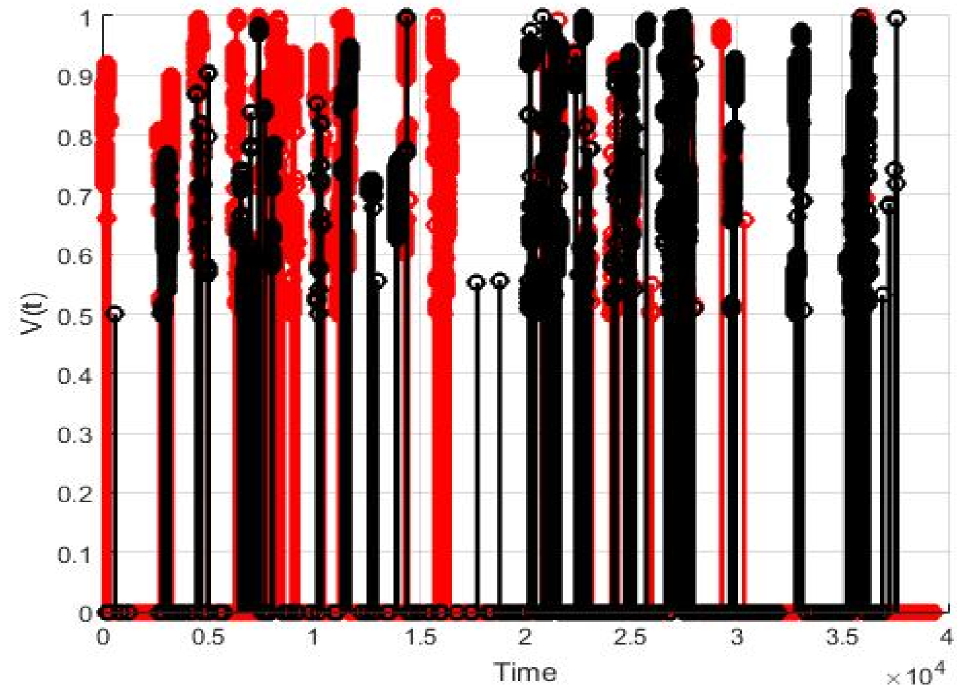

Figure 47: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

Partition 1(Future window = 600, Past window 100, $V_{thres} = 50\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39300	61.9%
Percantage of upward spread-crossing events	23596/39300	60.0%
Percantage of downward spread-crossing events($V_{thres} = 50\%$)	7525/39300	19.1%
Percantage of upward spread-crossing events($V_{thres} = 50\%$)	7640/39300	19.4%

Table 9: Partition 1(Future window = 600, Past window 100, $V_{thres} = 50\%$)

Partition 1(Future window = 600, Past window 100, $V_{thres} = 50\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	15521/19650	79.0%
Percantage of downward spread-crossing events(testing)	14670/19650	74.7%
Percantage of upward spread-crossing events(training)	15260/19650	77.7%
Percantage of upward spread-crossing events(testing)	18358/19650	93.4%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,training)	11037/19650	56.2%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,testing)	11243/19650	57.2%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,training)	11564/19650	58.8%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,testing)	13698/19650	69.7%

Table 10: Partition 1(Future window = 600, Past window 100, $V_{thres} = 50\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 5093 & 2170 \\ 1959 & 10428 \end{matrix}$	Ctest Down = $\begin{matrix} 3370 & 4338 \\ 4275 & 7667 \end{matrix}$
Ctrain Up = $\begin{matrix} 6743 & 735 \\ 4245 & 7927 \end{matrix}$	Ctest Up = $\begin{matrix} 4963 & 3263 \\ 5144 & 6280 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 11387 & 3671 \\ 719 & 3873 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 10645 & 6072 \\ 2014 & 919 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 15156 & 492 \\ 800 & 3202 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 12939 & 3073 \\ 2879 & 759 \end{matrix}$

5.1.6 Scenario 1.6 (Future window = 600, Past Window = 200, $V_{thres} = 50\%$)

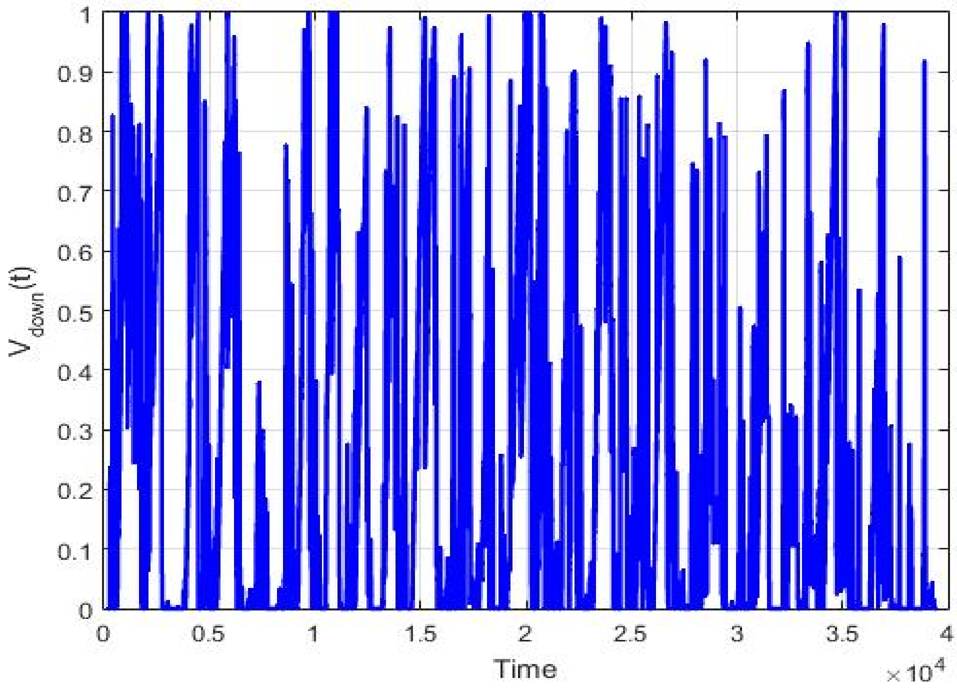

Figure 48: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

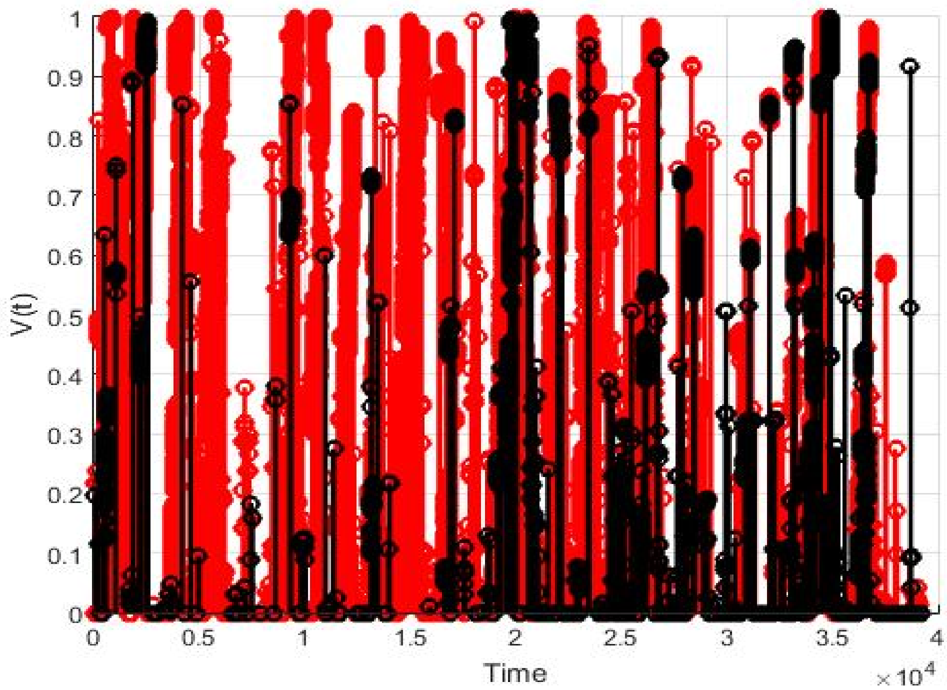

Figure 49: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

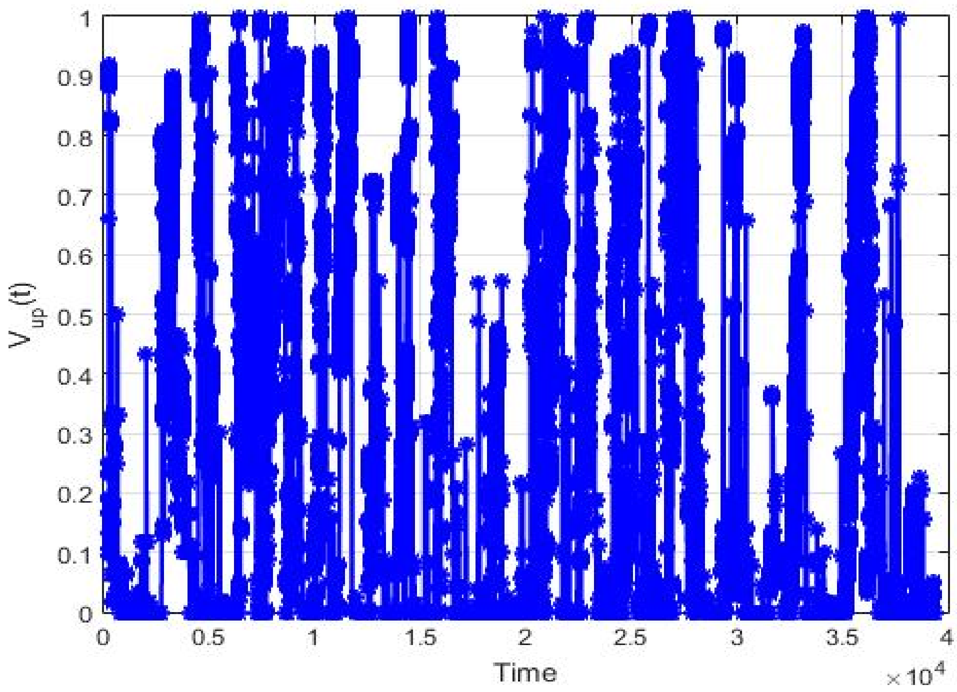

Figure 50: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

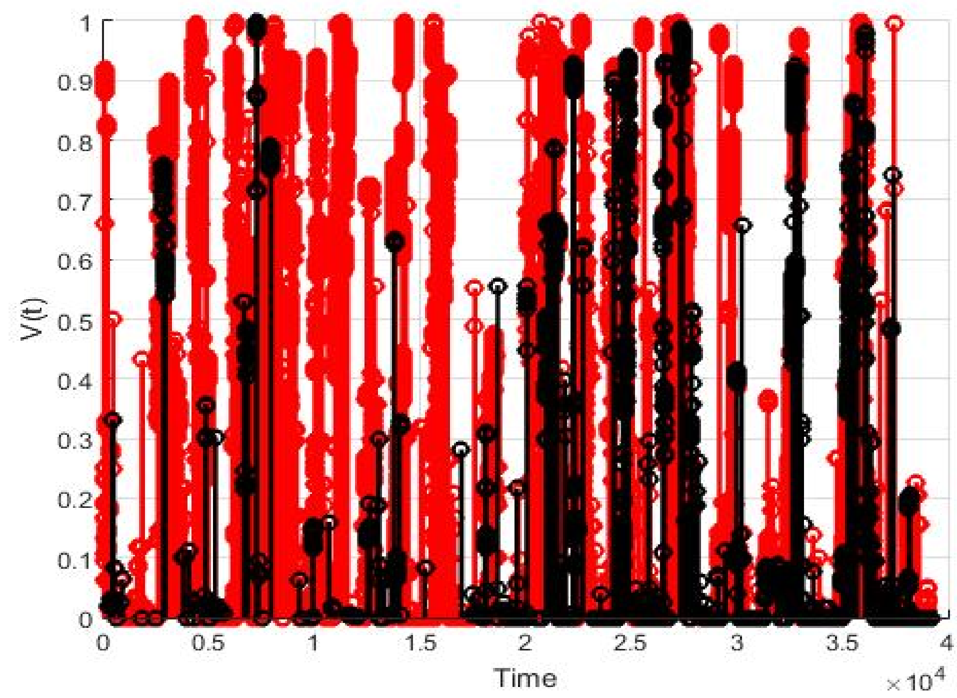

Figure 51: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

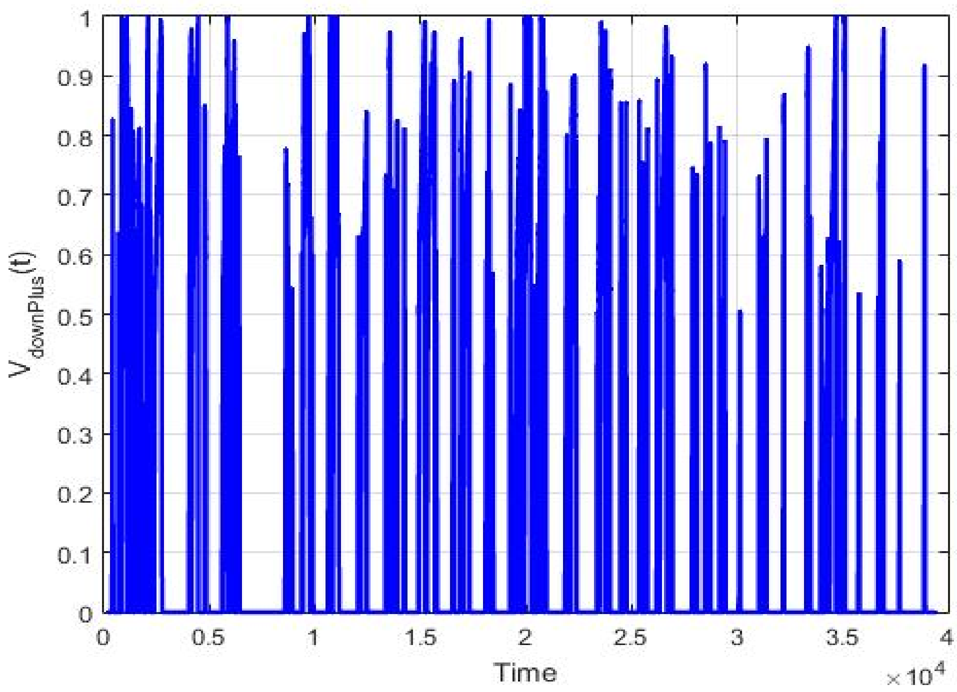

Figure 52: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

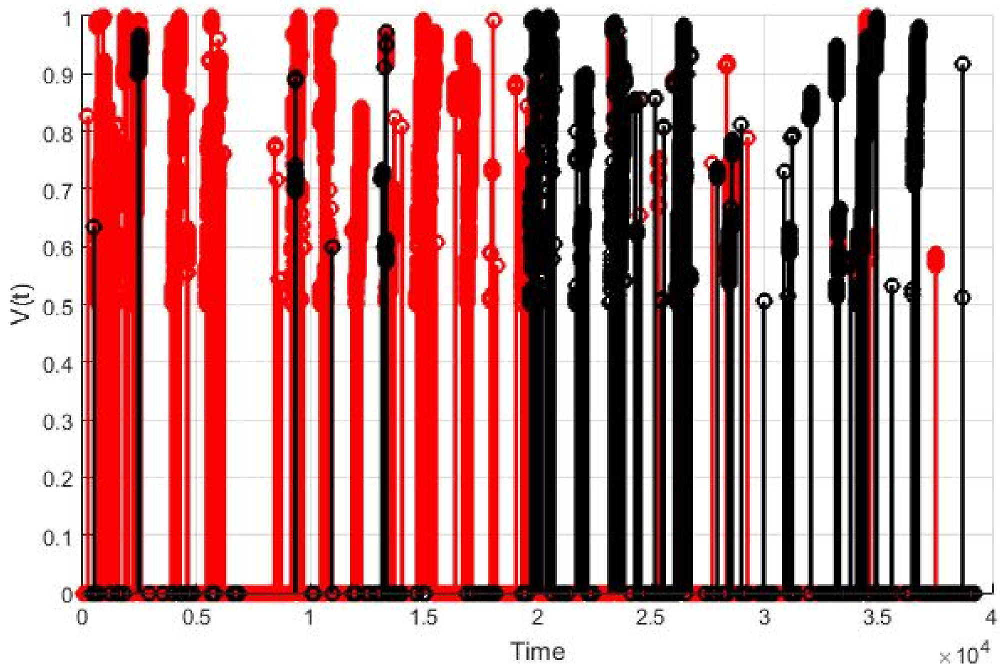

Figure 53: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

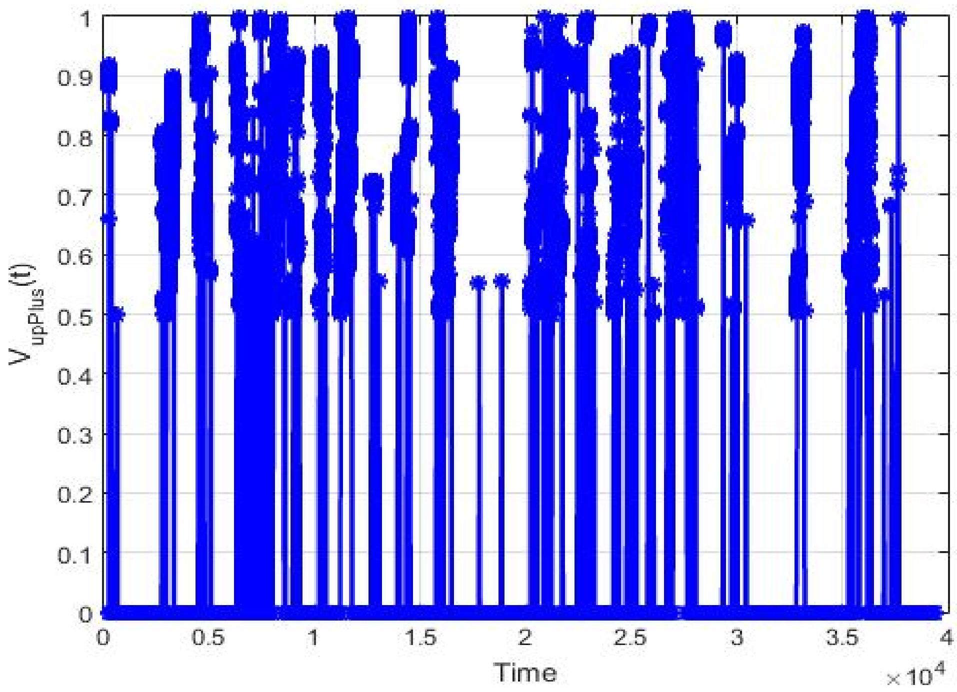

Figure 54: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

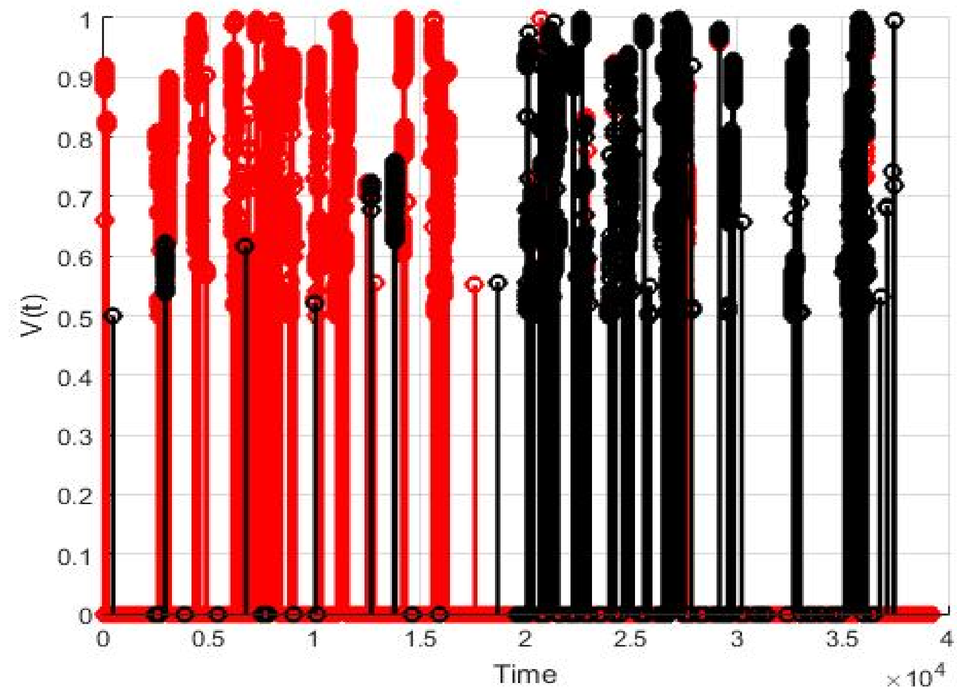

Figure 55: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

Partition 1(Future window = 600, Past window 200, $V_{thres} = 50\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39200	62.1%
Percantage of upward spread-crossing events	23496/39200	59.9%
Percantage of downward spread-crossing events($V_{thres} = 50\%$)	7525/39200	19.2%
Percantage of upward spread-crossing events($V_{thres} = 50\%$)	7540/39200	19.2%

Table 11: Partition 1(Future window = 600, Past window 200, $V_{thres} = 50\%$)

Partition 1(Future window = 600, Past window 200, $V_{thres} = 50\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	18068/19600	92.2%
Percantage of downward spread-crossing events(testing)	18550/19600	94.6%
Percantage of upward spread-crossing events(training)	18089/19600	92.3%
Percantage of upward spread-crossing events(testing)	19416/19600	99.1%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,training)	11061/19600	56.4%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,testing)	11261/19600	57.5%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,training)	12911/19600	65.9%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,testing)	15018/19600	76.6%

Table 12: Partition 1(Future window = 600, Past window 200, $V_{thres} = 50\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 6478 & 685 \\ 847 & 11590 \end{matrix}$	Ctest Down = $\begin{matrix} 2595 & 5113 \\ 3426 & 8466 \end{matrix}$
Ctrain Up = $\begin{matrix} 7212 & 295 \\ 755 & 11338 \end{matrix}$	Ctest Up = $\begin{matrix} 2649 & 5548 \\ 2791 & 8612 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 13609 & 1399 \\ 112 & 4480 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 12466 & 4201 \\ 2488 & 445 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 15651 & 47 \\ 137 & 3765 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 14731 & 1231 \\ 3351 & 287 \end{matrix}$

5.1.7 Scenario 1.7 (Future window = 600, Past Window = 50, $V_{thres} = 80\%$)

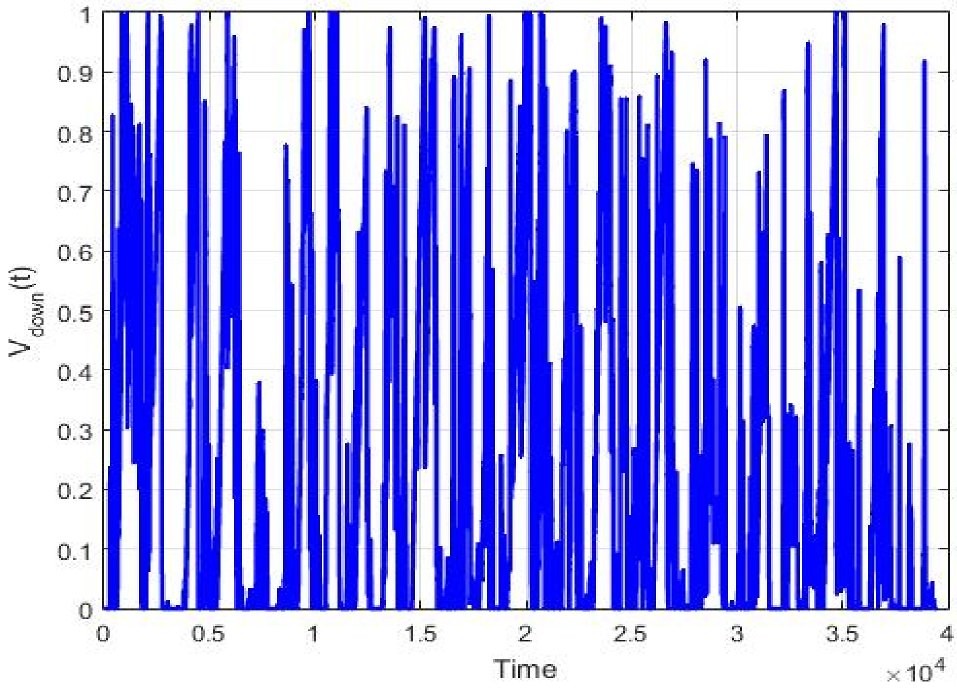

Figure 56: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

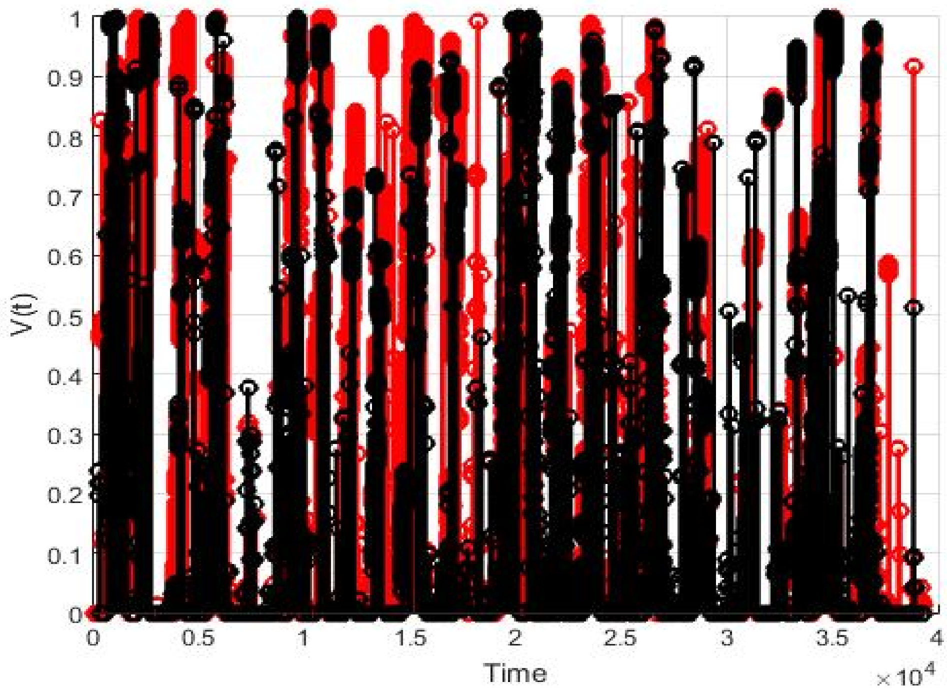

Figure 57: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

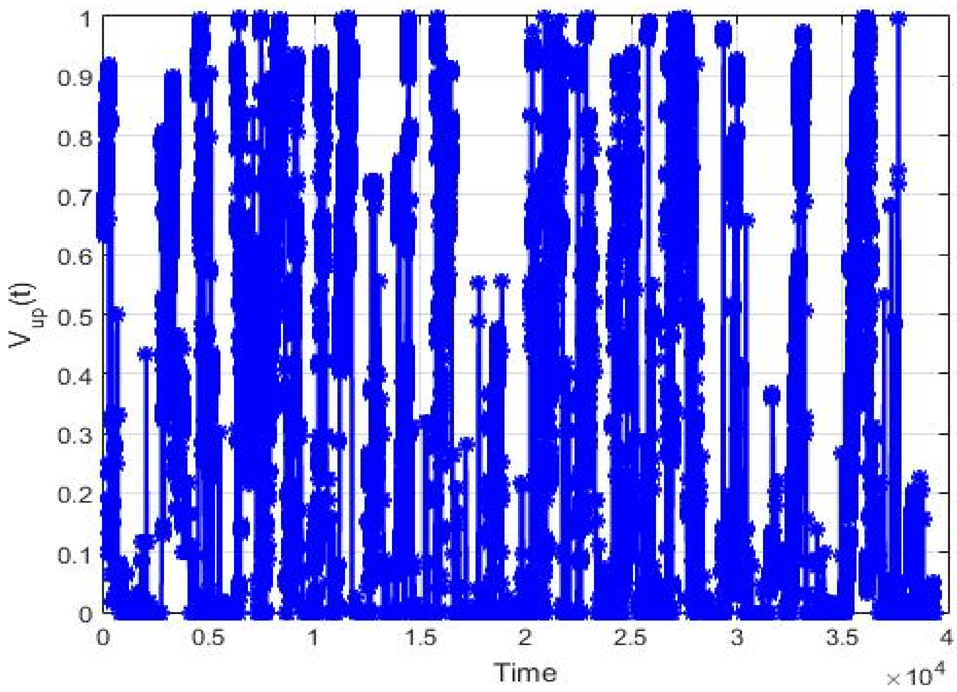

Figure 58: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

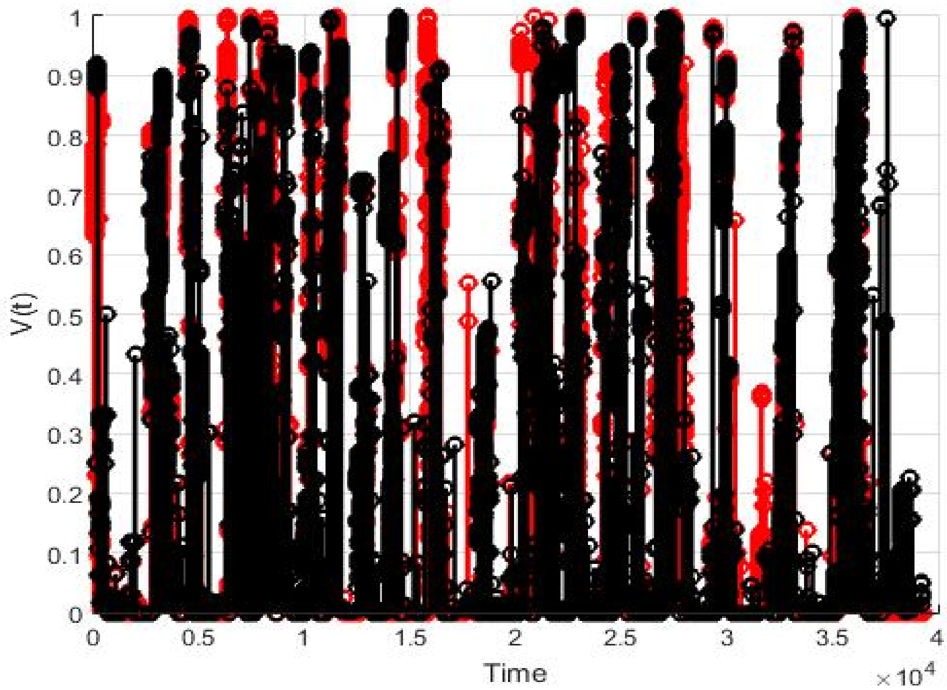

Figure 59: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

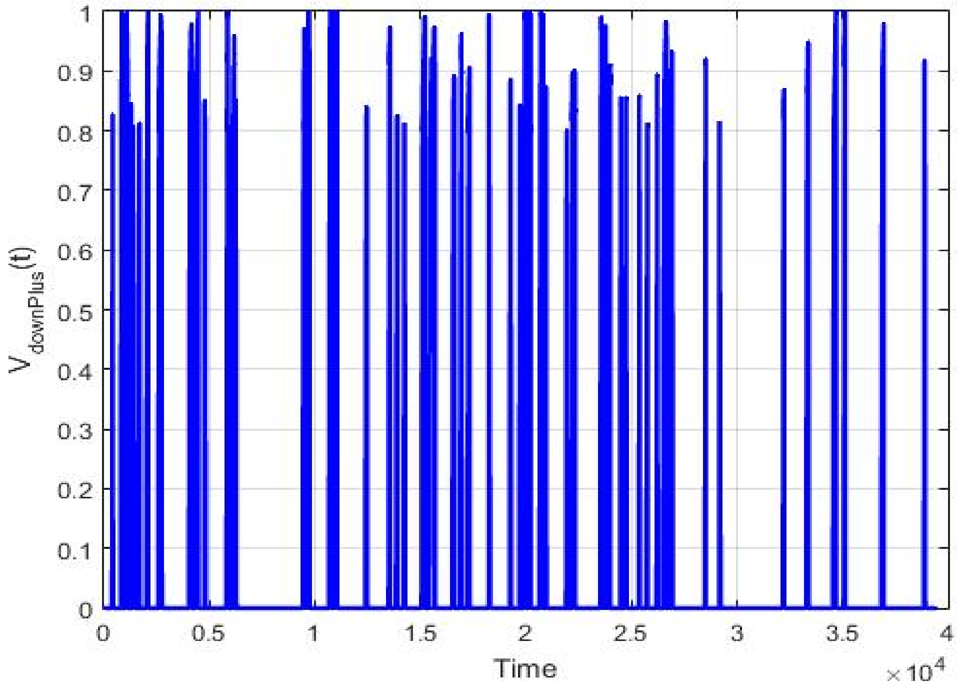

Figure 60: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

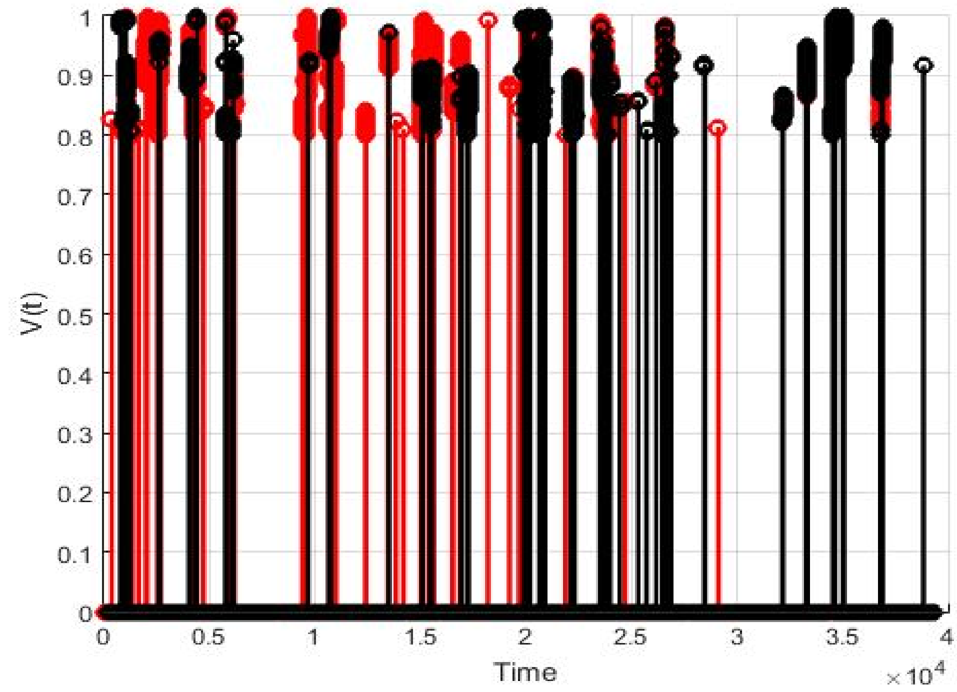

Figure 61: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

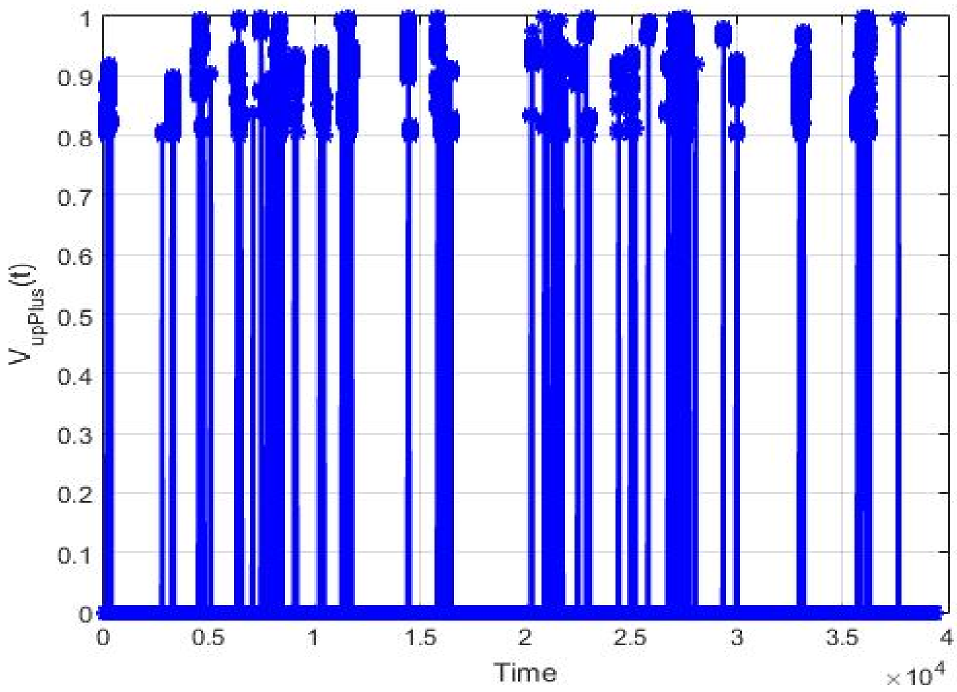

Figure 62: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

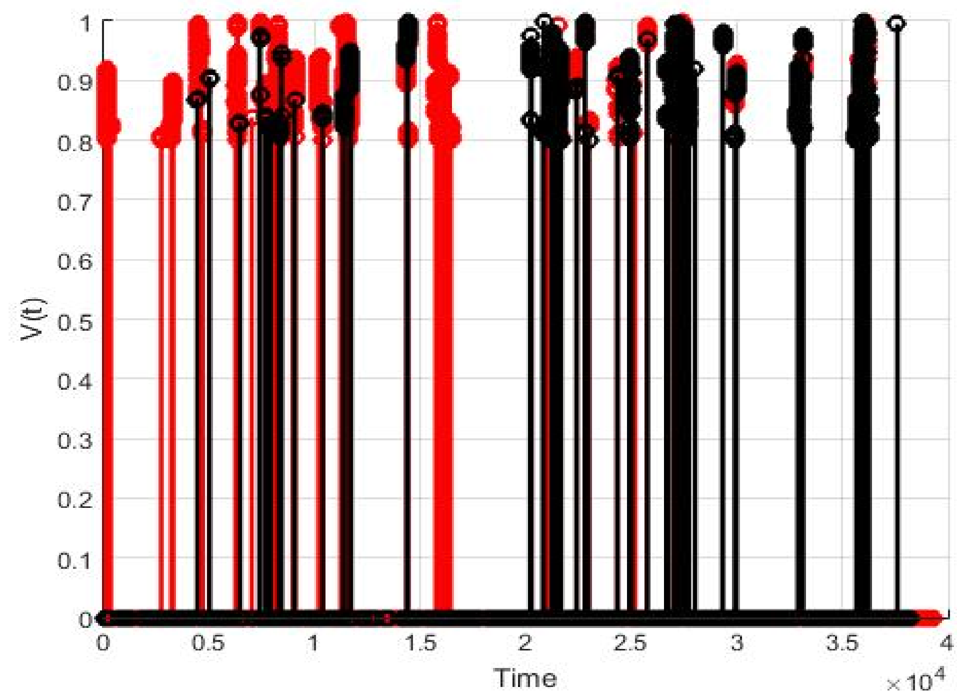

Figure 63: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

Partition 1(Future window = 600, Past window 50, $V_{thres} = 80\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39350	61.8%
Percantage of upward spread-crossing events	23646/39350	60.1%
Percantage of downward spread-crossing events($V_{thres} = 80\%$)	2853/39350	7.3%
Percantage of upward spread-crossing events($V_{thres} = 80\%$)	2978/39350	7.6%

Table 13: Partition 1(Future window = 600, Past window 50, $V_{thres} = 80\%$)

Partition 1(Future window = 600, Past window 50, $V_{thres} = 80\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	13337/19675	67.8%
Percantage of downward spread-crossing events(testing)	11900/19675	60.5%
Percantage of upward spread-crossing events(training)	14194/19675	72.1%
Percantage of upward spread-crossing events(testing)	16855/19675	85.7%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,training)	10296/19675	52.3%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,testing)	10905/19675	55.4%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,training)	13805/19675	70.2%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,testing)	14115/19675	71.7%

Table 14: Partition 1(Future window = 600, Past window 50, $V_{thres} = 80\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 4881 & 2432 \\ 3906 & 8456 \end{matrix}$	Ctest Down = $\begin{matrix} 4100 & 3608 \\ 5771 & 6196 \end{matrix}$
Ctrain Up = $\begin{matrix} 6349 & 1109 \\ 6666 & 5551 \end{matrix}$	Ctest Up = $\begin{matrix} 5836 & 2410 \\ 6360 & 5069 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 12976 & 5068 \\ 413 & 1218 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 13369 & 5084 \\ 786 & 436 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 15638 & 2652 \\ 168 & 1217 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 13678 & 4404 \\ 1156 & 437 \end{matrix}$

5.1.8 Scenario 1.8 (Future window = 600, Past Window = 100, $V_{thres} = 80\%$)

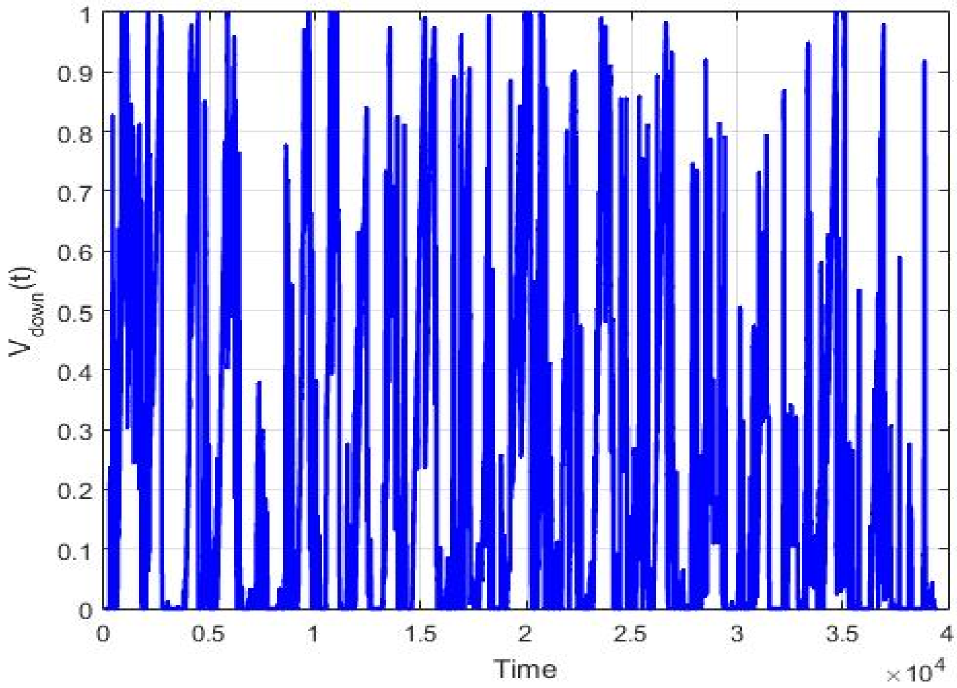

Figure 64: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

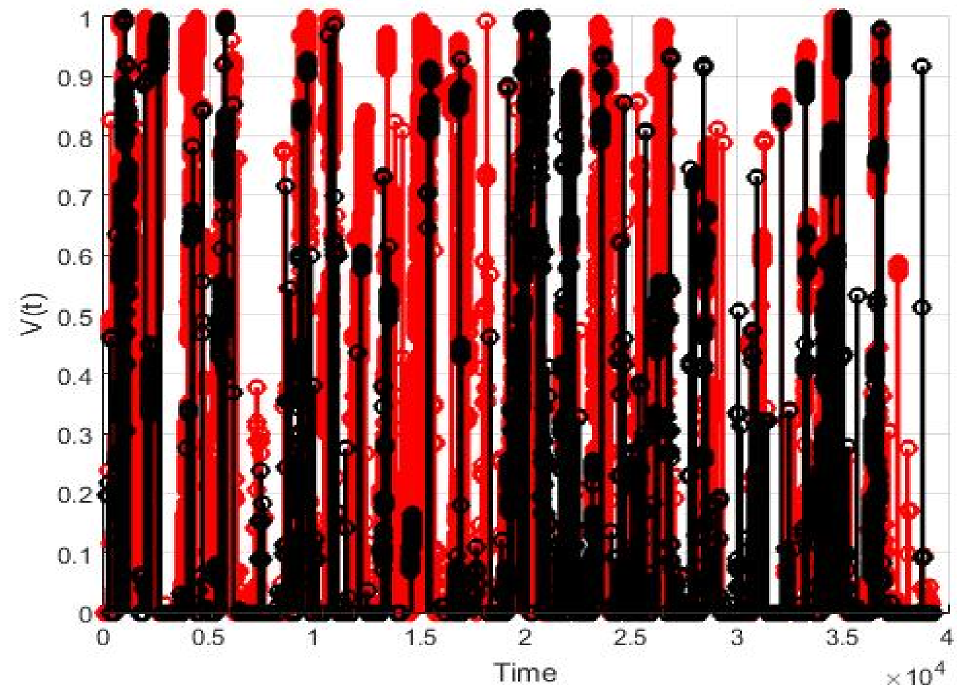

Figure 65: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

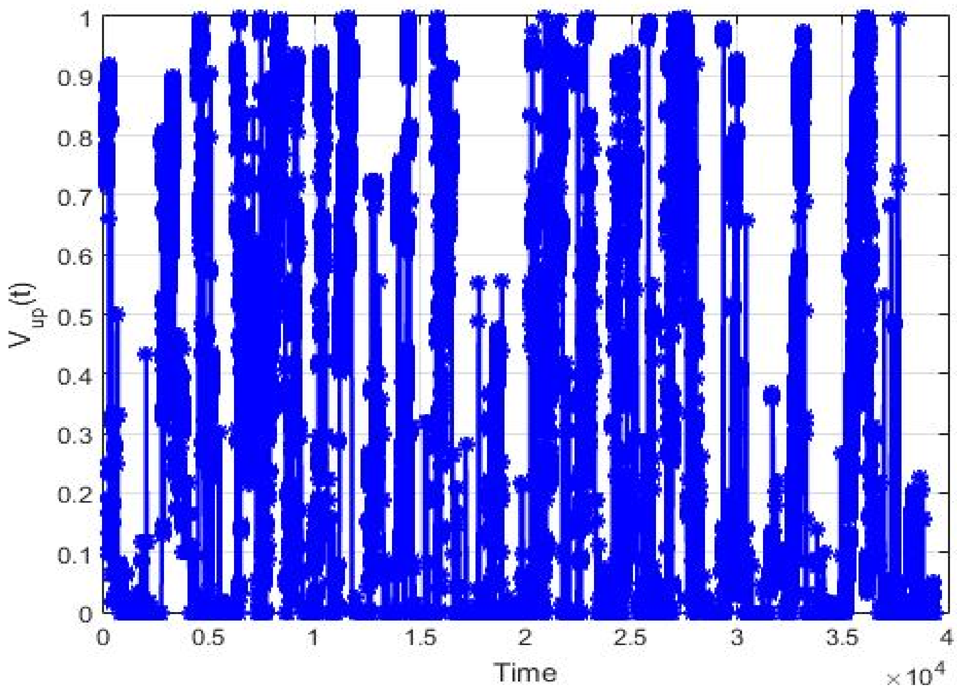

Figure 66: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

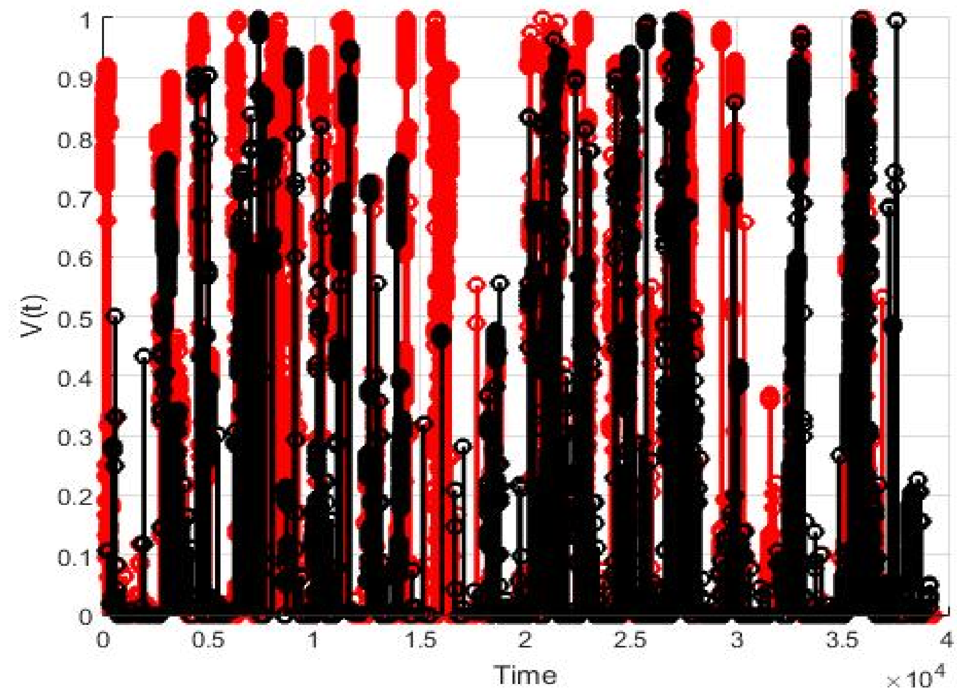

Figure 67: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

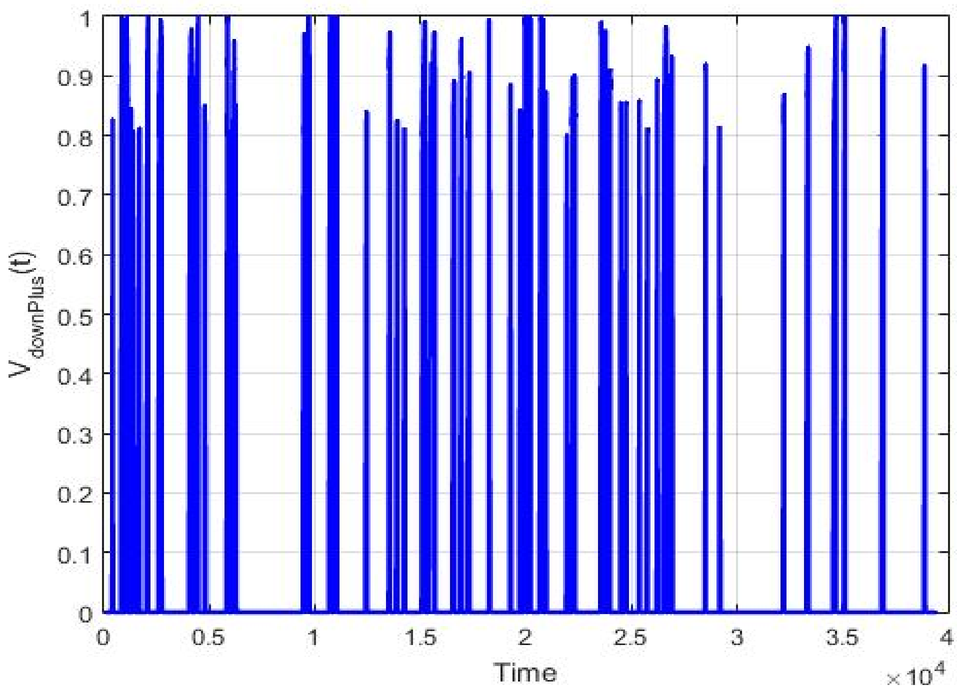

Figure 68: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

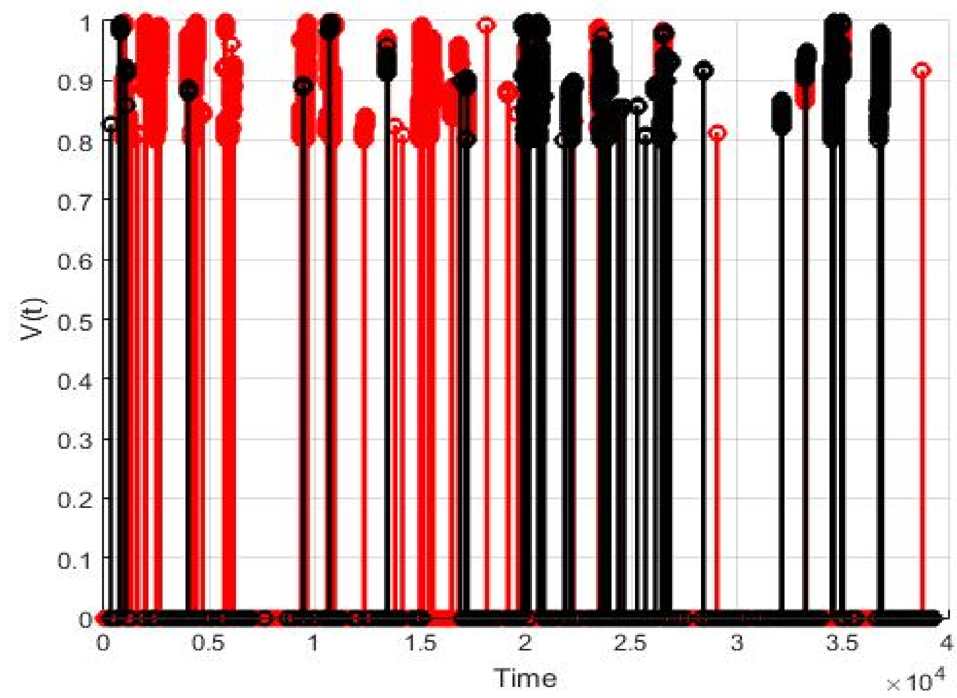

Figure 69: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

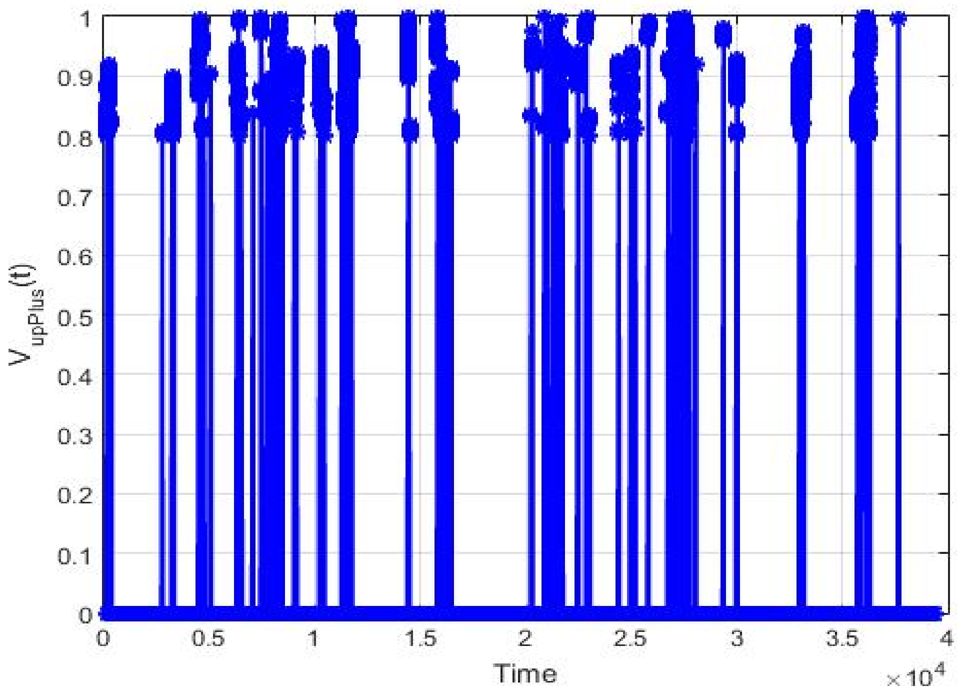

Figure 70: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

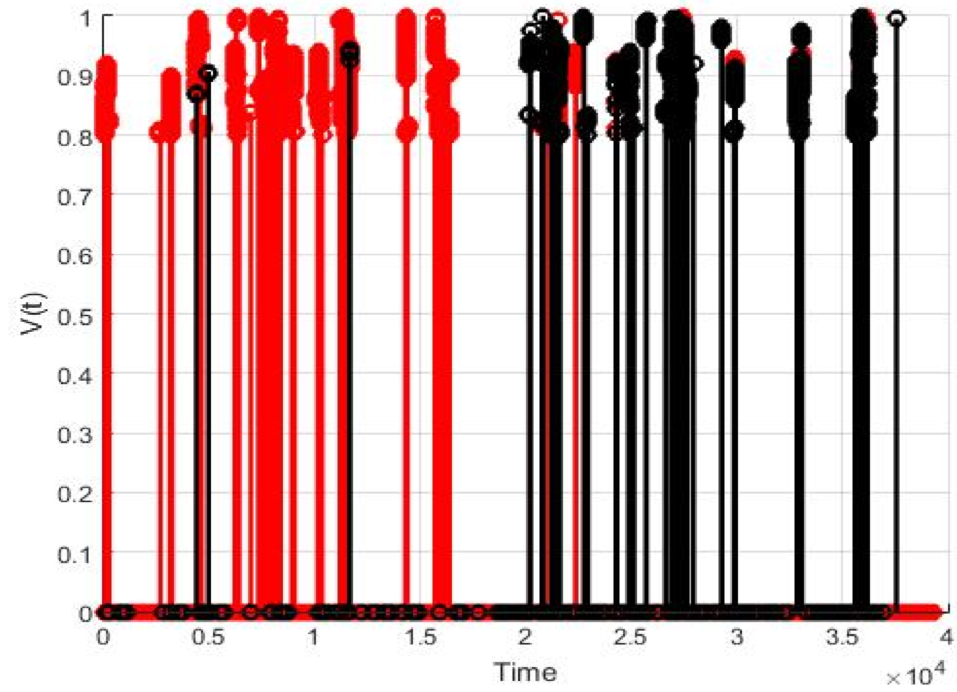

Figure 71: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

Partition 1(Future window = 600, Past window 100, $V_{thres} = 80\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39300	61.9%
Percantage of upward spread-crossing events	23596/39300	60.0%
Percantage of downward spread-crossing events($V_{thres} = 80\%$)	2853/39300	7.3%
Percantage of upward spread-crossing events($V_{thres} = 80\%$)	2978/39300	7.6%

Table 15: Partition 1(Future window = 600, Past window 100, $V_{thres} = 80\%$)

Partition 1(Future window = 600, Past window 100, $V_{thres} = 80\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	15521/19650	79.0%
Percantage of downward spread-crossing events(testing)	14670/19650	74.7%
Percantage of upward spread-crossing events(training)	16612/19650	84.5%
Percantage of upward spread-crossing events(testing)	19136/19650	97.4%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,training)	11037/19650	56.2%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,testing)	11243/19650	57.2%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,training)	15456/19650	78.7%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,testing)	16638/19650	84.7%

Table 16: Partition 1(Future window = 600, Past window 100, $V_{thres} = 80\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 5093 & 2170 \\ 1959 & 10428 \end{matrix}$	Ctest Down = $\begin{matrix} 3370 & 4338 \\ 4275 & 7667 \end{matrix}$
Ctrain Up = $\begin{matrix} 6743 & 735 \\ 4245 & 7927 \end{matrix}$	Ctest Up = $\begin{matrix} 4963 & 3263 \\ 5144 & 6280 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 15056 & 2963 \\ 75 & 1556 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 15175 & 3253 \\ 941 & 281 \end{matrix}$
Ctrain Up $V_{thres} = 17760 \quad 5059 \quad 1376$	Ctest Up $V_{thres} = \begin{matrix} 16406 & 1651 \\ 1361 & 232 \end{matrix}$

5.1.9 Scenario 1.9 (Future window = 600, Past Window = 200, $V_{thres} = 80\%$)

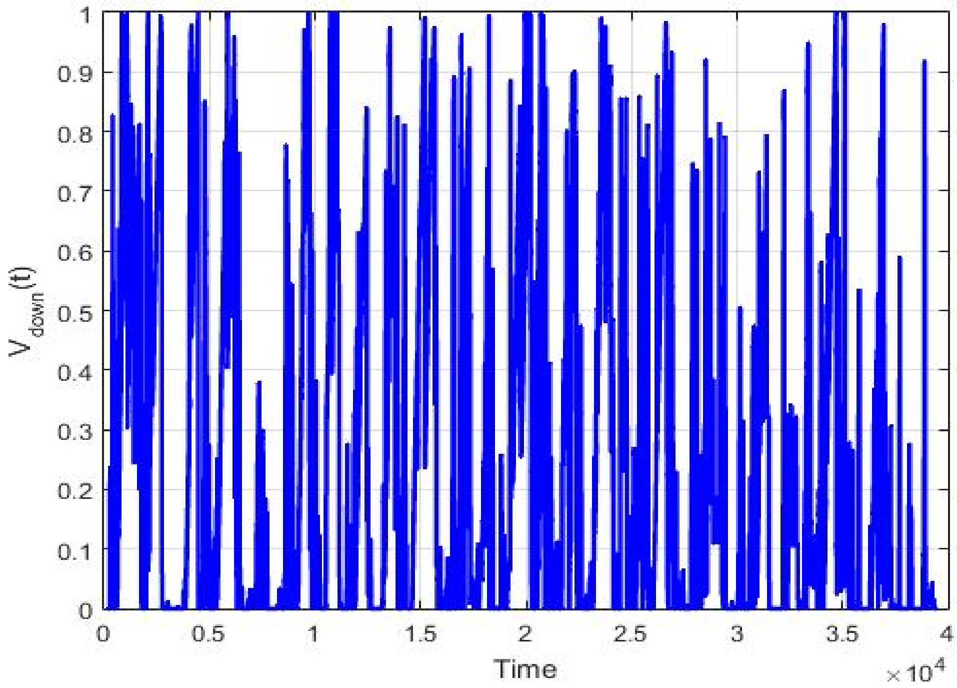

Figure 72: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

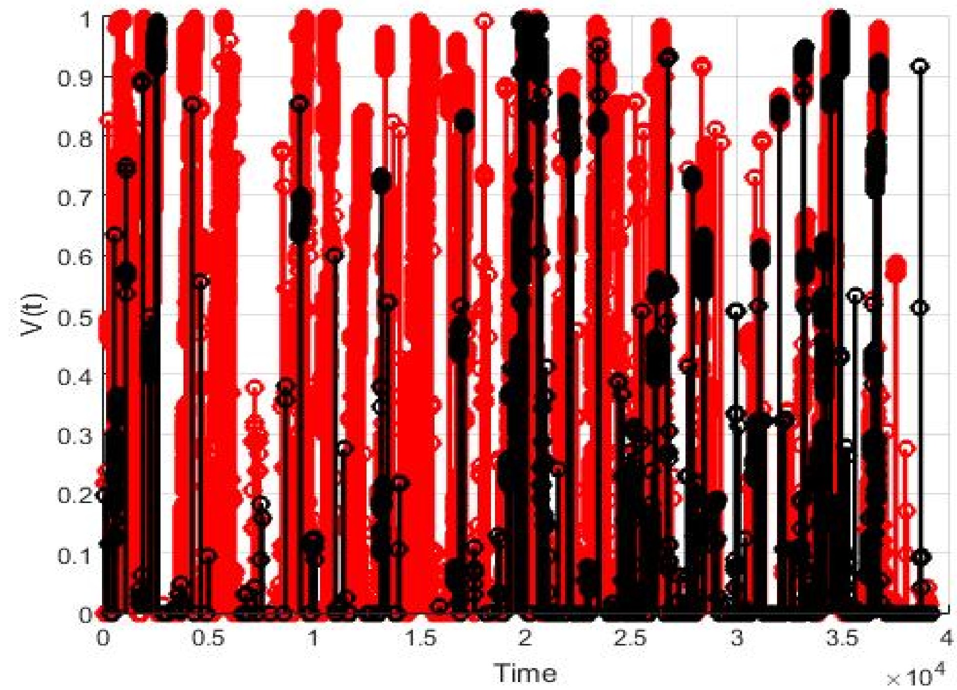

Figure 73: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

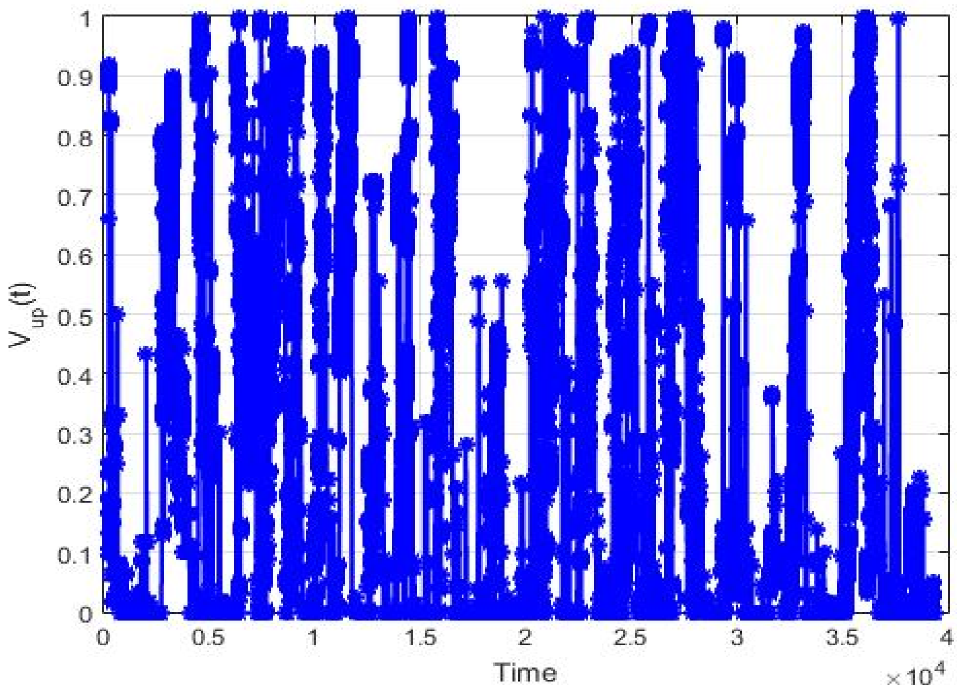

Figure 74: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

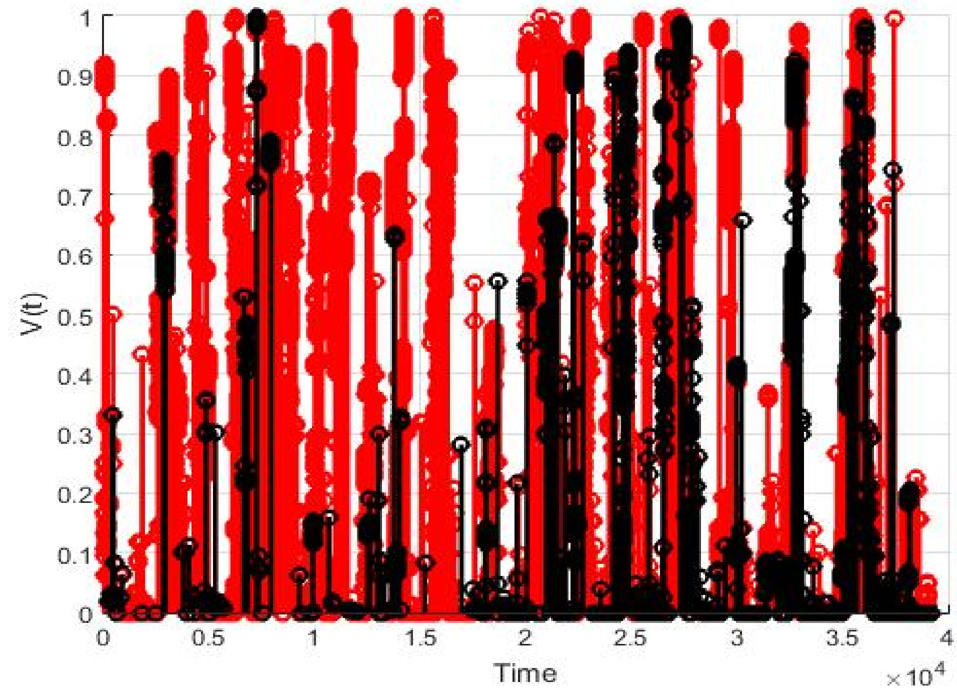

Figure 75: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

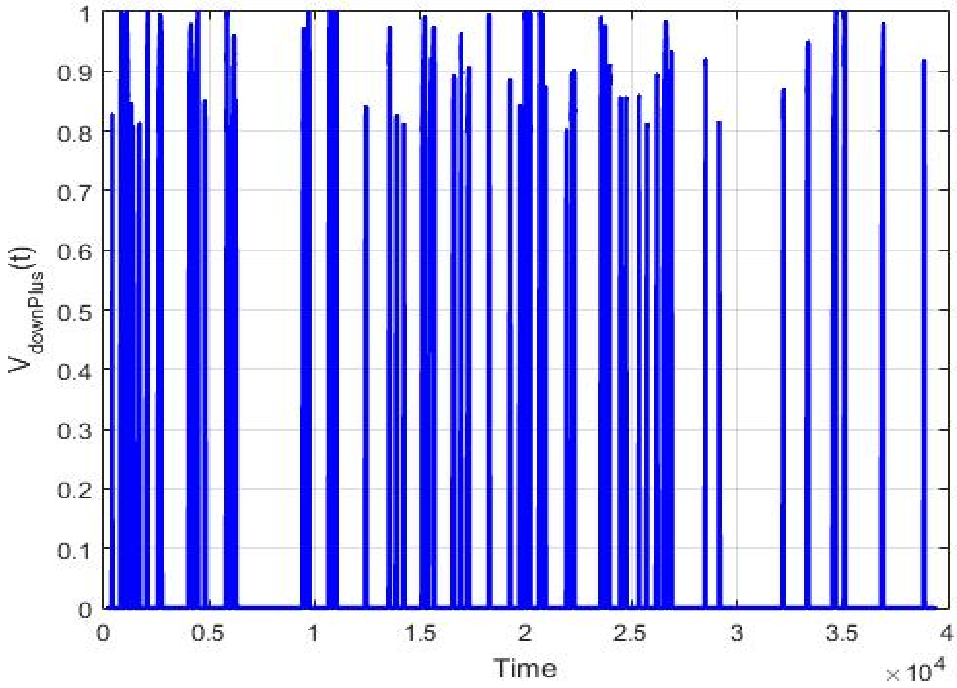

Figure 76: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

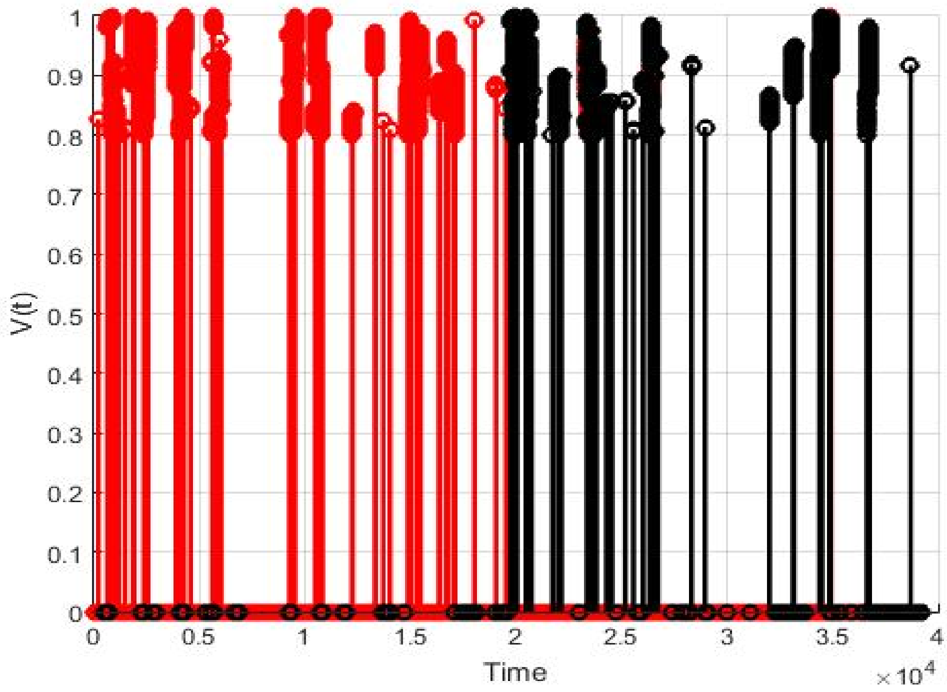

Figure 77: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

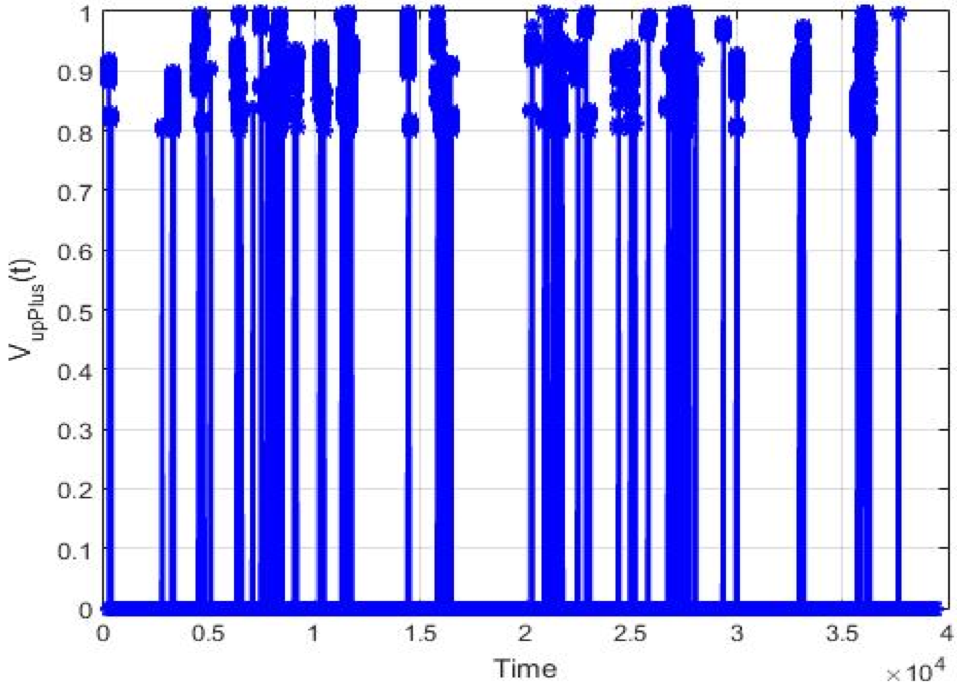

Figure 78: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

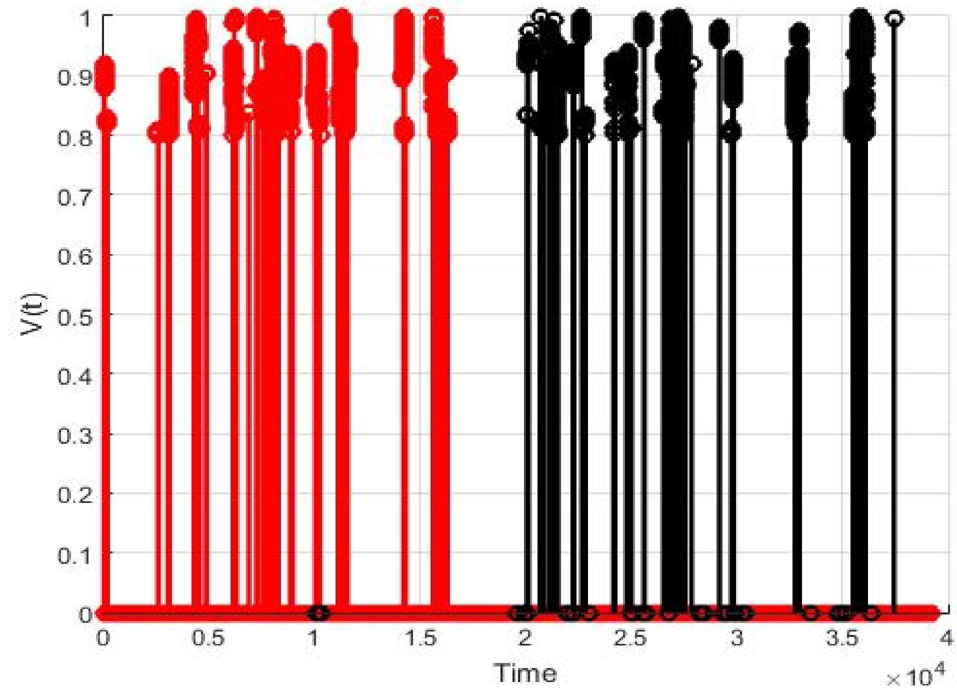

Figure 79: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 1, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

Partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	24329/39200	62.1%
Percantage of upward spread-crossing events	23496/39200	60.0%
Percantage of downward spread-crossing events($V_{thres} = 80\%$)	2853/39200	7.3%
Percantage of upward spread-crossing events($V_{thres} = 80\%$)	2922/39200	7.5%

Table 17: Partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$)

partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	18068/19600	92.2%
Percantage of downward spread-crossing events(testing)	18550/19600	94.6%
Percantage of upward spread-crossing events(training)	18944/19600	96.7%
Percantage of upward spread-crossing events(testing)	19582/19600	99.9%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,training)	11061/19600	56.4%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,testing)	11261/19600	57.5%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,training)	17040/19600	86.9%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,testing)	17652/19600	90.1%

Table 18: partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 6478 & 685 \\ 847 & 11590 \end{matrix}$	Ctest Down = $\begin{matrix} 2595 & 5113 \\ 3426 & 8466 \end{matrix}$
Ctrain Up = $\begin{matrix} 7212 & 295 \\ 755 & 11338 \end{matrix}$	Ctest Up = $\begin{matrix} 2649 & 5548 \\ 2791 & 8612 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 17313 & 656 \\ 0 & 1631 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 16902 & 1476 \\ 1084 & 138 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 18253 & 18 \\ 0 & 1329 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 17635 & 372 \\ 1576 & 17 \end{matrix}$

5.2 Partition 2

5.2.1 Scenario 2.1 (Future window = 600, Past Window = 50, $V_{thres} = 10\%$)

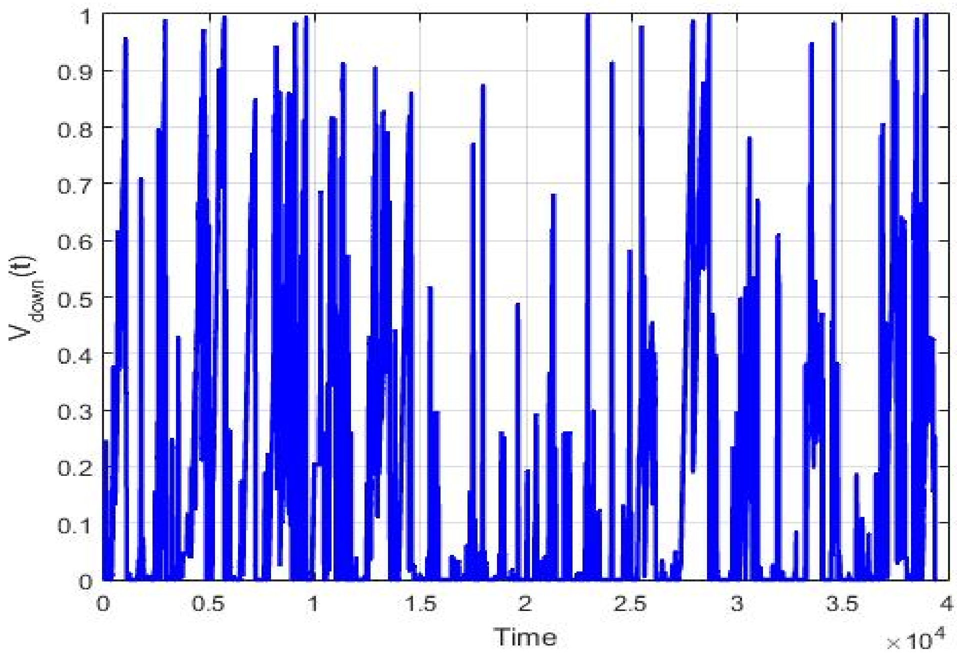

Figure 80: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50

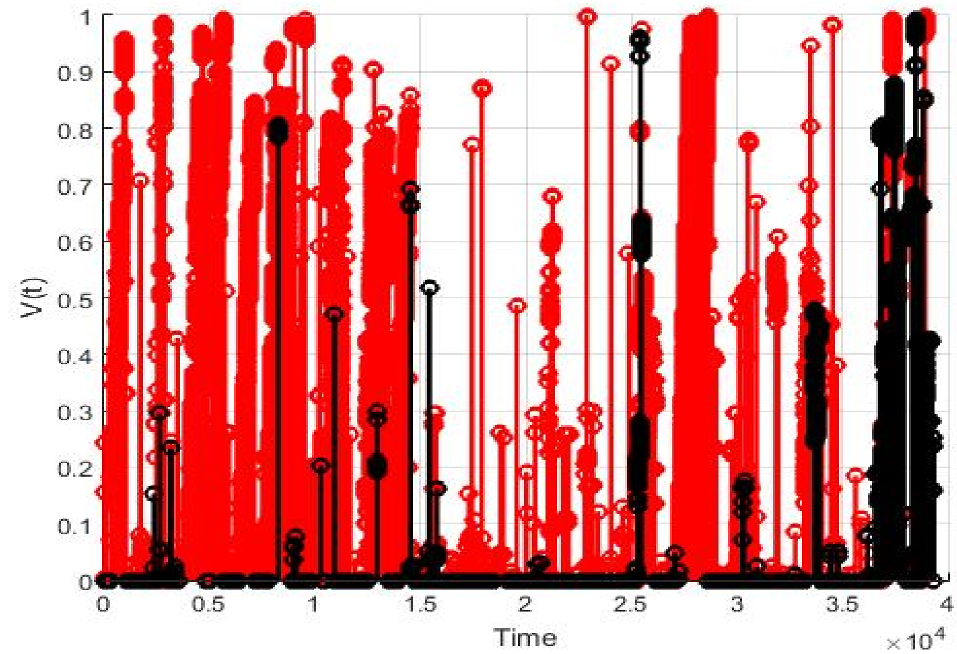

Figure 81: **Plot volume of Downward spread-crossing events after Classification within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50

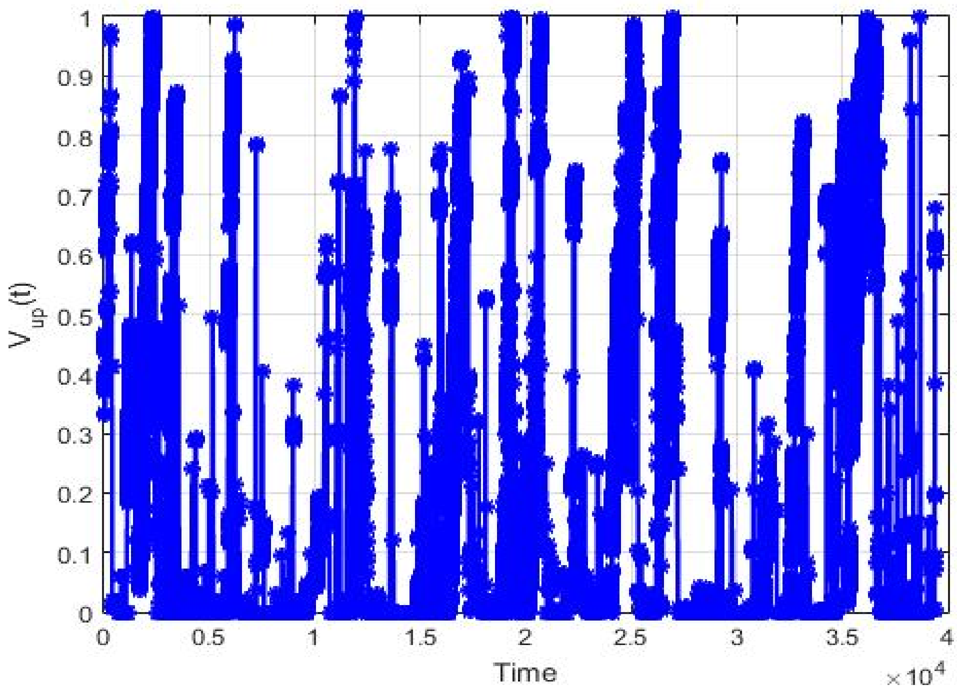

Figure 82: **Plot volume of Upward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50

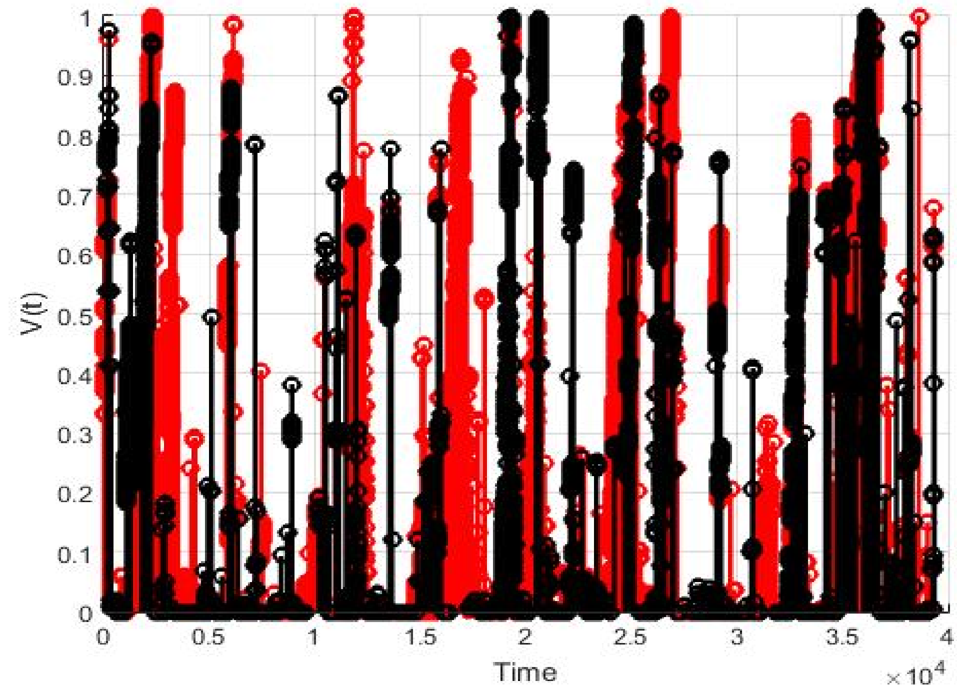

Figure 83: **Plot volume of Upward spread-crossing after Classification events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50

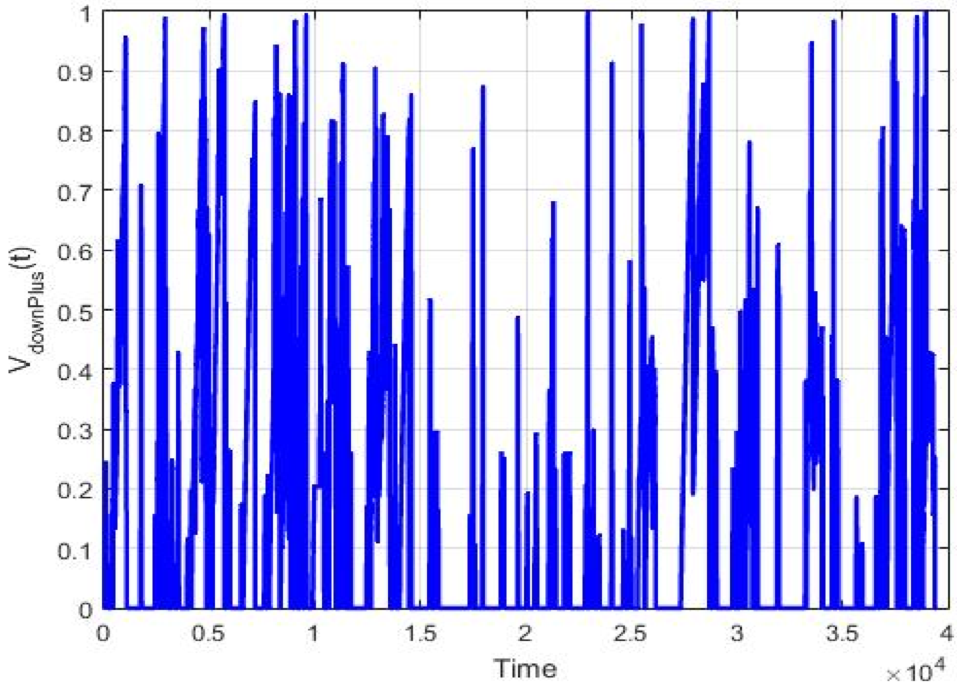

Figure 84: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

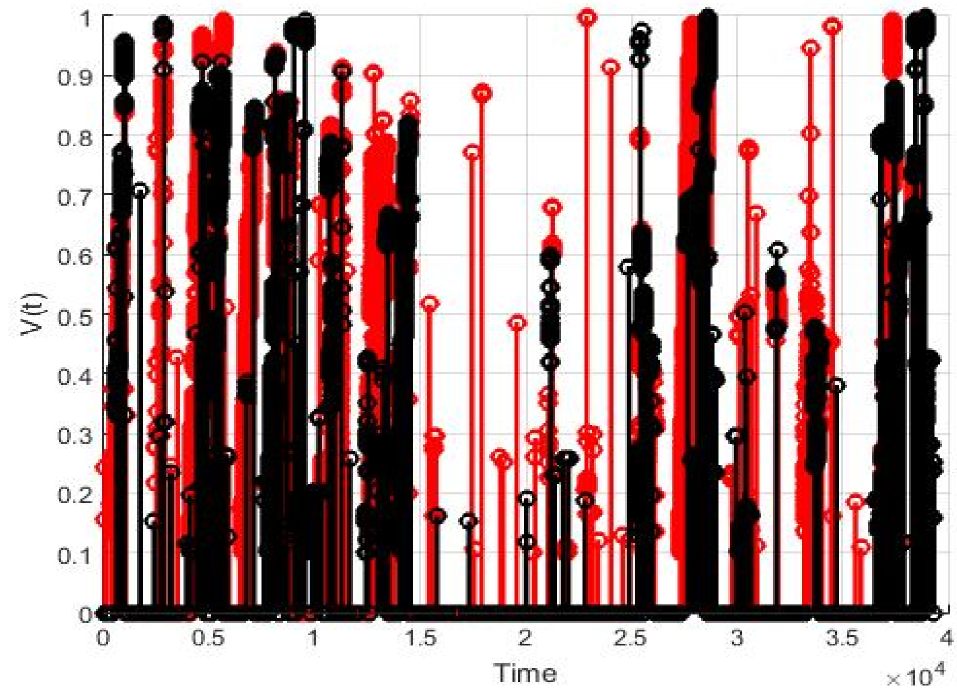

Figure 85: **Plot volume of Downward spread-crossing after Classification events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

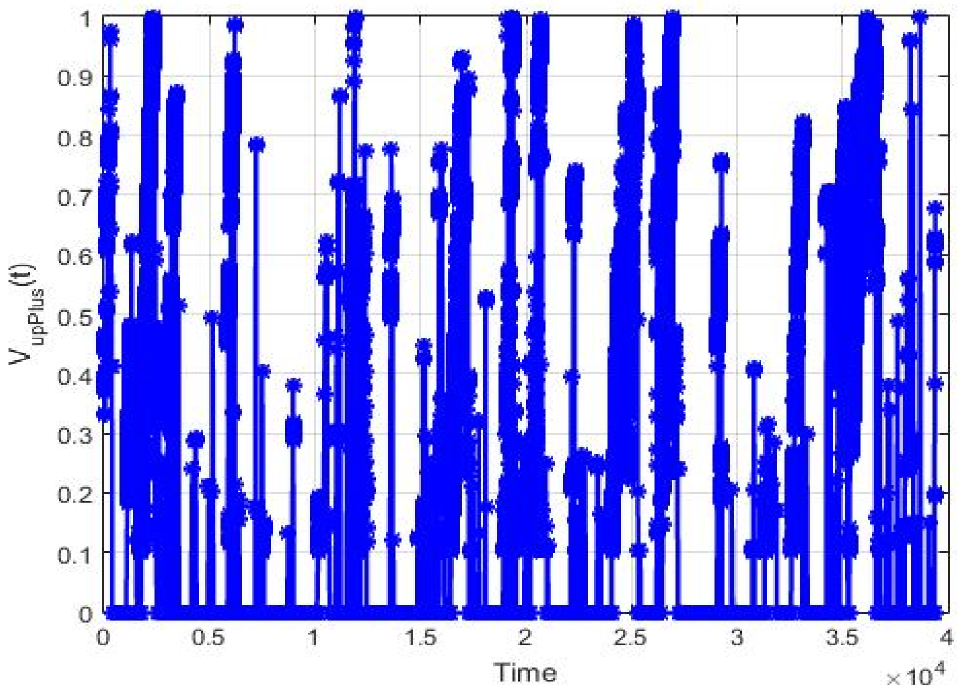

Figure 86: **Plot volume of Upward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

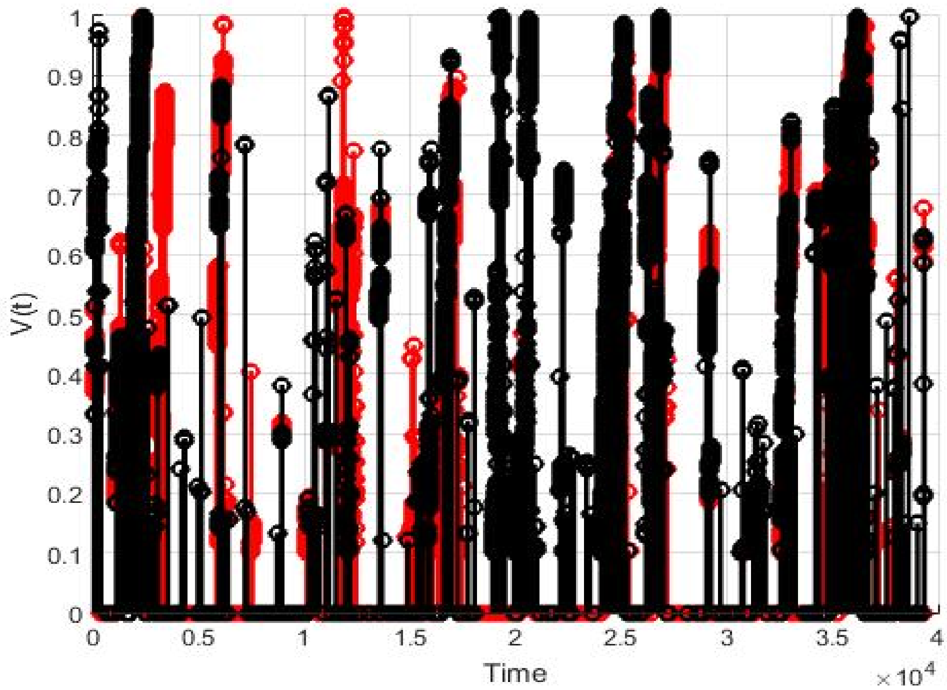

Figure 87: **Plot volume of Upward spread-crossing after Classification events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 10\%$

Partition 2(Future window = 600, Past window 50, $V_{thres} = 10\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18603/39350	47.3%
Percantage of upward spread-crossing events	22012/39350	60.0%
Percantage of downward spread-crossing events($V_{thres} = 10\%$)	12442/39350	31.6%
Percantage of upward spread-crossing events($V_{thres} = 10\%$)	13580/39350	34.5%

Table 19: Partition 2(Future window = 600, Past window 50, $V_{thres} = 10\%$)

Partition 2(Future window = 600, Past window 50, $V_{thres} = 10\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	11546/19675	58.7%
Percantage of downward spread-crossing events(testing)	12510/19675	63.6%
Percantage of upward spread-crossing events(training)	11812/19675	60.0%
Percantage of upward spread-crossing events(testing)	12517/19675	63.6%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,training)	7416/19675	37.7%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,testing)	11671/19675	59.3%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,training)	7917/19675	40.2%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,testing)	11861/19675	60.3%

Table 20: Partition 2(Future window = 600, Past window 50, $V_{thres} = 10\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 1315 & 7826 \\ 303 & 10231 \end{matrix}$	Ctest Down = $\begin{matrix} 1627 & 9979 \\ 2280 & 5789 \end{matrix}$
Ctrain Up = $\begin{matrix} 5335 & 3474 \\ 3691 & 7175 \end{matrix}$	Ctest Up = $\begin{matrix} 3920 & 4609 \\ 3395 & 7751 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 7558 & 5342 \\ 2521 & 4254 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 4802 & 9206 \\ 2552 & 3115 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 9480 & 3836 \\ 3322 & 3037 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 9795 & 2659 \\ 5155 & 2066 \end{matrix}$

5.2.2 Scenario 2.2 (Future window = 600, Past Window = 100, $V_{thres} = 10\%$)

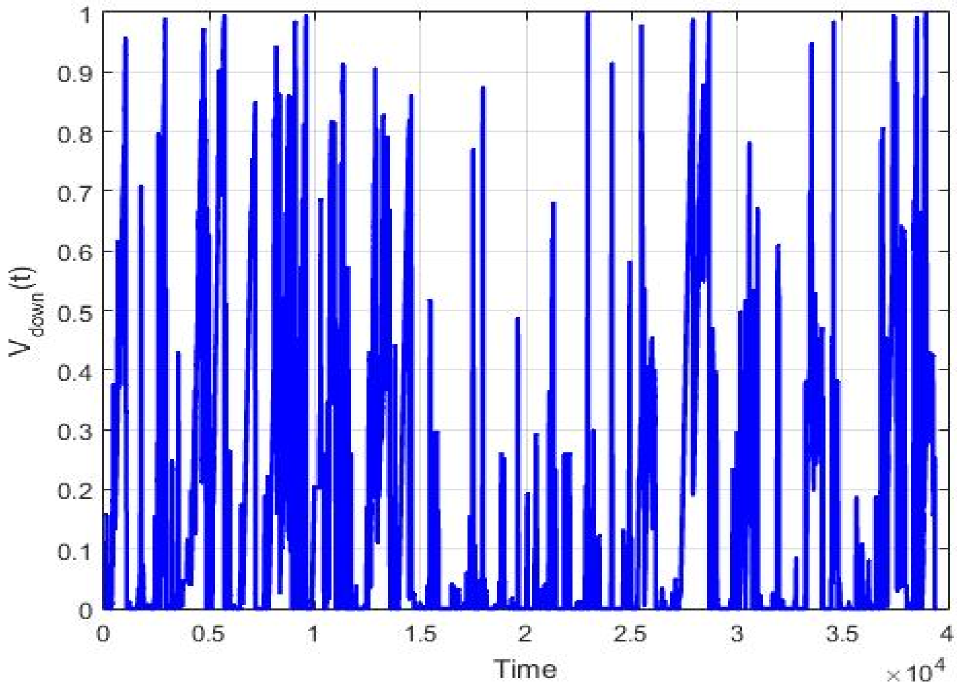

Figure 88: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

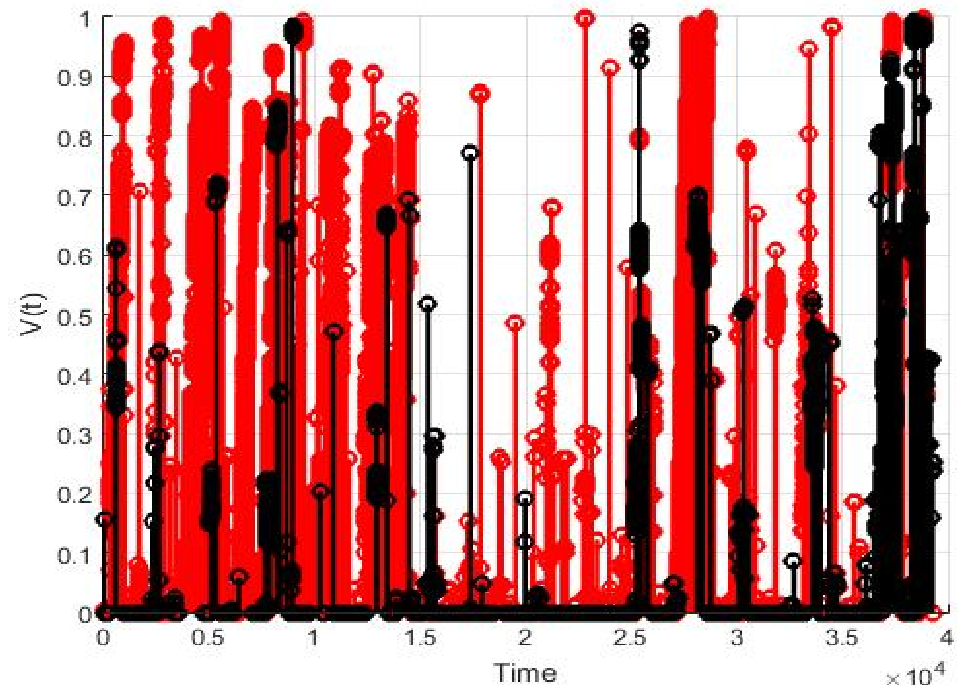

Figure 89: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

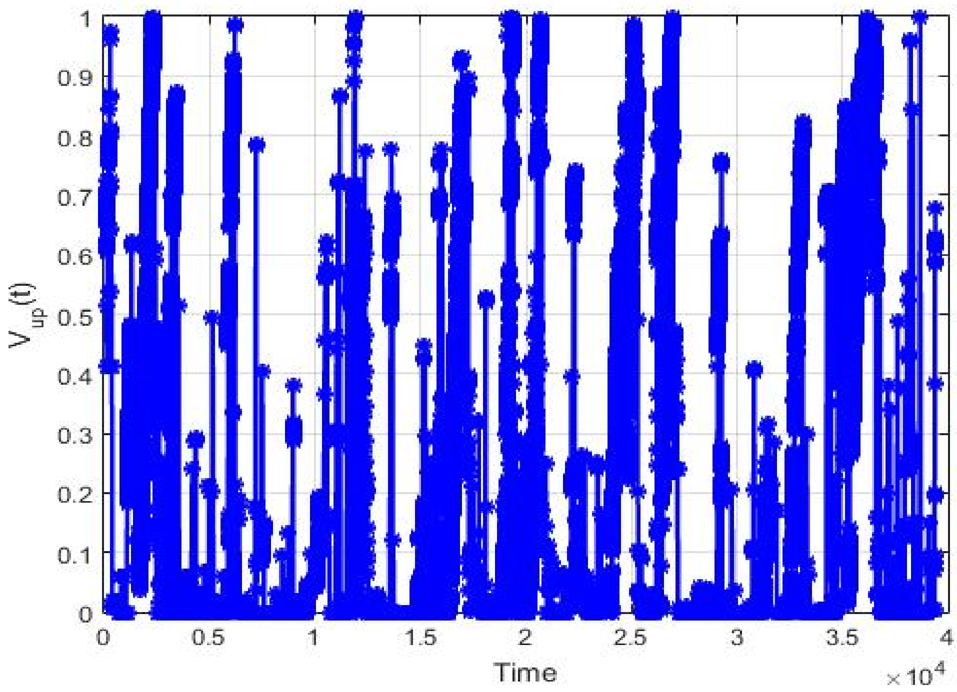

Figure 90: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

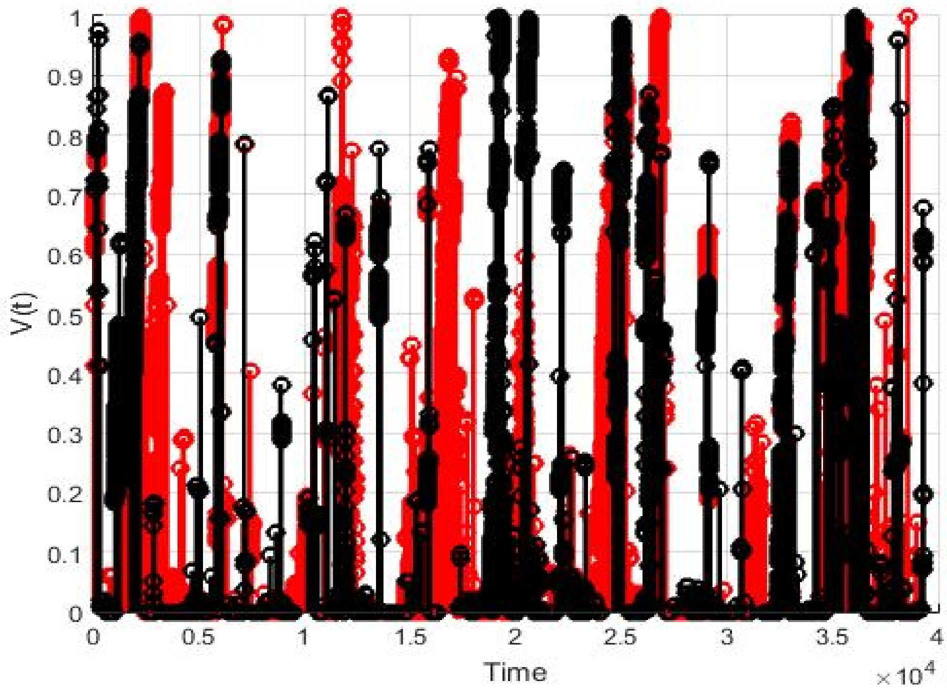

Figure 91: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

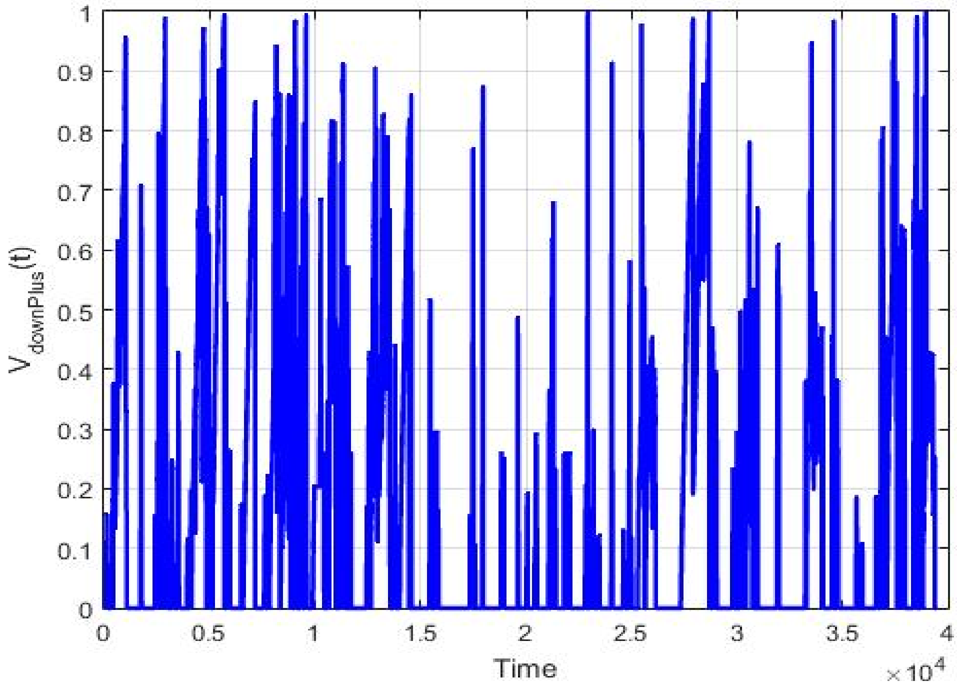

Figure 92: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

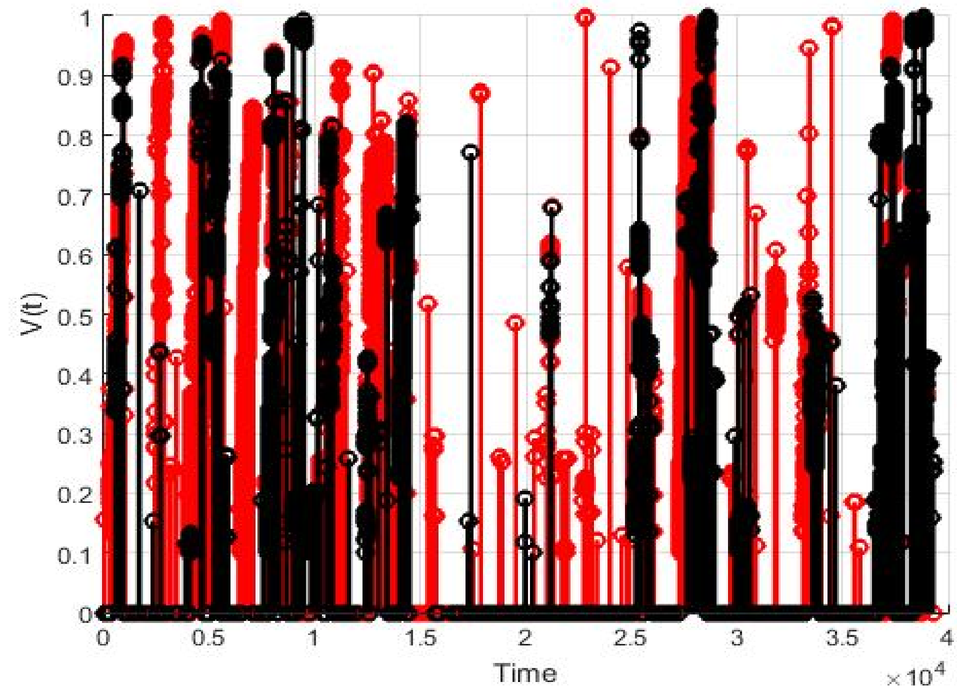

Figure 93: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

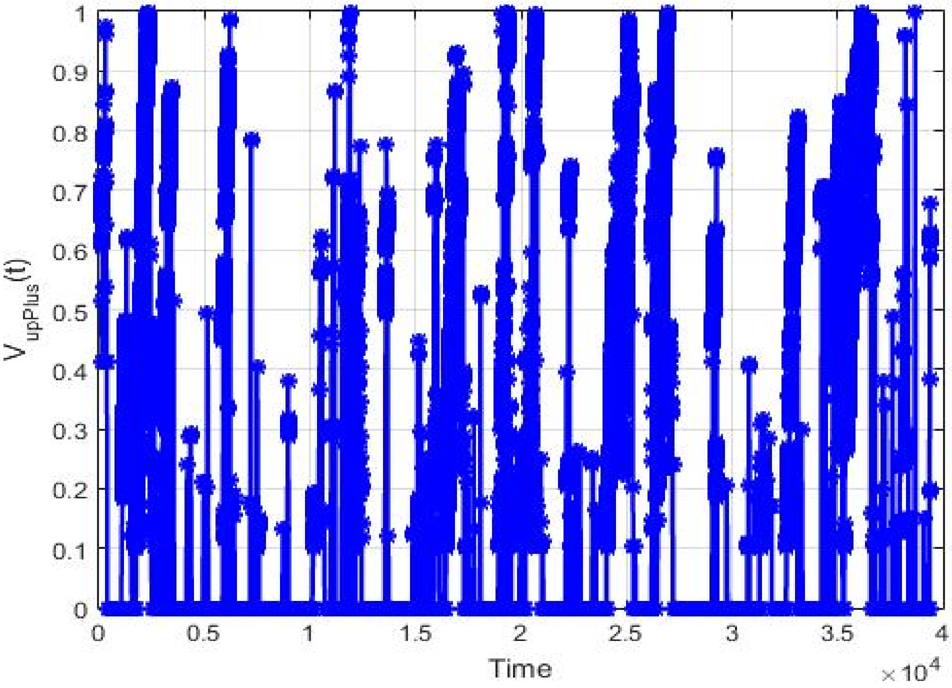

Figure 94: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

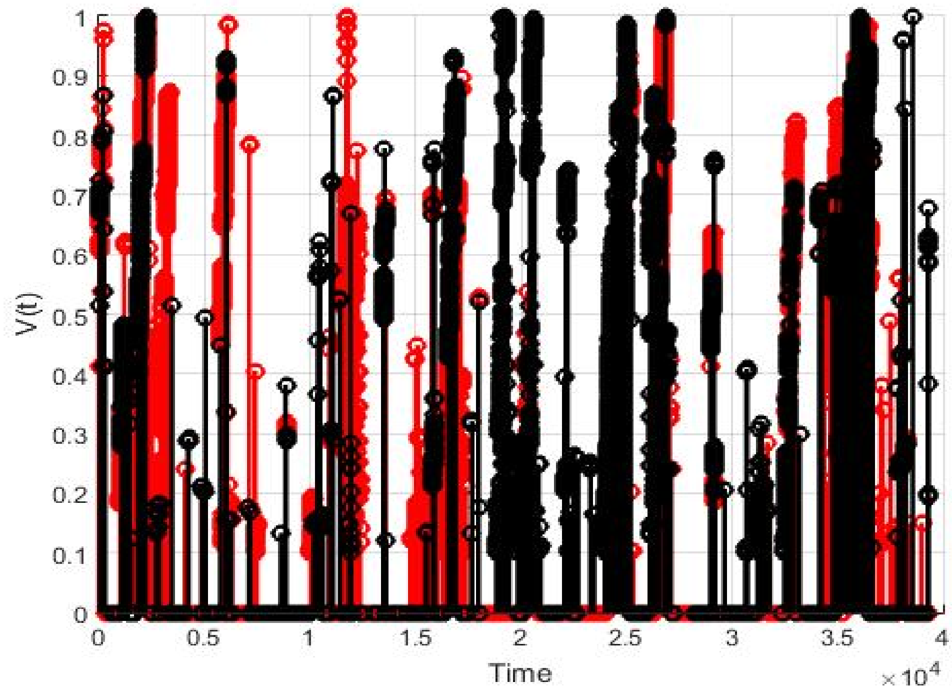

Figure 95: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 10\%$

Partition 2(Future window = 600, Past window 100, $V_{thres} = 10\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18594/39300	47.3%
Percantage of upward spread-crossing events	21962/39300	55.9%
Percantage of downward spread-crossing events($V_{thres} = 10\%$)	12433/39300	31.6%
Percantage of upward spread-crossing events($V_{thres} = 10\%$)	13530/39300	34.4%

Table 21: Partition 2(Future window = 600, Past window 100, $V_{thres} = 10\%$)

Partition 2(Future window = 600, Past window 100, $V_{thres} = 10\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	12068/19650	61.4%
Percantage of downward spread-crossing events(testing)	13047/19650	66.4%
Percantage of upward spread-crossing events(training)	12387/19650	63.0%
Percantage of upward spread-crossing events(testing)	12792/19650	65.0%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,training)	7495/19650	38.1%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,testing)	11621/19650	59.1%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,training)	7447/19650	37.9%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,testing)	11241/19650	57.2%

Table 22: Partition 2(Future window = 600, Past window 100, $V_{thres} = 10\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 2401 & 6724 \\ 858 & 9667 \end{matrix}$	Ctest Down = $\begin{matrix} 2488 & 9093 \\ 3062 & 5007 \end{matrix}$
Ctrain Up = $\begin{matrix} 6044 & 2765 \\ 3838 & 7003 \end{matrix}$	Ctest Up = $\begin{matrix} 4345 & 4184 \\ 3845 & 7276 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 7736 & 5148 \\ 2115 & 4651 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 4643 & 9340 \\ 2863 & 2804 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 8794 & 4547 \\ 2311 & 3998 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 9130 & 3299 \\ 5110 & 2111 \end{matrix}$

5.2.3 Scenario 2.3 (Future window = 600, Past Window = 200, $V_{thres} = 10\%$)

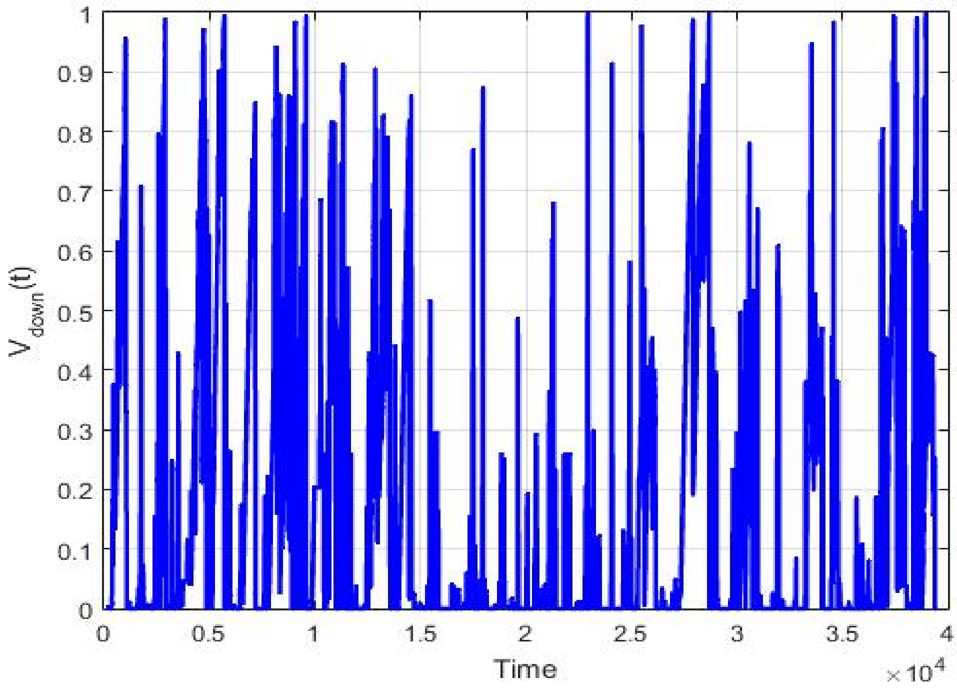

Figure 96: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

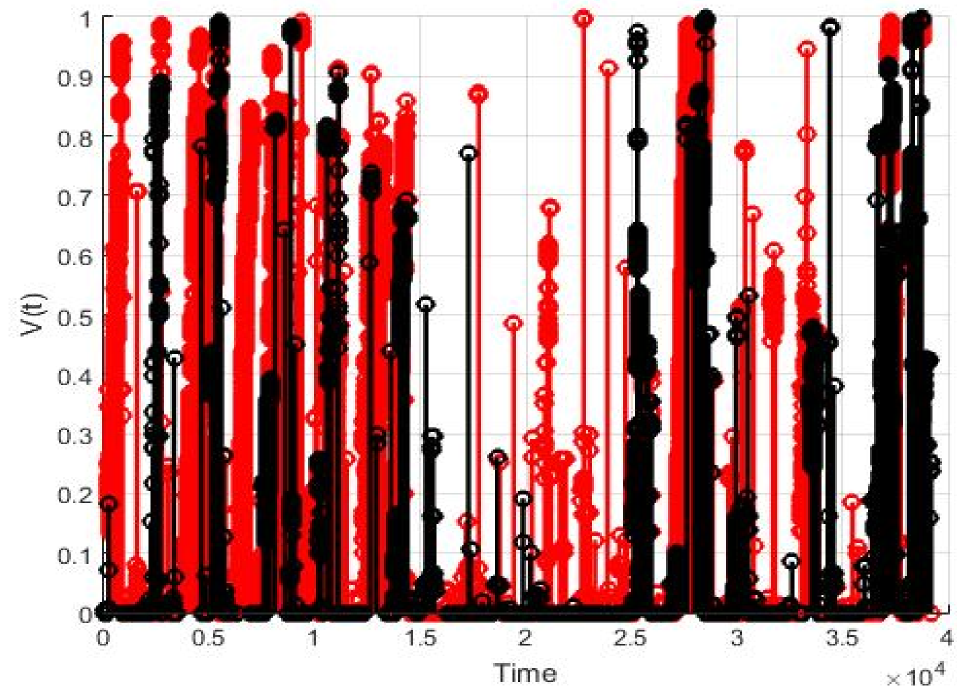

Figure 97: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

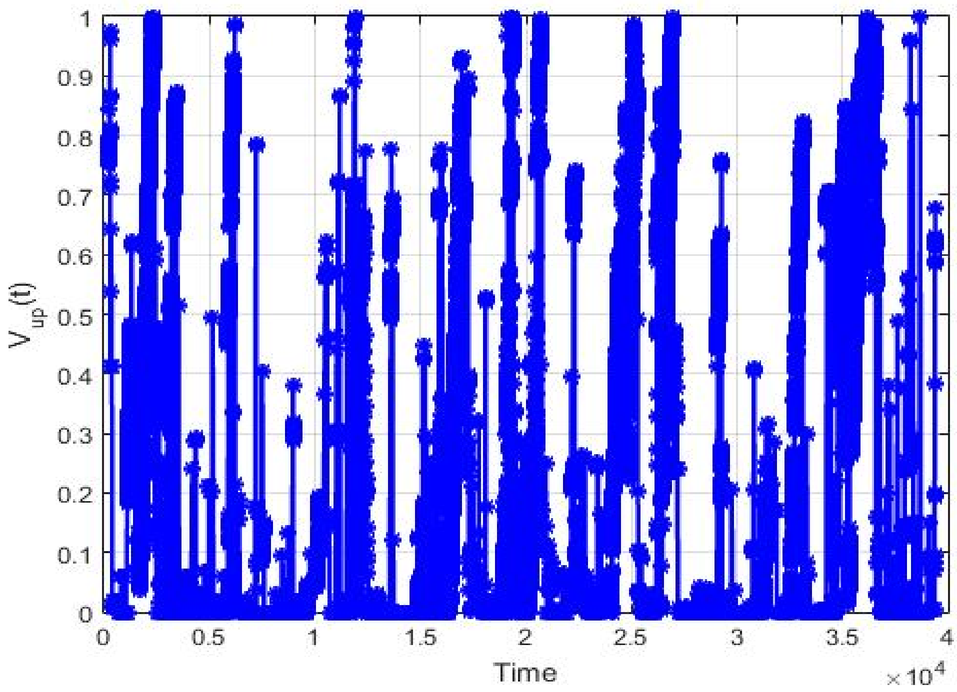

Figure 98: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

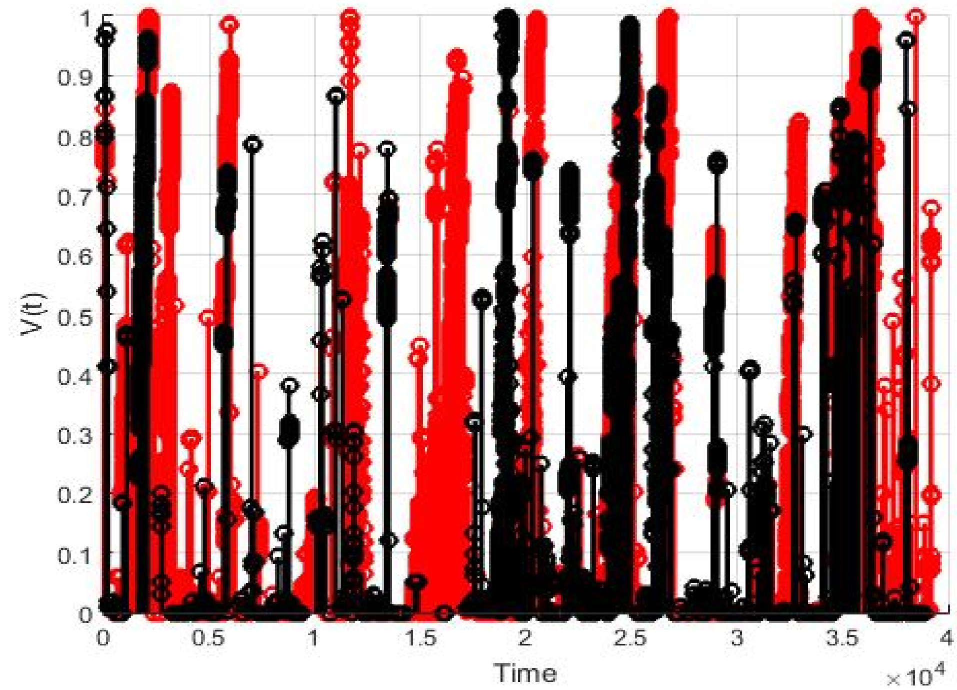

Figure 99: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

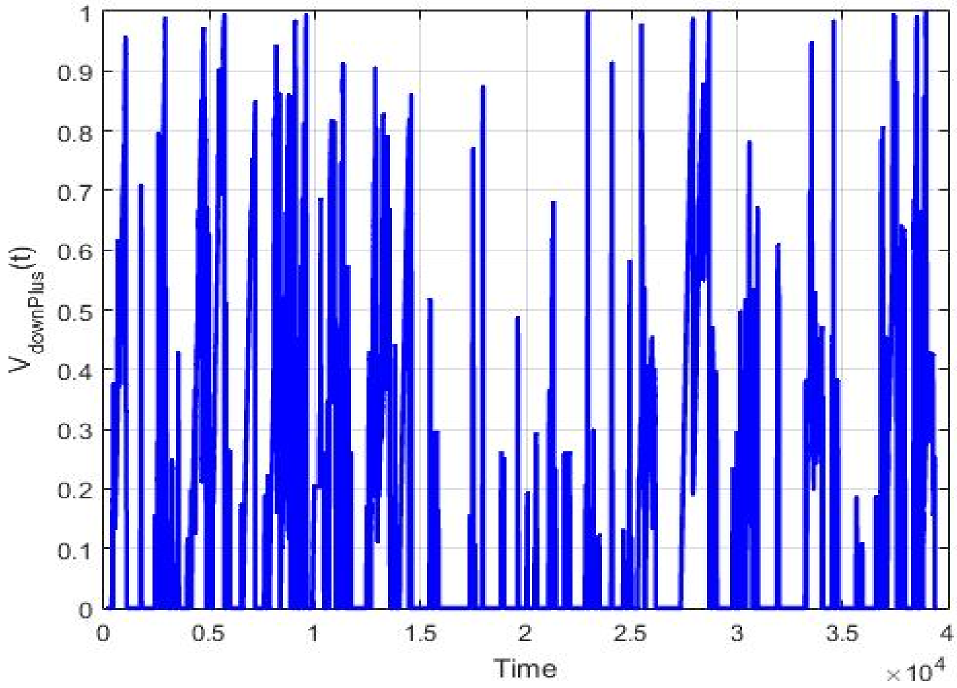

Figure 100: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

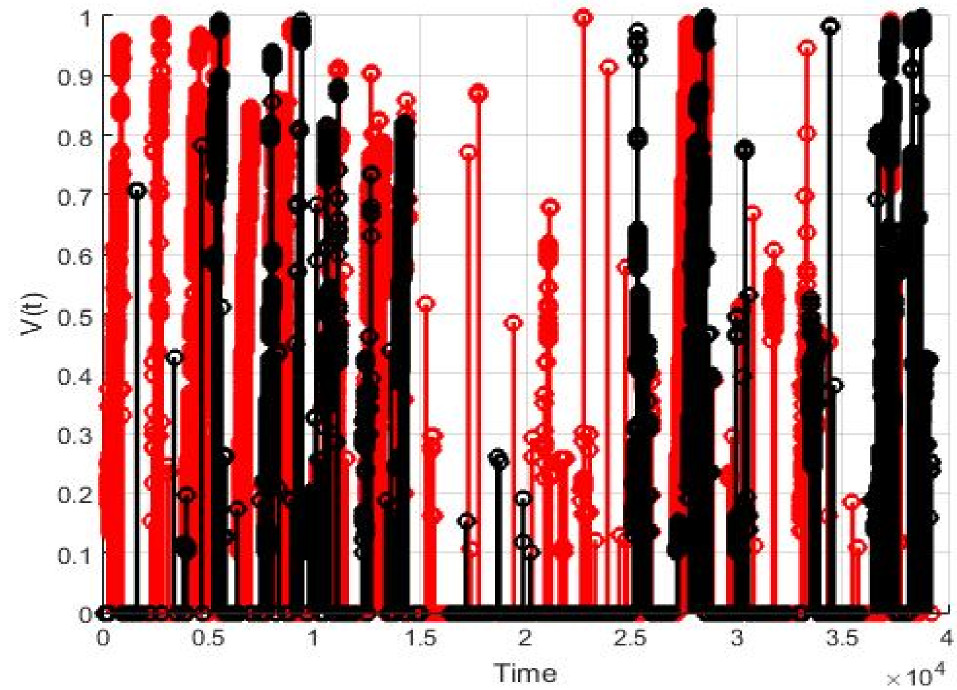

Figure 101: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

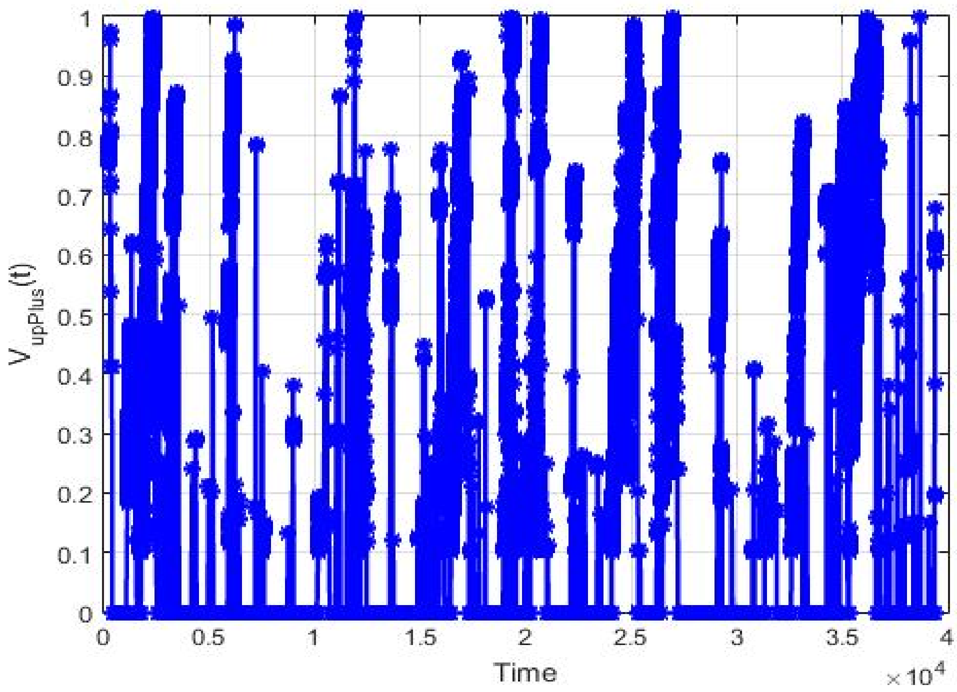

Figure 102: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

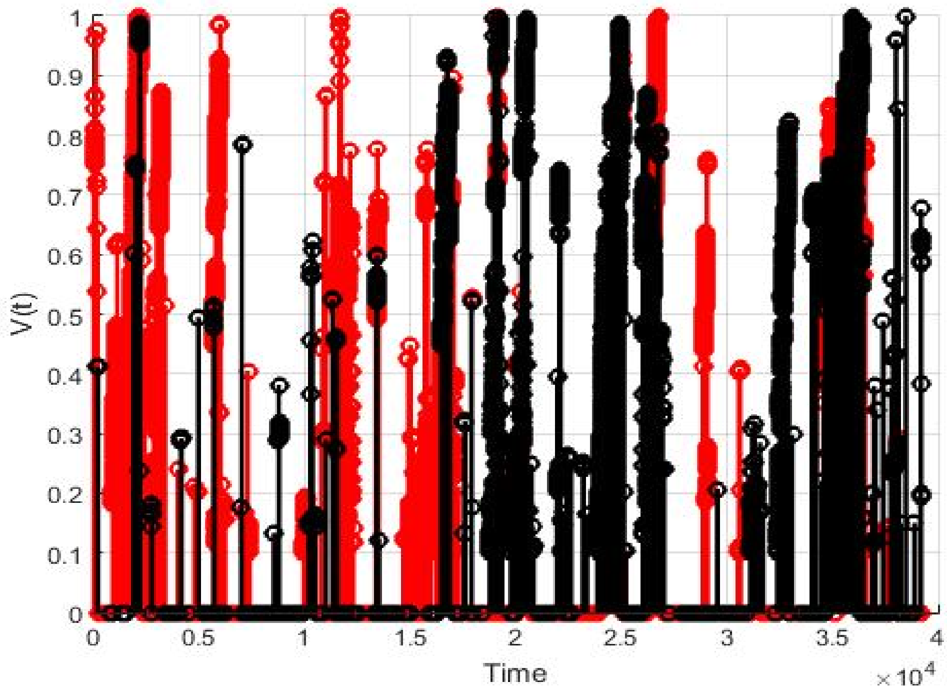

Figure 103: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 10\%$

Partition 2(Future window = 600, Past window 200, $V_{thres} = 10\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18574/39200	47.3%
Percantage of upward spread-crossing events	21862/39200	55.8%
Percantage of downward spread-crossing events($V_{thres} = 10\%$)	12432/39200	31.7%
Percantage of upward spread-crossing events($V_{thres} = 10\%$)	13430/39200	34.3%

Table 23: Partition 2(Future window = 600, Past window 200, $V_{thres} = 10\%$)

Partition 2(Future window = 600, Past window 200, $V_{thres} = 10\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	13554/19600	69.2%
Percantage of downward spread-crossing events(testing)	14547/19600	74.2%
Percantage of upward spread-crossing events(training)	14357/19600	73.3%
Percantage of upward spread-crossing events(testing)	13877/19600	70.8%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,training)	7500/19600	38.3%
Percantage of downward spread-crossing events($V_{thres} = 10\%$,testing)	11701/19600	59.7%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,training)	7703/19600	39.3%
Percantage of upward spread-crossing events($V_{thres} = 10\%$,testing)	10668/19600	54.4%

Table 24: Partition 2(Future window = 600, Past window 200, $V_{thres} = 10\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 4894 & 4201 \\ 1845 & 8660 \end{matrix}$	Ctest Down = $\begin{matrix} 3747 & 7784 \\ 4316 & 3753 \end{matrix}$
Ctrain Up = $\begin{matrix} 6820 & 2029 \\ 3024 & 7727 \end{matrix}$	Ctest Up = $\begin{matrix} 4604 & 3885 \\ 4014 & 7097 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 9301 & 3534 \\ 1709 & 5056 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 5429 & 8504 \\ 3393 & 2274 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 9047 & 4344 \\ 1379 & 4830 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 9439 & 2940 \\ 5992 & 1229 \end{matrix}$

5.2.4 Scenario 2.4 (Future window = 600, Past Window = 50, $V_{thres} = 50\%$)

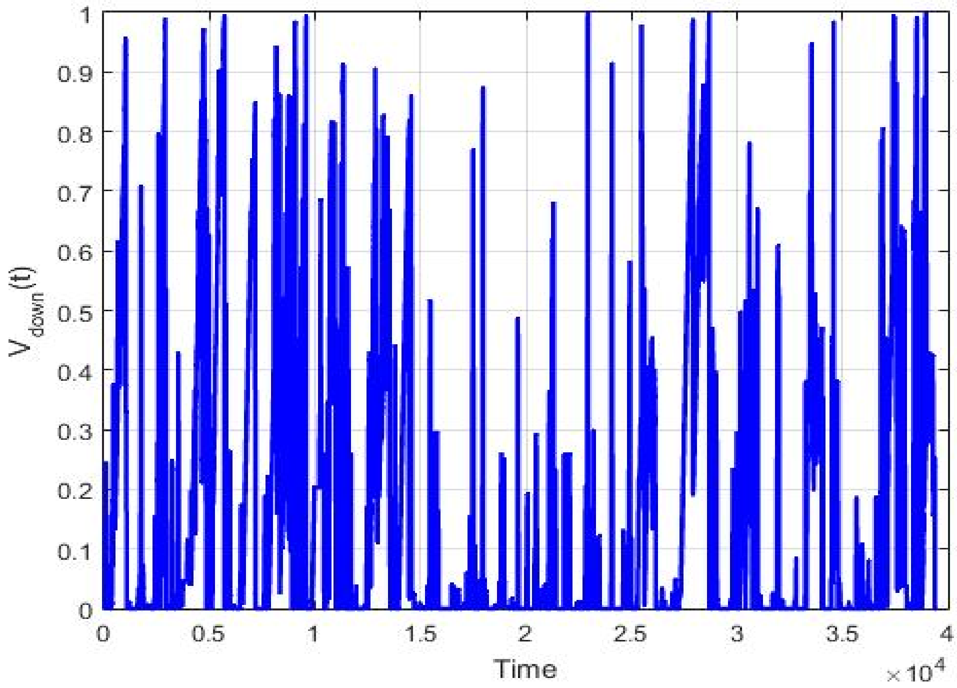

Figure 104: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

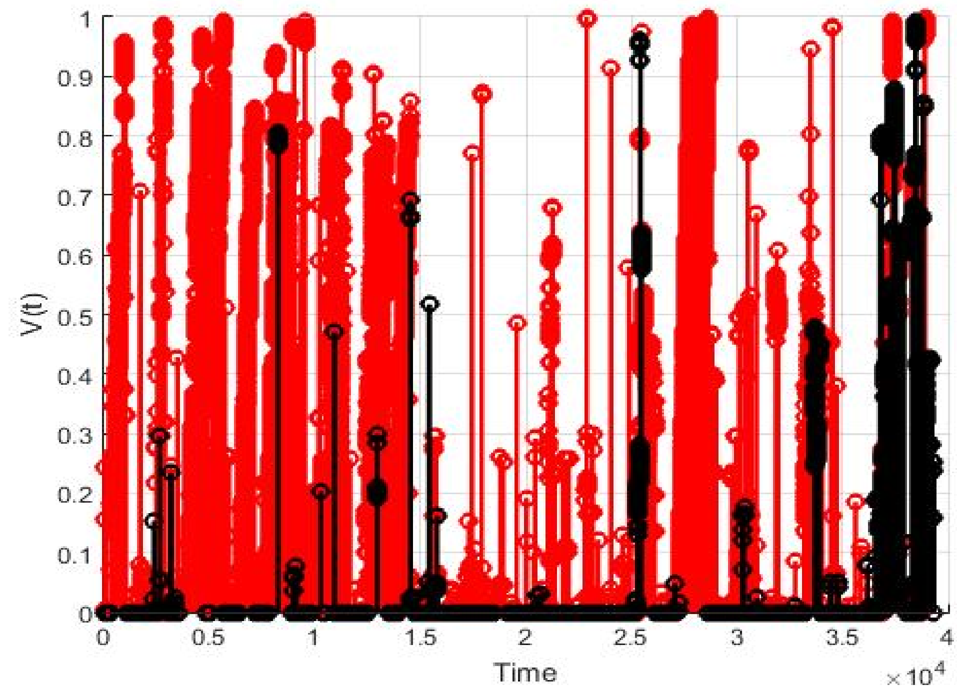

Figure 105: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

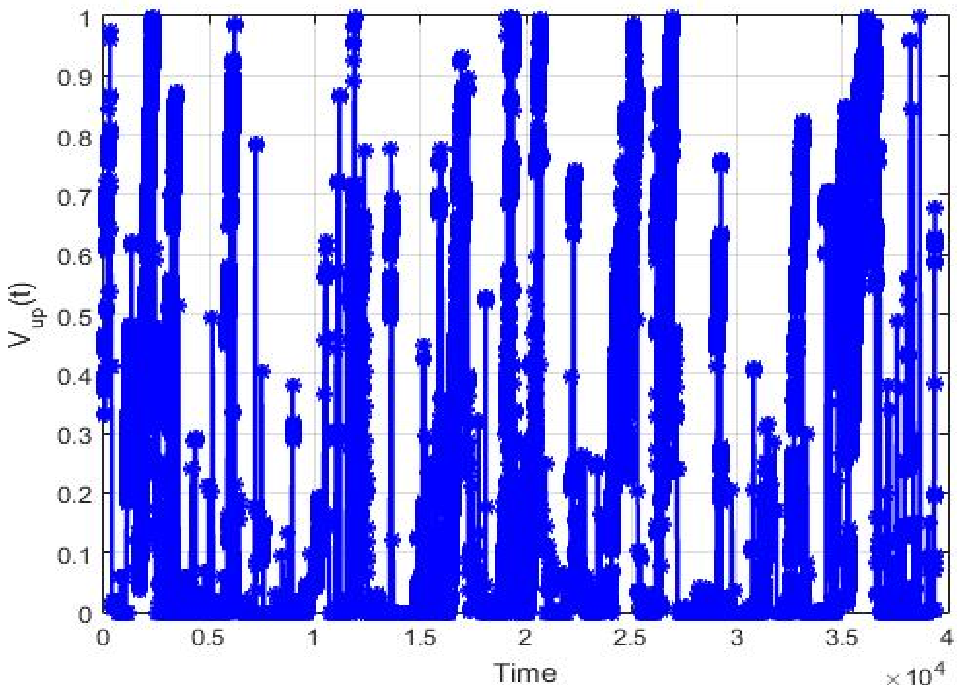

Figure 106: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

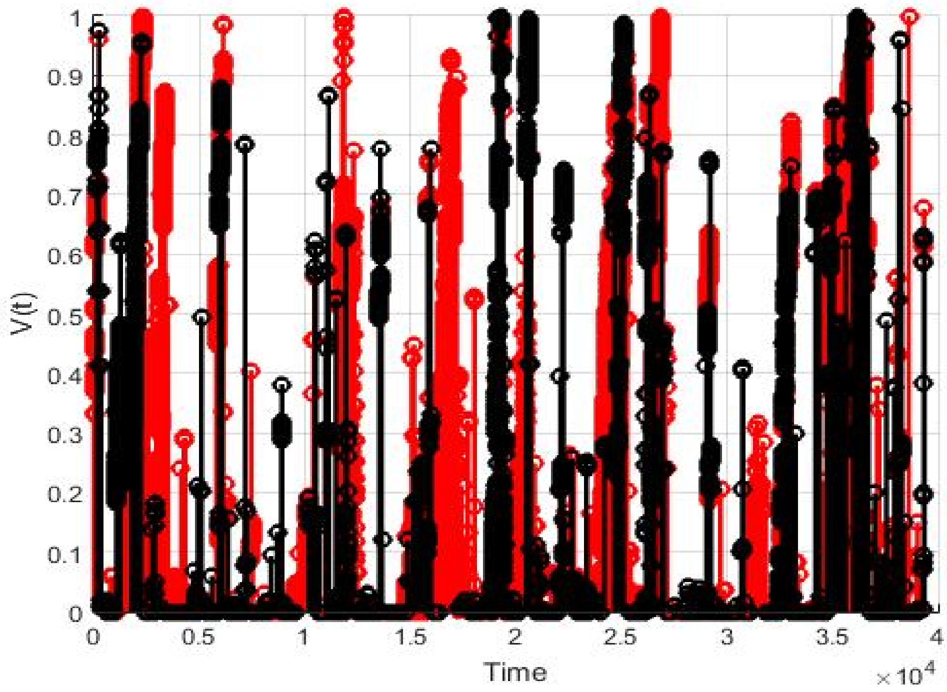

Figure 107: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

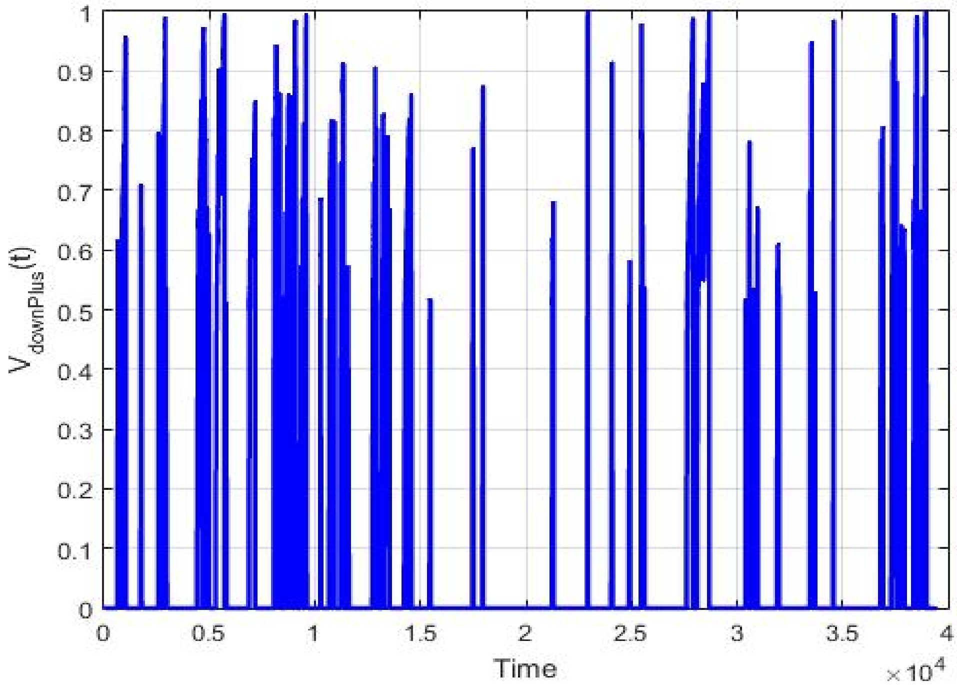

Figure 108: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

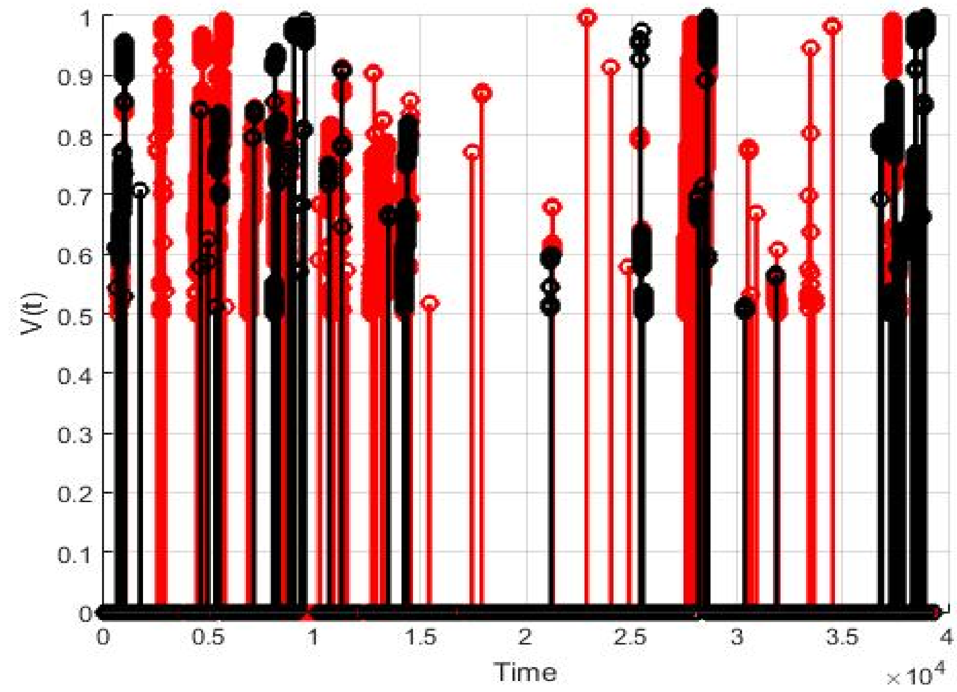

Figure 109: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

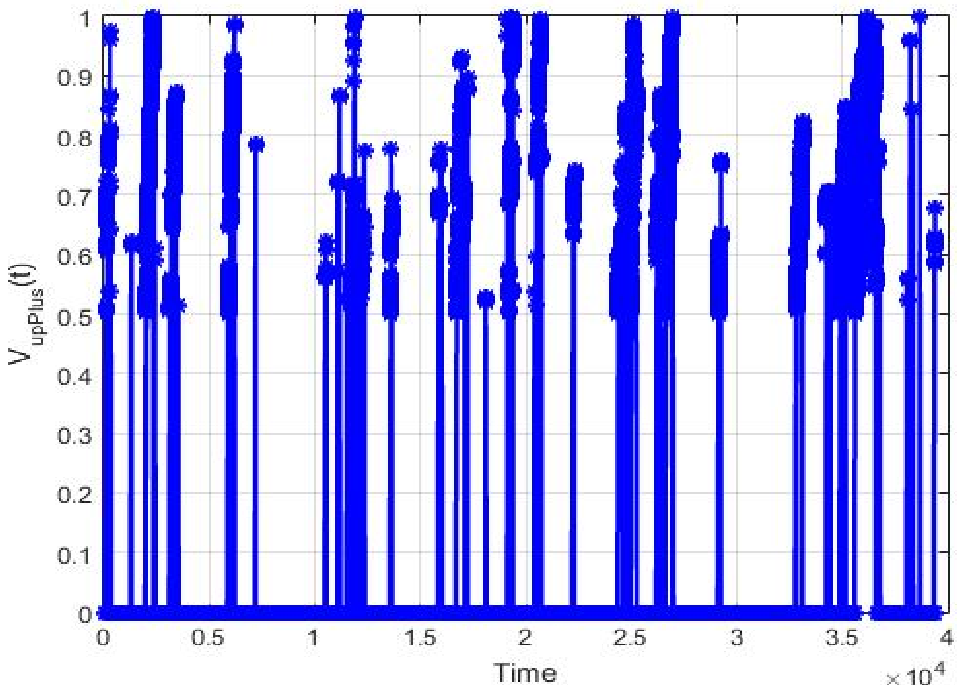

Figure 110: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

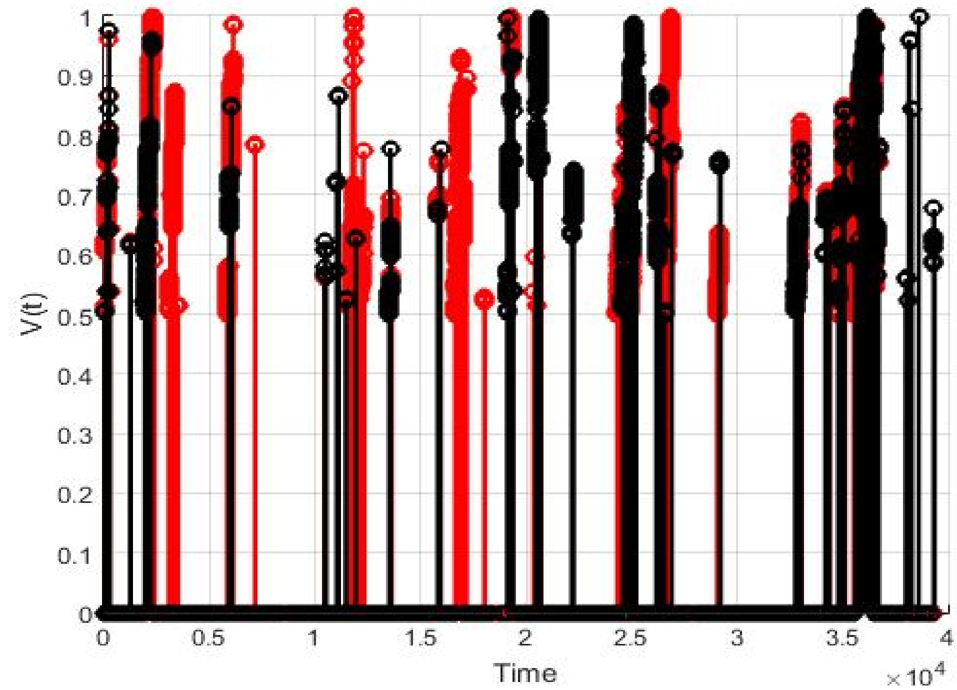

Figure 111: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 50\%$

Partition 2(Future window = 600, Past window 50, $V_{thres} = 50\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18603/39350	47.3%
Percantage of upward spread-crossing events	22012/39350	55.9%
Percantage of downward spread-crossing events($V_{thres} = 50\%$)	4371/39350	11.1%
Percantage of upward spread-crossing events($V_{thres} = 50\%$)	5460/39350	13.9%

Table 25: Partition 2(Future window = 600, Past window 50, $V_{thres} = 50\%$)

Partition 2(Future window = 600, Past window 50, $V_{thres} = 50\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	11546/19675	58.7%
Percantage of downward spread-crossing events(testing)	12510/19675	63.6%
Percantage of upward spread-crossing events(training)	9592/19675	48.8%
Percantage of upward spread-crossing events(testing)	10322/19675	52.5%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,training)	7416/19675	37.7%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,testing)	11671/19675	59.3%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,training)	7032/19675	35.7%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,testing)	9824/19675	49.9%

Table 26: Partition 2(Future window = 600, Past window 50, $V_{thres} = 50\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 1315 & 7826 \\ 303 & 10231 \end{matrix}$	Ctest Down = $\begin{matrix} 1627 & 9979 \\ 2280 & 5789 \end{matrix}$
Ctrain Up = $\begin{matrix} 5335 & 3474 \\ 3691 & 7175 \end{matrix}$	Ctest Up = $\begin{matrix} 3920 & 4609 \\ 3395 & 7751 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 7440 & 9494 \\ 589 & 2152 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 5942 & 12103 \\ 540 & 1090 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 8859 & 8734 \\ 619 & 1463 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 8040 & 8257 \\ 1594 & 1784 \end{matrix}$

5.2.5 Scenario 2.5 (Future window = 600, Past Window = 100, $V_{thres} = 50\%$)

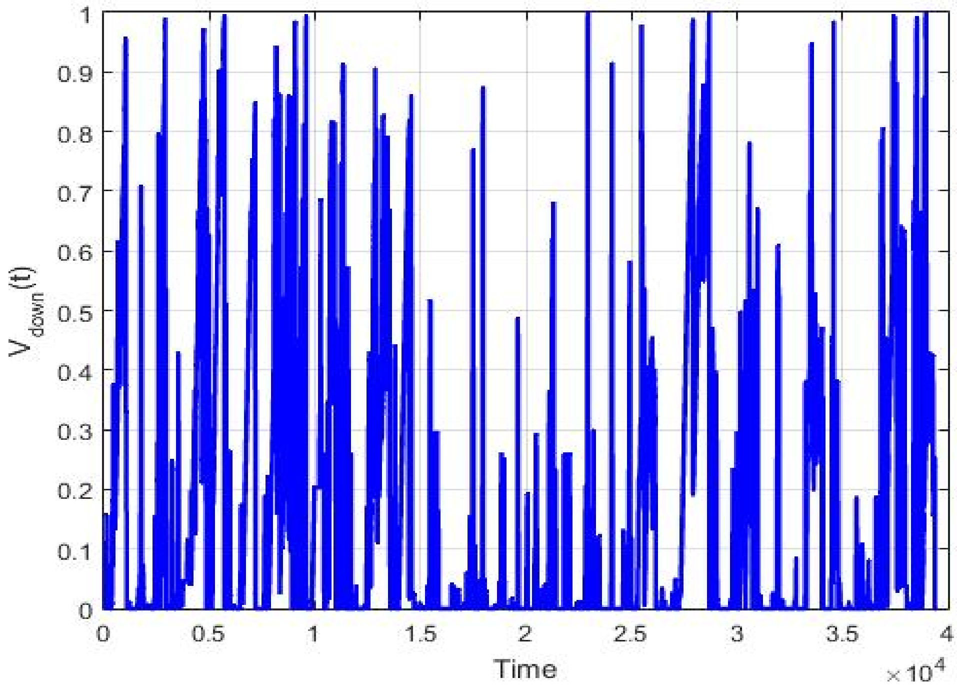

Figure 112: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

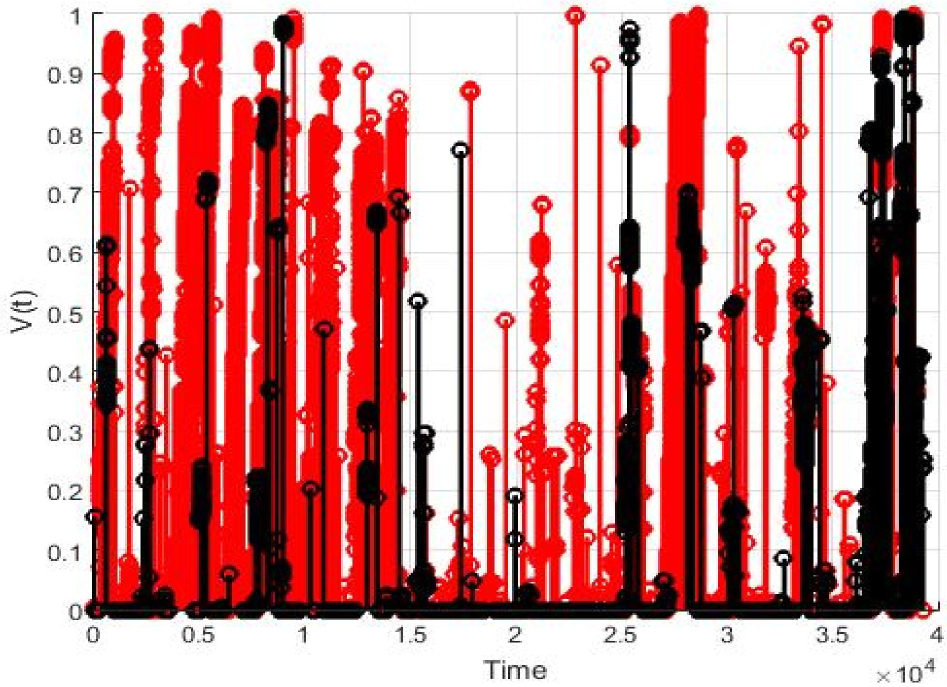

Figure 113: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

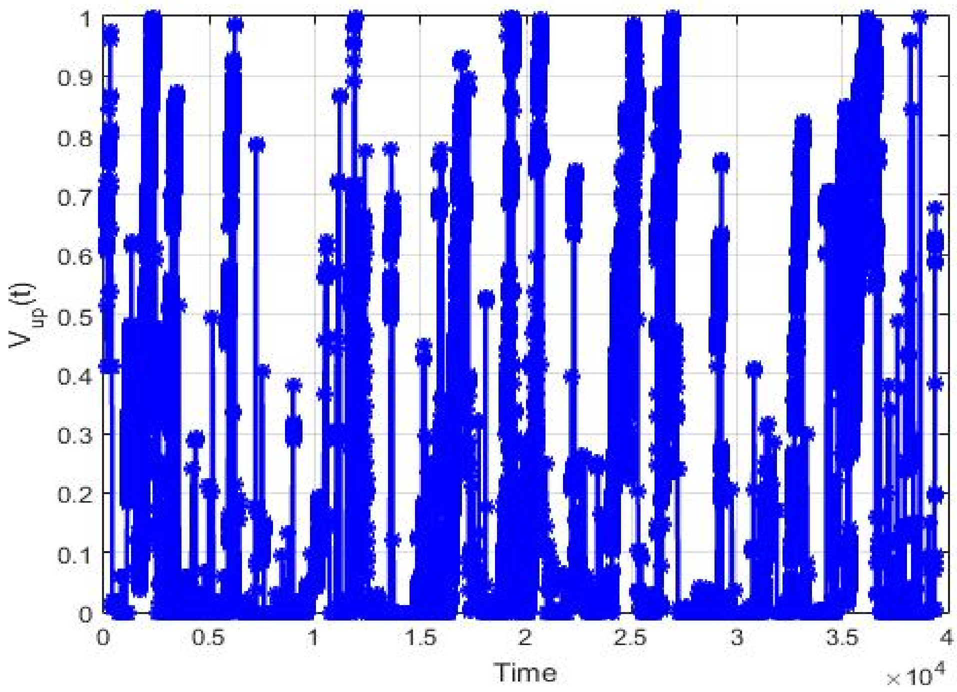

Figure 114: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

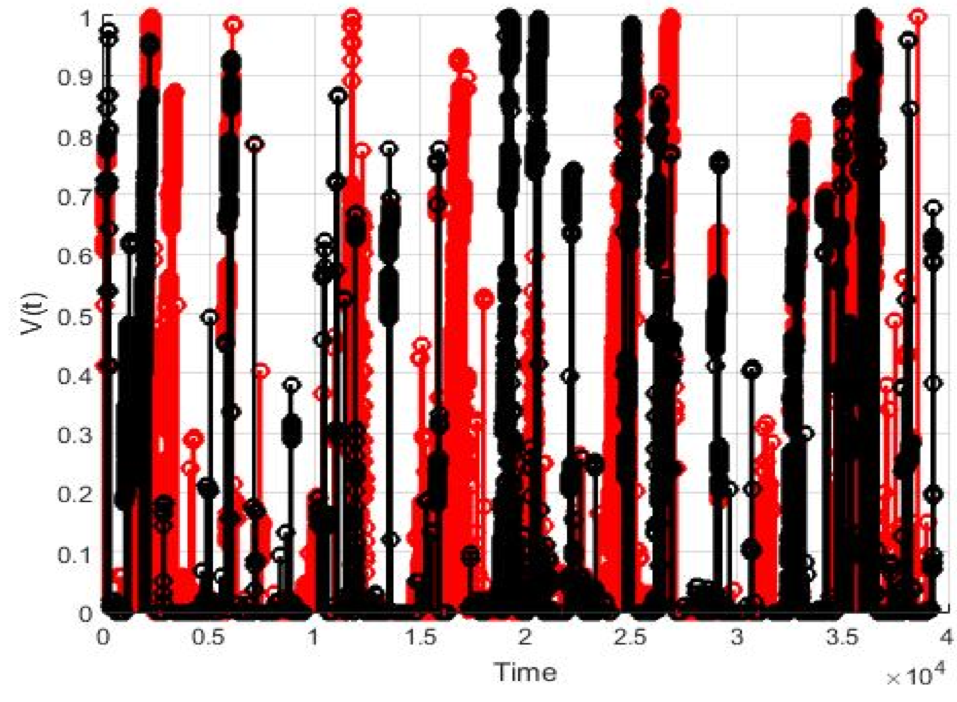

Figure 115: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

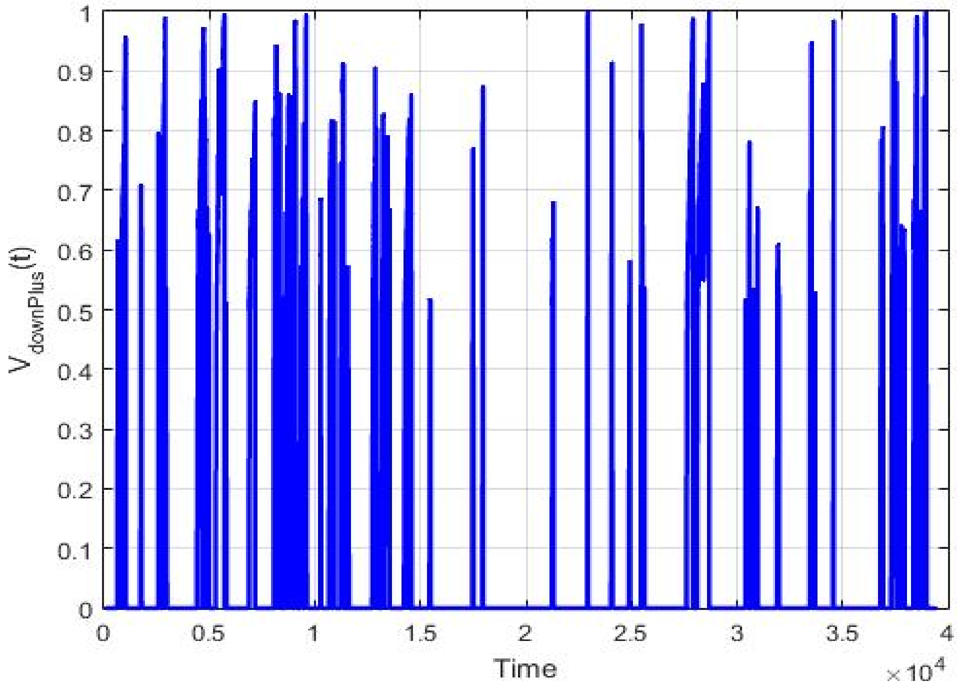

Figure 116: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

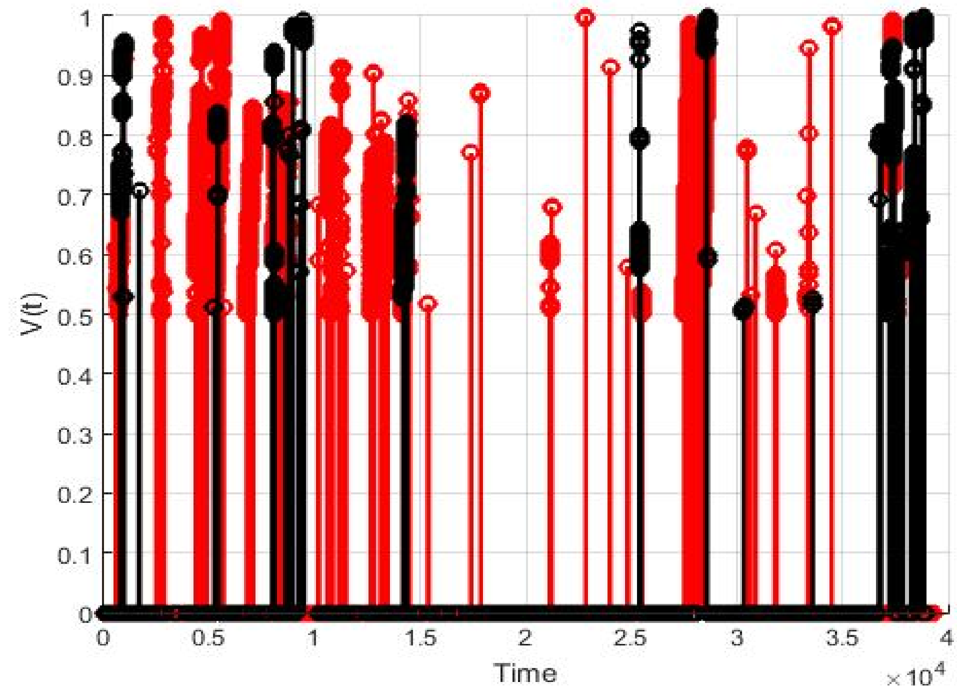

Figure 117: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

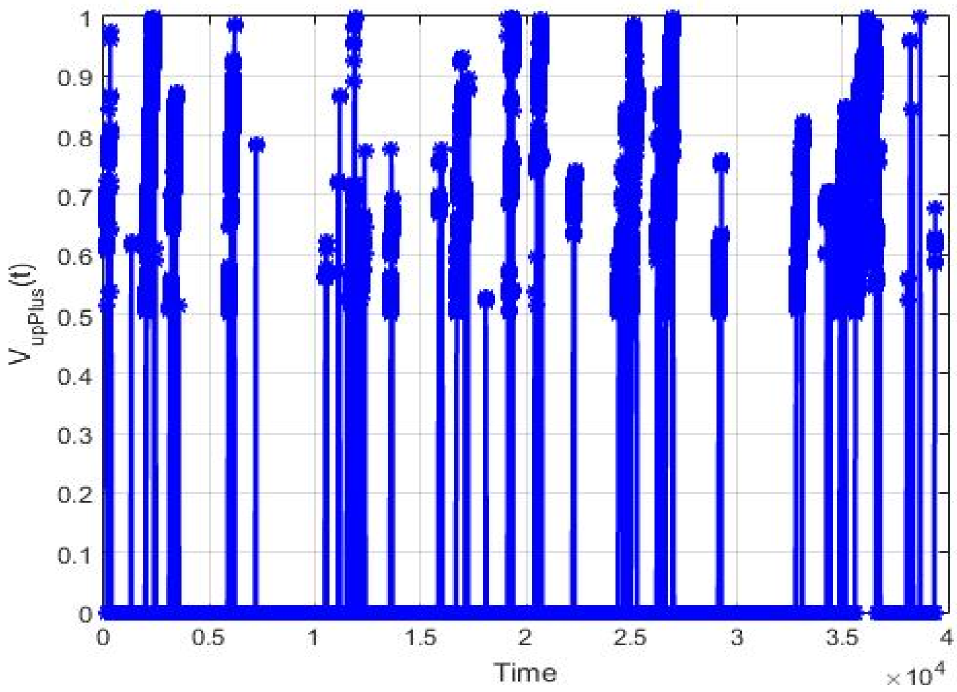

Figure 118: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

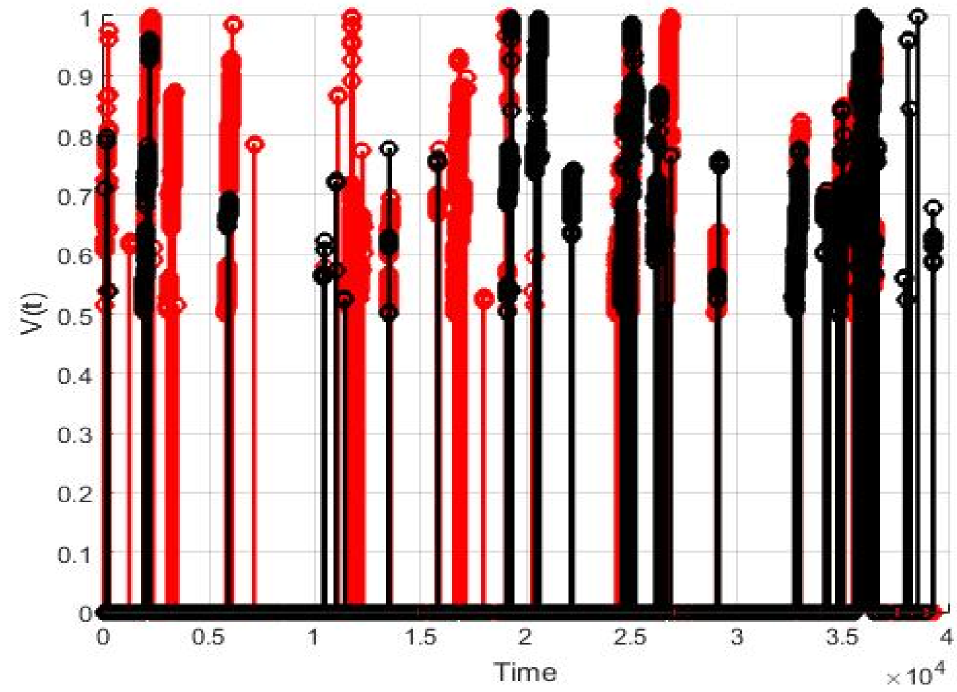

Figure 119: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 50\%$

Partition 2(Future window = 600, Past window 100, $V_{thres} = 50\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18594/39300	47.3%
Percantage of upward spread-crossing events	21962/39300	55.9%
Percantage of downward spread-crossing events($V_{thres} = 50\%$)	4371/39300	11.1%
Percantage of upward spread-crossing events($V_{thres} = 50\%$)	5452/39300	13.9%

Table 27: Partition 2(Future window = 600, Past window 100, $V_{thres} = 50\%$)

Partition 2(Future window = 600, Past window 100, $V_{thres} = 50\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	12068/19650	61.4%
Percantage of downward spread-crossing events(testing)	13047/19650	66.3%
Percantage of upward spread-crossing events(training)	10090/19650	51.3%
Percantage of upward spread-crossing events(testing)	10697/19650	54.4%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,training)	7495/19650	38.1%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,testing)	11621/19650	59.1%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,training)	7376/19650	37.5%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,testing)	10099/19650	51.4%

Table 28: Partition 2(Future window = 600, Past window 100, $V_{thres} = 50\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 2401 & 6724 \\ 858 & 9667 \end{matrix}$	Ctest Down = $\begin{matrix} 2488 & 9093 \\ 3062 & 5007 \end{matrix}$
Ctrain Up = $\begin{matrix} 6044 & 2765 \\ 3838 & 7003 \end{matrix}$	Ctest Up = $\begin{matrix} 4345 & 4184 \\ 3845 & 7276 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 7833 & 9076 \\ 484 & 2257 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 6311 & 11709 \\ 565 & 1065 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 9007 & 8569 \\ 384 & 1690 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 8608 & 7664 \\ 1887 & 1491 \end{matrix}$

5.2.6 Scenario 2.6 (Future window = 600, Past Window = 200, $V_{thres} = 50\%$)

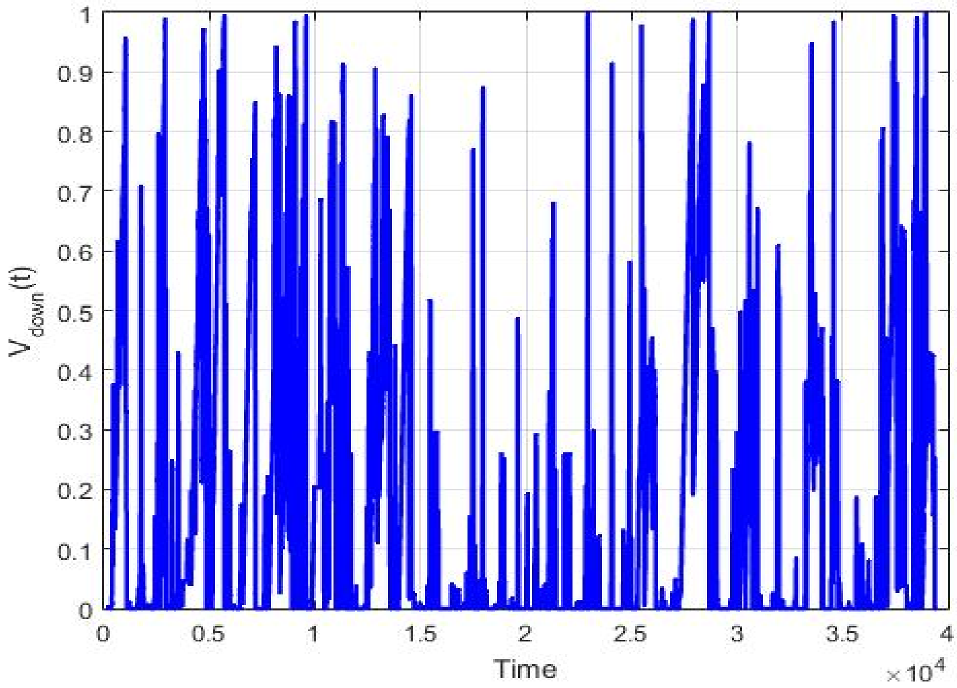

Figure 120: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

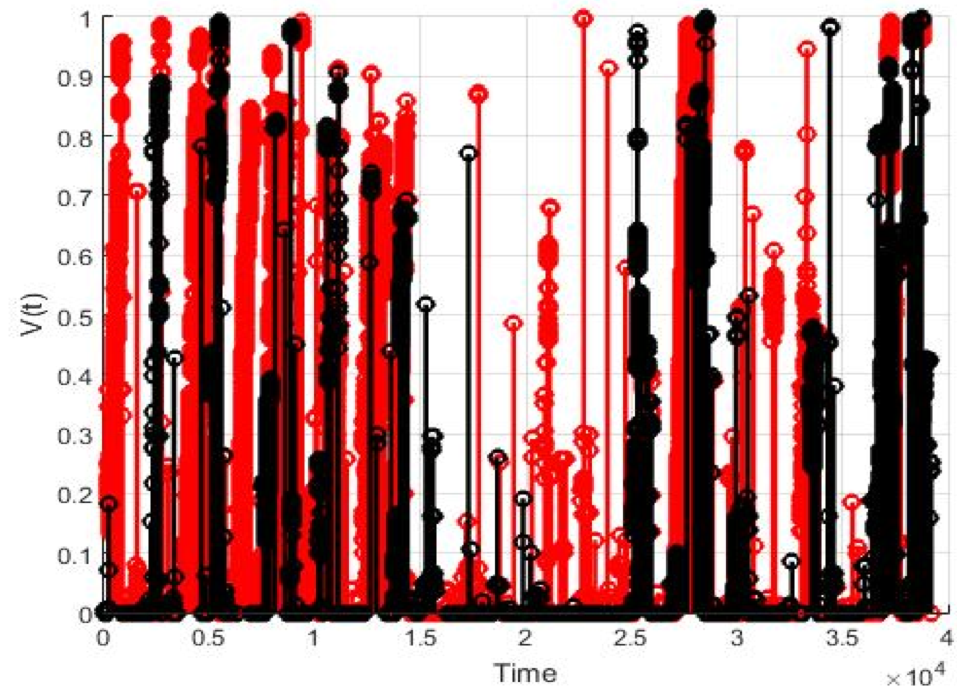

Figure 121: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

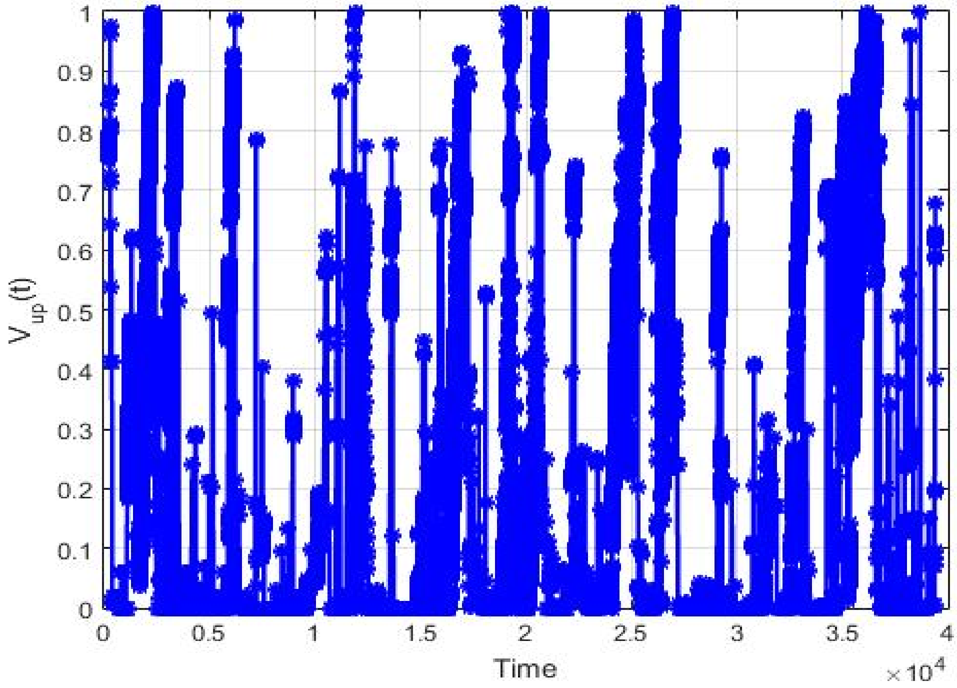

Figure 122: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

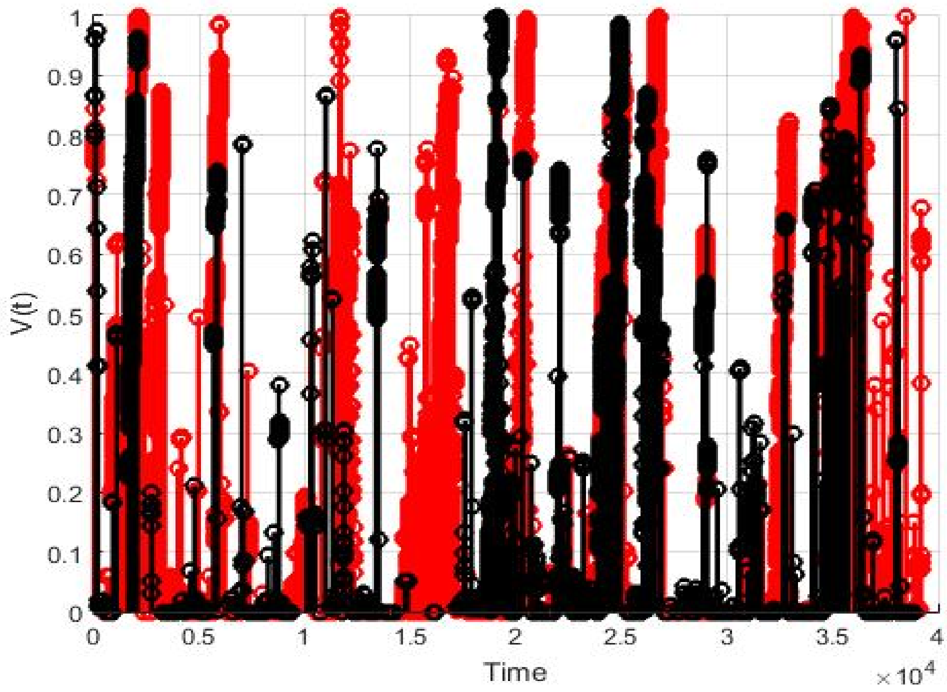

Figure 123: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

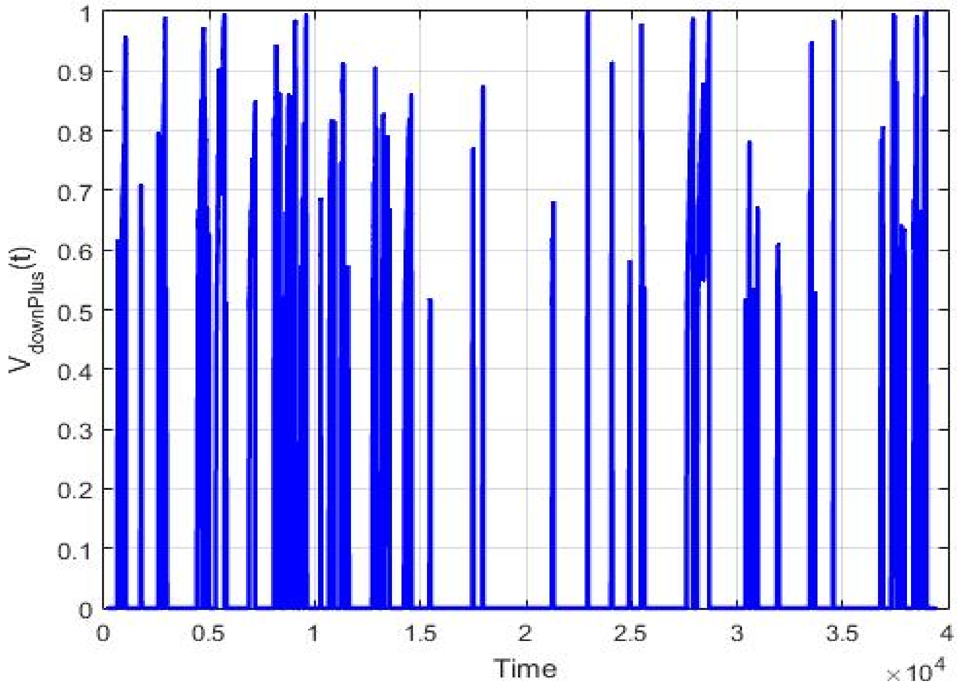

Figure 124: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

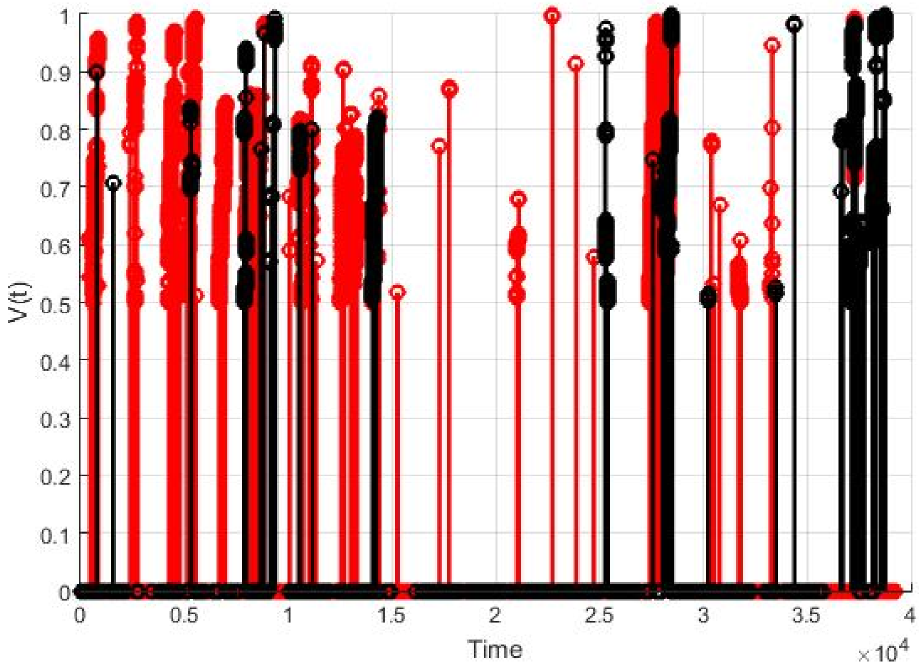

Figure 125: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

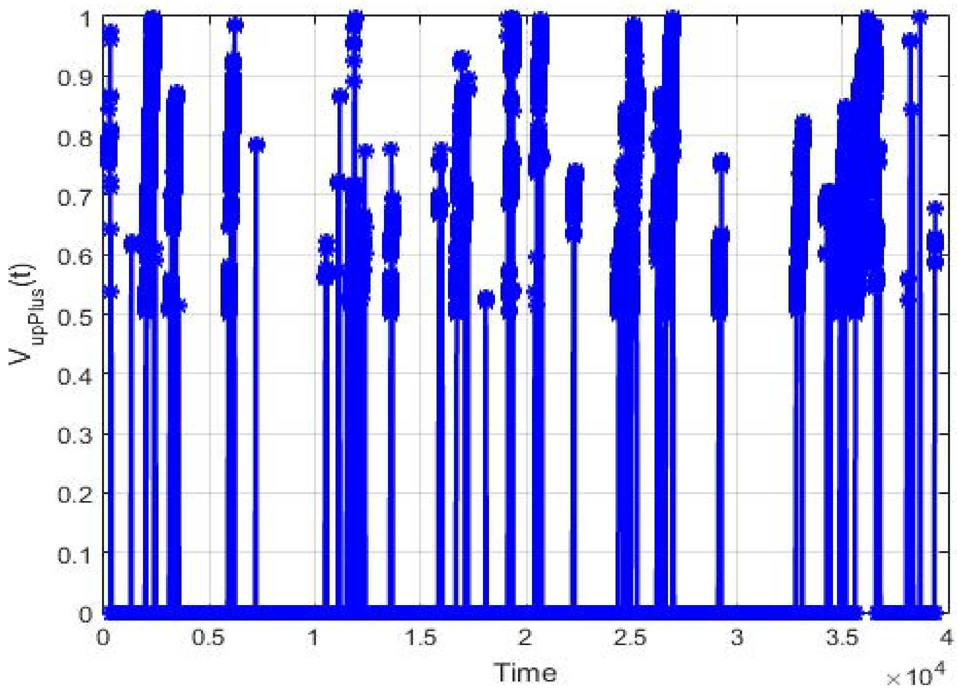

Figure 126: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

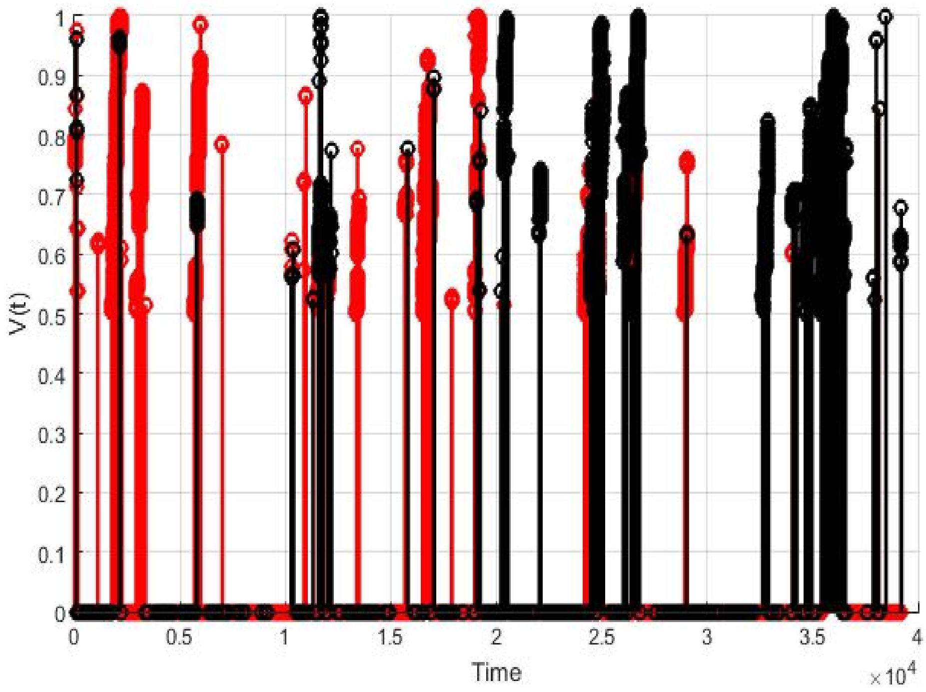

Figure 127: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 50\%$

Partition 2(Future window = 600, Past window 200, $V_{thres} = 50\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18574/39200	47.3%
Percantage of upward spread-crossing events	21862/39200	55.8%
Percantage of downward spread-crossing events($V_{thres} = 50\%$)	4371/39200	11.2%
Percantage of upward spread-crossing events($V_{thres} = 50\%$)	5360/39200	13.7%

Table 29: Partition 2(Future window = 600, Past window 200, $V_{thres} = 50\%$)

Partition 2(Future window = 600, Past window 200, $V_{thres} = 50\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	13554/19600	69.2%
Percantage of downward spread-crossing events(testing)	14547/19600	74.2%
Percantage of upward spread-crossing events(training)	12894/19600	65.7%
Percantage of upward spread-crossing events(testing)	15311/19600	78.1%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,training)	7500/19600	38.3%
Percantage of downward spread-crossing events($V_{thres} = 50\%$,testing)	11701/19600	59.7%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,training)	8901/19600	45.4%
Percantage of upward spread-crossing events($V_{thres} = 50\%$,testing)	10637/19600	54.3%

Table 30: Partition 2(Future window = 600, Past window 200, $V_{thres} = 50\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 4894 & 4201 \\ 1845 & 8660 \end{matrix}$	Ctest Down = $\begin{matrix} 3747 & 7784 \\ 4316 & 3753 \end{matrix}$
Ctrain Up = $\begin{matrix} 6820 & 2029 \\ 3024 & 7727 \end{matrix}$	Ctest Up = $\begin{matrix} 4604 & 3885 \\ 4014 & 7097 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 10542 & 6317 \\ 389 & 2352 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 8071 & 9899 \\ 800 & 830 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 13570 & 4048 \\ 241 & 1741 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 10053 & 6169 \\ 2794 & 584 \end{matrix}$

5.2.7 Scenario 2.7 (Future window = 600, Past Window = 50, $V_{thres} = 80\%$)

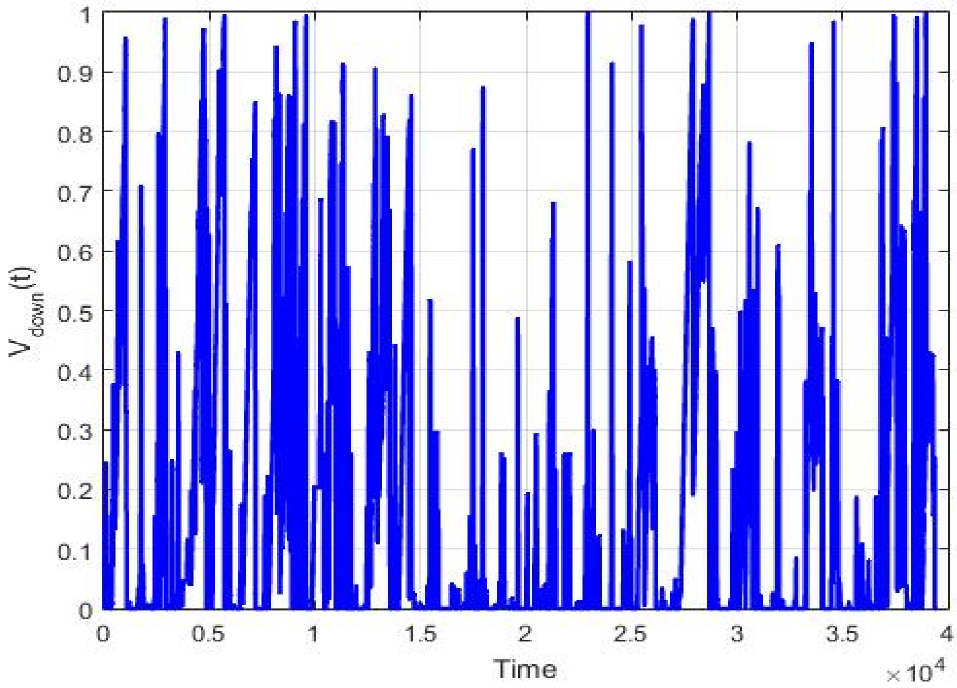

Figure 128: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

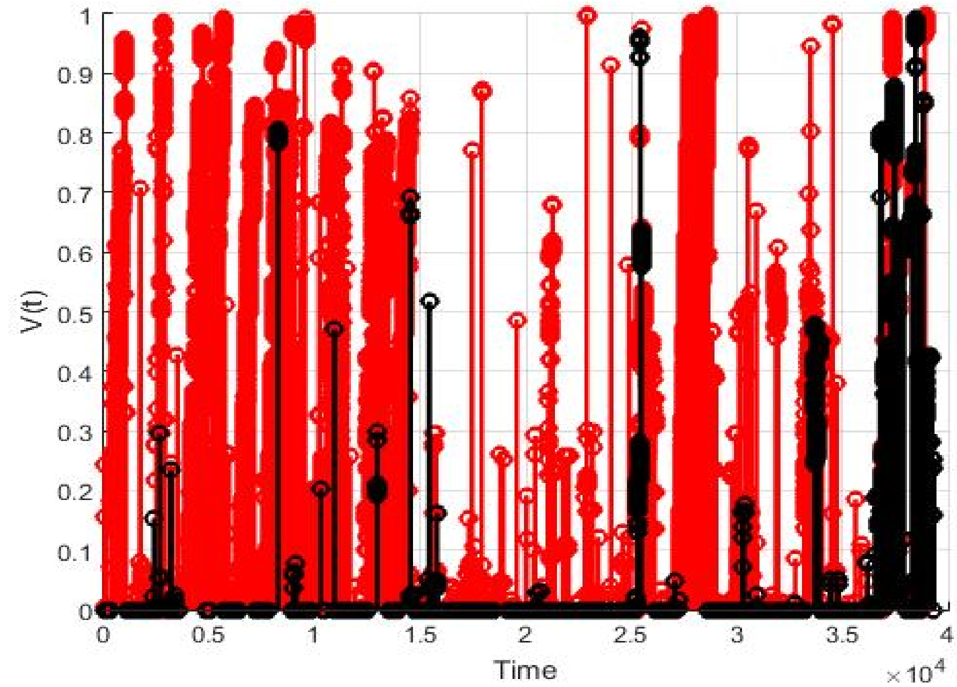

Figure 129: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

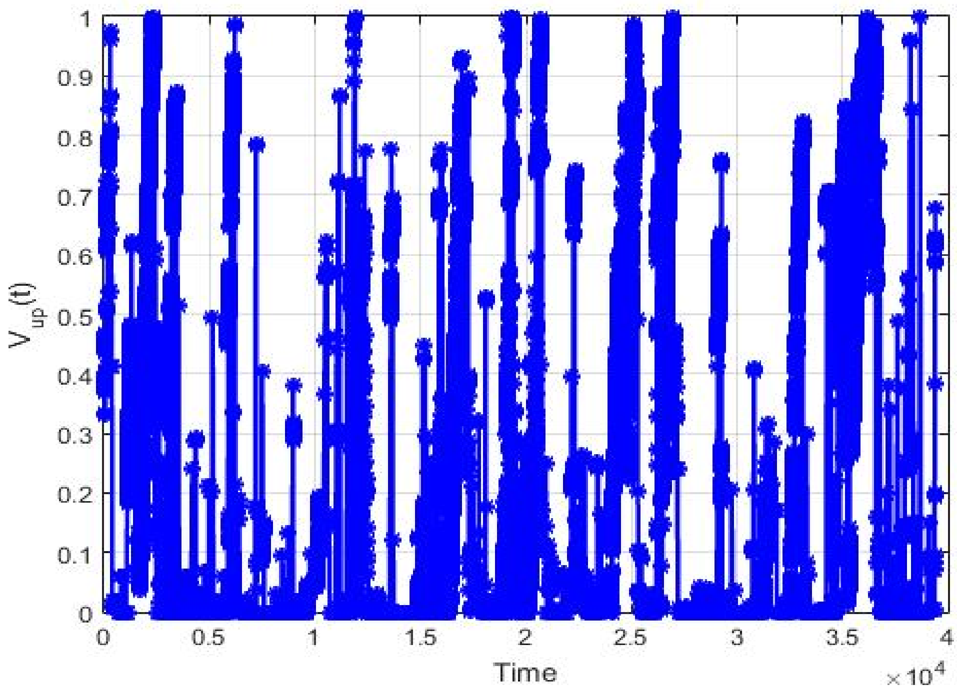

Figure 130: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

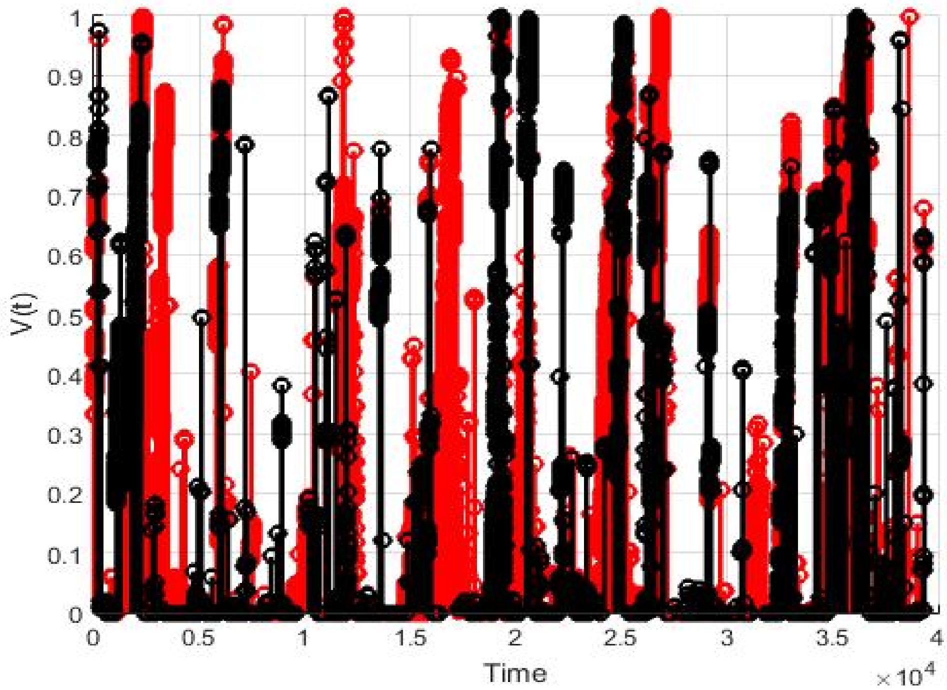

Figure 131: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

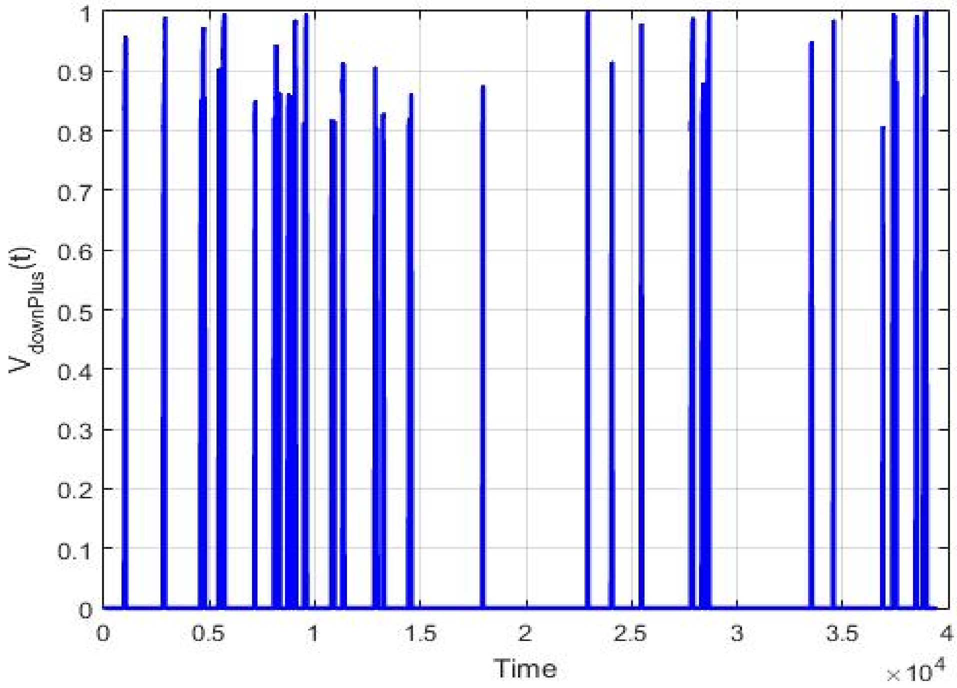

Figure 132: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

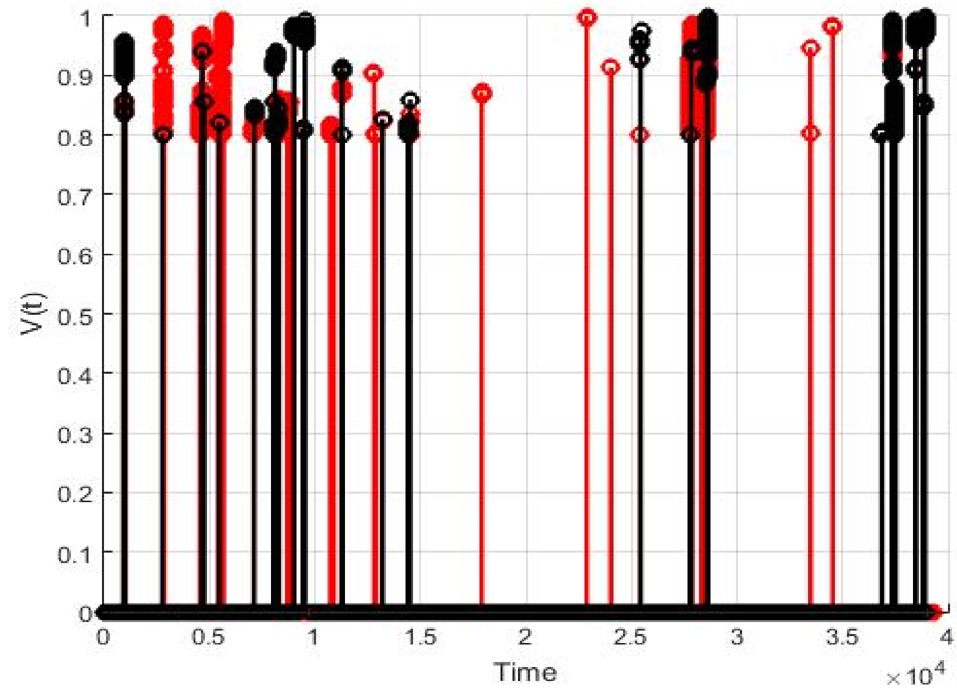

Figure 133: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

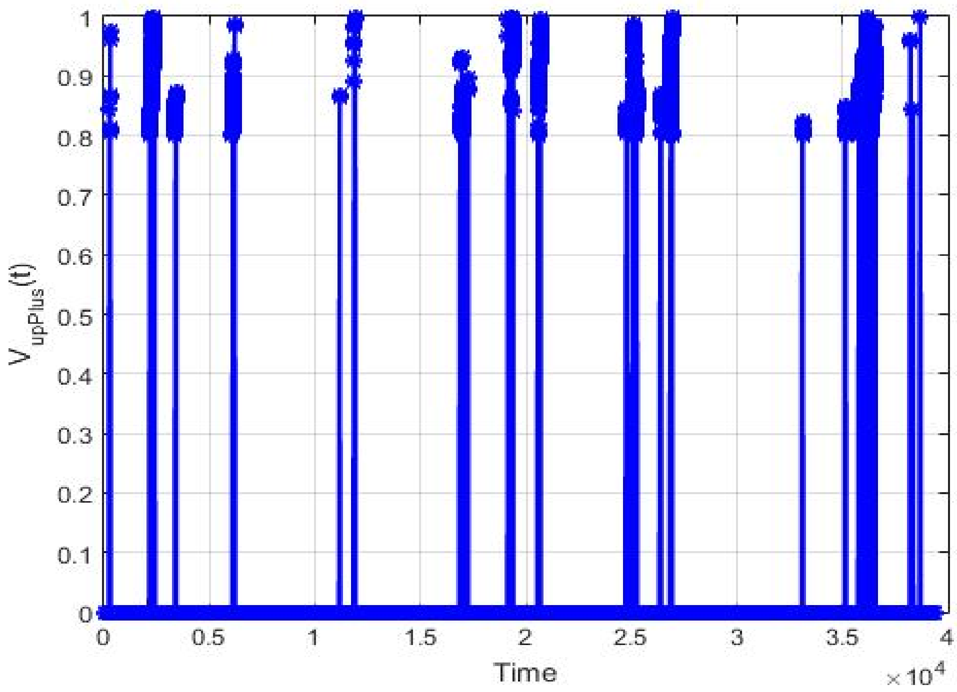

Figure 134: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

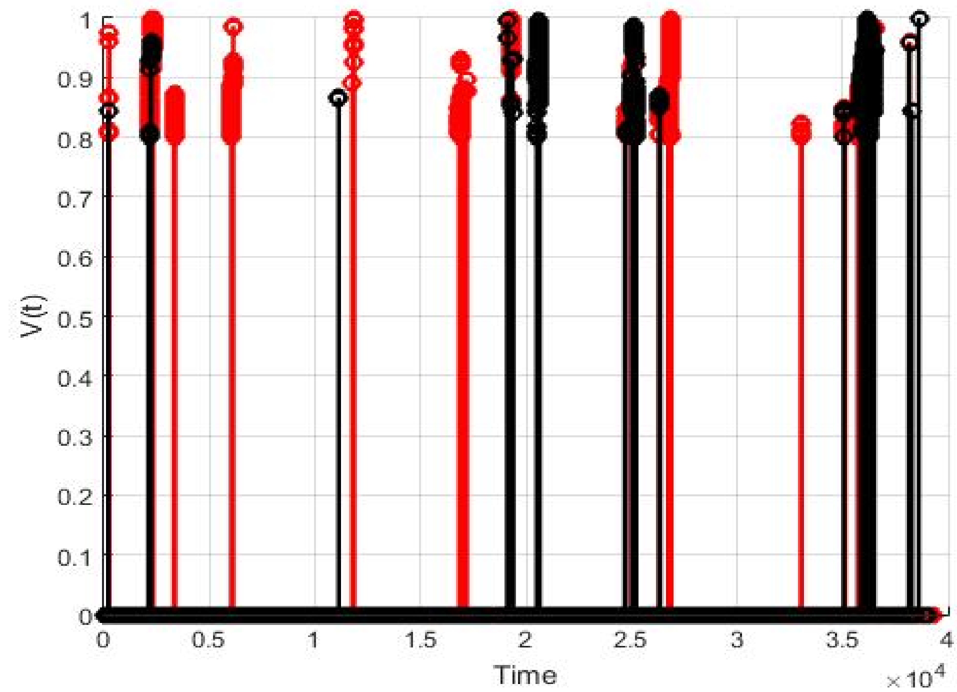

Figure 135: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 50, $V_{thres} = 80\%$

Partition 2(Future window = 600, Past window 50, $V_{thres} = 80\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18603/39350	47.3%
Percantage of upward spread-crossing events	22012/39350	55.9%
Percantage of downward spread-crossing events($V_{thres} = 80\%$)	1133/39350	2.9%
Percantage of upward spread-crossing events($V_{thres} = 80\%$)	1540/39350	3.9%

Table 31: Partition 2(Future window = 600, Past window 50, $V_{thres} = 80\%$)

Partition 2(Future window = 600, Past window 50, $V_{thres} = 80\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	11546/19675	58.7%
Percantage of downward spread-crossing events(testing)	12510/19675	63.6%
Percantage of upward spread-crossing events(training)	10250/19675	52.1%
Percantage of upward spread-crossing events(testing)	10076/19675	51.2%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,training)	7416/19675	37.7%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,testing)	11671/19675	59.3%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,training)	10703/19675	54.4%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,testing)	9815/19675	49.9%

Table 32: Partition 2(Future window = 600, Past window 50, $V_{thres} = 80\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 1315 & 7826 \\ 303 & 10231 \end{matrix}$	Ctest Down = $\begin{matrix} 1627 & 9979 \\ 2280 & 5789 \end{matrix}$
Ctrain Up = $\begin{matrix} 5335 & 3474 \\ 3691 & 7175 \end{matrix}$	Ctest Up = $\begin{matrix} 3920 & 4609 \\ 3395 & 7751 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 9746 & 9267 \\ 158 & 504 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 10453 & 8751 \\ 221 & 250 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 9579 & 9557 \\ 42 & 497 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 9509 & 9165 \\ 695 & 306 \end{matrix}$

5.2.8 Scenario 2.8 (Future window = 600, Past Window = 100, $V_{thres} = 80\%$)

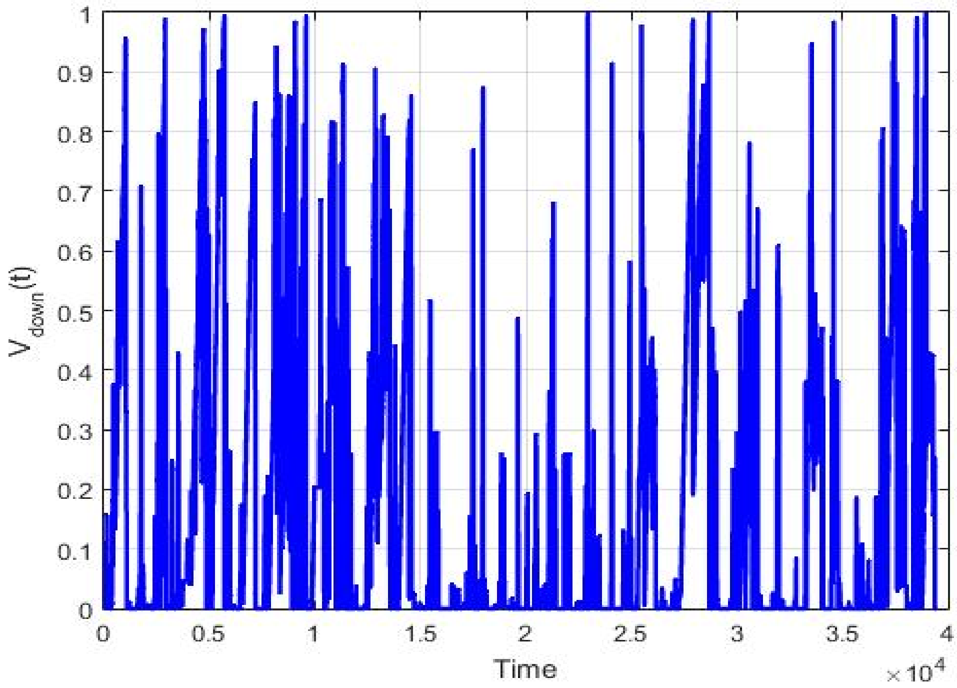

Figure 136: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

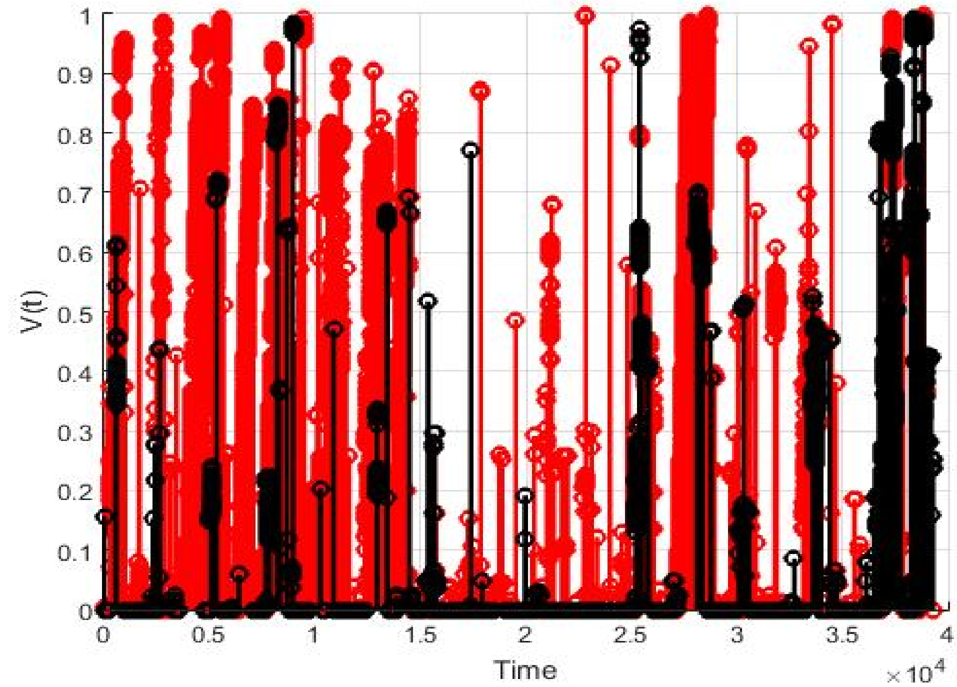

Figure 137: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

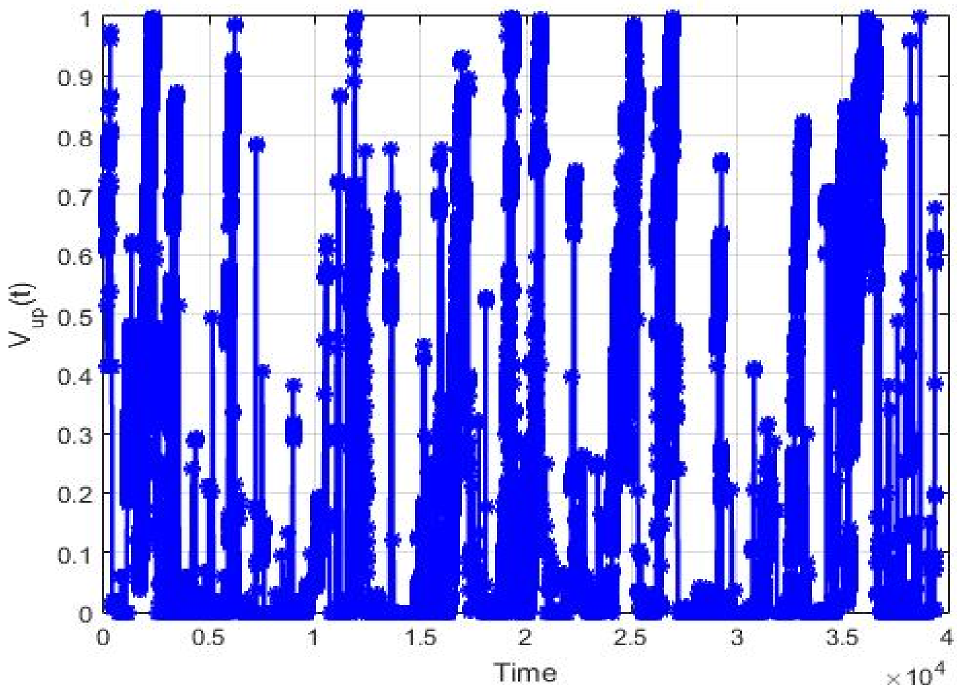

Figure 138: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

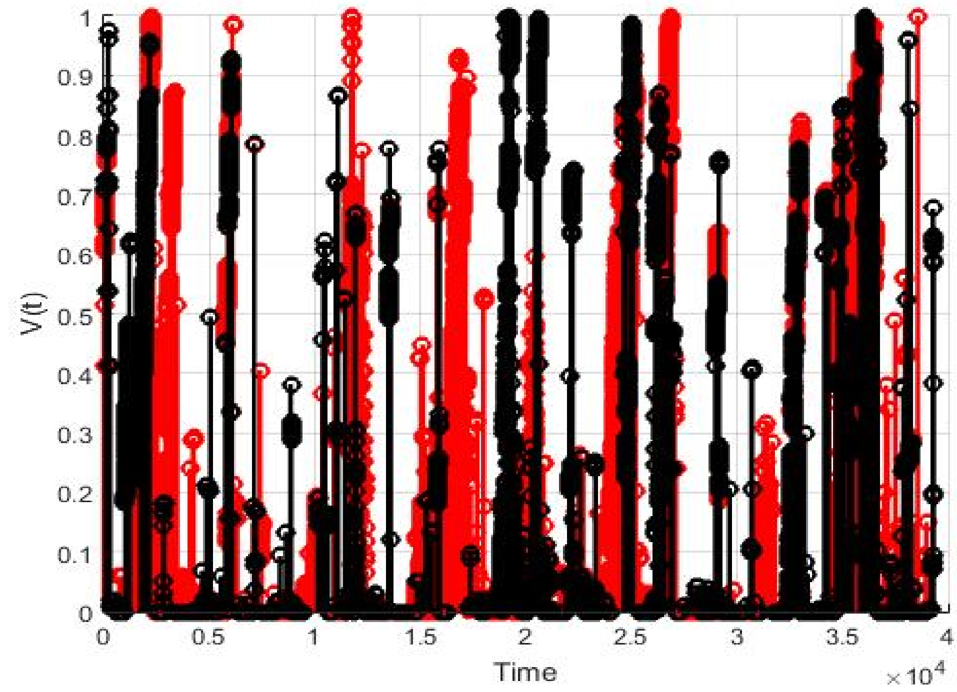

Figure 139: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

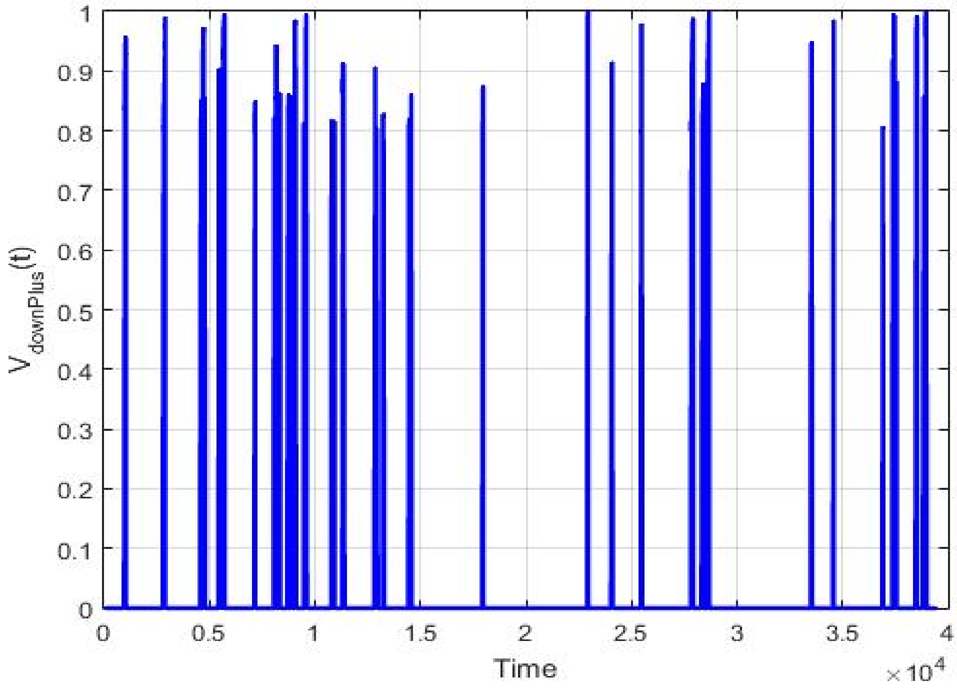

Figure 140: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

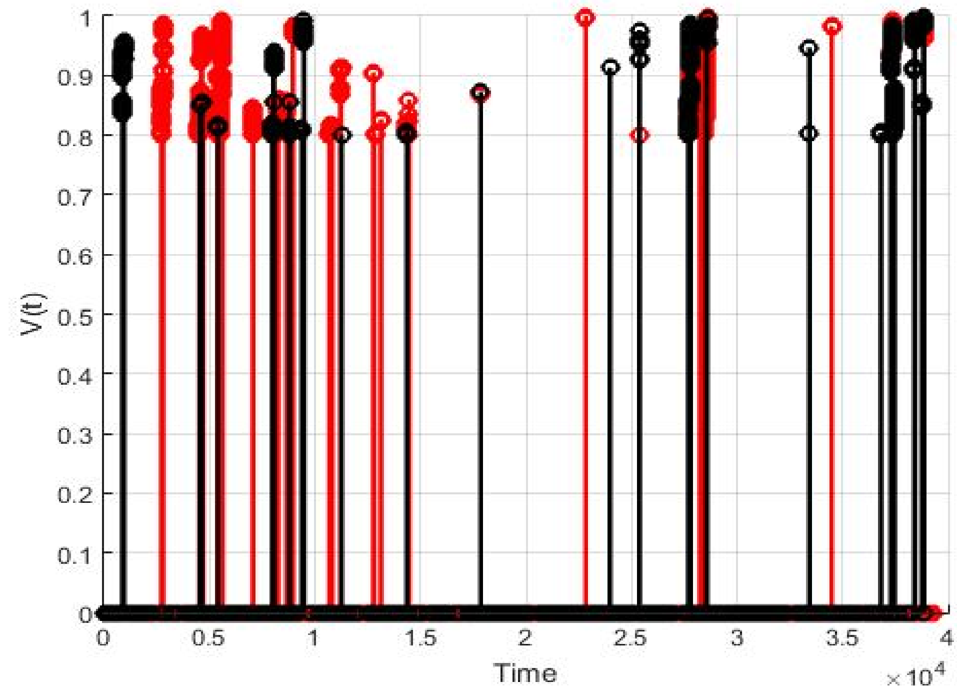

Figure 141: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

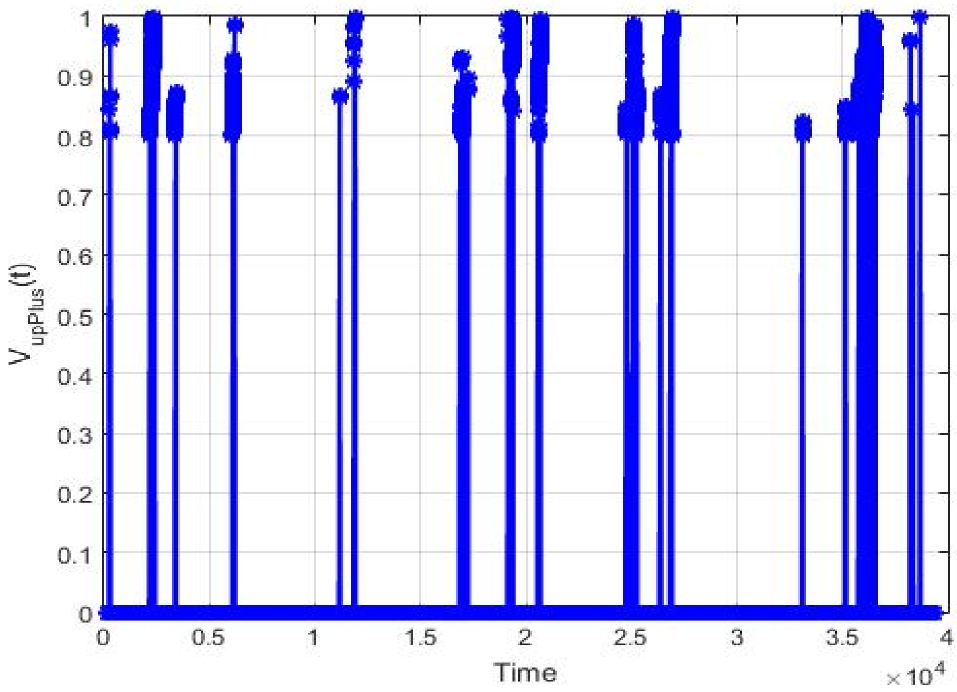

Figure 142: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

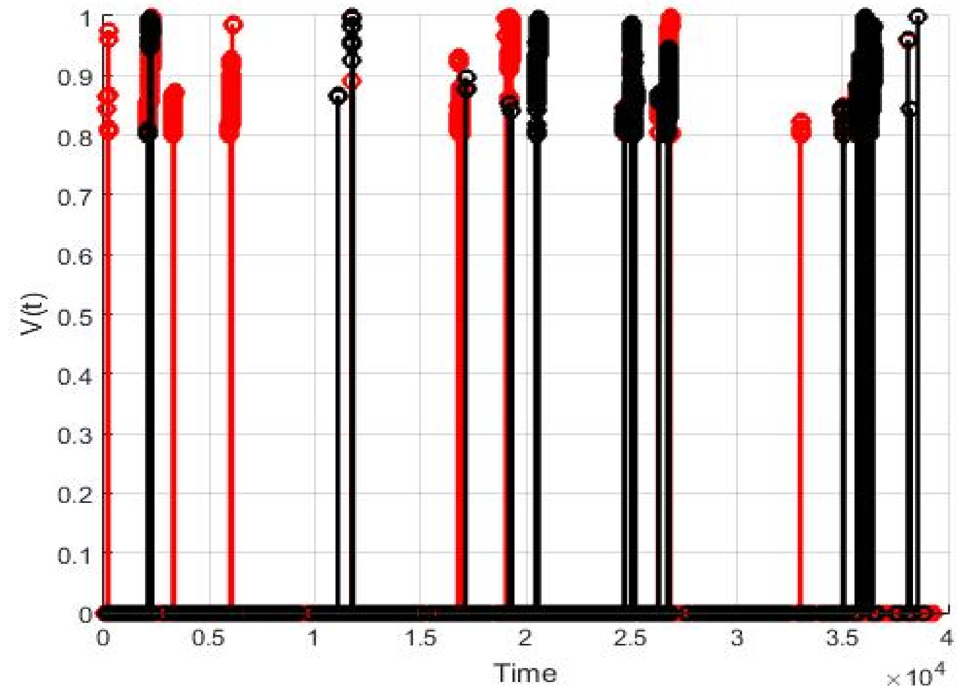

Figure 143: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 100, $V_{thres} = 80\%$

Partition 2(Future window = 600, Past window 100, $V_{thres} = 80\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18594/39300	47.3%
Percantage of upward spread-crossing events	21962/393000	55.9%
Percantage of downward spread-crossing events($V_{thres} = 80\%$)	1133/39300	2.9%
Percantage of upward spread-crossing events($V_{thres} = 80\%$)	1540/39300	3.9%

Table 33: Partition 2(Future window = 600, Past window 100, $V_{thres} = 80\%$)

Partition 2(Future window = 600, Past window 100, $V_{thres} = 80\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	12068/19650	61.4%
Percantage of downward spread-crossing events(testing)	13047/19650	66.4%
Percantage of upward spread-crossing events(training)	12640/19650	64.3%
Percantage of upward spread-crossing events(testing)	14858/19650	75.6%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,training)	7495/19650	38.1%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,testing)	11621/19650	59.1%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,training)	13037/19650	66.3%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,testing)	12688/19650	64.6%

Table 34: Partition 2(Future window = 600, Past window 100, $V_{thres} = 80\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 2401 & 6724 \\ 858 & 9667 \end{matrix}$	Ctest Down = $\begin{matrix} 2488 & 9093 \\ 3062 & 5007 \end{matrix}$
Ctrain Up = $\begin{matrix} 6044 & 2765 \\ 3838 & 7003 \end{matrix}$	Ctest Up = $\begin{matrix} 4345 & 4184 \\ 3845 & 7276 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 12106 & 6882 \\ 128 & 534 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 12822 & 6357 \\ 256 & 215 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 14384 & 4727 \\ 65 & 474 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 12544 & 6105 \\ 857 & 144 \end{matrix}$

5.2.9 Scenario 2.9 (Future window = 600, Past Window = 200, $V_{thres} = 80\%$)

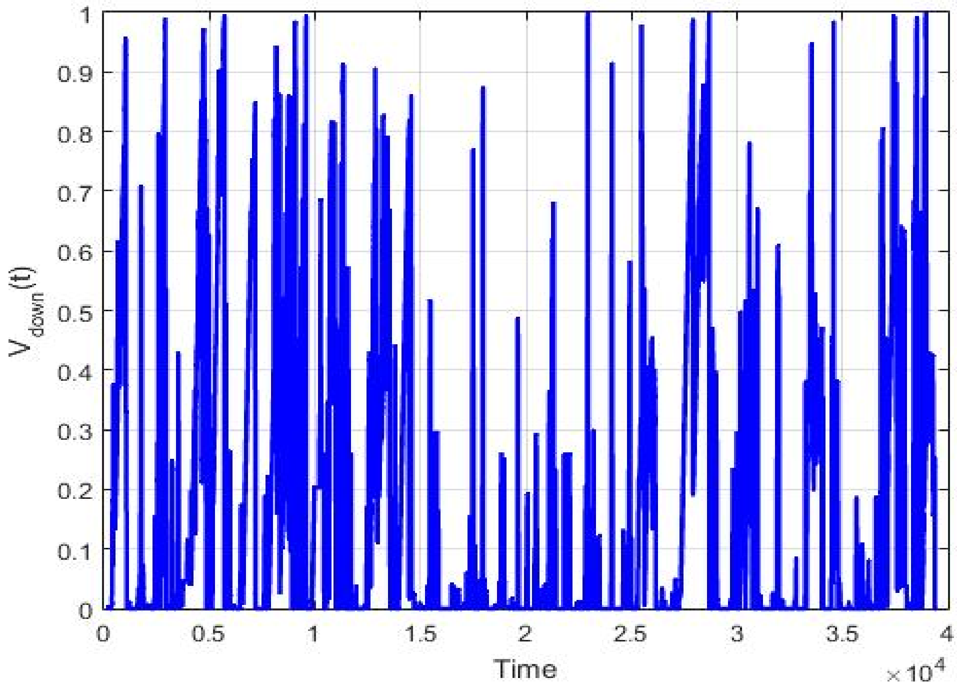

Figure 144: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

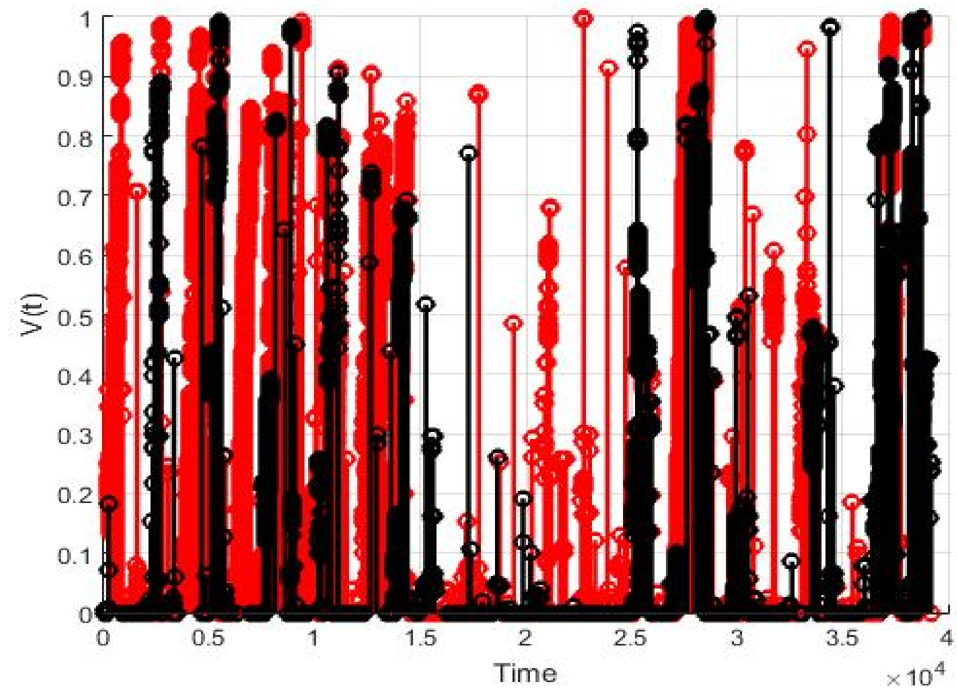

Figure 145: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

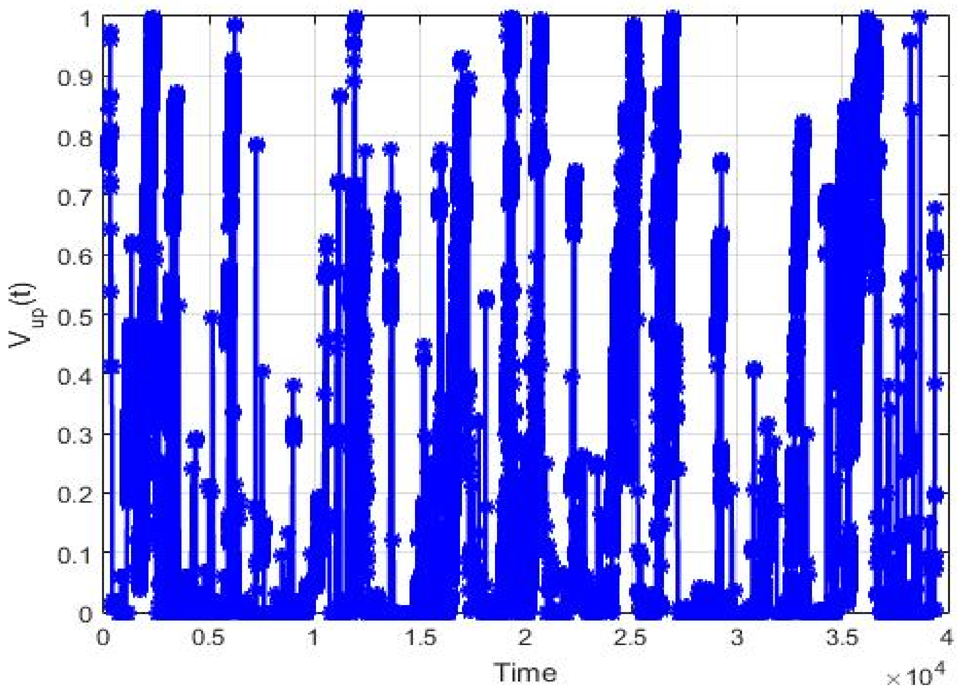

Figure 146: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

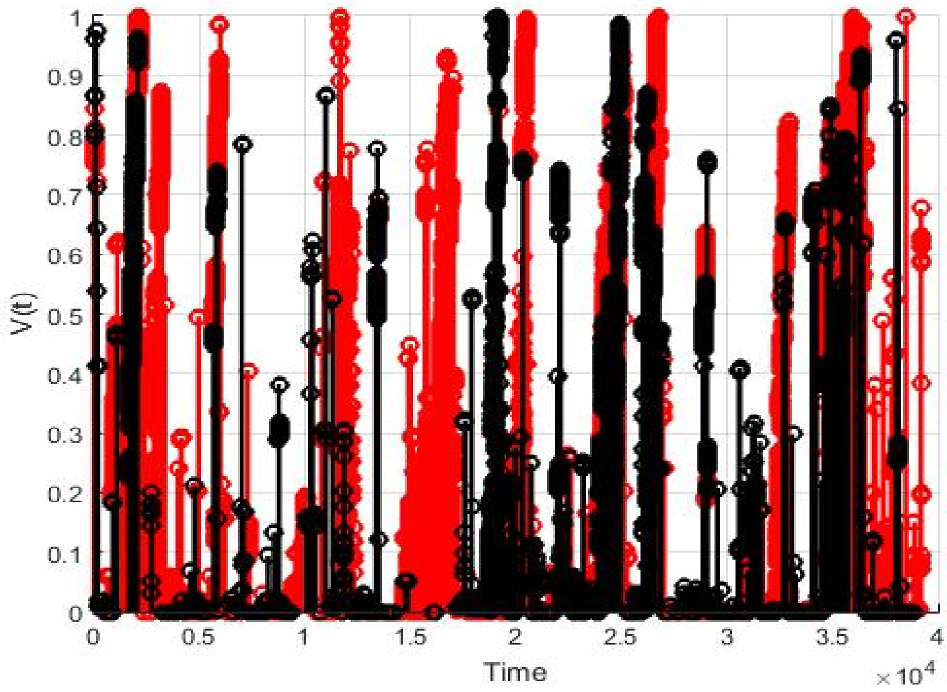

Figure 147: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

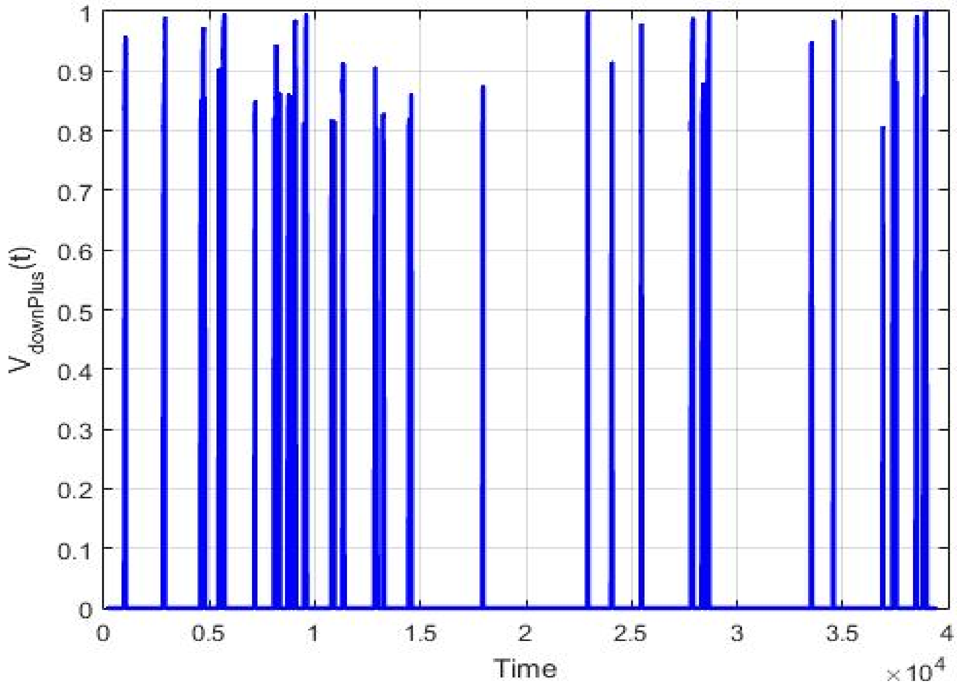

Figure 148: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

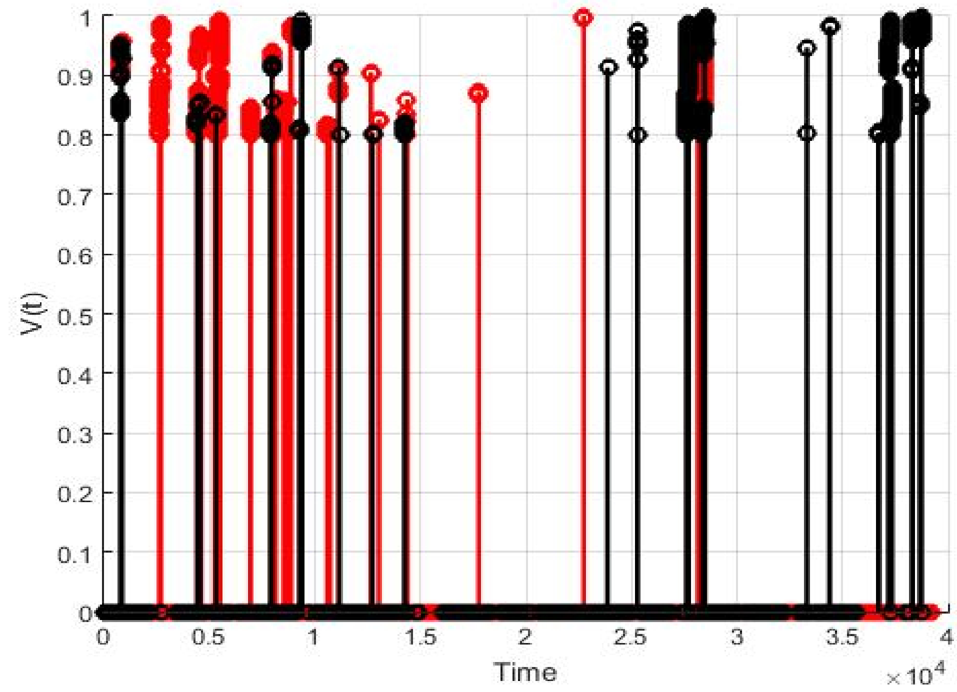

Figure 149: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

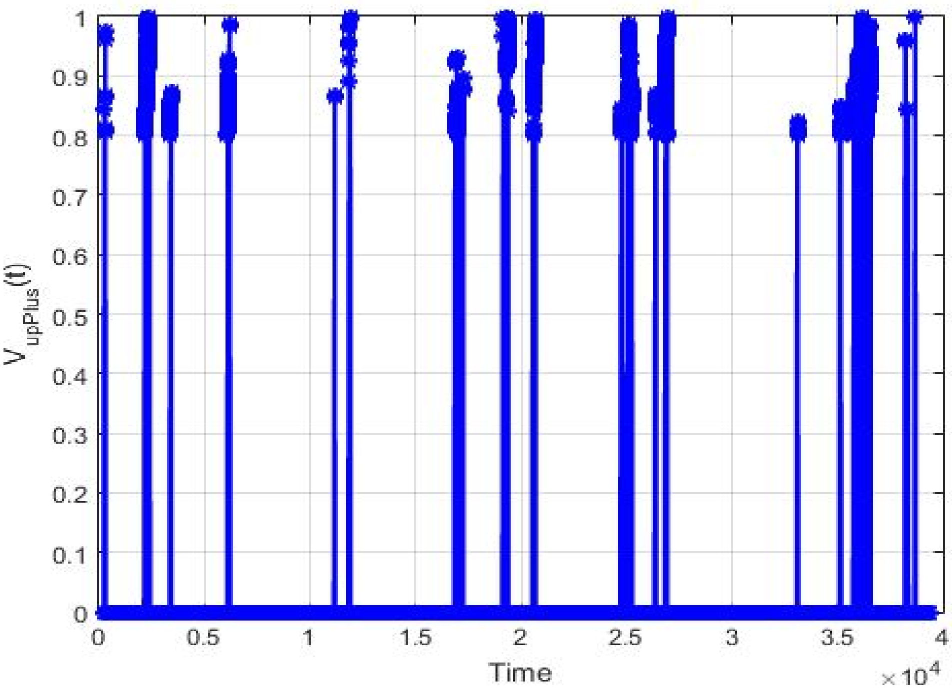

Figure 150: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

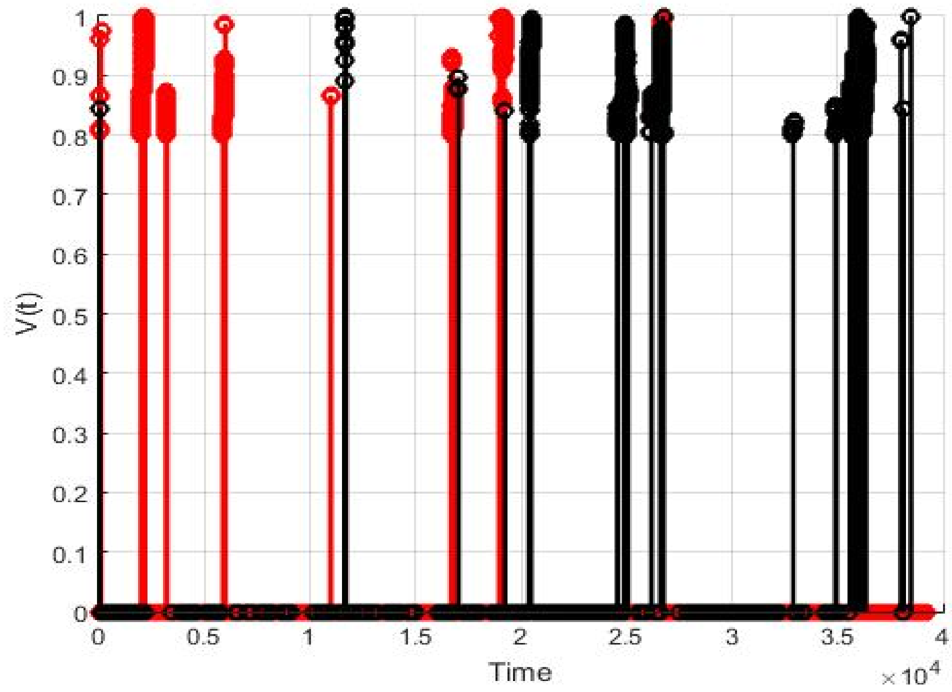

Figure 151: **Plot volume of Downward spread-crossing events within the future time window as a function of time within the adjusted time interval.**
Partition = 2, Future Window = 600, Past Window = 200, $V_{thres} = 80\%$

Partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events	18574/39200	47.4%
Percantage of upward spread-crossing events	21862/39200	55.8%
Percantage of downward spread-crossing events($V_{thres} = 80\%$)	1133/39200	2.9%
Percantage of upward spread-crossing events($V_{thres} = 80\%$)	1540/39200	3.9%

Table 35: Partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$)

partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$, Training / Testing)		
Description	**Timestamps with events**	**Accuracy**
Percantage of downward spread-crossing events(training)	13554/19600	69.2%
Percantage of downward spread-crossing events(testing)	14547/19600	74.2%
Percantage of upward spread-crossing events(training)	16276/19600	83.0%
Percantage of upward spread-crossing events(testing)	16533/19600	84.4%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,training)	7500/19600	38.2%
Percantage of downward spread-crossing events($V_{thres} = 80\%$,testing)	11701/19600	59.7%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,training)	16035/19600	81.8%
Percantage of upward spread-crossing events($V_{thres} = 80\%$,testing)	14763/19600	75.3%

Table 36: partition 1(Future window = 600, Past window 200, $V_{thres} = 80\%$, Training / Testing)

Ctrain Down = $\begin{matrix} 4894 & 4201 \\ 1845 & 8660 \end{matrix}$	Ctest Down = $\begin{matrix} 3747 & 7784 \\ 4316 & 3753 \end{matrix}$
Ctrain Up = $\begin{matrix} 6820 & 2029 \\ 3024 & 7727 \end{matrix}$	Ctest Up = $\begin{matrix} 4604 & 3885 \\ 4014 & 7097 \end{matrix}$
Ctrain Down $V_{thres} = \begin{matrix} 15716 & 3222 \\ 102 & 560 \end{matrix}$	Ctest Down $V_{thres} = \begin{matrix} 15922 & 3207 \\ 358 & 113 \end{matrix}$
Ctrain Up $V_{thres} = \begin{matrix} 16013 & 3048 \\ 19 & 520 \end{matrix}$	Ctest Up $V_{thres} = \begin{matrix} 14721 & 3878 \\ 959 & 42 \end{matrix}$

6 Conclusion

Starting with partition 1 and specifically the scenario 1.1, where the V_{thres} is set to 10%, the figures 8 - 15 present the results of Downward - Upward events and the classification of each case. There is also the V_{thres} parameter that allows only volumes with price over the 10%.

The figures are the result of the program and because of the number of values is extremely high, there are three tables that personalize the data more representive. In table 1, the Downward events in the 100% of their data have accuracy $61,8$%. The Downward events with $V_{thres} = 10$%, so the 90% of the highest volume data have accuracy $45,1$%. That is a loss of $16,7$% in only 10% less data. Also, the $45,1$% are more useful and exploitable, because the highest volumes are the data closer closer to happen in the future window.

This is one of the best attributes of V_{thres}, as is filtering the events and permits only the more useful data to be counted and shown. Even bigger gap, up to 20% drop, exists in the Upward events. It is worth mentioning that the percentages between the Downward - Upward events is very small, approximately $\pm 0,7$% and $\pm 2,5$% to V_{thres}.

After the separation to half for the classification to Training - Testing points, the accuracy of the timestamps with events, remains over 50%, which is accually very good and useful data. Someone can invest in this range with high possibility of profit.

In the last matrix of the same section, there are four values. The positive values are the upper which came true. For example, in the Ctrain Down case, the upper(TP) left value (4.881 / 19.675) with percentage $29,8$%, are the sum of the Downward events that predicted correctly. Also, the upper(TN) right value (2.432 / 19.675) with percentage $12,4$% are the sum of the Non - Downward events that predicted correctly.

The last one can be used to avoid possible loss of profit in the case of short position as it is mentioned in chapter 2.1.3. The lower cases can also be useful for each case. Their sum percentage present the cases that did not predicted correctly.

In the same table, the Ctrain Down V_{thres}(TP) has value (6.633 / 19.675) with percentage $33,7$%. This TP value has more representive data, so is more possible to happen than the last case.

In chapter 5.1.9, the past window and the V_{thres} have increased to 200 and 80%, respectively. The TP of the Ctrain Down is (6.478 / 19.600) with percentage $33,1$%. This $8,3$% increment occurs because of the past window set to 200. This accually provides much better results, but the time needed is also larger. The TP in the Ctrain Down V_{thres} case is (17.313/19.600) with percentage $88,3$% ,which obviously is the best case of all. In the same table, the positive volumes are (2.853/39.200) and the zero volumes are (36.347/39.200). In the $88,3$%, they are included the zero volumes which are predicted correctly.

It is also worth mentioning, the difference in results in the two partitions. The second partition has less timestaps with volumes over 80% and more timestaps with volumes over 10%. So, the data in partition 2 useful for a conservative buyer or seller, who does not want to risk, because medium volume values are more stable. The partition 1 is for buyers or sellers who prefer high risk and more profits. In the second case the chances of win are less possible.

References

[1] F. Abergel, M. Anane, A. Chakraborti, A. Jedidi, and I. Toke, *Limit Order Books*, 1st ed., ser. Physics of Society: Econophysics and Sociophysics. Cambridge University Press, 2016.

[2] A. Ranaldo, "Order aggressiveness in limit order book markets," *Journal of Financial Markets*, vol. 7, no. 1, pp. 53–74, 2004.

[3] T. Foucault, O. Kadan, and E. Kandel, "Limit order book as a market for liquidity," *The review of financial studies*, vol. 18, no. 4, pp. 1171–1217, 2005.

[4] I. Roşu, "A dynamic model of the limit order book," *The Review of Financial Studies*, vol. 22, no. 11, pp. 4601–4641, 2009.

[5] Akansu, A. N, Torun, and M. U, *A primer for financial engineering: financial signal processing and electronic trading*. Academic Press, 2015.

[6] N. Gandevani, *Winning Edge Trading: Successful and Profitable Short and Long-Term Systems and Strategies (Wiley Trading)*, ser. Wiley Trading. Wiley, 2009.

[7] L.Williams., *Long-Term Secrets to Short-Term Trading*.

[8] A. N. Kercheval and Y. Zhang, "Modelling high-frequency limit order book dynamics with support vector machines," *Quantitative Finance*, vol. 15, no. 8, pp. 12–20, 2015.

[9] S. R. Gunn *et al.*, "Support vector machines for classification and regression," *ISIS technical report*, vol. 14, no. 1, pp. 5–16, 1998.

[10] K.-M. Chung, W.-C. Kao, C.-L. Sun, L.-L. Wang, and C.-J. Lin, "Radius margin bounds for support vector machines with the rbf kernel," *Neural computation*, vol. 15, no. 11, pp. 2643–2681, 2003.

[11] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Advances in neural information processing systems*, 2003, pp. 577–584.

[12] D. Sotiropoulos and G. A. Tsihrintzis, *Machine Learning Paradigms Artificial Immune Systems and their Applications in Software Personalization*. Springer, 2016.

[13] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[14] A. Lampropoulos and G. Tsihrintzis, *Machine Learning Paradigms*. Springer, 2015.

[15] K. Oku, S. Nakajima, J. Miyazaki, and S. Uemura, "Context-aware svm for context-dependent information recommendation," in *Proceedings of the 7th international Conference on Mobile Data Management*. IEEE Computer Society, 2006, p. 109.

[16] D.-R. Chen, Q. Wu, Y. Ying, and D.-X. Zhou, "Support vector machine soft margin classifiers: error analysis," *Journal of Machine Learning Research*, vol. 5, no. Sep, pp. 1143–1175, 2004.

[17] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[18] L. Wang, *Support vector machines: theory and applications*. Springer Science & Business Media, 2005, vol. 177.

[19] Q. Wu and D.-X. Zhou, "Svm soft margin classifiers: linear programming versus quadratic programming," *Neural computation*, vol. 17, no. 5, pp. 1160–1187, 2005.

[20] C.-L. Huang, M.-C. Chen, and C.-J. Wang, "Credit scoring with a data mining approach based on support vector machines," *Expert systems with applications*, vol. 33, no. 4, pp. 847–856, 2007.

[21] S. Theodoridis, K. Koutroumbas *et al.*, "Pattern recognition," *IEEE Transactions on Neural Networks*, vol. 19, no. 2, p. 376, 2008.

[22] P. J. Moreno, P. P. Ho, and N. Vasconcelos, “A kullback-leibler divergence based kernel for svm classification in multimedia applications,” in *Advances in neural information processing systems*, 2004, pp. 1385–1392.

[23] Y. Bazi and F. Melgani, “Toward an optimal svm classification system for hyperspectral remote sensing images,” *IEEE Transactions on geoscience and remote sensing*, vol. 44, no. 11, pp. 3374–3385, 2006.

[24] D. Yermack, “Is bitcoin a real currency? an economic appraisal,” in *Handbook of digital currency*. Elsevier, 2015, pp. 31–43.

[25] A. H. Dyhrberg, “Bitcoin, gold and the dollar–a garch volatility analysis,” *Finance Research Letters*, vol. 16, pp. 85–92, 2016.

[26] A. Greaves and B. Au, “Using the bitcoin transaction graph to predict the price of bitcoin,” *No Data*, 2015.

[27] S. McNally, J. Roche, and S. Caton, “Predicting the price of bitcoin using machine learning,” in *Parallel, Distributed and Network-based Processing (PDP), 2018 26th Euromicro International Conference on*. IEEE, 2018, pp. 339–343.

[28] I. Georgoula, D. Pournarakis, C. Bilanakos, D. Sotiropoulos, and G. M. Giaglis, “Using time-series and sentiment analysis to detect the determinants of bitcoin prices,” 2015.

[29] A. Greaves and B. Au, *Using the Bitcoin Transaction Graph to Predict the Price of Bitcoin*, 2015.

[30] D. Ron and A. Shamir, “Quantitative analysis of the full bitcoin transaction graph,” in *International Conference on Financial Cryptography and Data Security*. Springer, 2013, pp. 6–24.

[31] J. Cox, *Bitcoin and Digital Currencies: The New World of Money and Freedom*, ser. LFB. Laissez Faire Books, 2013.

[32] M. S. Daniel Forrester, *Bitcoin Explained: Today’s Complete Guide to Tomorrow’s Currency*. CreateSpace Independent Publishing Platform, 2013.

[33] I. Eyal and E. G. Sirer, “Majority is not enough: Bitcoin mining is vulnerable,” *Communications of the ACM*, vol. 61, no. 7, pp. 95–102, 2018.

[34] S. Barber, X. Boyen, E. Shi, and E. Uzun, “Bitter to better—how to make bitcoin a better currency,” in *International Conference on Financial Cryptography and Data Security*. Springer, 2012, pp. 399–414.

[35] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008.

[36] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[37] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “Liblinear: A library for large linear classification,” *Journal of machine learning research*, vol. 9, no. Aug, pp. 1871–1874, 2008.

[38] C.-W. Hsu, C.-C. Chang, C.-J. Lin *et al.*, “A practical guide to support vector classification,” 2003.

[39] O. L. Mangasarian and E. W. Wild, “Multisurface proximal support vector machine classification via generalized eigenvalues,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 28, no. 1, pp. 69–74, 2006.

[40] K. Pelckmans, J. A. Suykens, T. Van Gestel, J. De Brabanter, L. Lukas, B. Hamers, B. De Moor, and J. Vandewalle, “Ls-svmlab: a matlab/c toolbox for least squares support vector machines,” *Tutorial. KULeuven-ESAT. Leuven, Belgium*, vol. 142, pp. 1–2, 2002.