กระบวนการเกาส์เซียนสำหรับการทำนายจุดเปลี่ยนและการประยุกต์ในกลยุทธ์การซื้อขายหุ้น

นายดนุพล ต้นกิ่ง

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาคณิตศาสตร์ประยุกต์และวิทยาการคณนา ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2558

GAUSSIAN PROCESS FOR TURNING POINTS PREDICTION AND APPLICATION IN STOCK TRADING STRATEGY

Mr. Danuphon Tonking

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Applied Mathematics and Computational Science
Department of Mathematics and Computer Science
Faculty of Science
Chulalongkorn University
Academic Year 2015

Thesis Title	GAUSSIAN PROCESS FOR TURNING POINTS PREDICTION AND APPLICATION IN STOCK TRADING STRATEGY
By	Mr. Danuphon Tonking
Field of Study	Applied Mathematics and Computational Science
Thesis Advisor	Assistant Professor Petarpa Boonserm, Ph.D.
Thesis Co-Advisor	Chaiyakorn Yingsaeree, Ph.D.

Accepted by the Faculty of Science, Chulalongkorn University in Partial Fulfillment of the Requirements for the Master's Degree

....Dean of the Faculty of Science

(Associate Professor Polkit Sangvanich, Ph.D.)

THESIS COMMITTEE

....Chairman

(Assistant Professor Khamron Mekchay, Ph.D.)

....Thesis Advisor

(Assistant Professor Petarpa Boonserm, Ph.D.)

....Thesis Co-Advisor

(Chaiyakorn Yingsaeree, Ph.D.)

....Examiner

(Boonyarit Intiyot, Ph.D.)

....External Examiner

(Suttipong Thajchayapong, Ph.D.)

ดนุพล ต้นกิ่ง : กระบวนการเกาส์เซียนสำหรับการทำนายจุดเปลี่ยนและการประยุกต์ในกลยุทธ์การซื้อขายหุ้น (GAUSSIAN PROCESS FOR TURNING POINTS PREDICTION AND APPLICATION IN STOCK TRADING STRATEGY) อ.ที่ปรึกษาวิทยานิพนธ์หลัก: ผศ. ดร. เพชรอาภา บุญเสริม, อ.ที่ปรึกษาวิทยานิพนธ์ร่วม: ดร.ชัยกร ยิ่งเสรี, 60 หน้า.

การทำนายจุดเปลี่ยน หรือการทำนายจุดสูงสุด/จุดต่ำสุด ของข้อมูลอนุกรมเวลาของราคาของหุ้นในประเทศไทย โดยผู้ค้าสามารถที่จะซื้อหุ้นในราคาต่ำสุด และขายในราคาที่สูงสุดได้อย่างแม่นยำที่จุดกลับตัวของแนวโน้มราคาหุ้น เพื่อที่จะได้รับผลกำไรสูงสุดในช่วงเวลานั้น ในงานวิจัยนี้ได้นำเสนอวิธีการของเกาส์เซียนเพื่อสร้างแบบจำลองทางคณิตศาสตร์ เพื่อทำนายจุดเปลี่ยน และเปรียบเทียบเทคนิคกับแบบจำลองที่ใช้กระบวนการซัพพอร์ตเวกเตอร์แมชชีน และโครงข่ายประสาทเทียม พร้อมทั้งนำแบบจำลองทางคณิตศาสตร์ที่สร้างขึ้นไปประยุกต์ใช้ในการออกแบบกลยุทธ์การซื้อขายหลักทรัพย์ที่สร้างผลกำไรได้จริง

ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	ลายมือชื่อนิสิต	....
		ลายมือชื่อ อ.ที่ปรึกษาหลัก	....
สาขาวิชา	คณิตศาสตร์ประยุกต์และวิทยาการคณนา	ลายมือชื่อ อ.ที่ปรึกษาร่วม	....
ปีการศึกษา	2558		

5671960023 : MAJOR APPLIED MATHEMATICS AND COMPUTATIONAL SCIENCE
KEYWORDS: TURNING POINT / GAUSSIAN PROCESS / NON-LINEAR REGRESSION

DANUPHON TONKING: GAUSSIAN PROCESS FOR TURNING POINTS PREDICTION AND APPLICATION IN STOCK TRADING STRATEGY. ADVISOR: ASST. PROF. PETARPA BOONSERM, Ph.D., CO-ADVISOR: CHAIYAKORN YINGSAEREE, Ph.D., 60 pp.

In the financial price sequence, the turning points prediction using the time series data form stock prices in Thai stock exchange. Turning points are critical local extreme points along a series. A trader who is able to buy stocks at trough prices and sell at peak prices to enter/exit the market precisely at the turning points would gain the maximum possible profit. In addition to using the Gaussian process model for predicting the turning points, in this project, we also test and compare the efficient techniques of modeling between the Gaussian Process Model, Neural Network and Support Vector Regression, respectively. Finally, we utilize the developed turning point prediction model to create trading strategy for the derivation of maximum profit.

Department:	Mathematics and Computer Science	Student's Signature	....
		Advisor's Signature	....
Field of Study:	Applied Mathematics and Computational Science	Co-Advisor's Signature	....
Academic Year:	2015		

ACKNOWLEDGEMENTS

I would like to thank the department of Mathematics and Computer Science of Chulalongkorn University for giving me the opportunity to pursue and finally finish my master’s degree. I would like to express my gratitude to Assistant Professor Dr. Petarpa Boonserm, my thesis advisor, and Dr. Chaiyakorn Yingsaeree, my thesis co-advisor, for their invaluable help and constant encouragement throughout the course of this research.

In addition, I am sincerely grateful to all my dissertation committee members, Assistant Professor Dr. Khamron Mekchay, Dr. Boonyarit Intiyot and Dr. Suttipong Thajchayapong, for their contribution and advice in the completion of this dissertation.

I would like to thank my friends and colleagues in the AMCS program, especially Panote Songwattanasiri for his suggestions, encouragement and good advice. Finally, I am extremely grateful to my family for giving me the opportunity to follow my dreams and for supporting me throughout the period of this research.

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER I
INTRODUCTION

Our research study focuses on predicting the start of a new swing of a financial price sequence. This swing is referred to as the turning point of a financial price sequence, with a local minimum called the *trough,* and a local maximum called the *peak.* Turning points are critical local extreme points along a series, which act as indicators to reflect the trends in the financial markets. A trader who is able to buy stocks at trough prices and sell at peak prices in order to enter/exit the market precisely at the turning points would gain the maximum possible profit. The ability to anticipate the trends of the stock market and the turning points can help in guiding financial decisions, and subsequently, in establishing stock market trading strategies. Turning points have been an interest since 1946 when Arthur and Wesley [1] described them as intrinsic parts of business cycles in their publication entitled 'Measuring Business Cycles'. Following Arthur and Wesley, Bry and Boshan [2] proposed a process for the detection of turning points in time series, in hindsight. Overtime, research about turning point prediction has gradually shifted toward the use of another prediction model. This prediction model has been shown to be compatible with the other models such as the standard linear least square method and the standard Bayesian approach [3, 4]. Currently, there are two research interests in the field of computational finance, namely, Artificial Neural Network (ANN) and Support Vector Regression (SVR). In 2009, Li et al [5] proposed a model for short term prediction using the Artificial Neural Network model, which has been presented as an ensemble of learning based turning point prediction frameworks. This Ensemble Artificial Neural Network (EANN) model functions by approximating nonlinear maps. The regression problem uses feed-forward with a back propagation algorithm as part of the EANN. Moreover, there is a genetic algorithm (GA) based threshold optimization, together with a newly defined performance measure, named *TpMSE*, which has been used as a cost function in our research. In addition, in 2012, Ran and Alexandra [6] also presented an autoregressive short-term method for the prediction of turning points using Support Vector Regression (SVR), and confirmed the viability of their method using a long history of the Dow Jones Industrial average. The autoregressive model has been shown to be useful in predicting turning points of small swings. This demonstrates that the current model is significantly more effective than the previous neural network model, in respect to directing trade performance.

In this research, we propose a new model for predicting the turning points in financial sequence, and also to examine the applicability of this new method, which is based on the Gaussian Process and the turning point prediction framework proposed by Li et al [5], and El-Yaniv and Faynburd [6]. The proposed model is utilized in the study to predict the turning points within a time series of stock prices obtained from the Stock Exchange of Thailand (SET). In addition, we also test and compare the efficiency in the techniques of modeling among the Gaussian Process Model, the Neural Network and the Support Vector Regression methods [5, 6], respectively. Besides, the methodology also entails the implementation of codes to generate the model using the *RStudio* program. This can be used to import the time series data of stock prices from the Stock Exchange of Thailand. An algorithm to predict the turning points will then be generated. Finally, we apply the Gaussian Process to the real world trading strategy for the derivation of maximum profit.

The data provided in the model is a segment of the financial price sequence of stock prices from a time series master data, from the Stock Exchange of Thailand. The data has been generated using the *RStudio* program. We have segregated the financial price information as either training data or test data. The training data set is used in fitting the price data with the GP model, and the testing data set is used in predicting the turning points as well as in evaluating the performance of the models.

1.1 Objectives

We aim to predict the turning points in the end-of-day time series of stock prices within the Stock Exchange of Thailand through a modification of the Gaussian Process model, and to compare the efficiency of the model with the Neural Network and the Support Vector Regression methods, when applied to stock trading.

1.2 Thesis Overview

This thesis is divided into 5 chapters. In chapter 2, we present the background knowledge and relevant researches related to the concept of turning point prediction, which is required for this thesis. In chapter 3, the work process involved in the construction of the models will be explained. We present the algorithm for the model and the experiments performed in the study to assess the validity of the Gaussian Process turning point prediction model. In chapter 4, we describe the dataset and the program needed to implement the model, along with its potential application as a simple trading strategy analyzer. The model is then evaluated for its performance in trading application, and to determine the error function TpRMSE. Finally, chapter 5 draws a conclusion and suggests possible future work.

CHAPTER II
BACKGROUND KNOWLEDGE

In this chapter, we present the background knowledge and the relevant concepts related to turning point prediction based on the Gaussian Process and the turning point prediction framework proposed by the Neural Network models by Li et. al. [5] and the Support Vector Regression model by El-Yaniv and Faynburd [6]. The proposed model is utilized in the study to predict the turning points within a time series of stock prices from the Stock Exchange of Thailand.

2.1 Literature review

The section presents the related research for turning point prediction within financial price sequences. There are several fields of study that also focus on the determination of turning points such as business studies, economics, financial engineering and computational sciences. In addition, the predicting of a turning points require the use of mathematical models such as the RBF neural network (RBF), the ensemble artificial neural network (ANN), and the support vector regression method (SVR). Moreover, the Gaussian Process Model has been used in financial predictions before. To illustrate the concept, researchers such as Dejan Petelin and Sofiane Brahim-Belhouari [7, 8] presented prediction techniques in terms of the Gaussian Process Model related researches are as follows;

El-Yaniv and Faynburd [6] have presented autoregressive short-term prediction of turning points using the support vector regression (SVR) method over a long history of the Dow Jones Industrial average. The autoregressive model is advantageous in predicting turning points of small swings, and when the efficiency of the model is compared with the support vector regression model and the artificial neural network model, the better performance of the model is statistically significant [5, 6]. Nevertheless, the SVR model has been able to yield a higher average return.

Li and Deng [9] presented a machine learning approach to predict turning points within a chaotic financial time series. This is a proposed turning point prediction method based on chaotic theory and machine learning that utilizes the radial basis function (RBF) neural network to build a nonlinear model. The nonlinear mapping between different data points is conducted for primitive time series. The result shows that the approach has good performance but not highly precise in its exact value prediction.

Brahim-Belhouari [8] focused on a research study problem of time series prediction. This problem relates to a Bayesian procedure based on the Gaussian process models and a nonstationary covariance function. The performance of the GP model is quantified using the root mean square error. For this reason, the GP model, using a nonstationary covariance function, show good tracking performance owing to the time series that is often nonstationary.

From literature, we have reviewed a range of researches about the prediction of turning points that can be applied to our research. The turning points and the models used in the study are introduced below.

2.2 Turning point

A turning point (TP) or a pivot in a financial price sequence is considered a time index, denoted by t. The turning point is a local extremum. This is called a *peak* if it is a local maximum and a *trough* if it is a local minimum, which marks the start of a new swing, as shown in figure (2.1). A turning point can be categorized according to its "size", which is reflected by the duration and magnitude of the trends before and after the reversal [2]. We can consider the usefulness of predicting a turning point since each time index in a financial price sequence may point out the beginning of a downturn. Turning point prediction has been of interest since 1946 when Burns and Michell [6] described a turning point in terms of business cycles in their Measuring Business Cycles model. The linear least squares prediction methods were first introduced in 1979 to predict the turning points of a time series. This research discusses the relationship between the theory of minimum mean square error linear prediction and the turning point prediction problem [5].

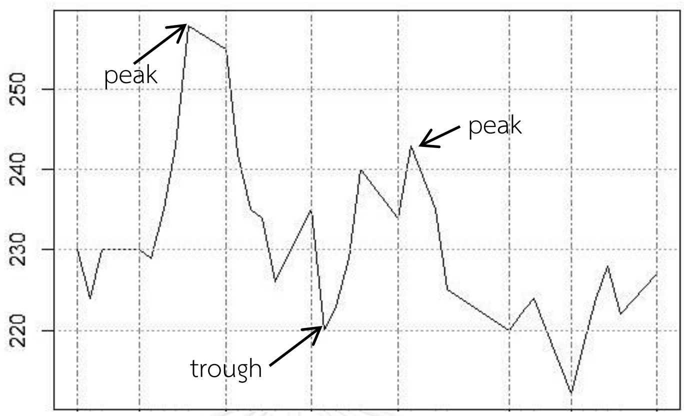


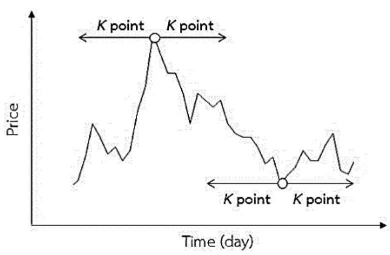
Figure 2.1: An examples for peak and trough.

We now consider three definitions including the pivot of degree *K*, the impact of a turning point and the momentum of turning points. Let $X = x_1, x_2, \ldots, x_t, \ldots$ be a real sequence.

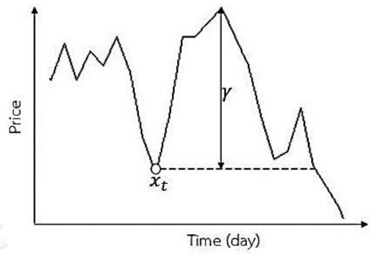
Definition 2.1 Pivot of degree K [5]. The time index *t* in a time series is an upper pivot or a *peak* of degree K, if $x_t \geq x_{t-i}$ and $x_t \geq x_{t+i}$ for $i = 1, 2, \ldots, K$. In the same way, t is a lower pivot or a *trough* of degree K, if $x_t \leq x_{t-i}$ and $x_t \leq x_{t+i}$ for $i = 1, 2, \ldots, K$.

Definition 2.2 Impact of a turning point [5]. The upward impact of a *trough* t is the ratio $\max\{x_t, \ldots, x_n\} / x_t$, where n is the first index greater than t, such that $x_n \leq x_t$. That is, if the sequence increases after the *trough* t to some maximal value $x_{\max}$ and then decreases below x_t; the impact is the ratio $x_{\max} / x_t$. If x_t is the global minimum of the sequence, then the numerator is taken as the global maximum appearing after time t. The downward impact of a peak is defined conversely.

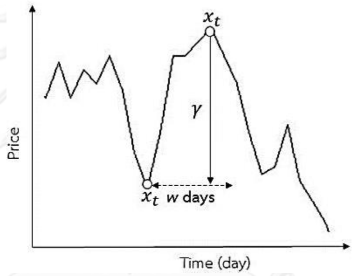
Definition 2.3 Momentum of a turning point [5]. The upward momentum of *a trough* t, with respect to a lookahead window of length w, is the percentage increase from x_t to the maximal value in the window $X_{t+1}^{t+w} = x_{t+1},...,x_{t+w}$. That is, the upward momentum is $\max\{x_{t+1},...,x_{t+w}\} / x_t$. The downward momentum of a *peak* is defined conversely.



(a) Pivot of degree K



(b) Turning point with impact γ



(c) Turning point with momentum γ

With respect to a lookahead window of length w

Figure 2.2: Example of the turning point scheme.

Three definitions of turning points have been described above. We find a local maximum *(peak)* and a local minimum (*trough*) for the construction of the alternating pivotal sequence, which is a characteristic feature of turning points in financial price sequences, through the consideration of the pivots in financial prices. Details have been described in section 2.2.1.

2.2.1 Alternating Sequence Pivots

In this section, we will detect characteristic features of a turning point that we call alternative pivot sequence. Alternating sequence pivots $A(X)$ function by detecting existing turning points to predict upcoming turning points. Alternating sequence pivots have a class of pivotal points considered as local maximum and local minimum, and alternate between peaks and troughs. Figure 2.2 shows points of alternating sequence with pivot of degree K=15. The sequence $A(X)$ is then used to construct a turning point oscillator.

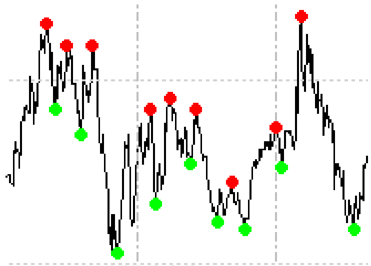


Figure 2.3: An example for alternative pivot sequence with pivot of degree K=15

By alternating the pivot sequence above, the construction of the turning point oscillator was achievable, which provides the target function for the prediction model. Details regarding the turning point oscillator has been described in the next section.

2.2.2 A Turning point oscillator

The oscillator will be normalizing the prices for the numerical interval values between 0 and 1. The reason for the construction of an *oscillator* is related to the existence of swings and extreme points within financial prices sequences. The main function of the oscillator is essentially to normalize the financial prices. The value of TP oscillator is bounded in the interval $[0,1]$.

Definition 2.4 A Turning point oscillator. Let X be a price sequence and let $A(X)$ be its alternating sequence pivots, which is a list of turning points with time in sequence X. The TP oscillator is a mapping $\Gamma : \mathbb{N} \to [0,1]$,

$$\Gamma(t) = \begin{cases} 0, & \text{if } t \text{ is a trough}, \\ 1, & \text{if } t \text{ is a } peak, \\ \dfrac{P(t) - x_t}{P(t) - T(t)} & \text{otherwise}, \end{cases}$$

where $P(t)$ and $T(t)$ are the values of the time series at the nearest peak and trough located on opposite sides of time t. The oscillator has to normalize the price in order to assign value 0 to all troughs and value 1 to all peaks.

To obtain the '*turning point oscillator*', we directed the turning point oscillator as the target function in our model. The *TP oscillator* has been shown to be useful in turning point prediction; meaning, if we are unable to use other means to predict the turning points directly, we can also use the turning point oscillator (TP oscillator) to make the predictions instead.

2.3 Neural Network models

In this section, we briefly describe the artificial neural network (ANN). The model for predicting turning points with neural network has been presented by Li et al [5]. An Artificial neural network (ANN) is a mathematics models that can be used as forecasting models. A neural network can be thought of as a network of "neurons" organized in layer. The input from the bottom layer and the output from the top layer are used as the predictor for forecasts. There might also be an intermediate layer containing hidden neurons (hidden layer). A simple network would have no hidden layers and would be equivalent to linear regression. The coefficient of input (predictor) is called weights. At the access point of artificial neurons, the inputs are weighted. This means that every input value is multiplied with its individual weight. The forecasts are then obtained by a linear combination of these inputs. The weights are selected in the neural network model using a learning algorithm. We can also use linear regression, which is a more efficient method for training model. The neural network adds an intermediate layer as a hidden layer shown in figure 2.3. With this hidden layer, the network becomes associated with non-linear regression for training model. This is known as a *multilayer feed-forward network* where each layer of nodes receives inputs from the previous layers. This can be used as a *backpropagation* algorithm. The backpropagation works by approximating the non-linear relationship between the input and the output by adjusting the weight values internally. It can further be generalized for the input that is not included in the training patterns (predictive abilities). The outputs of nodes in one layer are inputs to the next layer. The inputs to each node are combined using a weighted linear combination. The result is then modified by a nonlinear function before being converted to output. The following figure 2.4 shows the topology of the backpropagation neural network that includes an input layer, one hidden layer and an output layer. It should be noted that backpropagation neural networks can also have more than one hidden layer.

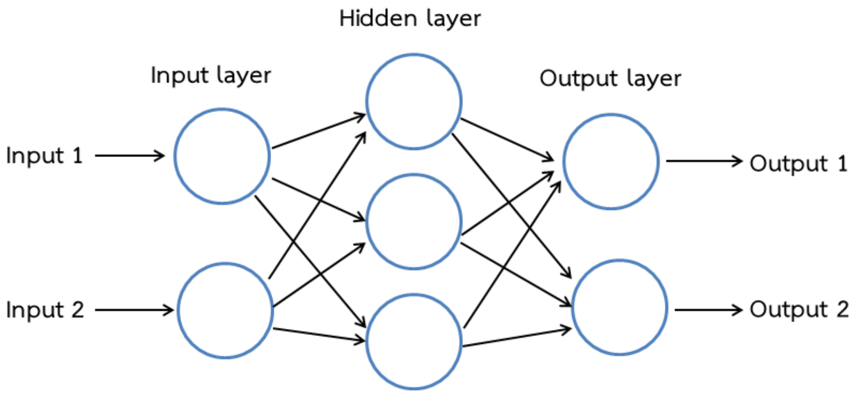


Figure 2.4: A neural network with two input and three hidden layers with two neurons [10].

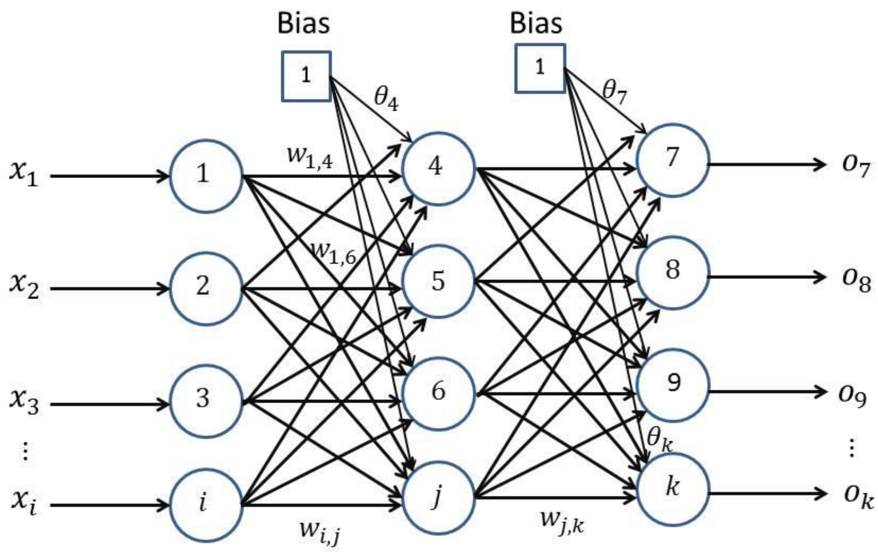


Figure 2.5: Backpropagation Neural Network with one hidden layer [10].

The following describes the learning algorithm and the equations used to train a neural network. For *feedforward,* the inputs to j^{th} node in the hidden neuron i on figure 2.4 are linearly combined to give

$$z_j = \theta_j + \sum_{i=1}^{n} w_{i,j} x_j . \tag{2.1}$$

The equation is used to calculate the inputs to the neuron. The θ_j term is the weighted value from a *bias* node that always has an output value of 1. The bias node is considered a "pseudo input" to each neuron in the hidden layer and the output layer, and is used to overcome the problems associated with situations where the values of an input pattern are zero. If an input pattern has zero values, the neural network could not be trained without a bias node. Within the hidden layer, this is then modified using a nonlinear function such as a sigmoid equation,

$$s(z) = \frac{1}{1+e^{-z}} . \tag{2.2}$$

The equation gives the input for the next layer. This tends to reduce the effect of extreme input values, thus making the network somewhat robust to outliers. In addition, the output value for node *K* can also be computed from the equations (2.1) and (2.2).

2.4 Support vector regression model

In machine learning, support vector machines (SVM) represent supervised learning models with associated learning algorithms that analyze data and recognize patterns, and is used for the classification and regression analysis proposed by Boser, Guyon and Vapnik [11, 12]. The basic concept algorithm is designed to work on dataset with two target classes by finding the linear hyper plane, which is used as the decision boundary to linearly separate instances in dataset into two sides. The hyper plane gives the largest minimum distance to the training set. This distance receives the important name of margin within the SVM theory. Therefore, the optimal separation of the hyper plane *maximizes* the margin of the training data. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM, which would have already been requested from the problem.

We will consider the problem of approximating the set of data. Suppose we are given a set of training data $\{(x_1, y_1), \ldots, (x_l, y_l)\} \subset \mathrm{X} \times \mathbb{R}$, where X denotes the space of input patterns. Within the $\varepsilon - SV$ regression, the target is to find a function $f(x)$ that has the highest deviation from the actual obtained targets y_i for all the training data. We begin by describing the case of a linear function f taking the form

$$f(x) = \langle w, x \rangle + b \quad \text{with } w \in X, b \in \mathbb{R}, \tag{2.3}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product in X, and where b is the “bias” term. Often the data is assumed to have a zero mean (this can be achieved via preprocessing), so the bias term can be dropped. From the equation (2.3), a small w is desired to minimize the Euclidean norm. We can specify the problem as a convex optimization problem by requiring

$$\text{minimize} \quad \frac{1}{2}\|w\|^2$$
$$\text{subject to} \begin{cases} y_i - \langle w, x_i \rangle - b \le \varepsilon \\ \langle w, x_i \rangle + b - y_i \le \varepsilon \end{cases}. \tag{2.4}$$

SVM regression performs linear regression in a high-dimensional feature space using ε-insensitive loss and, at the same time, tries to reduce the model complexity by minimizing $\|w\|^2$. This can be described by introducing (non-negative) slack variables ξ_i, ξ_i^*, $i = 1,..,n$, to measure the deviation of the training samples outside ε-insensitive zone. Thus, SVM regression is formulated as a minimization of the following function

$$\text{minimize} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*)$$
$$\text{subject to} \begin{cases} y_i - \langle w, x_i \rangle - b \le \varepsilon + \xi_i \\ \langle w, x_i \rangle + b - y_i \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases}. \tag{2.5}$$

The constant $C > 0$ determines the trade between the function f and the amount up to which deviations larger than ε are tolerated. The quality of estimation is measured by the loss function $|\xi|_\varepsilon$. The SVM regression uses a new type of loss function called ε-insensitive loss function, which is describe by

$$|\xi|_\varepsilon = \begin{cases} 0 & \text{if } |\xi| \le \varepsilon \\ |\xi| - \varepsilon & \text{otherwise} \end{cases}. \tag{2.6}$$

This optimization problem can be transformed into a dual problem for which the solution is given by

$$f(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\, k(x_i, x) + b \quad \text{s.t.} \quad 0 \le \alpha_i^* \le C, 0 \le \alpha_i \le C. \tag{2.7}$$

The kernel function is

$$k(x_i, x_j) = \exp\left(-\left\|\frac{x_i - x_j}{2\sigma^2}\right\|^2\right). \quad (2.8)$$

The SVM generalization performance (estimation accuracy) depends on a good setting of meta-parameters of C and ε, together with the kernel parameters.

2.5 Gaussian process model

A Gaussian process with supervised learning is the problem of establishing an input-output map from a training dataset. In this problem, we denote the input as x and the output as y. We consider a dataset D of n observations $D = \{(x_i, y_i) | i = 1,...,n\}$. The training data set can be used to make predictions for a new input X^* that is not seen in the training set. The finite training data can be transformed into a function f that makes predictions for all possible input values. A Gaussian process is thus a generalization of the Gaussian probability distribution [13].

Let the i^{th} input and the corresponding output of the computer simulator be denoted by a d-dimensional vector $x_i = (x_{i1},...,x_{id})^T$ and $y_i = y(x_i)$, respectively. The experimental design $D_0 = \{x_1,...,x_n\}$ is the set of n input trials stored in a $n \times d$ matrix X.

We assume $x_i \in [0,1]^d$, where the outputs are held in the $n \times 1$ vector $Y = y(x) = (y_1,...,y_n)^T$. Assuming a relationship of the form

$$y(x_i) = \mu + z(x_i), i = 1,...,n, \quad (2.9)$$

where μ is the mean, $z(x_i)$ is a Gaussian Process with

$$E(z(x)) = 0,\ Var(z(x_i)) = \sigma^2,$$

$$Cov(z(x_i), z(x_j)) = \sigma^2 R_{ij}. \quad (2.11)$$

The simulator output has a multivariate normal distribution ($N_n(1_n\mu,\Sigma)$), where $\Sigma=\sigma^2 R$ and 1_n is an $\mathbf{X}$ vector of all ones. The Gaussian correlation function is a special case of the power exponential correlation family

$$R_{ij} = corr(z(x_i), z(x_j)) = \prod_{k=1}^{d} \exp\{-\theta_k \left| x_{ik} - x_{jk} \right|^{p_k}\} \text{ for all } i, j, \quad (2.12)$$

where $\theta=(\theta_1,\dots,\theta_d) \in [0,\infty)^d$ is a vector of hyper-parameters and $p_k \in (0,2]$ is the smooth parameter. The mean μ and the variance σ^2 are given by

$$\hat{\mu}(\theta) = (1_n^{\mathrm{T}} R^{-1} \hat{\mu}(\theta))^{-1} (1_n^{\mathrm{T}} R^{-1} Y), \quad (2.13)$$

$$\sigma^2(\theta) = \frac{(Y - 1_n \hat{\mu}(\theta))^{\mathrm{T}} R^{-1} (Y - 1_n \hat{\mu}(\theta))}{n}. \quad (2.14)$$

The covariance function is required to be optimized in order to find the maximization point of the probabilistic value for this model. We use the maximum likelihood approach to estimate the hyper-parameters θ, followed by

$$-2\log(L_\theta) \propto \log(|R|) + n\log\left[\left(Y - 1_n \hat{\mu}(\theta)\right)^{\mathrm{T}} R^{-1} \left(Y - 1_n \hat{\mu}(\theta)\right)\right], \quad (2.15)$$

where $|R|$ is the determinant of R.The linear equation predicted at x^* is.

$$\hat{y}(x^*) = \hat{\mu} + r^{\mathrm{T}} R^{-1} (Y - 1_n \hat{\mu}) = \left[\frac{(1 - r^{\mathrm{T}} R^{-1} 1_n)}{1_n^{\mathrm{T}} R^{-1} 1_n} 1_n^{\mathrm{T}} + r^{\mathrm{T}} \right] R^{-1} Y = C^T Y. \quad (2.16)$$

where $r = (r_1(x^*), \dots, r_n(x^*))'$, $r_i(x^*) = \mathrm{corr}(z(x^*), z(x_i))$, while the mean square error $s^2(x^*)$ is defined by

$$\begin{aligned} s^2(x^*) &= \mathrm{E}\left[(\hat{y}(x^*) - y(x^*))^2 \right] \\ &= \sigma^2 (1 - 2C^T R C). \end{aligned} \quad (2.17)$$

As defined earlier, the Gaussian process (GP) models are a generalization of the Gaussian probability distribution. It is a supervised learning problem in a machine learning format. We will use the Gaussian process to predict the turning points in chapter 3. Through the package of the *Rstudio* program, we obtain the package of *kernlab* for the regression model.

2.6 Problems related to specific error functions and optimization

Peaks and *troughs* are considered as the sequence turning points of our dataset. We introduced $\hat{\Gamma}(t)$ is an approximate of $\Gamma(t)$ and the thresholds T_{low}, T_{high} as hyper-parameter, where $\hat{\Gamma}(t) < T_{low}$ are all considered as troughs, and $\hat{\Gamma}(t) > T_{high}$ are all considered as peaks. The parameter space (Grid) for the thresholds is as follow

$$\Theta\ (T_{low}, T_{high}) = [0,1]^2 \quad \text{with step 0.01.} \tag{2.18}$$

In addition, we also introduced a specific error function. The basic error function used in the model is the root mean square error (RMSE), where RMSE is used in conjunction with the predicted values. Li et al [6]. had proposed the use of specialized error function (*TpRMSE*) based on *RMSE*, which is given as follows

$$\text{TpRMSE} \triangleq \text{TpRMSE}\left(\Gamma'(t), \hat{\Gamma}(t), T_{low}, T_{high}\right) \triangleq \left(\frac{1}{n}\sum_{t=t_1}^{t_2}\left(\Gamma'(t) - \hat{\Gamma}(t)\right)\right)^{1/2}, \tag{2.19}$$

where $\Gamma'(t)$ is defined in terms of following the trimmed reference function

$$\Gamma'(t) = \begin{cases} T_{high} & \text{if } \Gamma(t) = 1 \text{ and } \hat{\Gamma}(t) < T_{high} \quad (\text{high false negative}) \\ T_{high} & \text{if } \Gamma(t) \neq 1 \text{ and } \hat{\Gamma}(t) > T_{high} \quad (\text{high false negative}) \\ T_{low} & \text{if } \Gamma(t) = 0 \text{ and } \hat{\Gamma}(t) > T_{low} \quad (\text{low false positive}) \\ T_{low} & \text{if } \Gamma(t) \neq 0 \text{ and } \hat{\Gamma}(t) < T_{high} \quad (\text{low false positive}) \\ \hat{\Gamma}(t) & \text{otherwise} \end{cases}.$$

From the model, we need to optimize the thresholds T_{low} and T_{high} so that we can optimize the *TpRMSE* error.

$$\left[T_{low}, T_{high}\right] = \underset{T'_{low}, T'_{high}}{\operatorname{argmin}} TpRMSE\left(\Gamma(t), \hat{\Gamma}(t), T'_{low}, T'_{high}\right). \tag{2.20}$$

To address problem specific errors in the study, we must use the *TpRMSE* error function because *RMSE* considers errors from all data points (non-specific). The *TpRMSE* error function has been defined in terms of the reference function. This error function is programmed to consider the turning points only.

2.7 Performance measures and trading strategies

Performance measures are composed of a number and a unit of measure. The number gives us a magnitude and the unit gives a meaning to the number. Performance measures are always tied to a goal or the ultimate target of the work.

In our research, we could evaluate our model using the error function *(TpRMSE)* defined above, and assess trading performance using cumulative return, maximum drawdown, Sharpe ratio and success rate. Next, we now explain a very simple trading strategy that can be used to specify buying and selling signals. A trader who is able to buy stocks at trough prices and sell at peak prices (buy low and sell high) to enter/exit the market can obtain the maximum possible profit. The idea of a trigger is to generate predictions $\hat{\Gamma}(t)$, with well-defined thresholds T_{low} and T_{high}. The simplest trigger, which is often used in general, is associated with predictions based on the following rules.

$$Trigger(t) = \begin{cases} Buy, & if\ \hat{\Gamma}(t) < T_{low} \quad and\ not\ in\ position \\ Sell, & if\ \hat{\Gamma}(t) > T_{high} \quad and\ in\ position \end{cases}. \tag{2.21}$$

These trading strategies are based on fundamental analysis, with the ‘buy signal’ being triggered when we are out of the market.

There are various types of measures that evaluate trading performance of the models. The trading performance is determined using cumulative return, maximum drawdown, the Sharpe ratio and the success rate. Let x_t^{t+n} be a financial prices sequence of length n. There are L pairs of times corresponding to matching buy and sell signals. Let $\{(b_i, s_i), b_i, s_i \in [t, t+n], i = 1, \ldots, L\}$ be defined in relation to buying the stock at price x_{b_i} and selling the stock at price x_{s_i}, which is generated by a trading strategy S with respect to the price sequence x_t^{t+n}.

The Cumulative return of a trading strategy S is the entire amount of money an investment has returned for an investor. The Cumulative return expresses the total percentage increase in the value of investment from the time it was purchased.

$$\text{The Cumulative return is, } ret_S(x_t^{t+n}) = \prod \frac{x_{s_i}}{x_{b_i}}. \tag{2.22}$$

The annualized cumulative return is the entire amount of money the investment has returned for the investor in a single year. The number of trading days per year is assumed to be 252.

$$\text{The annualized cumulative return is, } RET_S(x_t^{t+n}) = [ret_S(x_t^{t+n})]^{252/n}. \tag{2.23}$$

The rolling cumulative return is the curve of cumulative return through time.

The rolling cumulative return is, $ROC_s(t)$,

$$ROC_s(t) = \begin{cases} 1, & \text{if } t \notin \bigcup([b_i, s_i]) \\ x_t / x_{t-1}, & \text{if } t \in \bigcup([b_i, s_i]) \end{cases}, t = t+1, \ldots t+n. \tag{2.24}$$

The maximum drawdown (MDD) is a measure of the risk which is defined with respect to the rolling cumulative return curve. The MDD is the maximum decline of a series from a peak to a trough over a period of time. This is the accumulated loss of buying an investment at its highest local maximum price and selling it at its lowest minimum price. $ROC_s(t)$ is the cumulative return of the sequence. We consider the maximum drawdown in the time interval $[t, t+n]$ for MDD as follows;

$$MDD_S(x_t^{t+1}) = \max_{\tau\in[t,t+n]} \{ \max_{\tau\in[t,t+n]} [ROC_S(k) - ROC_S(\tau)]\}. \quad (2.24)$$

The Sharpe ratio (SR) has been developed. It is a measure for calculating risk-adjusted return. The SR is calculated using standard deviation of the rolling cumulative return and excess return to determine the associated reward per unit of risk. The Sharpe ratio (SR) is given as

$$\text{sharpe}_S(x_t^{t+1}) = \frac{ret_S(x_t^{t+1}) - 1}{std(\text{ROC}_S)(x_t^{t+1})}. \quad (2.25)$$

The annualized Sharpe ratio (ASR) is

$$\text{SHARPE}_S(x_t^{t+1}) = \frac{RET_S(x_t^{t+1}) - 1}{std(\text{ROC}_S)(x_t^{t+1}) \cdot \sqrt{252}}. \quad (2.26)$$

The rate of success is the faction or percentage of successful trades given by,

$$RATE_{S_s}(x_t^{t+n}) = \frac{\sum \text{II}(x_{s_i} > x_{b_i})}{L}. \quad (2.27)$$

The performance measurements represent a process toward a desirable outcome, including cumulative return, maximum drawdown, the Sharpe ratio and the success rate. Firstly, the cumulative return is defined as the total wealth accumulated overtime. Secondly, the maximum drawdown explains the risk of investment. Thirdly, the Sharpe ratio is a risk-adjusted return measure. Finally, the success rate is the determination of the number of successful trades amongst all the conducted trades. These are often used to evaluate the overall trading performance, and in generating trading strategies and tactical decisions prior to future trades. The measurements would enable traders to determine the effectiveness of their trades and the correlated success.

Chapter 2 has provided details regarding the use of turning points in preparing processes and in performance measure to determine the effectiveness of the models. The details of the work is described in chapter 3, where the turning point prediction and mathematical concepts are combined. Throughout the study, we construct the turning point predictions using algorithms.

CHAPTER III
PREDICTING TURNING POINTS

This chapter introduces the concept of turning point prediction utilized in this research. We firstly describe data preparation process based on three turning point definitions described in chapter 2. After that, the chapter will focus on the definition of the turning point oscillator as well as the method to construct this oscillator from alternating pivot sequence. Then, we will briefly describe the algorithm for predicting this oscillator using the Artificial Neural Network (ANN), the Support Vector Regression (SVR) method, and the Gaussian Process (GP) model. Finally, the framework for predicting turning points from the predicted oscillator as well as experimental designs for comparing the performance of the proposed framework are introduced.

3.1 Designing Algorithm

Let $X = x_1, x_2, \ldots, x_t, \ldots$ be a time series of stock price, with the index *t* representing time. To predict whether the stock price at time *t* is a turning point or not, we will use the backward window $W_t = X_{t-m}^{t-1}$, when m is the window size, as a sole information. The backward window may be transformed into a feature vector, by using price normalization process, and considered as an input of the models. The overview of the algorithm is illustrated as a flowchart in Figure 3.1. The detail of each step will be described be described in the rest of this chapter.

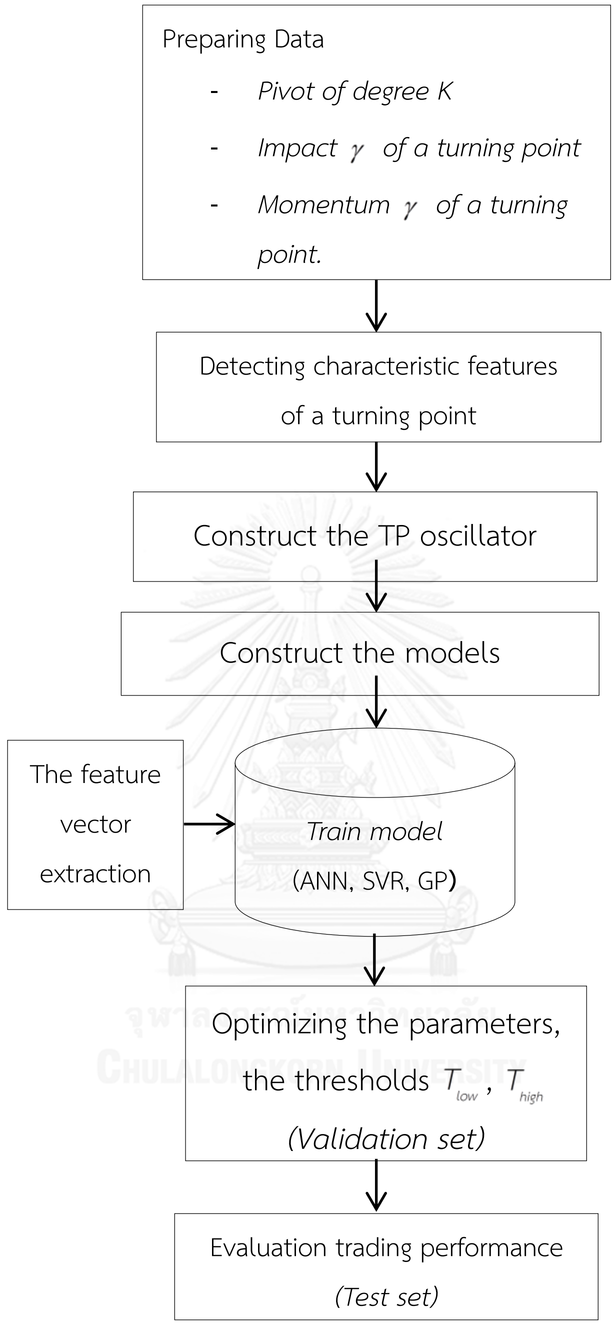
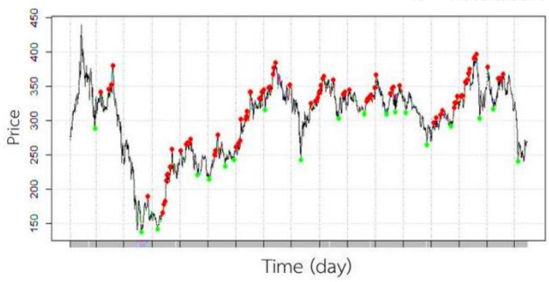


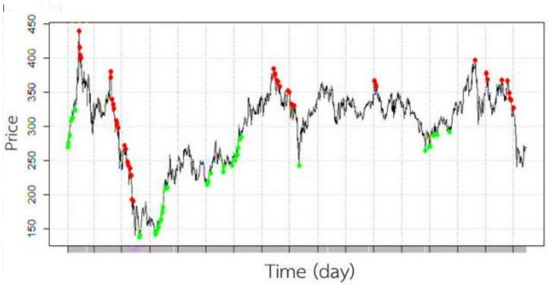
Figure 3.1: The flowchart of the entire turning point prediction model.

Algorithm for the prediction models

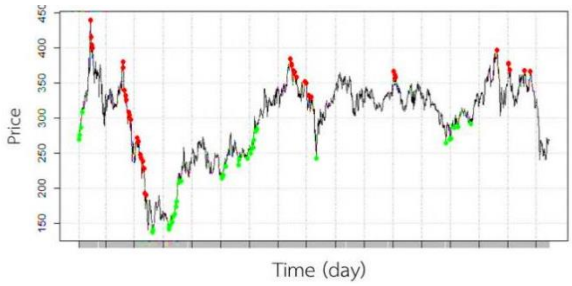
1. The turning point prediction algorithm utilized in this study begins with the process of preparing data for the detection of local maximum and local minimum in the time series of stock prices. These local maximums and minimums will be called pivots if it meets the criteria previously described in Chapter 2. These criteria include the pivot of degree K, the impact of a turning point, and the momentum of a turning point. It is important to note that the resulting pivot sequences obtained from these three turning point criteria can be very different. Figure 3.2 illustrates the resulted pivot sequences obtained from three different definitions which are the pivot of degree K (with K=35), the impact of a turning point (with $\gamma=1.35$), and the momentum of a turning point (with $\gamma=1.35$ and window of length $w=80$) using PTT stock price data. The green dots in the figure represent troughs while the red dots represent peaks in the sequence.



(a) Turning points of degree K=35



(b) Turning points with impact $\gamma=1.35$

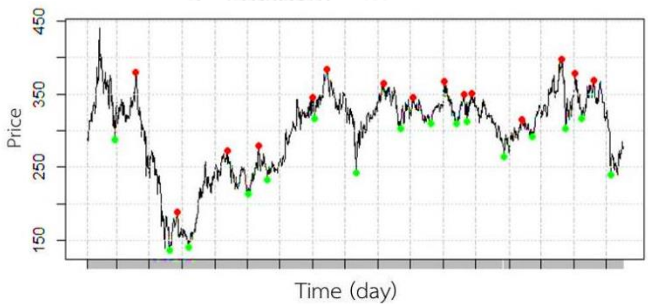


(c) Turning points with momentum $\gamma=1.35$

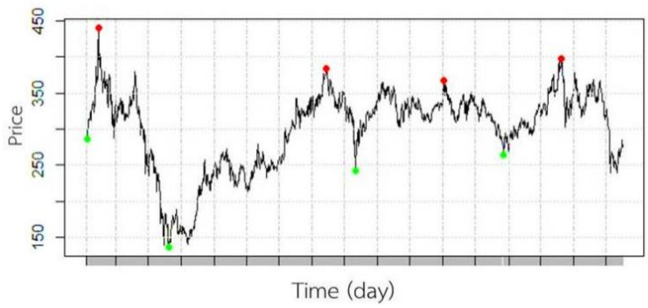
with respect to a lookahead window of length $w=80$ days

Figure 3.2: Examples of turning point types using PTT stock price data.

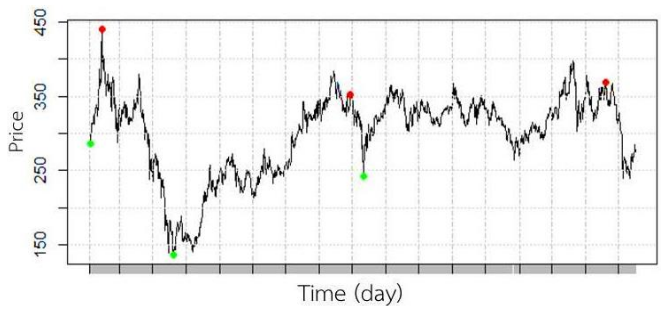
2. In this step, we consider the peaks and troughs as the sequence of turning points. We will extract an alternating pivot sequence $A(X)$ from the given pivot sequences. The extracted sequences will alternate between peaks and troughs (i.e. a peak will always follow by a trough and vice versa). In this sequence every trough can be considered as a global minimum within a certain time interval, with registered peaks surrounding it. Similarly, every peak can be considered as a global maximum within a time interval, with registered troughs surrounding it. After obtaining the alternative pivot sequences, we will utilize it to construct a turning point oscillator (TP oscillator) in the next step. The example of the alternating Figure 3.3 shows an example of the alternating pivot sequences obtained from the pivot sequences generated from the pivots of degree 35, pivot impact of $\gamma = 1.35$, and the pivot momentum of γ=1.35, with respect to a lookahead window of length w=80 using PTT stock price data.



(a) Turning points of degree K=35



(b) Turning points with impact $\gamma = 1.35$

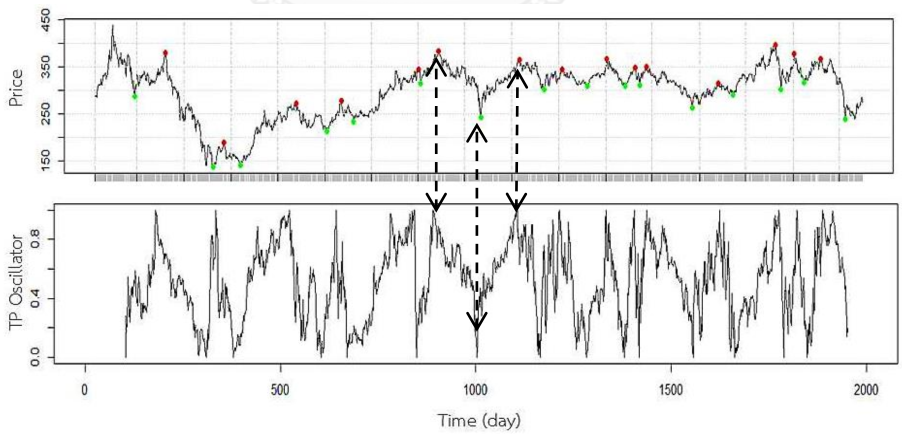


(c) Turning points with momentum γ=1.35

with respect to a lookahead window of length w=80

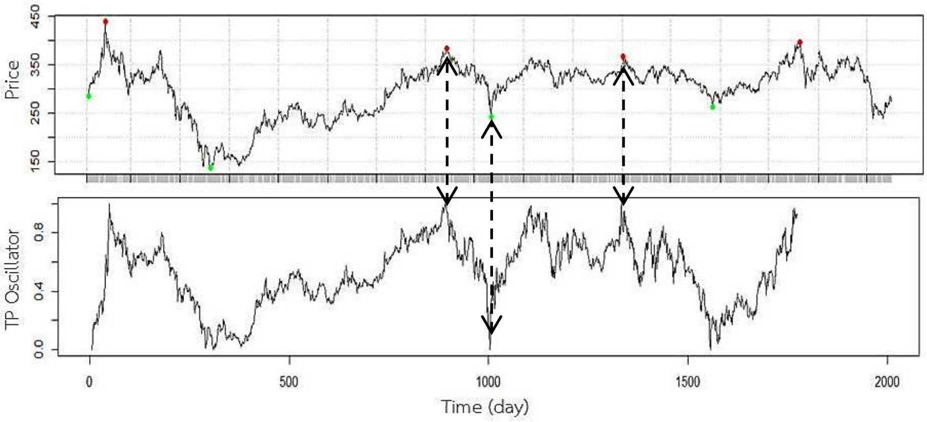
Figure 3.3: Examples of turning point types using PTT stock price data.

3. In this step, we construct a turning point oscillator (TP Oscillator) from the alternating pivot sequence using the definitions described in Chapter 2.The idea behind this oscillator is to normalize stock prices [0.1] interval so that 0 represents troughs in the sequence while 1 represents peaks in the sequence. Similar to the alternating pivot sequences, the TP oscillator also depends on the type of turning points. Figure 3.4 presents an example of a TP Oscillator for turning points with degree $\kappa = 35$ and momentum γ=1.35.



The upper graph shows the turning points with degree $\kappa = 35$

The lower graph shows the TP oscillator of turning point with degree $\kappa = 35$



The upper graph shows the turning point with impact γ=1.35

The lower graph shows the TP oscillator of turning point with impact γ=1.35

Figure 3.4: Examples of TP Oscillator types using PTT stock price data

The above figure is an example of varying TP oscillator types generated using the PTT stock price data. There are different types of TP oscillators. The type of TP oscillator depends on the construction process of turning points. Moreover, a TP oscillator computed with pivot degree $\kappa = 35$ or computed with impact γ=1.35, will have a lower number of peaks and troughs with the increase in the number of pivot points.

4. While previous steps focus on target values of the prediction, this step will focus on the input for making turning point prediction. To achieve this, we will use previous stock price information in a form of backward window, $w_t = x_{t-m}^{t-1}$ where m is the window size, as a sole information. The stock price in the backward window is then normalized into a [-1,1] range to create a feature vector. This process can be described by a feature extraction function $F(W_t)$ which can be defined by the following equation:

$$F(W_t) = \frac{2(w_t - \min(w_t))}{(\max(w_t) - \min(w_t))} - 1, \quad (3.1)$$

Figure 3.5 visualize this feature vector extraction process. In the next chapter, we will perform an experiment to find the best window size to make turning point prediction in our dataset as varying window size can greatly impact the performance of the models. This consequence of this framework is that the input of the model will be a vector whose dimension is m, and the output of the model will be a real value representing the value of the TP oscillator at some specified time. Consequently, the training dataset utilized to train the model in the next step can be defined as $S_n = \{(F(W_t), \Gamma(t)); t = 0,\ldots,n\}$.

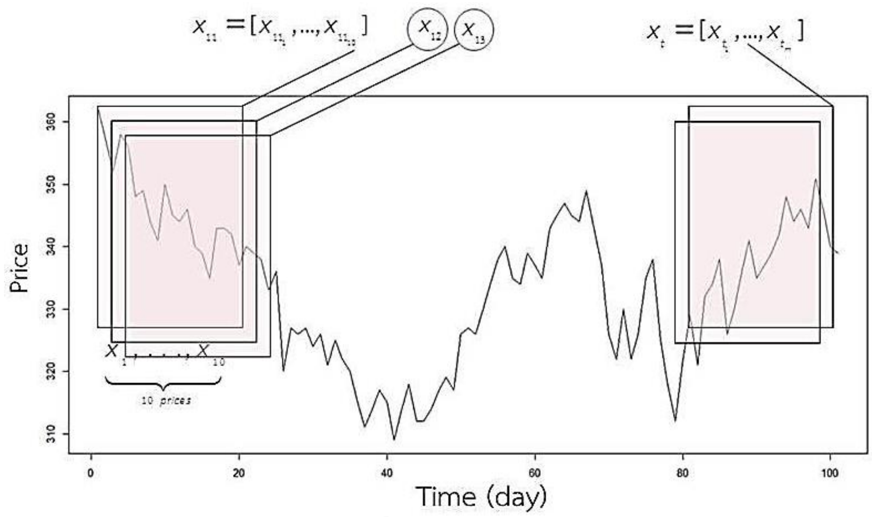


Figure 3.5: Examples of the feature vector extraction.

5. After obtaining the training dataset S_n in the previous step, this step will utilize this dataset to construct models to predict TP oscillator from past stock prices. As our target value is a real value, we can utilize any modern regression models to perform this tasks. In this study we will utilize the Artificial Neural Network (ANN), the Support Vector Regression (SVR) method, and the Gaussian process (GP) model to learn this function. After we obtained the model for predicting TP oscillator, we will utilized the resulted model to predict the turning point in the step subsequence sections.

6. After obtaining a model for predicting the value of TP oscillator, we need a mechanism to decide whether a given point is a turning point or not. According to the definition of the TP oscillator, this can be done easily as 0 represents troughs in the sequence while 1 represents peaks in the sequence. However, the predicted values of TP oscillator, $\hat{\Gamma}$(t), rarely touch 0 and 1 regions. To make this decision, we will introduce thresholds T_{low}, T_{high} as parameters where $\hat{\Gamma}(t) < T_{low}$ are all considered as troughs and $\hat{\Gamma}(t) > T_{high}$ are all considered as peaks. Consequently, we need to find the best thresholds for making this prediction and, in this study, we will utilize the thresholds that minimization the *TpRMSE* error function defined as the following,

$$\text{TpRMSE} \triangleq \text{TpRMSE}\left(\Gamma'(t), \hat{\Gamma}(t), T_{low}, T_{high}\right) \triangleq \left(\frac{1}{n}\sum_{t=t_1}^{t_2}\left(\Gamma'(t) - \hat{\Gamma}(t)\right)\right)^{1/2}$$

so that the optimization of the thresholds is given by

$$\left[T_{low}, T_{high}\right] = \underset{T_{low}, T_{high}}{\text{argmin}}\, TpRMSE\left(\Gamma(t), \hat{\Gamma}(t), T'_{low}, T'_{high}\right).$$

To achieve this, we will utilize a grid search method in a region of [0,1]x[0,1]with a step of 0.01.

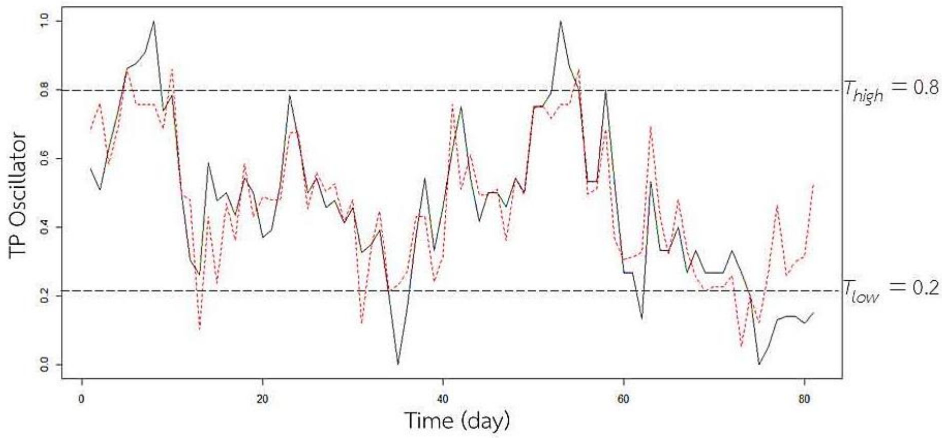


Figure 3.6: Example of the value of an actual TP Oscillator ($-\ \Gamma(t)$) **and the predicted turning points using the TP Oscillator** ($\text{----}\Gamma(t)$) **with** $T_{low} = 0.2$ **and** $T_{high} = 0.8$**.**

7. After obtaining the model and the threshold value for making turning point prediction, this final step will evaluate the performance of the prediction model. To achieve this, we will test the model with the testing dataset and evaluate the prediction results by trading simulation as described in Section 2.7. The trading performances measured in the study include cumulative return, maximum drawdown, the Shape ratio and the success rate.

Through the use of the algorithm to predict the turning points, we have been able to construct functional models to perform the experiments. From the algorithm, we have also been able to manage the training segments to fit the models, use the segments to estimate the parameters, as well as test segments for validation.

In the next chapter, we perform the experiments to test the algorithm. In addition, we also introduce the datasets for our experiments and the program to implement specific codes during the research. Furthermore, we introduce a simple trading strategy implemented for trading applications.

CHAPTER IV
EXPERIMENTS AND RESULTS

In this chapter, we briefly describe the dataset and the program required to implement the prediction model. In addition, we also describe a simple trading strategy implemented with the objective of trading applications. The dataset provided for testing the model is a real financial price sequence. The sequence contains a segment of the stock prices obtained from the master set of data from the Stock Exchange of Thailand. Because of the narrow field of computational sciences in Thailand, there has been a limit to the introduction of mathematical models for the prediction of turning points. As for the computational program, we will be using *the RStudio* program to implement the codes used during the research. The data used in this project have been obtained from a time series data of stock prices in Thailand. The conducted experiments involve the operation of trading. The models are evaluated using the error function *TpRMSE*, and the trading performance assessed using the cumulative return, the maximum drawdown, the Sharpe ratio and the success rate.

4.1 Dataset and experiment

In our research, we consider a time series dataset from the Stock Exchange of Thailand, where a data of PTT stock, were collected. These datasets are numeric, with 5 major attributes, containing 2000 days of record. The attributes include open prices, close price, high and low prices of each day, and the adjusted stock price. From the dataset, we have chosen the 'close prices' as the input data for our prediction models. The close price is defined as the last trading price recorded when the market closes at the end of a trading day. We evaluate the performance of the proposed methods over a long history of the PTT close prices, with an example of 20 data inputs shown in table 4.1.1. However, for the actual evaluation of the model, our research used all 2000 input data of the dataset for the PTT stocks.

Table 4.1: Example of a dataset of the PTT stocks

	Open	High	Low	Close	Volume	Adjusted
2/4/2014	303	307	301	307	4119000	307
3/4/2014	308	309	306	308	3662700	308
4/4/2014	308	309	304	306	2980200	306
8/4/2014	309	309	302	302	2028200	302
9/4/2014	303	306	303	304	1726000	304
10/4/2014	307	308	303	307	2042800	307
11/4/2014	305	307	305	307	973400	307
16/4/2014	309	316	309	315	5539100	315
17/4/2014	314	315	312	312	2381400	312
18/4/2014	312	314	311	311	1606900	311
21/4/2014	312	312	309	310	2139700	310
22/4/2014	311	312	306	307	3009800	307
23/4/2014	308	311	306	308	3951500	308
24/4/2014	308	309	306	309	2452700	309
4/4/2014	306	310	306	307	2402000	307
28/4/2014	307	309	306	306	1233700	306
29/4/2014	305	306	302	303	3613200	303
30/4/2014	306	313	304	313	7636800	313
2/5/2014	309	312	309	309	2692000	309
6/5/2014	310	311	308	309	1605800	309

In this research, we consider the autoregressive technique for the prediction of the turning points, and apply the method towards the trading of stocks in the stock exchange of Thailand. The preparation of data for the detection of local maximum and local minimum has been extracted from the financial price sequences. This conforms to the three principle definitions explained in chapter 2. Next, we detect the turning point and extract the characteristics of these turning points based on the pivots values of pivot degree K, pivot with impact γ , and pivot with momentum γ , with respect to a lookahead window of length w. This is an alternating sequence $A(X)$ of turning points, such that it can be used in the construction of a turning point oscillator (TP Oscillator). The problem with turning point prediction is that we cannot predict the turning points directly. To solve the problem, predictions are made using a turning point oscillator (TP oscillator). The oscillator has a value between 0 and 1, with the numerical value 0 assigned to all troughs and the numerical value of 1 assigned to all peak, as discussed in chapter 2. The information provided in the model is based on a given financial price sequence, which is a part of the stock price master data from the Stock Exchange of Thailand. We define the price sequence as X. Let $X = x_1, x_2, ..., x_t$ be a real sequence, $x_t \in \mathbf{X}$. We describe x_t as the financial price sequence, where t is the index of time. We have denote the backward window of day t as $w_t = x_{t-m}^{t-1}$, where m is the window size of the prices. This feature has specifically been generated for the normalization of prices. The prediction of turning points involves several features and backward window lengths. In this research, we distinctly consider these features necessary for the normalization of prices. We consider the backward window length using the past ten prices. Throughout this research, we divided the financial price information as either training data or testing data. The training dataset is used in fitting the price data with the Artificial Neural Network, the Support Vector Regression, and the Gaussian Process model, while the testing dataset is used in predicting the turning points and in evaluating the performance of the models.

4.1.1 Train, validation and test set

The financial price sequence of the PTT stock consists of train, validation and test segments, as shown in figure 4.1.1. The training segment is used in fitting the models, the validation segment to estimate the parameters T_{low} and T_{high}, and the test segment to assess the performance of the prediction models. The percentage of each data segment considered as train, validation and test segment is 60%, 20% and 20%, respectively.

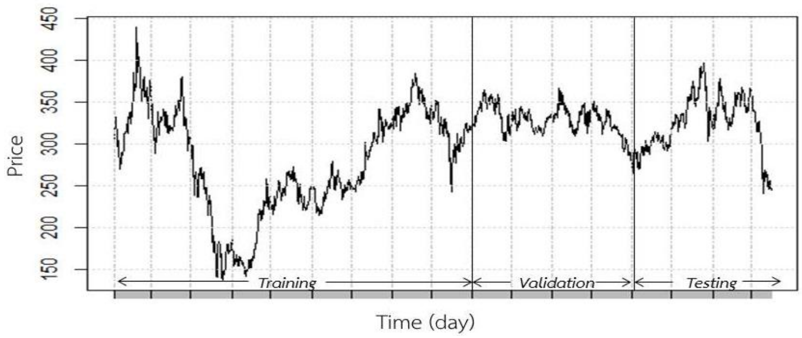


Figure 4.1: Separating data (PTT) into train and test segments.

4.2 Experimental settings

In this section, we will conduct three experiments to compare the performance of the proposed turning point prediction models on the PTT stock price.

In the first experiment, we make use of directed analyses to determine the most suitable definition out of the three definitions (i.e. turning points with pivot K, turning points with impact γ, and turning points with momentum γ, with respect to a lookahead window of length w days). We assume simulated trading by computing the commission rate that a trader will pay at 0.2% of the net sale price. We then perform a sample trading by buying at a trough and selling at a peak. Then, we select the definition that has highest cumulative return for further analysis in the subsequence experiments.

In the second experiment, we find the best backward window size for modeling TP oscillator. This is achieved by training the model with different window sizes using training set and selects the one that has lowest root mean square error in the validation set. The models utilized in this experiment include Artificial Neural Network, Support Vector Regression and Gaussian Process Regression model.

In the last experiment, we evaluate the performance of our turning point prediction model on the testing set. The performance of our models is evaluated using both TpRMSE and trading performance, which includes cumulative return, maximum drawdown, Shape ratio and success rate.

4.2.1 Experiment 1

The experimental in the study is comprised of the following; points of alternating pivot sequence, average time period, cumulative return and average of cumulative return from an different turning point of degree K, impact γ and momentum γ with respect to a lookahead window of length w days. In a particular scheme, we consider the cumulative return while keeping in mind that the ultimate objective is to pick the most suitable dataset. In the experiment, we also consider the simulation of stock trading. The simulated stock trading makes use of the PTT stock price. Additionally, if the pivot of a turning point has a degree of K, we consider an increase equivalent to the degree of K value. By definition, pivots of higher degrees correspond to larger price swings. Therefore, the points of alternating pivot sequences present as small. For this reason, the profitability of stock trading can be low. Nevertheless, the degree K value can also manifest as being small. As a result, traders make more frequent trades in order to make a certain amount of profit from stock trading. This explanation is applicable to the other two definitions as well.

Table 4.2: Experimental data of turning points with pivot degree K=1 to K=15.

Degree K	Points of alternating pivot sequence	Mean period time	Cumulative Return	Average Cumulative Return
Turning points with pivot degree K=1 to K=15				
1	1091	1.83211	13.26033	1.004754
2	591	3.377966	12.71224	1.008656
3	391	5.089744	11.70892	1.012697
4	305	6.529605	11.22745	1.016038
5	248	8.02834	10.4772	1.019283
6	221	8.972727	10.10788	1.021253
7	177	11.21591	9.506384	1.025921
8	147	13.47945	8.807815	1.030252
9	117	16.96552	8.004242	1.036512

10	107	18.38679	7.512015	1.03878
11	101	19.49	7.362883	1.040737
12	95	20.73404	7.231862	1.042994
13	89	22.14773	7.027595	1.045311
14	87	22.53488	6.918705	1.046009
15	79	24.84615	6.709076	1.050017

Table 4.3: Experimental data of turning points of impact γ =1.01-1.15.

γ	Points of alternating pivot sequence	Mean period time	Cumulative Return	Mean Cumulative Return
Turning points of impact γ =1.01-1.15				
1.01	566	3.532743	13.40152	1.009246
1.02	368	5.422343	12.32284	1.013818
1.03	268	7.453184	11.38446	1.018456
1.04	186	10.75676	10.05935	1.02541
1.05	140	14.31655	9.167361	1.032632
1.06	122	16.44628	8.702487	1.036718
1.07	98	20.51546	8.127758	1.044619
1.08	84	23.9759	7.712312	1.051087
1.09	65	30.71875	7.044938	1.062909
1.1	53	37.80769	6.524254	1.074801
1.11	49	40.95833	6.330131	1.079922
1.12	39	51.73684	5.771344	1.096648
1.13	37	53.86111	5.654985	1.101037
1.14	29	69.25	5.175137	1.124591
1.15	29	69.25	5.175137	1.124591

Table 4.4: Experimental data of turning points with momentum γ =1.01-1.15, with respect to a lookahead window of length 1 day.

γ	Points of alternating pivot sequence	Mean period time	Cumulative Return	Mean Cumulative Return
Turning points with momentum γ =1.01-1.15, with respect to a lookahead window of length 1 day				
1.01	565	3.535461	13.40152	1.009246
1.02	368	5.419619	12.32284	1.013818
1.03	268	7.449438	11.38446	1.018456
1.04	186	10.71892	10.05935	1.02541
1.05	140	14.26619	9.167361	1.032632
1.06	122	16.38843	8.702487	1.036718
1.07	98	20.4433	8.127758	1.044619
1.08	83	23.89024	7.712312	1.051087
1.09	65	30.60938	7.044938	1.062909
1.1	53	37.67308	6.524254	1.074801
1.11	49	40.8125	6.330131	1.079922
1.12	39	51.55263	5.771344	1.096648
1.13	37	54.41667	5.654985	1.101037
1.14	29	69.96429	5.175137	1.124591
1.15	29	69.96429	5.175137	1.124591

We have made use of the datasets from tables 4.2-4.4 to assure that we have selected the most appropriate definition. The definitions include pivot of degree K=1, turning point of impact γ =1.01 and turning point with momentum γ =1.01, with respect to a lookahead window of length 1 day. From these definitions, the objective of the study is to achieve the maximum cumulative return. This information will also be utilized as part of further examinations to discover the backward window length.

4.2.2 Experiment 2

In our experiment, we consider the backward window size for normalized prices. We based the operation of our developed model on the ANN, SVR and GP models. From experiment 1, the cumulative return of turning point of impact γ =1.01 was no different than the results achieved with momentum γ =1.01, with respect to a lookahead window of length 1 day. Throughout the study, turning point of impact γ =1.01 presented with the most comprehensive results. In addition, we were also able to find the size of the backward window with impact γ =1.01.

Table 4.5: Experimental data of the ANN model, with turning points of impact γ =1.01 and the feature vector consisting of the past 5-30 prices.

Backward window size	RMSE of training set	RMSE of validation set
5	0.261567	0.295412
6	0.256612	0.315298
7	0.252777	0.314972
8	0.246899	0.314689
9	0.246466	0.313333
10	0.244357	0.318461
11	0.238091	0.32634
12	0.236412	0.321028
13	0.236096	0.315785
14	0.227263	0.32611
15	0.224134	0.336539
16	0.226137	0.314598
17	0.226451	0.342214
18	0.221023	0.325664
19	0.224593	0.359878
20	0.209488	0.355116
21	0.217084	0.339934

22	0.206083	0.357662
23	0.203778	0.355387
24	0.210245	0.342973
25	0.201366	0.368187
26	0.200678	0.375248
27	0.194892	0.368655
28	0.201911	0.384932
29	0.190798	0.373902
30	0.185483	0.389509

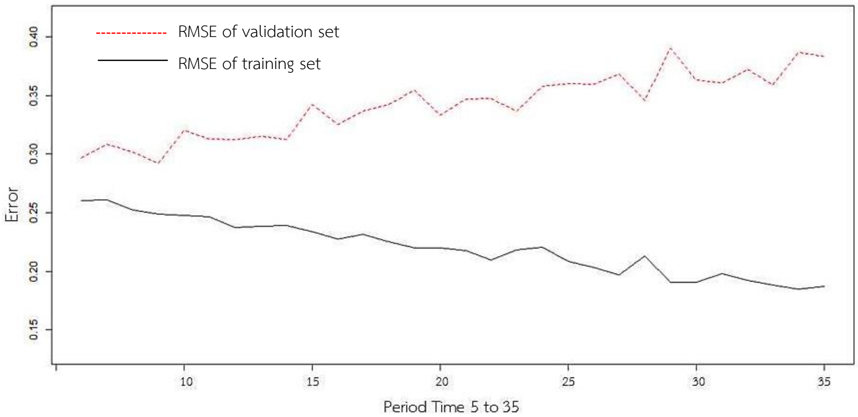


Figure 4.2: The RMSE between training set and validation set of the ANN model with turning point of impact γ =1.01.

Table 4.6: Experimental data of the SVR model, with turning points of impact γ =1.01 and the feature vector consisting of the past 5-30 prices.

Backward window size	RMSE of training set	RMSE of validation set
5	0.278801	0.294922
6	0.273275	0.291627
7	0.274171	0.301726
8	0.269271	0.300891
9	0.26442	0.303078
10	0.262658	0.312805

11	0.262969	0.305313
12	0.261951	0.308417
13	0.262458	0.30628
14	0.2619	0.307694
15	0.260215	0.309502
16	0.257545	0.314255
17	0.256569	0.313088
18	0.256488	0.309465
19	0.255117	0.312607
20	0.251462	0.310392
21	0.250305	0.31659
22	0.248438	0.321995
23	0.247626	0.324815
24	0.246318	0.320671
25	0.246467	0.318857
26	0.246557	0.316854
27	0.246714	0.316432
28	0.246511	0.319657
29	0.247273	0.325352
30	0.247053	0.327385

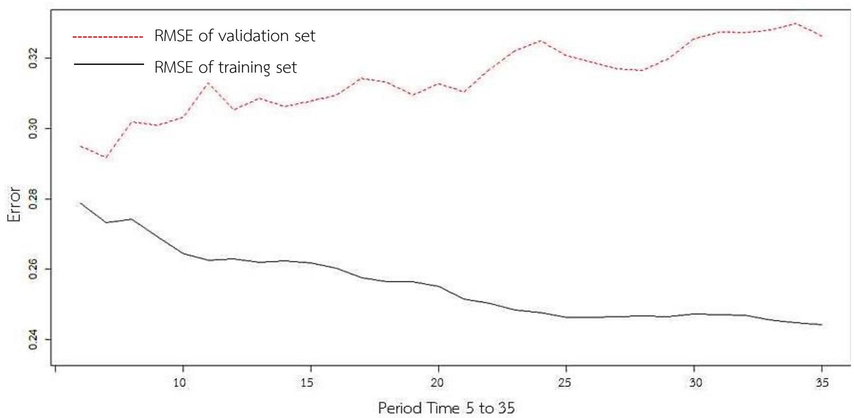


Figure 4.3: The RMSE between training set and validation set of the SVR model with turning point of impact γ =1.01.

Table 4.7: Experimental data of the GP model, with turning points of impact γ =1.01 and the feature vector consisting of the past 5-30 prices.

Backward window size	RMSE of training set	RMSE of validation set
5	0.679467	0.276986
6	0.670473	0.276691
7	0.667247	0.282992
8	0.661249	0.284526
9	0.659346	0.287587
10	0.658846	0.295222
11	0.653058	0.292477
12	0.651675	0.294024
13	0.646558	0.292982
14	0.645501	0.29325
15	0.642487	0.293973
16	0.640269	0.295448
17	0.638717	0.294738
18	0.635992	0.29272
19	0.637093	0.293537
20	0.634494	0.294225
21	0.630442	0.298053
22	0.632246	0.302175
23	0.630093	0.303595
24	0.627994	0.300692
25	0.628542	0.299527
26	0.625918	0.298325
27	0.624776	0.298629
28	0.622897	0.300903
29	0.620781	0.30292
30	0.619439	0.305056

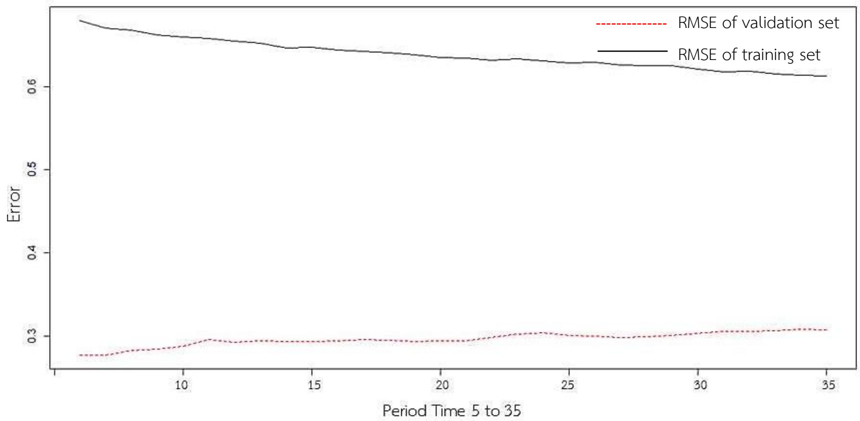


Figure 4.4: The RMSE between training set and validation set of the GP model with turning point of impact γ =1.01.

In table 4.5-4.7, we restricted attention to turning points with impact γ =1.01 with the ANN, SVR and GP models. In this experiment, the feature vector consisted of the past five, six and six price values in association with the ANN, SVR and GP models, respectively. This information will be utilized as part of further examinations in the construction of the TP oscillator model for turning point prediction in the last experiment.

From previous experiments, we restricted attention to turning points with impact γ =1.01 to be used with the ANN SVR and GP models, with the feature vectors consisting of the past five, six and six prices, respectively.

4.2.3 Experiment 3

From the outcomes of the previous experiments (1 and 2), the results of the current experiment have been determined. The results represent a comparative study of the ANN, SVR and GP models. The experimental data show the number of stocks traded, the mean cumulative return from the stocks, and the mean number of days each stock was held for. Moreover, this experiment also shows a comparison of the cumulative return, the annualized cumulative return, the Sharpe ratio, the annualized Sharpe ratio and the success rate generated using the three models. The definitions we considered in this study were based on turning point with impact γ =1.01, and used with the ANN, SVR and GP models, with the feature vectors constituting of the past three, five and three prices, respectively. The turning point identification has been achieved using two additional hyper-parameters, T_{low} and T_{high} .

Table 4.8: Performance of the models in predicting turning point of impact γ =1.01, with the ANN, SVR and GP model (using RBF for SVR and GP)

Models	ANN	SVR	GP
T_{low} and T_{high}	0.23, 0.75	0.33, 0.65	0.35, 0.67
TpRMSE	0.1103241	0.143029	0.0204369
cumulative return	0.8557061	1.093163	1.151083
Average cumulative return	0.9954273	1.001857	1.002562
Annualized cumulative return	0.9062701	1.05787	1.092933
Maximum drawdown	0.12638	0.1294585	0.1294585
Sharpe ratio	-10.16415	5.477872	8.897039
Annualized Sharpe ratio	-0.4159116	0.2143516	0.3447475
success rate	0.4857143	0.4375	0.4363636
Number of stock traded	35	48	55
Mean number of days each hold stock.	5.457143	4.145833	3.727273

Table 4.8 presents the results of these tests for the turning point with impact γ =1.01. The GP model performed better than the ANN and the SVR models in terms of *TpRMSE,* the cumulative return, maximum drawdown and the Sharpe ratio. However, the *maximum drawdown* is the lowest for the ANN model. When considering the success rate, we observe that the ANN model achieved a higher success than both the SVR and the GP models. Figure 4.5-4.10, shows predictions obtained by the ANN, SVR and GP models over a test set of PTT stock prices.

In figure 4.5-4.9, we show the turning point predictions obtained by the ANN, SVR and GP models over a test segment of the PTT stock prices. We are only interested in turning point with impact γ =1.01. In this experiment, we trained the models over 06-09-2007 to 27-07-2012. The thresholds T_{low} and T_{high} were selected over a validation segment from 07-08-2012 to 14-03-2014, and predictions were performed for test segment from 24-03-2014 to 04-11-15.

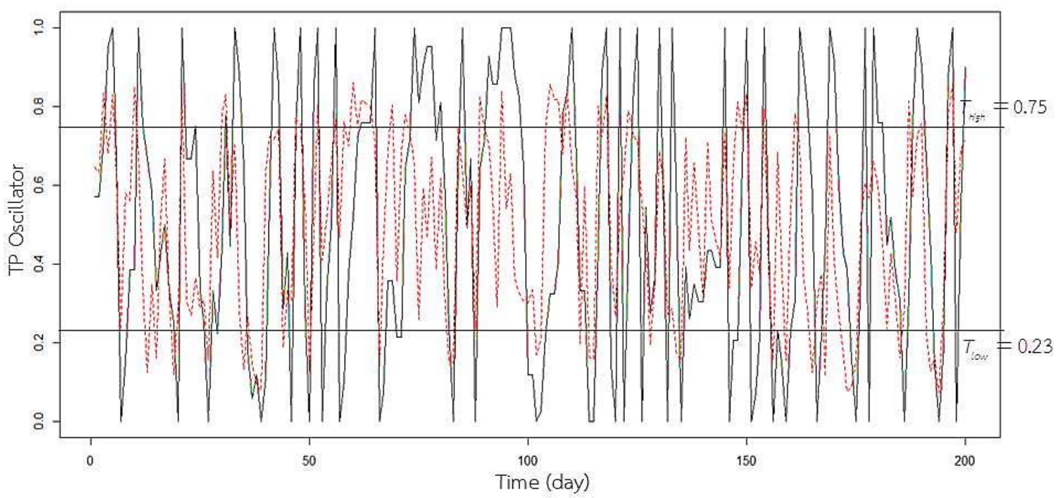


Figure 4.5: TP Oscillator and its prediction by the ANN model with T_{low} =0.23 and T_{high} =0.75 over the PTT, 24-03-2014 to 19-01-2015.

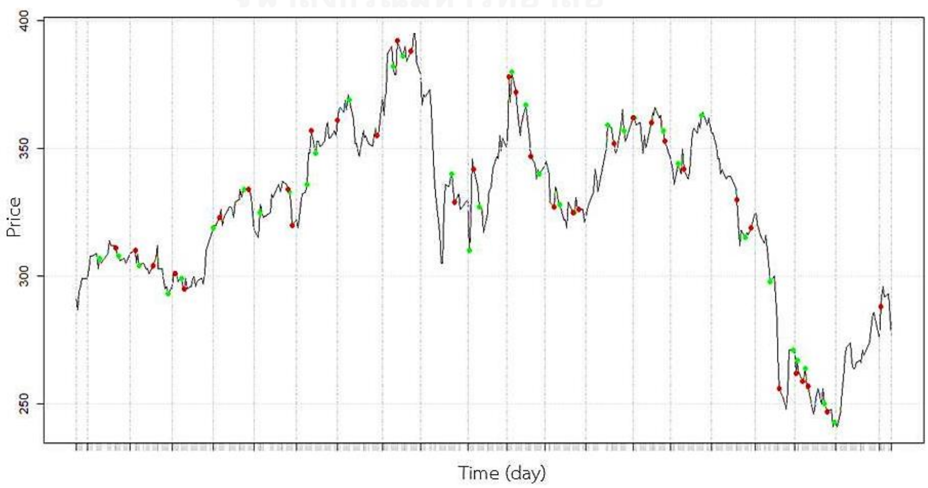


Figure 4.6: The circles mark the predicted turning points of the ANN model over a test set of PTT stock prices over the PTT, 24-03-2014 to 11-11-2015.

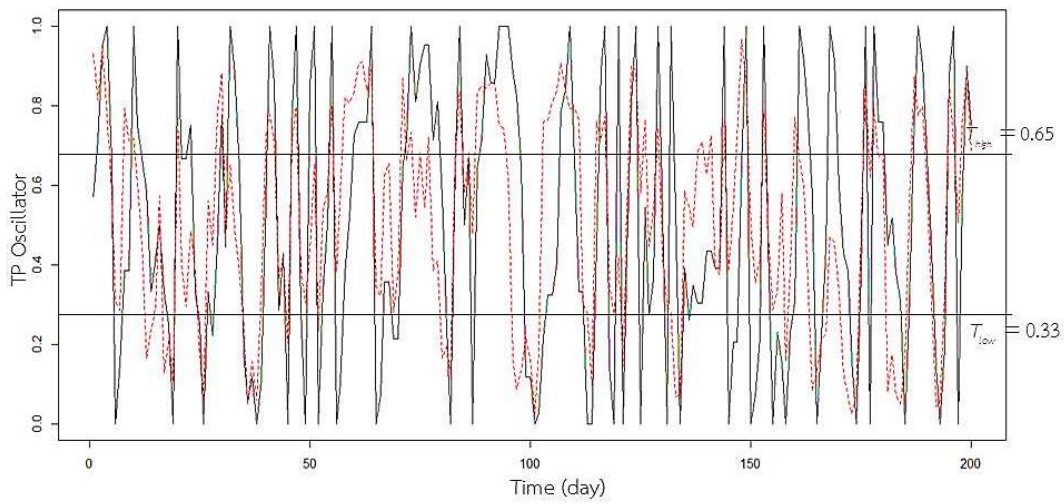


Figure 4.7: Figure 4.7: TP Oscillator and its prediction by the SVR model with T_{low} =0.33 and T_{high} =0.65 over the PTT, 24-03-2014 to 19-01-2015.

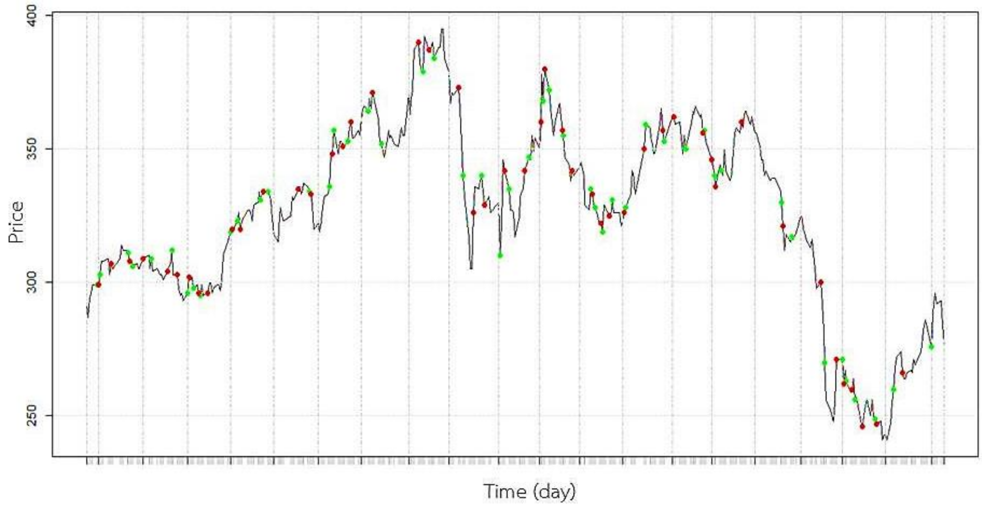


Figure 4.8: The circles mark the predicted turning points of the SVR model over a test set of PTT stock prices over the PTT, 24-03-2014 to 11-11-2015.

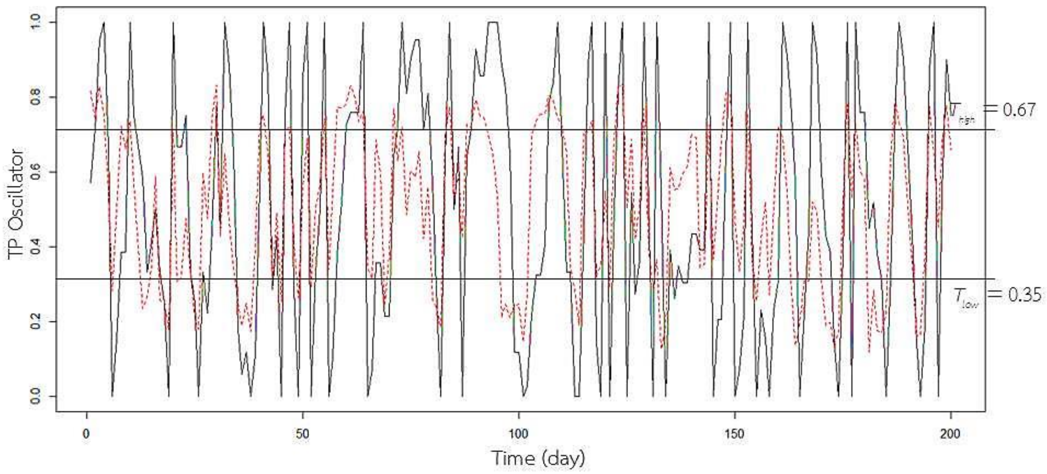


Figure 4.9: TP Oscillator and its prediction by the GP model with T_{low} =0.35 and T_{high} =0.67 over the PTT, 24-03-2014 to 19-01-2015.

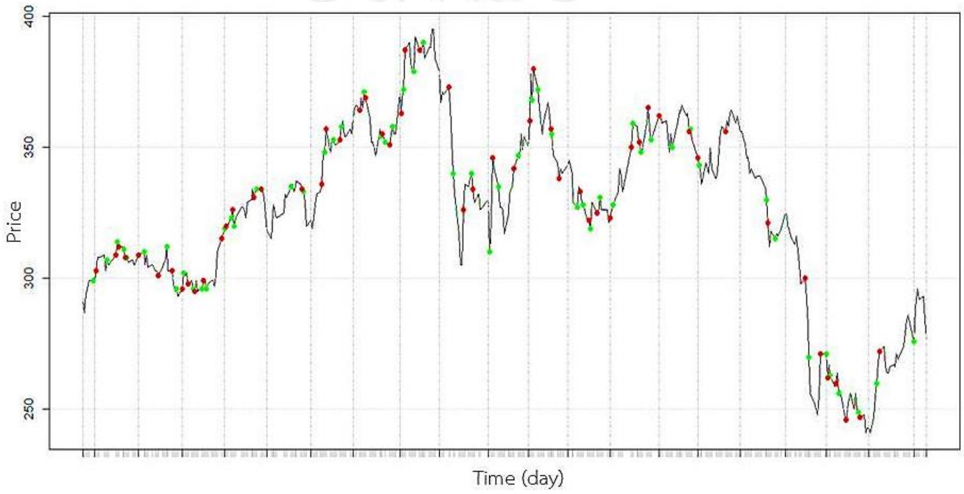


Figure 4.10: The circles mark the predicted turning points of the GP model over a test set of PTT stock prices over the PTT, 24-03-2014 to 11-11-2015.

From the comparative analysis, based on the cumulative return and the Sharpe ratio, we can conclude that the SVR model performed better than the ANN and the GP models. However, the GP model has an overall higher accuracy of prediction compared to the other two models. On the other hand, the ANN model has a higher rate of success. The GP model also showed the best performance in terms of *TpRMSE* and other performance measures, based on turning point with impact γ =1.01. The results additionally support the point that the proposed models are more useful for predicting turning points with turning point with impact γ =1.01, with the feature vectors consisting of the past six prices for the PTT stock.

From previous experiments, we restricted attention to turning point with impact γ =1.01 to be used with the ANN SVR and GP models, with the feature vectors consisting of the past five, six and six prices, respectively.

CHAPTER V
CONCLUSION

5.1 Conclusion

We began this research with a review of the concept of turning points (TP) in order to accumulate some basics necessary for the study of turning point prediction. In particular, we have reviewed the literature about turning point prediction using the Support Vector Regression method and the Artificial Neural network model. These models encompass the underlying knowledge for the study of turning point prediction. In this study, we proposed and studied a prediction model for turning points that relies on the ANN, SVR and GP models. We also made an attempt to gather useful information from the publications of Li et. al. and El-Yaniv and Faynburd [5, 6]. In the next step, we conducted experiments using the three models (ANN, SVR and GP). The experimental results pointed towards the performance of the models in predicting the turning points. We have also compared the efficiency of the three models, when applying them to stock trading.

In our research, we devised three experimental tasks for turning point prediction based on three definitions (i.e. turning points with pivot *K*, turning points with impact γ, and turning points with momentum γ, with respect to a lookahead window of length w days). The experiments were conducted in order to find the appropriate parameter for each definition in predicting turning points. For the purpose of evaluation, we used the PTT stock prices obtained from the Stock Exchange of Thailand. The tools for the prediction of turning points have been summarized in table 5.1.

Table 5.1 Summary of turning point properties and feature vector length.

Turning point with impact γ =1.01		
	Models	Feature vector length
Experiment	- *Artificial Neural Network* - *Support Vector Regression* - *Gaussian Process*	- *past 5 prices* - *past 6 prices* - *past 6 prices*

The conclusions we have drawn is in regards to the three turning point properties (table 5.1). In terms of the performance, the GP model performed better than the ANN and SVR models, when applied to stock trading. It has been observed that the performance of the GP model has a better outcome for cumulative returns. In the same way, the ANN model has also achieved a lowest risk just return (the Sharpe ratio), compared to the SVR and GP models. On the other hand, in terms of error values (TpRMSE), the three models presented with similar results. In brief, the GP model is the most suitable model (out of the three models) to utilize in the trading for PTT stocks.

5.2 Future work

In a future study, we will improve upon this technique and will investigate the possibility of using another model to support the current model in the prediction of turning points within a financial price sequence. Another extension will be to construct a mathematical model to test with a different type of stock price sequence. Furthermore, we will also utilize the developed turning point prediction model to device a more effective trading strategy for the derivation of maximum profit.

REFERENCES

1. A. F. Burns, W.C.M., *Measuring Business Cycles.* NBER, 1946.
2. G. Bry, C.B., *Cyclical Analysis of Time Series Selected Procedures and Computer Programs.* National Bureau of Economic Research, 1971.
3. Kling, J.L., *Predicting the Turning Points of Business and Economic Time Series.* The Journal of Business, 1971. **60**(2): p. 201-238.
4. DORFMAN, D.T.L.a.J.H., *Predicting Turning Points through The Integration Of Multiple Models.* Journal of Business & Economic Statistics. **14**(4): p. 421-428.
5. Li et, a., *Trading strategy design in financial investment through a turning Points prediction scheme.* Expert Systems with Applications, 2009. **36**: p. 7181-7826.
6. Faynburd, E.-Y.a., *Autoregressive short-term prediction of turning point using support vector regression.* Israel Institute of Technology, 2012.
7. Petelin, D., *Financial Modeling Using Gaussian Process Models.* The 16th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems, 2011.
8. Brahim-Belhouari, S., *Gaussian process for nonstationary time series prediction.* Computational Statistics & Data Analysis 2004. **47**: p. 705-712.
9. Deng, X.L.a.Z., *A Machine Learning Approach to Predict Turning Points for Chaotic Financial Time Series.* The 19thIEEE International Conference on Tools with Artificial Intelligence, 2007.
10. Haykin, S., *Neural Networks a Comprehensive Foundation*. 1999: New Jersey.
11. Vapnik, V., *The Nature of Statistical Learning Theory*. 1995: Springer.
12. Vapnik, V., *Statistical Learning Theory*. 1998, New York: Wiley-Interscience.
13. MacDonald, B., P. Ranjan, and H. Chipman, *GPfit: An R package for Gaussian Process Model Fitting using a New Optimization Algorithm.* Journal of Statistical Software, 2015. **VV**(12): p. 20.

APPENDIX

Appendix A: Software

We will be using the *RStudio* program as a tool in implementing the necessary codes during the research. The data used in this project have been obtained from a time series data of stock prices in Thailand. Dataset for this thesis have been generated from the packages provided in the *RStudio* program. This package is in the form of *quantmod*, where the package for R has been designed to assist a quantitative trader in the development, testing, and deployment of statistically based trading models. Figure (a) presents a template from the *RStudio* program. The conducted experiments involve the operation of trading. The GP model is evaluated using the error function *(TpRMSE),* and the trading performance assessed using *cumulative return, maximum drawdown, the Sharpe ratio* and *the success rate.*

We used the models to predict the turning points in chapter 3. Through the package of the *Rstudio* program, we obtain the package of *nnet, e1071 and kernlab* for the Neual Network, Support Vector Regression and Gaussian Process model, respectively.

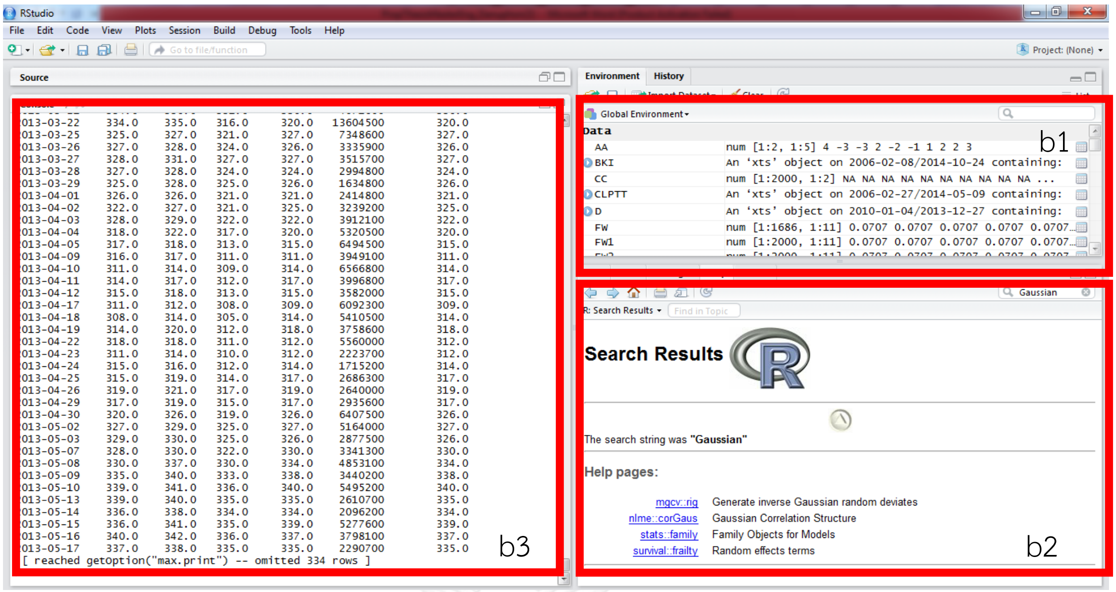


Figure A: Template from the *RStudio* program

Figure (b1) contains the workspace for a history of *R* commands. Any plot generated will show up in this region and will assist the *R* program shown in Figure (b2). Figure (b3) shows the *R* console including the inserted commands and the observable output.

Appendix B: Experimental results

Experiment B1

We consider the backward window size for normalized prices. We based the operation of our developed model on the ANN, SVR and GP models.

Table 1: Experimental data of the ANN model, with turning point with impact γ =1.01 and the feature vector consisting of the past 5-50 prices.

Backward window size	RMSE of training set	RMSE of validation set
5	0.260335	0.296978
6	0.260795	0.308332
7	0.252058	0.301903
8	0.24873	0.29193
9	0.247594	0.320221

10	0.246407	0.31291
11	0.23749	0.31221
12	0.238392	0.315434
13	0.238985	0.312224
14	0.234053	0.3426
15	0.227616	0.324911
16	0.2317	0.336684
17	0.225476	0.342253
18	0.220052	0.354673
19	0.220011	0.333045
20	0.21786	0.347102
21	0.209617	0.347368
22	0.218532	0.336545
23	0.220618	0.358176
24	0.208224	0.360535
25	0.203029	0.359512
26	0.197153	0.368283
27	0.213	0.345987
28	0.190387	0.390049
29	0.190455	0.363062
30	0.198181	0.360963
31	0.19223	0.372689
32	0.188112	0.359373
33	0.184959	0.386715
34	0.186894	0.3834
35	0.184701	0.372902
36	0.187192	0.391107
37	0.173453	0.390862
38	0.166628	0.37992
39	0.175045	0.376365

40	0.157202	0.383985
41	0.155796	0.3995
42	0.172397	0.384869
43	0.178108	0.378502
44	0.154586	0.402115
45	0.172034	0.391831
46	0.160643	0.386979
47	0.153069	0.415254
48	0.14764	0.393748
49	0.156857	0.395991
50	0.163968	0.368429

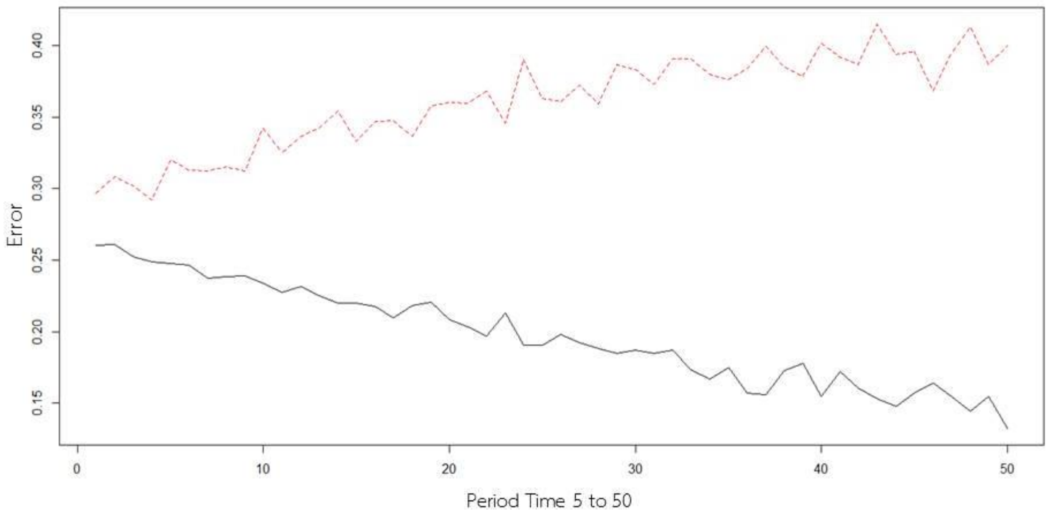


Figure 1: The RMSE between training set and validation set of the ANN model with turning point with impact γ =1.01.

Table 2: Experimental data of the SVR model, with turning point with impact γ =1.01 and the feature vector consisting of the past 5-50 prices.

Backward window size	RMSE of training set	RMSE of validation set
5	0.278801	0.294922
6	0.273275	0.291627

7	0.274171	0.301726
8	0.269271	0.300891
9	0.26442	0.303078
10	0.262658	0.312805
11	0.262969	0.305313
12	0.261951	0.308417
13	0.262458	0.30628
14	0.2619	0.307694
15	0.260215	0.309502
16	0.257545	0.314255
17	0.256569	0.313088
18	0.256488	0.309465
19	0.255117	0.312607
20	0.251462	0.310392
21	0.250305	0.31659
22	0.248438	0.321995
23	0.247626	0.324815
24	0.246318	0.320671
25	0.246467	0.318857
26	0.246557	0.316854
27	0.246714	0.316432
28	0.246511	0.319657
29	0.247273	0.325352
30	0.247053	0.327385
31	0.246993	0.327233
32	0.245607	0.327855
33	0.244859	0.329857
34	0.244297	0.32623
35	0.243134	0.321508
36	0.242341	0.314835

37	0.24303	0.31327
38	0.241233	0.312298
39	0.239206	0.312278
40	0.237406	0.31204
41	0.237151	0.316683
42	0.236871	0.316935
43	0.236674	0.314252
44	0.236664	0.311482
45	0.237205	0.309473
46	0.237249	0.310947
47	0.236962	0.311109
48	0.236743	0.311267
49	0.236475	0.311057
50	0.2365	0.315957

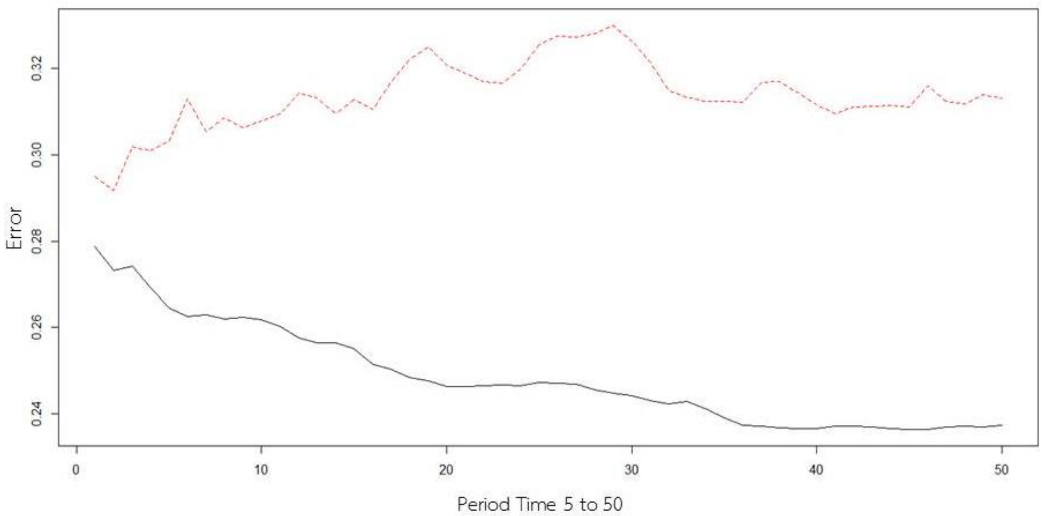


Figure 2: The RMSE between training set and validation set of the SVR model with turning point with impact γ =1.01.

Table 3: Experimental data of the GP model, with turning point with impact γ =1.01 and the feature vector consisting of the past 5-50 prices.

Backward window size	RMSE of training set	RMSE of validation set
5	0.6795	0.276991
6	0.6705	0.276657
7	0.667611	0.282785
8	0.662331	0.283952
9	0.659681	0.287525
10	0.658222	0.295396
11	0.654984	0.292377
12	0.652283	0.294025
13	0.646713	0.292994
14	0.647538	0.293393
15	0.644309	0.294096
16	0.642364	0.295582
17	0.640623	0.294728
18	0.638291	0.292849
19	0.635188	0.293502
20	0.633893	0.294276
21	0.631968	0.297767
22	0.633023	0.301953
23	0.630788	0.303378
24	0.628544	0.300544
25	0.628952	0.299408
26	0.626141	0.298261
27	0.62499	0.29857
28	0.625514	0.300067
29	0.620997	0.302856
30	0.61812	0.305487

31	0.618374	0.305365
32	0.614926	0.306625
33	0.613322	0.308084
34	0.612614	0.30677
35	0.610382	0.305247
36	0.608633	0.302126
37	0.605564	0.301193
38	0.602467	0.301668
39	0.598525	0.302044
40	0.596505	0.303044
41	0.596761	0.305395
42	0.59419	0.306276
43	0.59234	0.305261
44	0.590558	0.304735
45	0.588782	0.303427
46	0.586207	0.304102
47	0.583582	0.304804
48	0.583856	0.303934
49	0.582449	0.30383
50	0.579205	0.306115

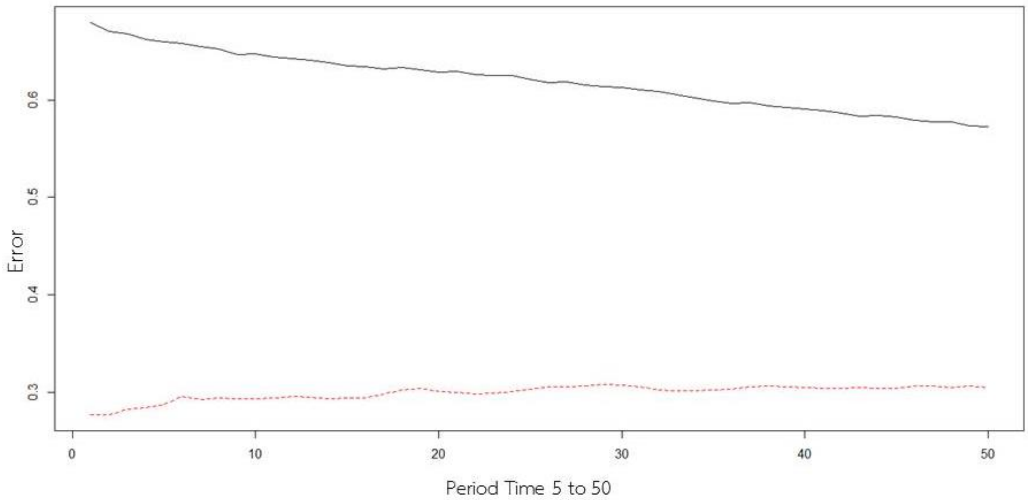


Figure 3: The RMSE between training set and validation set of the GP model with turning point with impact γ =1.01.

VITA

Mr. Danuphon Tonking was born in May 10, 1991, in Bangkok He received a bachelor degree in Applied Mathematics from Department of Mathematics, Faculty of science, king mongkut institute of technology Ladkrabang, Thailand 2012.