Abstract

This study provides a principal component analysis-fuzzy-support vector regression model for stock price prediction. Stocks with similar historical trends are selected using principal component analysis. Fuzzy information granulation is performed to construct a probability density for stock prices. Support vector regression is implemented to generate a regression function for future price prediction. This method suits for any sample size with any noise distribution type and eliminates the complicated fine tuning process compared with other Neural Network procedures. Besides, the use of fuzzy information granulation extends the prediction output form from a point to an interval with a certain probability assigned.

Keywords: Support vector regression, Fuzzy information granulation, Principal component analysis
# Contents

1 Introduction  

2 Principal Component Analysis  
   2.1 Principal Component Analysis  
   2.2 PCA correlation coefficients similarity search method  

3 Statistical learning theory  
   3.1 Empirical Risk Minimization Principle  
   3.2 Vapnik-Chervonenkis dimension  
   3.3 Structural Risk Minimization Principle  

4 Support Vector Regression  
   4.1 Loss Functions  
   4.2 Linear Regression  
   4.3 Non Linear Regression  
   4.4 Kernel Functions  
   4.5 Parameter Selection-Cross Validation  

5 Fuzzy Support Vector Regression  
   5.1 Fuzzy Sets and Fuzzy Membership Functions  
   5.2 Information Granulation  
      5.2.1 Basic Settings  
      5.2.2 W.Pedrycz Fuzzy Information Granulation Method  
   5.3 Fuzzy Support Vector Regression  

6 Application to Financial Forecasting  
   6.1 Research Data  
      6.1.1 Missing Data  
      6.1.2 Daily Return Conversion  
      6.1.3 Initial Public Offering  
   6.2 Stock Similarity Search  
      6.2.1 Principal Component Analysis Result  
      6.2.2 PCA Correlation Coefficient Similarity Search Result  
   6.3 Information Granulation Result  


Acknowledgements

Foremost, I would like to express my sincere gratitude to my supervisor Prof. dr. P.J.C. Spreij for continuous support of my research, for his patience, motivation, enthusiasm and immense knowledge. I am heartily thankful to my best friend Yang Li for helping get through the difficult times and all valuable recommendations to this thesis. I am grateful to all housemates in Botterweg: Ninghang Hu, Yu Tian, Zhenhao Li, Qingwen Chen and Fan Wang, for all the support and willingness to help me every time I needed. Besides, I would like to thank my parents and grandparents for supporting me spiritually throughout my life. Lastly, I would like to thank and offer my regards to all of those who supported me in any respect during the completion of my master programme.
Chapter 1

Introduction

Stock price prediction is regarded as a challenging task of financial time series prediction in the sense that a successful prediction of a stock’s future price could yield significant profit. Methodologies for stock price prediction vary while they are usually within three categories which can and often do overlap: fundamental analysis, technical analysis and technological methods.

Fundamental analysis [1] is concerned with the company that underlies the stock itself. A company’s past performance as well as the credibility of its accounts are evaluated. Many performance ratios [20] are created that aid the fundamental analyst with assessing the validity of a stock, such as the P/E ratio [16].

Technical analysis [15] is not concerned with any of the company’s fundamentals. It seeks to determine the future price of a stock based solely on the trends of the past price. Numerous patterns [14] are employed such as the head and shoulders. Alongside the patterns, statistical techniques are utilised such as the exponential moving average (EMA).

Technological methods are generated with the advent of the digital computer. It can be thought of as mathematical function approximation. In this study, we concern with technological methods. The goal of prediction is to generate a future stock price approximation function $f(x)$ based on the known information $x$, where $x$ can be any factors (both fundamental and technical) that influence stock prices. Evidence that daily stock returns are nonlinear [8] and so many options for $x$ makes prediction complicated and difficult.

In many studies, historical stock prices are selected as $x$ and different kinds of models are constructed for stock price prediction. French, K.R., Schwert, G.W. and Stambaugh, R.F. [6] provided a GARCH model for stock price forecasting. Awartani, B. and Corradi, V. [2] apply GARCH models to the S&P 500 index. However, the GARCH model is based on two assumptions: (1) sample size is sufficiently large (2) noise is normal distributed, which may not be satisfied in specific real situations. Neural

In all previous mentioned studies, there are two shortcomings which are worth paying attention to. Firstly, stock prices (returns) are measured based on its own lagged prices (returns). However, it has been found that there exists correlation between stocks due to cooperation or a competition relationship between companies and different companies may have different reaction time lags toward a shocking news in the market. Therefore, movement of one stock price may influence or indicate a relatively similar stock price movement in another stock. Hence interrelationships among different stocks should be considered and well addressed. Secondly, the prediction result is a price point which is not realistic in the stock market. It would be more meaningful if a model can provide an prediction interval with a probability distribution assigned on it that can indicate which price is most likely to be or be fluctuated around in the future.

In order to solve the above two problems using support vector regression (SVR) in stock price forecasting, this paper provides a principle component analysis (PCA)-fuzzy-SVR methodology. In chapter 2 a PCA coefficient similarity search method is described to select stocks with similar historical price trend. Chapter 3 presents a brief introduction to statistical learning theory which forms the base of SVR. Chapter 4 provides both a linear and a nonlinear SVR model and then a parameter selection method-cross validation is introduced. Chapter 5 extends a regular SVR model by using fuzzy information granulation(IG) in order to extract interrelationship information from the selected stocks and construct a probability density on predicted interval. In Chapter 6, an application of generated PCA-Fuzzy-SVR model in China’s stock market is presented.
Chapter 2

Principal Component Analysis

Principal component analysis (PCA) is a mathematical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of uncorrelated variables, the so-called principal components [10]. PCA is concerned with explaining the variance-covariance structure of a set of variables through a few linear combinations of these variables. Its general objectives are data reduction and interpretation. In this chapter, the basic idea of PCA, mainly based on Johnson, and Wichern [10] will be introduced first, then a similarity search method based on PCA correlation coefficients will be provided.

2.1 Principal Component Analysis

Let the random vector \( R' = [R_1, R_2, ..., R_p] \), the returns of \( p \) stocks, have the covariance matrix \( \Sigma \) with eigenvalues \( \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0 \). Consider the linear combinations

\[
\begin{align*}
Y_1 &= a'_1 R = a_{11} R_1 + a_{12} R_2 + \cdots + a_{1p} R_p \\
Y_2 &= a'_2 R = a_{21} R_1 + a_{22} R_2 + \cdots + a_{2p} R_p \\
&\vdots \\
Y_p &= a'_p R = a_{p1} R_1 + a_{p2} R_2 + \cdots + a_{pp} R_p.
\end{align*}
\]

(2.1)

Then

\[
\begin{align*}
Var(Y_i) &= a'_i \Sigma a_i \\
Cov(Y_i, Y_k) &= a'_i \Sigma a_k.
\end{align*}
\]

(2.2) (2.3)

The principal components are those uncorrelated linear combinations \( Y_1, Y_2, ..., Y_p \) whose variances are as large as possible.
The first principal component is the linear combination with maximum variance. That is, it maximizes $\text{Var}(Y_1) = a_1' \Sigma a_1$. It is clear that $\text{Var}(Y_1) = a_1' \Sigma a_1$ can be increased by multiplying $a_1$ by some constant. To eliminate this indeterminacy, it is convenient to restrict attention to coefficient vectors of unit length. We therefore define

First principal component = linear combination $a_1' R$
that maximizes $\text{Var}(a_1' R)$
such that $a_1' a_1 = 1$.

In general, for $i>1$,

The $i$th principal component = linear combination $a_i' R$
that maximizes $\text{Var}(a_i' R)$
such that $a_i' a_i = 1$ and $\text{Cov}(a_i' R, a_k' R) = 0$ for $k < i$.

The following property gives an explicit expression for each principle component.

**Property 2.1.** Let $\Sigma$ be the covariance matrix associated with the random vector $R' = [R_1, R_2, \ldots, R_p]$. Let $\Sigma$ have the eigenvalue-eigenvector pairs $(\lambda_1, e_1), (\lambda_2, e_2), \ldots, (\lambda_p, e_p)$ where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p \geq 0$. Then the $i$th principal component is given by

$$Y_i = e_i' R = e_{i1} R_1 + e_{i2} R_2 + \ldots + e_{ip} R_p, \text{ for } i=1,2,\ldots,p$$

(2.4)

with these choices,

$$\text{Var}(Y_i) = e_i' \Sigma e_i = \lambda_i \text{ for } i=1,2,\ldots,p$$

(2.5)

$$\text{Cov}(Y_i, Y_k) = e_i' \Sigma e_k = 0 \text{ for } i \neq k.$$  

(2.6)

Taking into account that each principal component is the linear combination of the $p$ stocks’ returns, the following property defines the PCA correlation coefficient which indicates how strong the relationship of the $i$th principal and $k$th stock return is.

**Property 2.2.** If $Y_1 = e_1' R, Y_2 = e_2' R, \ldots, Y_p = e_p' R$ are the principal components obtained from the covariance matrix $\Sigma$, then

$$\rho(Y_i, R_k) = \frac{e_{ik} \sqrt{\lambda_i}}{\sqrt{\sigma_{kk}}} \text{ for } i,k=1,2,\ldots,p$$

(2.7)

are the correlation coefficients between the components $Y_i$ and the variables $R_k$. Here $(\lambda_1, e_1), (\lambda_2, e_2), \ldots, (\lambda_p, e_p)$ are the eigenvalue-eigenvector pairs for $\Sigma$, $\sigma_{kk}$ is the variance of $k$th stock return.
2.2 PCA correlation coefficients similarity search method

Usually in a stock market, the first principal component is the market factor while other components sometimes can be interpreted from fundamental perspective, for instance, geographical factor, profitability factor and etcetera. While some principal components may not be easily interpreted, however, they do contain some information which contributes to the stock volatility to some extent. Taking into account that the correlation coefficient \( \rho(Y_i, R_k) \) indicates the relationship of the \( i \)th stock return to the \( k \)th principal component and each principal component reflects a class of independent indicators, thus the stocks with a similar historical trend may possess similar PCA correlation coefficient to each principal component. Based on such an idea, a time series similarity search method based on PCA correlation coefficients will be proposed in the following.

Taking the \( j \)th stock as the target variable, define the similarity coefficient of the \( i \)th stock and \( j \)th stock with respect to \( k \)th principal component as

\[
S_{Y_k}(R_i, R_j) = \frac{\rho(Y_k, R_i)}{\rho(Y_k, R_j)}
\]  

which measures how similar these two stocks are with respect to the \( k \)th principal component. The more \( S_{Y_k}(R_i, R_j) \) is close to 1, the more similar these two stocks are.

Moreover, each principal component contains a certain portion of comprehensive information. Taking into account such portion, we define the weighted similarity coefficient \( S(R_i, R_j) \) as

\[
S(R_i, R_j) = \frac{\sum_{k=1}^{q} S_{Y_k}(R_i, R_j) \times \lambda_k}{\sum_{k=1}^{q} \lambda_k} = \frac{\sum_{k=1}^{q} S_{Y_k}(R_i, R_j) \times \lambda_k}{\sum_{k=1}^{q} \lambda_k} \times \frac{\sum_{k=1}^{q} \lambda_k}{\sum_{k=1}^{q} \lambda_k}
\]  

(2.9)

where \( \lambda_1, \lambda_2, ..., \lambda_q \) are the first \( q \) largest eigenvalues which explain 80% of total variance. \( S(R_i, R_j) \) measures the similarity based on the first \( q \) principal components’ impact. The more \( S(R_i, R_j) \) is close to 1, the more similar these two stocks are.

Taking all above into account, we present the following algorithm in order to search for stocks with similar historical trend while taking \( j \)th stock as target.

Algorithm 2.1

1. Set the target stock, for instance the \( j \)th stock.
2. Find first \( q \) principal components \( \{Y_i\}_{i=1,...,q} \) which explain 80% of total
variance.
3. Pick the principal components \{Y_i\} such that the correlation coefficient 
\(\rho(Y_i, R_j)\) is larger than 0.05.
4. Calculate \(S(R_i, R_j)\) based on selected principal components \{Y_i\} accord-
ing to step 3 and select all \(R_i\) which satisfies \(S(R_i, R_j) \in [0.85, 1.15]\).
where 80%, 0.05 and [0.85,1.15] are adjustable case by case.

Therefore, by implementing Algorithm 2.1, several stocks which possess
similar historical returns compared with the target stock are selected from
the database. The price information of these stocks will be used to perform
fuzzy information granulation which will be introduced in Chapter 5.
Chapter 3

Statistical learning theory

This chapter contains a brief introduction to statistical learning theory. The aim is to generate and provide a structure risk minimization principle which forms the basic idea of support vector regression introduced in the next chapter. This chapter is mainly based on Vapnik [24].

The general model of learning can be described through three parts:
1. Input vector \( x \in \mathbb{R}^n \) (the variables of stock price).
2. Value \( y \), the real stock price.
3. A learning machine capable of implementing a set of functions \( \{ f(x, \alpha), \alpha \in \Lambda \} \), where \( \Lambda \) is a set of parameters.

The problem of learning is to select from the given set of functions \( \{ f(x, \alpha), \alpha \in \Lambda \} \), the one that best approximates the real value of stock price \( y \).

3.1 Empirical Risk Minimization Principle

In order to choose the best available approximation to the real stock price, one measures the loss \( L(y, f(x, \alpha)) \) between real stock price \( y \) and output \( f(x, \alpha) \) to a given input \( x \) provided by the learning machine.

Consider the expected value of the loss, given by the risk function

\[
R(\alpha) = \int L(y, f(x, \alpha))dF(x, y)
\]  

(3.1)

where \( F(x, y) \) is unknown.

The goal is to find the function \( f(x, \alpha_0) \) that minimizes the risk function \( R(\alpha) \) over the class of functions \( \{ f(x, \alpha), \alpha \in \Lambda \} \). To apply them to specific problems, one has to specify the corresponding loss functions (see section 4.1). For instance, we may take the quadratic loss function \( L = (y - f(x, \alpha))^2 \), which forms the least-squares method.
Given observed data set \((x_i, y_i), i = 1, ..., l\), the expected value of loss (3.1) is not able to be computed directly due to unknown distribution \(F(x, y)\). The following inductive principle can be applied in order to minimize the risk function (3.1) with unknown \(F(x, y)\).

**Empirical Risk Minimization (ERM) Principle**

The risk function (3.1) is replaced by the so-called empirical risk function

\[
R_{\text{emp}}(\alpha) = \frac{1}{l} \sum_{i=1}^{l} L(y_i, f(x_i, \alpha))
\] (3.2)

constructed on the basis of the training set \((x_i, y_i), i = 1, ..., l\). One approximates the function \(f(x, \alpha_0)\) that minimizes the risk (3.1) by the function \(f(x, \alpha_l)\) minimizing the empirical risk (3.2).

In learning theory the ERM principle is quite general and plays a crucial role. Note that the ERM principle makes sense only if,

\[
\lim_{l \to \infty} R_{\text{emp}}(\alpha) = R(\alpha)
\] (3.3)

which is based on the law of large numbers. Therefore, an efficient large sample size is required. However, this is not always the case. When the sample size is relatively small, fortunately, the theory of uniform convergence in probability provides a bound on the deviation of the empirical risk (3.2) from the expected risk (3.1). The following property provides an upper bound valid with probability at least \(1 - \delta\).

**Property 3.1.** Uniform Vapnick-Chervonenkis (VC) bound

\[
R(\alpha) \leq R_{\text{emp}}(\alpha) + \sqrt{\frac{h \ln\left(\frac{2l}{h} + 1\right)}{l}} - \ln \frac{\delta}{4}
\] (3.4)

holds with probability \(1 - \delta\), where \(h\) is the Vapnik-Chervonenkis dimension of \(f(x, \alpha)\) (see section 3.2), and \(l\) is the number of examples.

The first term in the right-hand side of inequality (3.4) is the empirical risk and the second term is referred as confidence interval.

### 3.2 Vapnik-Chervonenkis dimension

In statistical learning theory, the Vapnik-Chervonenkis dimension is a scalar value that measures the capacity of a set of functions. It is a core concept in Vapnik-Chervonenkis theory. The capacity is related to how rich a set of functions can be. For instance, consider a regression problem using polynomial functions, a high-degree polynomial can be wiggly thus it can fit a
given set of training points well. However, one can expect that the regression function will provide larger error in regressing new data because it is too wiggly. Such a polynomial has a high capacity and is likely to cause an overfitting problem since it is more accurate in fitting known data but less accurate in predicting new data. Our goal is to make a balance and Structural Risk Minimization (SRM) Principle provides such a method. Below we will give the definition of the VC dimension first and then introduce SRM principle.

**Definition 3.2.** A classification model $f$ with some parameter vector $\alpha$ is said to shatter a set of data points if, for all assignments of labels to those points $(x_1, y_1), ..., (x_n, y_n)$, there exists a $\alpha$ such that the model $f$ makes no errors when evaluating that set of data points.

**Definition 3.3.** The VC dimension of a set of indicator functions $\{Q(z, \alpha), \alpha \in \Lambda\}$, is the maximum number $h$ of vectors $(x_1, y_1), ..., (x_h, y_h)$ that can be separated into two classes in all $2^h$ possible ways using functions of the set (i.e., the maximum number of vectors that can be shattered by the set of functions).

**Example 3.4**

The VC dimension of the set of linear indicator functions

$$Q(z, \alpha) = \theta(\sum_{i=1}^{n} \alpha_i z_i + \alpha_0)$$

where

$$\theta(z) = \begin{cases} 0 & \text{if } z < 0 \\ 1 & \text{if } z \geq 0 \end{cases}$$

in $n$-dimensional coordinate space $Z = (z_1, ..., z_n)$ is equal to $h = n + 1$, since by using functions of this set one can shatter at most $n + 1$ vectors (Figure 3.1).

**Definition 3.5.** Let $A \leq Q(z, \alpha) \leq B, \alpha \in \Lambda$, be a set of real functions bounded by constants $A$ and $B$. Consider along with the set of real functions $Q(z, \alpha), \alpha \in \Lambda$, the set of indicators

$$I(z, \alpha, \beta) = \theta(Q(z, \alpha) - \beta), \alpha \in \Lambda, \beta \in (A, B)$$

where $\theta(z)$ is the step function as in (3.6).

The VC dimension of a set of real functions $Q(z, \alpha), \alpha \in \Lambda$ is defined to be the VC dimension of the set of corresponding indicators (3.7) with parameters $\alpha \in \Lambda, \beta \in (A, B)$.  

12
Figure 3.1: The VC dimension of the lines in the plane is equal to 3, since they can shatter three vectors, but not four: the vectors $z_2, z_4$ cannot be separated by a line from the vectors $z_1, z_3$

Example 3.6
The VC dimension of the set of linear functions

$$Q(z, \alpha) = \sum_{i=1}^{n} \alpha_i z_i + \alpha_0,$$

with $\alpha_0, ..., \alpha_n \in (-\infty, \infty)$ (3.8)

in $n$-dimensional coordinate space $Z = (z_1, ..., z_n)$ is equal to $h = n + 1$, because the VC dimension of the corresponding linear indicator functions is equal to $n + 1$.

3.3 Structural Risk Minimization Principle

From the inequality (3.4) one can see that a small value of the empirical risk does not necessarily imply a small value of expected risk. The following structural risk minimization (SRM) principle is intended to minimize the right-hand side of inequity (3.4) simultaneously over both terms. It is an inductive principle for model selection and provides a trade off between the VC dimension of regression functions (the hypothesis space complexity) and empirical risk (the quality of fitting the training data). The SRM principle is based on the observation that, in order to make the expected risk small, the confidence interval and empirical risk should be minimized at the same time.

The procedure is outlined below (Figure 3.2).
1. Using a priori knowledge of the domain, create a structure such that $S_h$ is a hypothesis space of VC dimension $h$. Choose a class of functions, such as polynomials of degree $h$.
2. Divide the class of functions into a hierarchy of nested subsets $S_1 \subset S_2 \subset$
... $\subset S_n$ in order of increasing VC dimension. For example, polynomials of increasing degree.

3. Perform empirical risk minimization on each subset $S_i, i = 1, ..., n$.

4. Select the model in the series whose sum of empirical risk and VC confidence is minimal.

Figure 3.2: The bound on the risk is the sum of the empirical risk and the confidence interval. The empirical risk decreases as the complexity of structure increases, while the confidence interval increases. The smallest bound of the risk is achieved on some appropriate complexity of structure $S^*$ [24].
Chapter 4

Support Vector Regression

The concept of Support Vector Machine (SVM), proposed by Vapnik (1995) is based on statistical learning theory. It was originally employed to solve a classification problem and soon has been extended to the domain of regression problems (Vapnik, 1997). The method used for regression is called Support Vector Regression (SVR) and is gaining popularity, due to many attractive features and promising empirical performance in widely used applications.

In this chapter, we first introduce a loss function, then explain the linear SVR method and extend to nonlinear case. Later kernel functions will be discussed and finally, the parameter selection method-Cross Validation will be introduced. This chapter is mainly based on Vapnik [24], Chang & Lin [4] and Refaeilzadeh,Tang& Liu [21].

4.1 Loss Functions

As mentioned in chapter 3, to describe the problem of selecting a real-valued function from \( \{f(x, \alpha), \alpha \in \Lambda\} \), a specific loss function should be chosen, for example, a quadratic loss function

\[
L(y, f(x, \alpha)) = (y - f(x, \alpha))^2
\]

In statistics, classical methods rely heavily on assumptions which are often not met in practice. In particular, it is often assumed that the data errors are normally distributed, at least approximately such that the central limit theorem can be relied on to produce normally distributed estimates. Unfortunately, when there are outliers in the data, classical methods often have very poor performance. Robust statistics seeks to provide methods that emulate popular statistical methods, but which are not unduly affected by outliers or other small departures from model assumptions.
To construct an SVR for real-valued functions which is robust, we use a new type of loss functions, the so-called $\varepsilon$-insensitive loss functions. These loss functions describe the $\varepsilon$-insensitive model: The loss is equal to 0 if the discrepancy between the predicted and the observed values is less than $\varepsilon$.

Below we will introduce two loss functions of this family.

**Linear $\varepsilon$-insensitive loss function**

\[
L(y, f(x, \alpha)) = |\xi|_\varepsilon = \begin{cases} 
0 & \text{if } |\xi| \leq \varepsilon \\
|\xi| - \varepsilon & \text{if } |\xi| > \varepsilon 
\end{cases}
\]  

(4.2)

where $\xi = y - f(x, \alpha)$. It coincides with the loss function (4.1) if $\varepsilon = 0$

**Quadratic $\varepsilon$-insensitive loss function**

\[
L(y, f(x, \alpha)) = |\xi|^2_\varepsilon = \begin{cases} 
0 & \text{if } |\xi| \leq \varepsilon \\
(|\xi| - \varepsilon)^2 & \text{if } |\xi| > \varepsilon 
\end{cases}
\]  

(4.3)

where $\xi = y - f(x, \alpha)$. It coincides with the loss function (4.1) if $\varepsilon = 0$

4.2 Linear Regression

Support vector regression (SVR) is a statistical method for creating regression functions of arbitrary type from a set of training data. Let $y$ be the real stock price, $\{f(x, \alpha), \alpha \in \Lambda\}$ be a set of real functions that contains the regression function. During the learning process, the learning machine observes pairs $(x, y)$ (the training set). After training, the machine must on any given $x$ return a value $f(x, \alpha)$. The goal is to return a value $f(x, \alpha_0)$
that is close to \( y \), where \( \alpha_0 \) is chosen as the minimizer of the loss function. Linear SVR takes place if:

(i) One estimates the regression in the set of linear functions

\[
f(x, \alpha) = \omega^T x + b
\]

where \( \alpha = (\omega, b) \).

(ii) One defines the problem of regression estimation as the problem of risk minimization with respect to a specific loss function for instance linear \( \varepsilon \)-insensitive loss function.

(iii) One minimizes the risk using the SRM principle, where elements of the structure \( S_n \) are defined by the inequality

\[
\omega^T \omega \leq c_n.
\]

Consider the problem of approximating the set of data,

\[
\mathcal{D} = \{(x_1, y_1), ..., (x_l, y_l)\}, x \in \mathbb{R}^n, y \in \mathbb{R}
\]

with a linear function,

\[
f(x) = \omega^T x + b
\]

where \( \omega \) and \( b \) are unknown parameters. By minimizing the risk using SRM principle, the standard form of Linear SVR is

\[
\min_{\omega, b, \xi, \xi^*} \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{l} \xi_i + C \sum_{i=1}^{l} \xi_i^*
\]

subject to

\[
y_i - \omega^T x_i - b \leq \varepsilon + \xi_i
\]
\[
\omega^T x_i + b - y_i \leq \varepsilon + \xi_i^*
\]
\[
\xi_i, \xi_i^* \geq 0, i = 1, ..., l.
\]

where, \( C \) is a pre-specified nonnegative value determining the trade off between the flatness of the \( f(x) \) and the amount up to which deviations larger than \( \varepsilon \) are tolerated and \( \xi, \xi^* \) are slack variables representing upper and lower constraints on the outputs of the system.

By constructing a Lagrange function, specifically,

\[
L = \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{l} \xi_i + C \sum_{i=1}^{l} \xi_i^* - \sum_{i=1}^{l} \alpha_i (\varepsilon + \xi_i - y_i + \omega^T x_i + b) - \sum_{i=1}^{l} \alpha_i^* (\varepsilon + \xi_i^* + y_i - \omega^T x_i - b) - \sum_{i=1}^{l} \beta_i \xi - \sum_{i=1}^{l} \beta_i^* \xi^*
\]
where \( \alpha_i, \alpha_i^*, \beta_i, \beta_i^* \geq 0 \). To compute the optimal solution, we have:

\[
\frac{\partial L}{\partial \omega} = \omega - \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)x_i = 0 \\
\frac{\partial L}{\partial b} = \omega - \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) = 0 \\
\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \beta_i = 0 \\
\frac{\partial L}{\partial \xi_i^*} = C - \alpha_i^* - \beta_i^* = 0
\]

(4.10)

Thus we obtain the Lagrange dual problem [3] as follows:

\[
\max_{\alpha_i, \alpha_i^*} \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)x_ix_j - \varepsilon \sum_{i=1}^{l} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \\
\text{subject to} \\
\sum_{i=1}^{l} (\alpha_i - \alpha_i^*) = 0 \\
\alpha_i, \alpha_i^* \in (0, C).
\]

(4.11)

Solving (4.10) for \( \omega \), we obtain

\[
\omega^* = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)x_i \tag{4.13}
\]

where \( \alpha_i, \alpha_i^* \) are the optimal solution of Lagrange dual problem (4.11). By (4.7), we obtain

\[
f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)x_ix + b^* \tag{4.14}
\]

The parameter \( b \) can be computed from the complementary slackness conditions. Specifically,

\[
\alpha_i(\varepsilon + \xi_i - y_i + \omega^*T x_i + b) = 0 \\
\alpha_i^*(\varepsilon + \xi_i^* + y_i - \omega^*T x_i - b) = 0 \\
(C - \alpha_i)\xi_i = 0 \\
(C - \alpha_i^*)\xi_i^* = 0
\]

(4.15)

Therefore \( b \) can be computed as follows.

\[
b^* = y_i - \omega^*T x_i - \varepsilon \text{ for } \alpha_i, \alpha_i^* \in (0, C). \tag{4.16}
\]
Figure 4.2: The support vector machine maps the input space into a high dimensional feature space and performs linear regression in the feature space.

4.3 Non Linear Regression

In the case where linear regression is inappropriate, the SVR can map the input vector $x$, into a high dimensional feature space via a nonlinear mapping and then linear regression is performed in this space (see Figure 4.2). By choosing a nonlinear mapping $\phi$ as a so called kernel function [24], non-linear regression problems can be transferred into a linear one using the so called kernel approach. There are some restrictions on such non-linear mappings that can be employed, but it turns out, that most commonly employed functions are acceptable (see section 4.4).

Suppose $f(x)$ takes the following form,

$$f(x) = \omega^T \phi(x) + b$$ (4.17)

The optimisation problem of (4.8) becomes

$$\min_{\omega, b, \xi, \xi^*} \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{l} \xi_i + C \sum_{i=1}^{l} \xi_i^*$$ (4.18)

subject to

$$\omega^T \phi(x_i) + b - y_i \leq \varepsilon + \xi_i$$
$$y_i - \omega^T \phi(x_i) - b \leq \varepsilon + \xi_i^*$$
$$\xi_i, \xi_i^* \geq 0, i = 1, ..., l.$$
The dual problem of (4.11) becomes

\[
\max_{\alpha_i, \alpha_i^*} \frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) \phi(x_i) \phi(x_j) \\
- \varepsilon \sum_{i=1}^{l} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i (\alpha_i - \alpha_i^*)
\]

subject to

\[
\sum_{i=1}^{l} (\alpha_i - \alpha_i^*) = 0 \quad \alpha_i, \alpha_i^* \in (0, C)
\]

(4.19)

The approximate function of (4.14) becomes:

\[
f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) K(x_i, x) + b
\]

(4.21)

where \(K(x_i, x) = \phi(x_i)^T \phi(x)\).

### 4.4 Kernel Functions

This section discusses the method that can be used to construct a mapping into a high dimensional feature space by the use of reproducing kernels. The idea of the kernel function is to enable operations to be performed in the input space rather than the potentially high dimensional feature space. Hence the inner product does not need to be evaluated in the feature space.

**Mercer’s Conditions** [24]

To guarantee that the symmetric function \(K(x, x')\) from \(L_2\) has an expansion

\[
K(x, x') = \sum_{k=1}^{\infty} a_k \phi_k(x) \phi_k(x')
\]

(4.22)

with positive coefficients \(a_k > 0\) (i.e., \(K(x, x')\) describes an inner product in some feature space), it is necessary and sufficient that the condition

\[
\int \int K(x, x') h(x) h(x') dx dx' > 0
\]

(4.23)

be valid for all \(h \neq 0\) for which

\[
\int h^2(x) dx < \infty.
\]

Below we will provide two kinds of kernel functions.

**Polynomial Kernel Function**

A polynomial mapping is a popular method for non-linear modelling

\[
K(x, x') = (x \cdot x')^d
\]

(4.24)
Radial Basis Function

\[ K(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2g^2}\right) \] (4.25)

Usually we have more than one kernel to choose, the question which is best for a particular problem arises. We could not say that one kernel outperforms the others, however, one can use more than one kernel function for a particular problem and make a comparison.

### 4.5 Parameter Selection-Cross Validation

Even if a particular kernel function is decided, it is well known that SVR estimation accuracy depends on a good setting of meta-parameters \( C \) in (4.8) and the kernel parameters. Existing software implementations of SVR usually treat SVR meta-parameters as user-defined inputs and kernel function parameters are usually based on application-domain knowledge. For instance, Radial basis function has a parameter \( g \) and one has to decide its value before the experiment. Note that a bad parameter choice can heavily influence regression performance. For instance if \( C \) is too large (infinity), then the objective is to minimize the empirical risk only, without regard to model the complexity part in the optimization formulation (4.8) which will lead to an overfitting problem.

Cross validation [22] might be the most reliable way for parameter selection, especially for small and middle size problems. It is a statistical method of evaluating and comparing learning algorithms by dividing data into two segments: one considered as the validation set and the rest for training the model. In typical cross validation, the training and validation sets must cross over in successive rounds such that each data point has a chance of being validated against. The basic form of cross validation is \( k \)-fold cross validation. The idea is that the data is first partitioned in to \( k \) equally(or nearly equally) sized folds. Subsequently \( k \) iterations of training and validation are performed such that within each iteration a different fold of the data is held-out for validation while the remaining \( k - 1 \) folds are used for learning. Figure 4.3 demonstrates an example with \( k = 3 \). The performance of each learning algorithm on each fold can be tracked using an accuracy measure such as mean square error. Upon completion, \( k \) samples of the mean square error will be available for each algorithm. Different methodologies such as averaging can be used to obtain a measure to show which algorithm is superior to the others.

Taking all above into account, we present the following algorithm for parameter selection process,
Figure 4.3: 3-fold cross validation. The darker section of the data are used for training while the lighter sections are used for validation [21].

Algorithm 4.1

1. The training data is separated to $k$ folds.
2. Provide a possible interval of parameters $C$ and $g$ with the grid space.
3. Each grid point (pair of $C$ and $g$) are tried to implement $k$-fold cross validation and average mean square error is generated for each pair of $C$ and $g$.
4. Select the pair $C$ and $g$ which gives the lowest average mean square error as the best parameters.

Therefore, by implementing Algorithm 4.1, a pair of parameters $C$ and $g$ are selected as the best parameters in a support vector regression model. Then they are used to train the whole training set and generate the final model.
Chapter 5
Fuzzy Support Vector Regression

As the stock price can be predicted using the SVR method provided in Chapter 4, the problem arises that the predicted price is only a point compared with the reality that the price fluctuates within an interval. Thus very precise prediction seems not realistic and is not required for such practical problem. In this chapter, a fuzzy SVR method based on fuzzy information granulation is provided to generate a price fluctuation interval to which a probability is assigned.

5.1 Fuzzy Sets and Fuzzy Membership Functions

Since the conceptual framework for dealing with information granularity bears a close relation to that of fuzzy sets and possibility distributions, we shall begin our exposition with a brief introduction.

Definition 5.1 [27]. A fuzzy set $G$ defined in $X$ is a set whose elements have degrees of membership defined by its membership function $A: X \rightarrow [0, 1]$ where $A(x)$ denotes a degree of membership of $x$ in $G$.

There are several forms of fuzzy membership function such as triangle form, Gaussian form, trapezoidal form and parabolic form. Figure 5.1 is an example of triangle fuzzy membership function of the form

$$A(x, a, m, b) = \begin{cases} 
0 & \text{if } x < a \\
\frac{x-a}{m-a} & \text{if } a \leq x \leq m \\
\frac{b-x}{b-m} & \text{if } m < x \leq b \\
0 & \text{if } x > b 
\end{cases} \quad (5.1)$$
with parameters $a = 0.2$, $m = 0.6$, $b = 0.8$

Since the fuzzy membership function has range $[0,1]$, after normalizing it can be viewed as a probability density. For instance, in Figure 5.1, suppose on the horizontal axis is the stock price, we can see that the stock price can take value in $(0.2,0.8)$ and $0.6$ is with the highest probability. In another world, the future price is going to be fluctuated within the interval $(0.2, 0.8)$ around $0.6$.

Thus, in order to extend our prediction output to an interval with probability assigned, our goal is to find three parameters $a, m, b$ in such that the triangle fuzzy membership function provides upper and lower bounds of the interval together with an triangle density on it. In next section, a information granulation method will be introduced to figure out these three parameters.

### 5.2 Information Granulation

#### 5.2.1 Basic Settings

Fuzzy information granulation (IG) proposed by Lotfi A. Zadeh [26] is one of the basic tasks in designing fuzzy models for solving predictive tasks. The goal is to decompose the whole information sequence into separate parts. Each part is said to be granular in the sense that the data points within a granule have to be dealt with as a whole, rather than individually because they are drawn together by indistinguishability, similarity, proximity or functionality. A general framework of information granulation is primarily based on three kinds of theory: fuzzy set theory [27], rough set theory [17] and entropy space theory [13]. In this study we will describe a fuzzy information granulation methodology.
Definition 5.2 [26]. Let $X$ be a variable taking values in $U$ and let $G$ be a fuzzy subset of $U$. (Usually, but not universally, $U = \mathbb{R}^n$, and $G$ is a convex fuzzy subset of $U$.) A fuzzy granule, $g$, in $U$ is induced (or characterized) by a proposition of the form

$$g \triangleq X \text{ is } G \text{ is } \lambda$$

where $\lambda$ is a fuzzy probability which is characterized by a possibility distribution over the unit interval.

For instance,

$$g \triangleq X \text{ is small is unlikely}$$

Essentially, in the case of stock price prediction, the goal of information granulation is to generate the fuzzy membership function $A$ which can reasonably describe the probability that a stock will be priced in a possible price interval $G$. For example,

$$g \triangleq \text{(stock price } X) \text{ is (going to be in } G) \text{ is (with probability } \lambda)$$

To generate such fuzzy membership function, there are two steps: first to divide the whole time series into separate time windows and next manipulate fuzzy information granulation. Below we will introduce W.Pedrycz’ fuzzy information granulation method.

5.2.2 W.Pedrycz Fuzzy Information Granulation Method

For given time series, consider the case with a single time window first, the basic idea of W. Pedrycz [19] is that the granular should satisfying two conditions:

1. Fuzzy granular should reasonably explain original data.
2. Fuzzy granular should be specific to some extent.

In order to meet above requirements, consider a function with respect to a triangle fuzzy membership function $A(a, m, b)$

$$Q_A = \frac{N_A}{M_A},$$

where $M_A$ satisfies condition 1 and $N_A$ satisfies condition 2. Specifically, we can take

$$M_A = \sum_{x \in X} A(x)$$

$$N_A = measure(support(A))$$
where $X$ is the set of historical prices, $\text{Support}(A) = \{ x \in X | A(x) > 0 \}$ and $\text{measure}(A) = \text{sup}(A) - \text{inf}(A)$

Thus $Q_A$ is a function of parameters $a, m$ and $b$

$$Q_A = \frac{\sum_{x \in X} A(x)}{\text{measure(\text{support}(A))}} = Q(a, m, b) \quad (5.6)$$

and by W.Pedrycz’ method [18], the goal is to maximize $Q_A$. Therefore, $a, m$ and $b$ are figured out by solving an optimization problem

$$\max_{a,b,m} Q_A \quad (5.7)$$

subject to

$$0 \leq a \leq m \leq b. \quad (5.8)$$

When facing multiple time windows case, we may perform W.Pedrycz method on each single window and then a sequence of fuzzy membership function will be generated.

### 5.3 Fuzzy Support Vector Regression

Recall that in Chapter 2, the PCA correlation coefficient search method provides a search result of several similar stocks with respect to the target stock, providing additional information which can be used in the process of fuzzy information granulation. The information granulation objective is to split the whole time series sequence into three sequences generated by W.Pedrycz method, R-the price with highest probability, Low- the lower bound of stock price and Up- the upper bound of the stock price, which correspond $m, a$ and $b$ in Figure 5.1. Then, support vector regression is performed separately on R, Low and Up sequence. The algorithm is as follows,
Figure 5.2: PCA-Fuzzy-SVR method outline
Chapter 6

Application to Financial Forecasting

In this chapter we use the method described in Chapter 5 to predict stock price given historical data of China’s stock market. The models have been implemented in the MATLAB environment.

6.1 Research Data

The data used in this study consist of average daily stock prices (6.1) of 64 stocks, in terms of local currency. These data are obtained from the transport sector of Shanghai Stock Exchange and Shenzhen Stock Exchange in China for the period Jan 4, 2010 through March 31, 2011, generating a total of 64 × 300 observations. All the data have been adjusted for dividends and stock splits. We use

\[ AP(i,t) = \frac{P_{\text{open}}(i,t) + P_{\text{close}}(i,t) + P_{\text{min}}(i,t) + P_{\text{max}}(i,t)}{4} \] (6.1)

where

- \( AP(i,t) \): average daily stock price of ith stock at time t
- \( P_{\text{open}}(i,t) \): price of the ith stock at the open of the markets at time t
- \( P_{\text{close}}(i,t) \): price of the ith stock at the close of the markets at time t
- \( P_{\text{min}}(i,t) \): minimum price of the day of the ith stock at time t
- \( P_{\text{max}}(i,t) \): maximum price of the day of the ith stock at time t

6.1.1 Missing Data

When multiple-day returns arise as a result of weekends or holidays or stock suspension because of for instance a general meeting of shareholders, an important announcement and abnormal fluctuations, the observations are
estimated by mean of interpolation:

\[ P(i, t + k) = P(i, t) + \frac{k(P(i, t + j) - P(i, t))}{j}, k \in [1, j - 1]. \tag{6.2} \]

### 6.1.2 Daily Return Conversion

For precision and convenience, stock prices are converted into daily stock returns in order to eliminate the scale influence derived from high price stocks. Define the return of ith stock during the period \([t-1, t]\) as

\[ r(i, t) = \frac{P(i, t) - P(i, t - 1)}{P(i, t - 1)} = \frac{P(i, t)}{P(i, t - 1)} - 1. \tag{6.3} \]

### 6.1.3 Initial Public Offering

**Definition.** An initial public offering (IPO) is when a company issues common stock or shares to the public for the first time.

For the stocks who had its IPO after Jan 4, 2010 result in a lack of data prior to its IPO date. Average returns of other stocks are used to estimate the daily returns and to restore the stock prices.

### 6.2 Stock Similarity Search

#### 6.2.1 Principal Component Analysis Result

Taking each stock as a random variable, we implemented the principal component analysis. The result is shown in Table 6.1. The first column is the serial number of principal component. Column 2 states the eigenvalue sorted by descending values. Column 3 describes the proportion of each principal component’s variance to the total variance. Column 4 indicates the cumulative proportion of total variance that can be explained. We can see that the first 20 principal components explain more than 80% of the total variance. Thus we selected the first 20 principal components in the sense that they characterize the stock return in the transportation sector.

Randomly choose a stock as the target stock, for instance here stock SZ600009 is chosen. Taking into account that the former chosen 20 potential principal components may not have significant influence to this specific stock. Therefore, by cutting those components whose correlation coefficient \(\rho\) is too small (absolute value less than 0.05), a set of principal components are selected in order to perform PCA correlation coefficient similarity search. The selection result is given in Table 6.2, 6th, 10th, 11th and 14th principal components will be removed.
<table>
<thead>
<tr>
<th>No.</th>
<th>Eigenvalue</th>
<th>Proportion</th>
<th>Cumulative</th>
<th>No.</th>
<th>Eigenvalue</th>
<th>Proportion</th>
<th>Cumulative</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>31.10</td>
<td>48.60</td>
<td>48.60</td>
<td>33</td>
<td>0.38</td>
<td>0.60</td>
<td>90.27</td>
</tr>
<tr>
<td>2</td>
<td>2.95</td>
<td>4.62</td>
<td>53.22</td>
<td>34</td>
<td>0.37</td>
<td>0.58</td>
<td>90.85</td>
</tr>
<tr>
<td>3</td>
<td>1.87</td>
<td>2.93</td>
<td>56.14</td>
<td>35</td>
<td>0.36</td>
<td>0.56</td>
<td>91.41</td>
</tr>
<tr>
<td>4</td>
<td>1.67</td>
<td>2.61</td>
<td>58.76</td>
<td>36</td>
<td>0.35</td>
<td>0.54</td>
<td>91.95</td>
</tr>
<tr>
<td>5</td>
<td>1.46</td>
<td>2.29</td>
<td>61.04</td>
<td>37</td>
<td>0.35</td>
<td>0.54</td>
<td>92.49</td>
</tr>
<tr>
<td>6</td>
<td>1.10</td>
<td>1.72</td>
<td>62.76</td>
<td>38</td>
<td>0.31</td>
<td>0.48</td>
<td>92.97</td>
</tr>
<tr>
<td>7</td>
<td>1.06</td>
<td>1.66</td>
<td>64.43</td>
<td>39</td>
<td>0.30</td>
<td>0.47</td>
<td>93.44</td>
</tr>
<tr>
<td>8</td>
<td>1.01</td>
<td>1.58</td>
<td>66.01</td>
<td>40</td>
<td>0.28</td>
<td>0.44</td>
<td>93.89</td>
</tr>
<tr>
<td>9</td>
<td>0.93</td>
<td>1.45</td>
<td>67.46</td>
<td>41</td>
<td>0.28</td>
<td>0.44</td>
<td>94.33</td>
</tr>
<tr>
<td>10</td>
<td>0.91</td>
<td>1.43</td>
<td>68.89</td>
<td>42</td>
<td>0.28</td>
<td>0.43</td>
<td>94.76</td>
</tr>
<tr>
<td>11</td>
<td>0.86</td>
<td>1.34</td>
<td>70.22</td>
<td>43</td>
<td>0.25</td>
<td>0.39</td>
<td>95.15</td>
</tr>
<tr>
<td>12</td>
<td>0.84</td>
<td>1.31</td>
<td>71.53</td>
<td>44</td>
<td>0.24</td>
<td>0.37</td>
<td>95.52</td>
</tr>
<tr>
<td>13</td>
<td>0.81</td>
<td>1.27</td>
<td>72.80</td>
<td>45</td>
<td>0.23</td>
<td>0.36</td>
<td>95.89</td>
</tr>
<tr>
<td>14</td>
<td>0.77</td>
<td>1.20</td>
<td>74.00</td>
<td>46</td>
<td>0.23</td>
<td>0.35</td>
<td>96.24</td>
</tr>
<tr>
<td>15</td>
<td>0.73</td>
<td>1.15</td>
<td>75.15</td>
<td>47</td>
<td>0.21</td>
<td>0.33</td>
<td>96.57</td>
</tr>
<tr>
<td>16</td>
<td>0.73</td>
<td>1.14</td>
<td>76.28</td>
<td>48</td>
<td>0.20</td>
<td>0.32</td>
<td>96.88</td>
</tr>
<tr>
<td>17</td>
<td>0.71</td>
<td>1.11</td>
<td>77.39</td>
<td>49</td>
<td>0.19</td>
<td>0.29</td>
<td>97.18</td>
</tr>
<tr>
<td>18</td>
<td>0.68</td>
<td>1.06</td>
<td>78.45</td>
<td>50</td>
<td>0.18</td>
<td>0.29</td>
<td>97.46</td>
</tr>
<tr>
<td>19</td>
<td>0.65</td>
<td>1.01</td>
<td>79.47</td>
<td>51</td>
<td>0.17</td>
<td>0.27</td>
<td>97.73</td>
</tr>
<tr>
<td>20</td>
<td>0.62</td>
<td>0.97</td>
<td><strong>80.44</strong></td>
<td>52</td>
<td>0.16</td>
<td>0.26</td>
<td>97.99</td>
</tr>
<tr>
<td>21</td>
<td>0.59</td>
<td>0.92</td>
<td>81.36</td>
<td>53</td>
<td>0.16</td>
<td>0.25</td>
<td>98.24</td>
</tr>
<tr>
<td>22</td>
<td>0.58</td>
<td>0.91</td>
<td>82.27</td>
<td>54</td>
<td>0.15</td>
<td>0.23</td>
<td>98.47</td>
</tr>
<tr>
<td>23</td>
<td>0.56</td>
<td>0.88</td>
<td>83.14</td>
<td>55</td>
<td>0.14</td>
<td>0.21</td>
<td>98.68</td>
</tr>
<tr>
<td>24</td>
<td>0.54</td>
<td>0.84</td>
<td>83.98</td>
<td>56</td>
<td>0.13</td>
<td>0.21</td>
<td>98.89</td>
</tr>
<tr>
<td>25</td>
<td>0.53</td>
<td>0.83</td>
<td>84.81</td>
<td>57</td>
<td>0.13</td>
<td>0.20</td>
<td>99.10</td>
</tr>
<tr>
<td>26</td>
<td>0.50</td>
<td>0.78</td>
<td>85.58</td>
<td>58</td>
<td>0.11</td>
<td>0.18</td>
<td>99.27</td>
</tr>
<tr>
<td>27</td>
<td>0.48</td>
<td>0.75</td>
<td>86.34</td>
<td>59</td>
<td>0.11</td>
<td>0.17</td>
<td>99.44</td>
</tr>
<tr>
<td>28</td>
<td>0.46</td>
<td>0.72</td>
<td>87.06</td>
<td>60</td>
<td>0.10</td>
<td>0.15</td>
<td>99.59</td>
</tr>
<tr>
<td>29</td>
<td>0.44</td>
<td>0.69</td>
<td>87.75</td>
<td>61</td>
<td>0.08</td>
<td>0.12</td>
<td>99.71</td>
</tr>
<tr>
<td>30</td>
<td>0.43</td>
<td>0.67</td>
<td>88.41</td>
<td>62</td>
<td>0.07</td>
<td>0.11</td>
<td>99.82</td>
</tr>
<tr>
<td>31</td>
<td>0.42</td>
<td>0.65</td>
<td>89.06</td>
<td>63</td>
<td>0.07</td>
<td>0.10</td>
<td>99.92</td>
</tr>
<tr>
<td>32</td>
<td>0.39</td>
<td>0.61</td>
<td>89.67</td>
<td>64</td>
<td>0.05</td>
<td>0.08</td>
<td>100.00</td>
</tr>
</tbody>
</table>

Table 6.1: Principal components analysis result
<table>
<thead>
<tr>
<th>NO.</th>
<th>Correlation coefficient</th>
<th>NO.</th>
<th>Correlation coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5979</td>
<td>11</td>
<td><strong>0.0219</strong></td>
</tr>
<tr>
<td>2</td>
<td>0.1411</td>
<td>12</td>
<td>-0.0804</td>
</tr>
<tr>
<td>3</td>
<td>0.0947</td>
<td>13</td>
<td>-0.1698</td>
</tr>
<tr>
<td>4</td>
<td>0.2275</td>
<td>14</td>
<td><strong>-0.0101</strong></td>
</tr>
<tr>
<td>5</td>
<td>0.2480</td>
<td>15</td>
<td>-0.0953</td>
</tr>
<tr>
<td>6</td>
<td><strong>-0.0009</strong></td>
<td>16</td>
<td>0.0634</td>
</tr>
<tr>
<td>7</td>
<td>0.1515</td>
<td>17</td>
<td>-0.1294</td>
</tr>
<tr>
<td>8</td>
<td>-0.0532</td>
<td>18</td>
<td>-0.0835</td>
</tr>
<tr>
<td>9</td>
<td>-0.1673</td>
<td>19</td>
<td>0.1466</td>
</tr>
<tr>
<td>10</td>
<td><strong>-0.0500</strong></td>
<td>20</td>
<td>-0.0862</td>
</tr>
</tbody>
</table>

Table 6.2: Correlation coefficient

6.2.2 PCA Correlation Coefficient Similarity Search Result

As can be seen from Table 6.2, 16 principal components are selected based on which the weighted similarity coefficients are computed. The result is shown in Table 6.3

Taking $[0.85, 1.15]$ as similarity confidence interval, according to Table 6.3, two stocks SZ600279 and SH000421 are selected to be further analysed for information granulation since the values are within the interval $[0.85, 1.15]$. The historical return and price track for target and selected stocks are shown in Figure 6.1.

![Selection Results](image1)

Figure 6.1: Stock similarity search result
<table>
<thead>
<tr>
<th>NO.</th>
<th>Stock Code</th>
<th>S(Ri,Rj)</th>
<th>NO.</th>
<th>Stock Code</th>
<th>S(Ri,Rj)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SZ600004</td>
<td>2.36</td>
<td>33</td>
<td>SZ600897</td>
<td>-2.58</td>
</tr>
<tr>
<td>2</td>
<td><strong>SZ600009(T)</strong></td>
<td><strong>1.00</strong></td>
<td>34</td>
<td>SZ600960</td>
<td>1.40</td>
</tr>
<tr>
<td>3</td>
<td>SZ600017</td>
<td>-5.36</td>
<td>35</td>
<td>SZ601000</td>
<td>-2.29</td>
</tr>
<tr>
<td>4</td>
<td>SZ600018</td>
<td>-7.54</td>
<td>36</td>
<td>SZ601006</td>
<td>1.67</td>
</tr>
<tr>
<td>5</td>
<td>SZ600026</td>
<td>-6.44</td>
<td>37</td>
<td>SZ601008</td>
<td>-0.26</td>
</tr>
<tr>
<td>6</td>
<td>SZ600029</td>
<td>0.12</td>
<td>38</td>
<td>SZ601018</td>
<td>1.37</td>
</tr>
<tr>
<td>7</td>
<td>SZ600087</td>
<td>-0.07</td>
<td>39</td>
<td>SZ601111</td>
<td>2.05</td>
</tr>
<tr>
<td>8</td>
<td>SZ600115</td>
<td>-1.44</td>
<td>40</td>
<td>SZ601299</td>
<td>-0.83</td>
</tr>
<tr>
<td>9</td>
<td>SZ600119</td>
<td>0.66</td>
<td>41</td>
<td>SZ601333</td>
<td>-0.34</td>
</tr>
<tr>
<td>10</td>
<td>SZ600125</td>
<td>4.52</td>
<td>42</td>
<td>SZ601766</td>
<td>-2.51</td>
</tr>
<tr>
<td>11</td>
<td>SZ600190</td>
<td>0.25</td>
<td>43</td>
<td>SZ601866</td>
<td>-2.09</td>
</tr>
<tr>
<td>12</td>
<td>SZ600221</td>
<td>1.35</td>
<td>44</td>
<td>SZ601872</td>
<td>-1.13</td>
</tr>
<tr>
<td>13</td>
<td>SZ600242</td>
<td>-2.78</td>
<td>45</td>
<td>SZ601880</td>
<td>-1.22</td>
</tr>
<tr>
<td>14</td>
<td>SZ600270</td>
<td>0.57</td>
<td>46</td>
<td>SZ601919</td>
<td>1.80</td>
</tr>
<tr>
<td>15</td>
<td><strong>SZ600279</strong></td>
<td><strong>1.10</strong></td>
<td>47</td>
<td>SH000022</td>
<td>-5.05</td>
</tr>
<tr>
<td>16</td>
<td>SZ600317</td>
<td>2.90</td>
<td>48</td>
<td>SH000088</td>
<td>-1.81</td>
</tr>
<tr>
<td>17</td>
<td>SZ600386</td>
<td>2.22</td>
<td>49</td>
<td>SH000089</td>
<td>4.78</td>
</tr>
<tr>
<td>18</td>
<td>SZ600387</td>
<td>0.36</td>
<td>50</td>
<td>SH000099</td>
<td>-4.80</td>
</tr>
<tr>
<td>19</td>
<td>SZ600428</td>
<td>0.73</td>
<td>51</td>
<td><strong>SH000421</strong></td>
<td><strong>0.95</strong></td>
</tr>
<tr>
<td>20</td>
<td>SZ600561</td>
<td>0.06</td>
<td>52</td>
<td>SH000520</td>
<td>-1.96</td>
</tr>
<tr>
<td>21</td>
<td>SZ600575</td>
<td>-0.90</td>
<td>53</td>
<td>SH000582</td>
<td>-0.06</td>
</tr>
<tr>
<td>22</td>
<td>SZ600611</td>
<td>-1.55</td>
<td>54</td>
<td>SH000905</td>
<td>-0.36</td>
</tr>
<tr>
<td>23</td>
<td>SZ600650</td>
<td>-0.64</td>
<td>55</td>
<td>SH000996</td>
<td>-2.91</td>
</tr>
<tr>
<td>24</td>
<td>SZ600662</td>
<td>2.31</td>
<td>56</td>
<td>SH002013</td>
<td>-1.12</td>
</tr>
<tr>
<td>25</td>
<td>SZ600692</td>
<td>-0.74</td>
<td>57</td>
<td>SH002023</td>
<td>-3.44</td>
</tr>
<tr>
<td>26</td>
<td>SZ600717</td>
<td>-0.26</td>
<td>58</td>
<td>SH002040</td>
<td>-0.63</td>
</tr>
<tr>
<td>27</td>
<td>SZ600751</td>
<td>0.48</td>
<td>59</td>
<td>SH002210</td>
<td>-0.54</td>
</tr>
<tr>
<td>28</td>
<td>SZ600787</td>
<td>-0.47</td>
<td>60</td>
<td>SH002320</td>
<td>0.67</td>
</tr>
<tr>
<td>29</td>
<td>SZ600794</td>
<td>0.12</td>
<td>61</td>
<td>SH002357</td>
<td>2.94</td>
</tr>
<tr>
<td>30</td>
<td>SZ600798</td>
<td>1.26</td>
<td>62</td>
<td>SH002492</td>
<td>-6.11</td>
</tr>
<tr>
<td>31</td>
<td>SZ600834</td>
<td>0.72</td>
<td>63</td>
<td>SH002510</td>
<td>8.15</td>
</tr>
<tr>
<td>32</td>
<td>SZ600896</td>
<td>1.91</td>
<td>64</td>
<td>SH300011</td>
<td>14.47</td>
</tr>
</tbody>
</table>

Table 6.3: Weighted similarity coefficients result. $R_j = R_2$ is the return of target stock SZ600009
6.3 Information Granulation Result

Divide the whole time series into 100 time windows with length of 3 days in which price information for both target stock and selected stocks are included. Using W. Pedrycz’ fuzzy granulation method described in Chapter 5, three sequences Up, Low and R are generated. Take into account that in the China stock market, there is a ±10% daily fluctuation restriction, the Low and Up are adjusted within the interval of [90%R, 110%R]. Specifying, Low is adjusted to 90%R if it is below 90%R and Up is adjusted to 110%R if it above 110%. The granulation result is shown in Figure 6.2

![Granulation Price](image)

Figure 6.2: Granulation price

6.4 Cross Validation

In order to get better SVR performance for Up, R and Low time series separately, the parameter selection process is implemented using the Cross Validation method. In the experiment a Radial basis kernel function (4.25) with parameter $g$ is selected and 3-fold Cross Validation is performed where $g$ takes values in interval $[2^{-8}, 2^8]$ with step 0.1 in the exponential growth, i.e., $g$ takes value $2^{-8}, 2^{-7.9}, ..., 2^8$. The same settings hold for parameter $C$. Cross Validation results are shown in Figure 6.3. The left figure is the contour view with the contour represents mean square error of SVR for given parameters $C$ and $g$. The right one is a 3D view of the mean squared error. The best parameter choice is given in each figure.
6.5 Regression result

Implement SVR using a radial basis kernel function with parameter $g$ and meta parameter $C$ selected by the previous cross validation result. All the granulation prices are taken both for training and validation. For computational convenience, assume that the next three days price depends on past three days performance. The result is shown in Figure 6.4. The left figure compares the original price with predicted price generated from a SVR.
model. The right figure gives the relative error which measures how accurate the model is. As we can see from these figures, the predicted price fit the original price track very well with most of relative error controlled in ±4%.

![Figure 6.4: Predicted result and error](image)

### 6.6 Prediction

After training, three SVR models are generated for Up, R and Low. We use it to predict for the next 3 days price and the result is shown in Table 6.4
Table 6.4: Predicted price for next 3 days

As we can see in the table, R=14.7586 is of the highest probability. Therefore, according to the fuzzy membership function, the real price should fluctuate around 14.7586 and be bounded in [13.2679, 16.2562]. The following Table 6.5 gives the real stock price obtained from market.

Table 6.5: Real stock price for next 3 days (2011.04.06-2011.04.08)

We compare the real price with predicted price and conclude that the prediction is quite promising especially for the nearest future (2011.04.06) with predicted price 14.7586 compared with average real price 14.7925.
Chapter 7

Conclusion

In this study a principal component analysis-fuzzy-support vector regression model for stock price prediction is provided. Stocks with similar historical trends are selected using principal component analysis. Fuzzy information granulation is performed to construct a probability density for stock prices. Support vector regression is implemented to generate a regression function for future price prediction. This method suits for any sample size with any noise distribution type and eliminates the complicated fine tuning process compared with other Neural Network methods. Besides, the use of fuzzy information granulation extends the prediction output form from a point to an interval with a probability assigned.
Bibliography


