UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

FUZZY CORRELATION AND REGRESSION ANALYSIS

A Dissertation

SUBMITTED TO THE GRADUATE FACULTY

In partial fulfillment of the requirements for the

Degree of

Doctor of Philosophy

By

YONGSHEN NI
Norman, Oklahoma
2005

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

FUZZY CORRELATION AND REGRESSION ANALYSIS

A Dissertation APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

____
Hong Liu, Committee Chair

____
John Y. Cheung, Co-chair

____
Joseph Havlicek

____
Monte Tull

____
Simin Pulat

ACKNOWLEDGEMENTS

Firstly I would like to express my deep gratitude to my advisor, Dr. John Y. Cheung for his support, patience, encouragement throughout my graduate studies. It is not often possible that one find an advisor that is always ready to discuss the small problems with me in the course of doing research. His technical and editorial advice is essential to the completion of this dissertation.

Secondly I would like to give my thankfulness to my advisory committee chair Dr. Hong Liu; His technical and financial help in the past several years plays a very important role during my Ph.D study.

My thanks also go to the members of my advisory committee, Dr. Joseph Havlicek, Dr. Monte Tull, and Dr. Simin Pulat for their time to serve as my committee.

The cooperation with Dr. Robert Chu and George Thomas is much appreciated and has led to some interesting applications to my research topic. I am also grateful to my colleague Dr. Yushan Li for the help with parts of the fuzzy regression research.

Last but not least, I would like to thank my family for their understanding and support during the past few years. It is impossible for me to finish my Ph.D study without their continuous encouragement and love.

TABLE OF CONTENTS

LISTS OF TABLES

LIST OF FIGURES

ABSTRACT

Correlation and regression analysis are widely used in all kinds of data mining applications. However many real world data have the characteristic of vagueness; the classical data analysis techniques have limitation in managing this vagueness systematically. Fuzzy sets theory can be applied to model this kind of data. New concepts and methods of correlation and regression analysis for data with uncertainty are presented in this dissertation. Recently, fuzzy correlation and regression have been applied to many applications. Successful examples include quality control, marketing, image processing, robot control, medical diagnosis etc. The purpose of this dissertation is to revisit the ongoing research work that people have already done on this issue and to develop some new models related to fuzzy data correlation and regression. In this dissertation, we define and conceptualize the correlation and regression concepts within the fuzzy context. Then the presently available methods are explored in light of their limitations. Then new concepts and new models are presented. Throughout this dissertation, a number of test data sets are used to verify how our ideas are implemented. Suggestions for further research will be provided.

The first half of the dissertation focuses on the motivation and concept of fuzzy correlation. Fuzzy data will be formulated in a mathematical way, and then we will build models of two types of fuzzy correlations, their computation methods are also presented in this dissertation. For the first type of fuzzy correlation problem we proposed an approximate bound as well as a number of computationally efficient algorithms. Monte Carlo sampling method is used to compute the second type of fuzzy correlation problem. The results provided by the second type of fuzzy correlation are more informative than the result of the classical correlation.

In the second part of the dissertation, eight fuzzy regression models are discussed. In order to enhance the central tendency and remove outliers which have important impact on the regression result, different techniques are used to improve the original model. The fuzzy regression method presented in this dissertation also applies to crisp data regression cases. Numerical examples are given for all the fuzzy correlation and fuzzy regression models we explored in this dissertation for illustration and verification purpose.

Some application examples are given at the end. Fuzzy regression models could be applied in short term stock price prediction. Intel Corp. 2003 stock price data are used in this demo. The Dosage-film response is estimated with a fuzzy regression model, this procedure is presented in detail in the last section. It is found that fuzzy regression gives more consistent results than the conventional regression model since it successfully models the inherent vagueness which exists in the application by formulated form.

CHAPTER 1

Introduction

Situations arise where the data under consideration consists of pairs of measurements. The essential feature of the data is that one observation can be paired with another observation for each member of the group, e.g. the input/output membership of an unknown system and the model. Studies of this type have two closely related aspects, correlation and regression.

1.1 Crisp Data and Fuzzy Data

Crisp data is also called precise data; it is very common in our everyday life. For example, when we say a student's high school GPA is 3.5, it is a single value without any ambiguity. Another example, if a weather station tells us that it is $30^\circ c$ outside for the time being, although it is possible there could be some small difference for the temperature measurement in different places, we still think it is precise information. The traditional science and technology pursuit for certainty in all its manifestations and almost all the mathematical theories are developed for handling such kind of data.

Figure 1.1 shows the definition of crisp data in terms of a membership plot between a linguistic variable and the independent variable. If the weather can be categorized of cold, warm and hot, then it is possible to say that when temperature is below $15^\circ c$, it is cold, when temperature is between $15^\circ c$ and $25^\circ c$, it is warm, and when temperature is above $30^\circ c$, it is hot. Note that the boundaries between cold, warm and hot are precise. Based on the definition, the weather can be cold or warm or hot. In other words, the weather is well defined by the temperature.

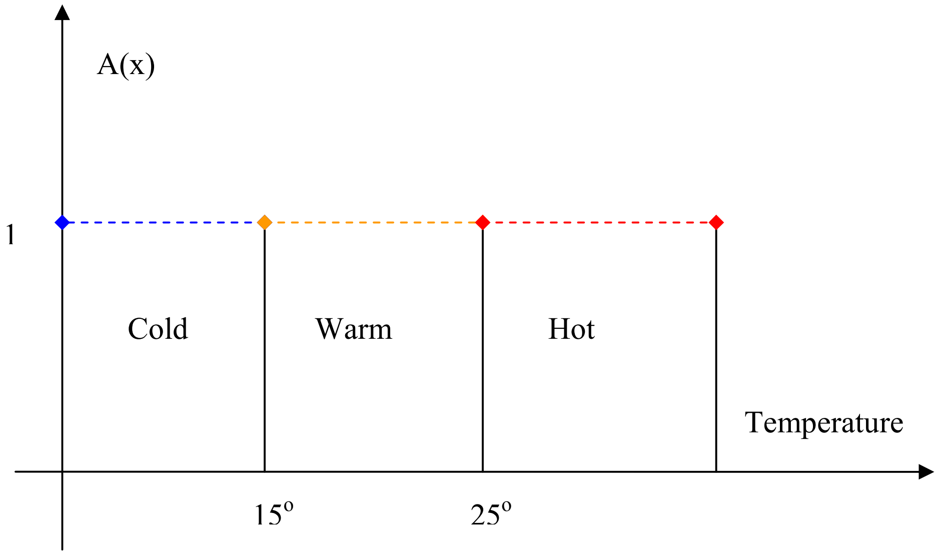

Fig. 1.1 Illustration of crisp data

However, uncertainty is not always avoidable. In many applications, data has the characteristic of vagueness or uncertainty. This is the case when data is derived from imprecise measurement instruments or from the description of human domain experts. In the presence of vagueness and uncertainty, precise boundaries as in the case of crisp data lose meaning.

The concept of "fuzzy" variable was first proposed by Dr. Lotfi Zadeh in 1965 [36]. He proposed that fuzzy set can be applied to represent data which has the characteristic of vagueness. This vagueness can be represented by the degree of participation to a set called a membership function. In contrast, a crisp data can either belong or not belong to a particular set.

We can think of a fuzzy variable as linguistic terms or some data coming out from imprecise measurement. In order to analyze the fuzzy data or explore the data relationship under a fuzzy environment, new concepts and new methods have to be developed to meet the challenges. Fuzzy data expresses the concept of "gradual transition", so it is natural to think about a membership function instead of a precise value to present a fuzzy data. A fuzzy set is an extension of the classical set theory, and it is characterized by a membership function which maps X into the interval [0, 1]. The value 0 means that the member does not participate in the given set, 1 indicates full participation. The membership function of a fuzzy set is denoted by $\mu_{A,}$

$$\mu_A : X \to [0,1]$$

where X is the domain the fuzzy data could locate, each fuzzy data is completely and uniquely defined by one particular membership function, the membership function may also be used as the label for the associated fuzzy data.

Natural phenomenon can be described more accurately by fuzzy data. For instance, we want to explore the property dark in a grayscale image. In classical set theory, we have to determine a threshold, all gray levels below the threshold will be thought as "dark", but the darkness is in fact a matter of degree. So, a fuzzy set can describe this data much better. Human languages do not express exact information either. When we state that somebody is tall, a context is necessary to describe the height. A tall person may be only 3 feet among preschool age group, but in an adult group, a tall person could be over six feet. Human mind can interpret information based on the context automatically, but computers can not handle imprecise information in the same way. We have to build an appropriate mathematical model to tell the computer about this imprecise information. Another example of imprecise data is data with measurement error. It is not always possible for us to measure a particular data very precisely; sometimes we only know that the value falls into an interval. This type of data is called fuzzy data. The uncertainty of a fuzzy data maybe uniformly distributed, Gaussian distributed or distributed in other forms. The exact distribution can be modeled by a fuzzy membership function.

Figure1.2 gives an illustration of fuzzy data in describing temperature. We use linguistic terms such as “cold”, “warm” or “hot”. These terms can be related to temperature by a fuzzy membership function. The “cold” membership function is seen in Figure 1.2, so that when the temperature is below $20^{\circ}c$, some people will consider the temperature to be cold. As the temperature drops, more people will consider the weather to be cold. When the temperature is below $10^{\circ}c$, everyone will consider the weather to be cold.

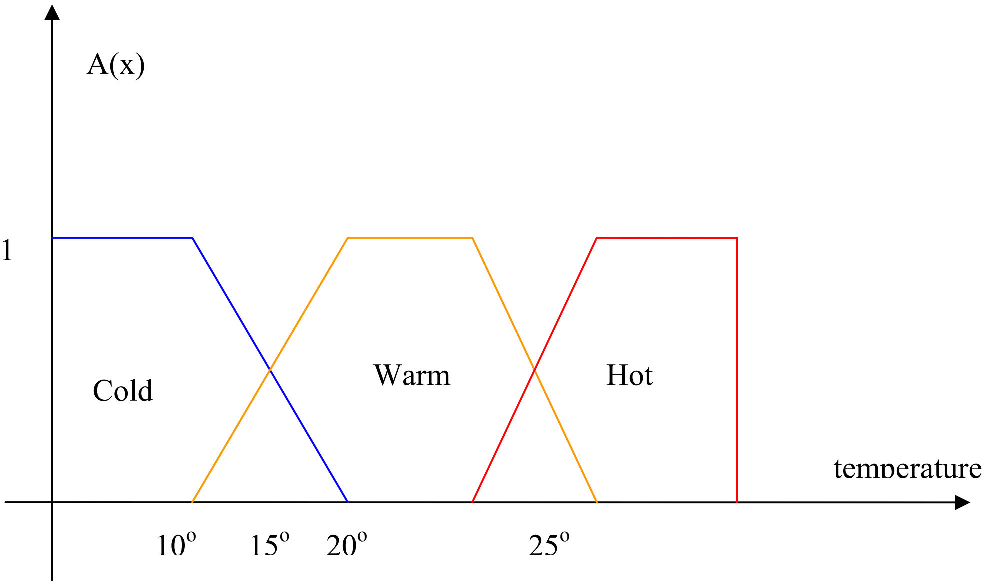

Fig. 1.2 Membership function for temperature

In Figure 1.2, three fuzzy sets are employed to capture the linguistic concepts cold, warm and hot temperatures. The fuzzy data facilitates gradual transition between states, hence the membership functions have a natural capability to express and handle observation and measurement uncertainties while the traditional crisp variables do not have such capability. So the fuzzy data are more attuned to reality than the crisp data. Obviously the fuzzy data in Figure 1.2 provides us more information than the crisp data in Figure 1.1. For example, according to the fuzzy classification in Figure 1.2, at a temperature of 15 °C the weather could be considered half warm and half cold, while with crisp data concept , 14.9°C is classified as cold and 15.1°C is classified as warm which is obviously not indicative of the actual situation.

1.2 Crisp Correlation and Fuzzy Correlation

Correlation is a measure of the association between two variables; it is a very important part of statistics. One of the most fundamental concepts in many applications is the concept of correlation. If two variables are correlated, this means that you can use information about one variable to predict the values of the other variable.

It is very common in statistical analysis of data to find the correlation between variables, the correlation defined on conventional crisp sets has been discussed in the classical statistics. However, the classical statistics cannot manage the data with uncertainty very well, new concepts and new methods have to be developed to find the correlation between fuzzy data.

Data with vagueness (or fuzzy data) can be modeled by a fuzzy set. Fuzzy sets have an advantage over the crisp representation in environments with a high degree of inaccuracy or uncertainty. We can define fuzzy measurements and mathematically manipulate those measurements with operations. For example, if the correlation coefficient on a fuzzy data set is defined, then we can know how the co-varying relationship between two variables will take on with the inherent uncertainty.

In this dissertation, I work on two types of fuzzy correlation models. The fuzzy data set, fuzzy correlation model is called Type I fuzzy correlation model in this dissertation. This model was reported in the literature [7], I propose six theorems to systematically elaborate the properties of Type I fuzzy correlation. I also develop three approximate methods; they are computationally efficient and the results are very close to the theoretic approach. The crisp data set, fuzzy correlation model is called Type II fuzzy correlation model in this dissertation; I develop the original idea and computation method of this model. Fuzzy correlation obtained from our model tells us not only the strength of the relationship between the random variables, but also the distribution of this correlation. The value of the correlation coefficient for our method also lies between the interval [-1, 1].

1.3 Crisp Regression and Fuzzy Regression

Correlation describes the strength of association between two random variables, and it is completely symmetrical, that is, the correlation between A and B is the same as the correlation between B and A. If the two variables are related it means that when one changes by a certain amount the other changes on an average by a corresponding amount. If we use y to represent the dependent variable and x the independent variable, this relationship could be described as the regression of y on x, and in the simplest case this is assumed to be a straight line. The slope of the line depends on whether the correlation is positive or negative.

Regression goes beyond correlation by adding prediction capabilities. For example, university admission office might want to use SAT score, high school GPA (independent variables) to predict a student's college GPA (dependent variable). The purpose of applying the regression is to find the relationship between the dependent variable and the independent variables. Then we can use that formula to predict values for the dependent variable when only the independent variables can be directly measured or observed.

There are many situations where observations cannot be described accurately, e.g., the observations resulting from human language or imprecise machine. In such cases, we only can give an approximate description about them. Classical statistical theory focuses on a kind of uncertainty which is called randomness, but we are concerned with another kind of uncertainty that is sometimes referred to as vagueness. When the data with vagueness (or fuzzy data) is analyzed through nonfuzzy techniques, it is regarded as if it is precise and the original vagueness is not taken into account in the analysis. So the model based on fuzzy data accommodates more information than models that just ignore the intrinsic vagueness of the data. Fuzzy set theory has been applied to manage the vagueness of the data and has been successfully demonstrated in many applications, such as: reliability, quality control, economical development forecast, etc.

Regression analysis has been a very popular method with many successful applications. The regression analysis dealing with fuzzy data is usually called fuzzy regression analysis. The aim of this dissertation is to develop regression models among fuzzy data variables. There are two motivations for developing fuzzy regression analysis. The first motivation results from the realization that it is not often realistic to assume that a crisp function of a given form can be used to represent the relationship between the given variables. Fuzzy relationship which is even though less precise seems intuitively more realistic. The second motivation results from the fact that the nature of data in many cases have inherent characteristic of uncertainty. Eight categories of regression models will be discussed in this dissertation including all the combinations of data characteristics in the input, the regression parameter and the output.

CHAPTER 2

Background of Fuzzy Correlation

2.1 Crisp Correlation Coefficient

The Pearson product-moment correlation coefficient (r) or correlation coefficient is a measure of the degree of linear relationship between two variables. The correlation coefficient possesses a number of interesting properties [5].

Firstly, the correlation coefficient takes on any value between plus and minus one.

$$-1 \le r_{xy} \le 1$$

where r_{xy} denotes the correlation between the variables x and y.

Secondly, the sign of the correlation coefficient (+, -) defines the direction of the relationship. A positive correlation coefficient means that as the value of one variable changes, the value of the other variable changes in the same direction; A negative correlation coefficient indicates that as one variable changes, the other changes in the opposite direction.

Thirdly, the absolute value of the correlation coefficient measures the strength of the relationship. Thus, a correlation coefficient of zero (r_{xy}=0.0) indicates the absence of a linear relationship and correlation coefficients of r_{xy}=+1.0 and r_{xy}=-1.0 indicate a perfect linear relationship.

Fourthly, the correlation coefficient may be interpreted by a data set scatter plot. The scatter plots perhaps best illustrate how the correlation coefficient changes as the linear relationship between the two variables is altered. When r_{xy}=0.0 the points scatter widely about the plot without any perceivable trend. As the linear relationship increases, the region containing all the data points becomes more and more elliptical in shape until the limiting case is reached (r_{xy}=1.00 or r_{xy}=-1.00) and all the points fall on a straight line.

2.2 Crisp Data, Crisp Correlation

In correlation analysis area, four cases are considered. The first case is to use crisp data to get a crisp correlation thus is the subject of classical correlation estimation.

The correlation coefficient can be formulated as follows.

$$r_{xy} = \frac{\sum_{i=1}^{m}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{m}\left(x_i - \overline{x}\right)^2}\sqrt{\sum_{i=1}^{m}\left(y_i - \overline{y}\right)^2}} \tag{2.1}$$

where i ranges from 1 to m, m is the cardinality of data set, ($x_{i,}$ y_i) is the i-th variable pairs, $\overline{x}$ is the mean of the independent variable, $\overline{y}$ is the mean of the dependent variable. We can use a statistical calculator or a computer program to compute the correlation coefficient for the given data set ($x_{i,}$ y_i).

When the input data is crisp and a crisp correlation is derived, then the Pearson correlation coefficient formula in (2.1) can be used.

2.3 Fuzzy Data, Crisp Correlation

Ding-An Chiang and Nancy P. Lin [8] developed a crisp correlation coefficient between two fuzzy sets. The crisp correlation coefficient lies in the interval [-1, 1]. Their method takes a random sample from a crisp set, with corresponding pairs of membership functions of the two fuzzy sets to compute the correlation between those two fuzzy sets. The formula used in this study is Pearson's product-sum correlation coefficient; a pair of membership function values replaces the original data values as follows.

$$\overline{\mu_A} = \frac{\sum_{i=1}^{n} \mu_A(x_i)}{n}$$

$$\overline{\mu_B} = \frac{\sum_{i=1}^{n} \mu_B(y_i)}{n}$$

$$S_A^2 = \frac{\sum_{i=1}^{n} \left(\mu_A(x_i) - \overline{\mu_A}\right)^2}{n-1}$$

$$S_B^2 = \frac{\sum_{i=1}^{n} \left(\mu_B(y_i) - \overline{\mu_B}\right)^2}{n-1}$$

$$r_{A,B} = \frac{\sum_{i=1}^{n} \left(\mu_A(x_i) - \overline{\mu_A}\right)\left(\mu_B(x_i) - \overline{\mu_B}\right)/(n-1)}{S_A \cdot S_B} \tag{2.2}$$

where n is the size of sample, μ_A, μ_B are membership function values for each sample respectively.

The correlation defined by their approach is in the interval [-1, 1]. As we have just described, the resultant correlation is a crisp value.

A major contribution of this model is the development of partial correlation of fuzzy sets. If a random sample with multiple fuzzy attributes, Chiang and Lin's method can compute the correlation coefficient between the two fuzzy attributes. For example, according to their theory, a correlation coefficient is calculated between each pair of the attributes A, B and C, so that we have r_{AC}, r_{BC} and r_{AB}, then the formula to compute the first-order partial correlation coefficient between the fuzzy attributes A and B holding fuzzy attribute C constant is defined in terms of the simple correlation coefficients:

$$r_{A.B\cdot C} = \frac{r_{A,B} - r_{A,C} r_{B,C}}{\sqrt{1 - r_{A,C}{}^2}\sqrt{1 - r_{B,C}{}^2}} \tag{2.3}$$

A limitation of the method is that it only applies when the membership functions of the fuzzy sets are well behaved, the left side (and the right side) of the membership function is monotonically increasing (decreasing).

Chaudhuri and Bhattacharya [12] also proposed a formula to measure fuzzy set association; his method qualifies the correlation relationship between two fuzzy attributes with crisp observation data. In fact, their correlation coefficient describes the similarity between two fuzzy sets.

$$r_{A,B} = 1 - \left[\sum_{x \in X} \left| \frac{\mu(x)}{(\sum_{x \in X} \mu^2(x))^{1/2}} - \frac{\eta(x)}{(\sum_{x \in X} \eta^2(x))^{1/2}} \right|^2 \right]^{1/2} \tag{2.4}$$

where an element $x \in X$ belong to set A with membership value μ and to set B with membership value η. The resultant correlation $r_{A,B}$ lies in the interval [0, 1] for all $x \in X$.

2.4 Fuzzy Correlation Coefficient

There are two general ways to develop fuzzy correlation model: models where the size of data set is small and re-sampling or bootstrapping gives fuzzy results and models where the variables themselves are fuzzy. Both of these models are explored in this chapter. We call the first model Type I fuzzy correlation and the second model Type II fuzzy correlation. In this section, we focus on models where the data points are fuzzy.

If the input data is fuzzy, then according to the Extension Principle, the correlation coefficient which is a function of the input data should also be fuzzy and is called a fuzzy correlation coefficient (FCC) and is denoted as $r_{xy}(\alpha)$, where α is the α-cut value. The notion of fuzzy data is formalized by introducing the concept of a fuzzy number based on the fuzzy set theory; fuzzy data can be represented by a membership function. Assume that $r_{xy}(\alpha)$ is the correlation coefficient at each α level. Because every fuzzy data has its own membership function (in both the x and y directions), two intervals are generated in both directions for each observation if we take the α-cut on the membership functions:

$$([x_{\alpha}^{L}, x_{\alpha}^{U}]_j \ [y_{\alpha}^{L}, y_{\alpha}^{U}]_j \ ) \quad j=1,2,\dots.,m$$

where m is the cardinality of data set A, x_{α}^{L} is the lower bound of the α-level for the x coordinate of jth fuzzy data, x_{α}^{U} is the upper bound of the α-level for the x coordinate of jth fuzzy data, y_{α}^{L} is the lower bound of the α-level for the y coordinate of jth fuzzy data, y_{α}^{U} is the upper bound of the α-level for the y coordinate of jth fuzzy data.

Now each of our data has been converted to a rectangular region. If α equals to 1, the fuzzy data degenerates to a single crisp value, and the fuzzy correlation coefficient degenerates to the crisp correlation coefficient accordingly. With increasing α, the rectangular region becomes bigger, and r_{xy} (α) is no longer a single value, but an interval.

$$r_{xy}(\alpha) = [\min r_{xy}(\alpha),\ \max r_{xy}(\alpha)]$$

Figure 2.1 shows a crisp data set when both x and y data points are crisp. Figure 2.2 shows that a crisp correlation coefficient is generated with the data set given in Figure 2.1. Figure 2.3 shows a data set when uncertainty exists in both the x and y directions. Figure 2.4 shows the resultant interval correlation coefficient. Figure 2.5 shows a fuzzy data set when there is a membership function describing the data points in both the x and y direction. Figure 2.6 shows the resultant fuzzy correlation coefficient.

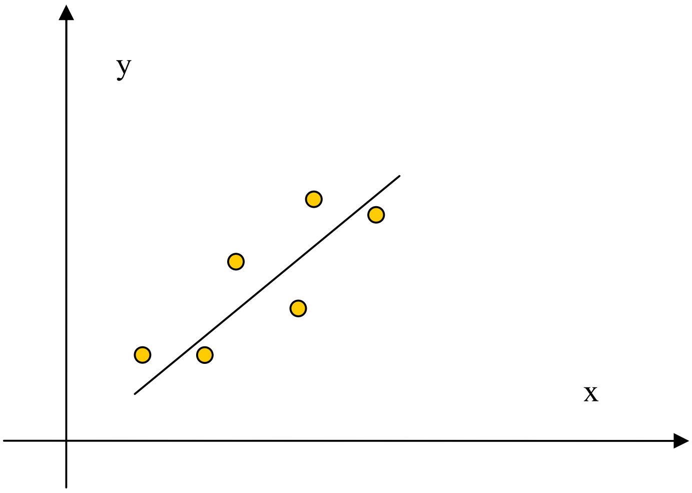

Fig. 2.1. A crisp data set

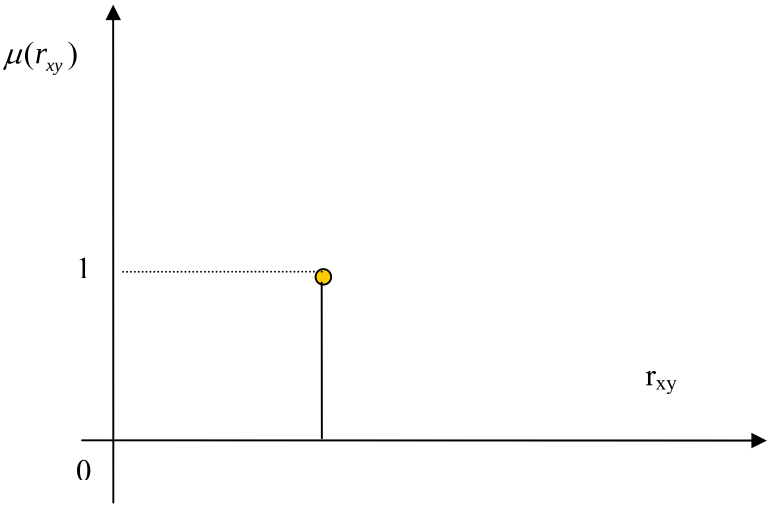

Fig. 2.2 A crisp correlation coefficient

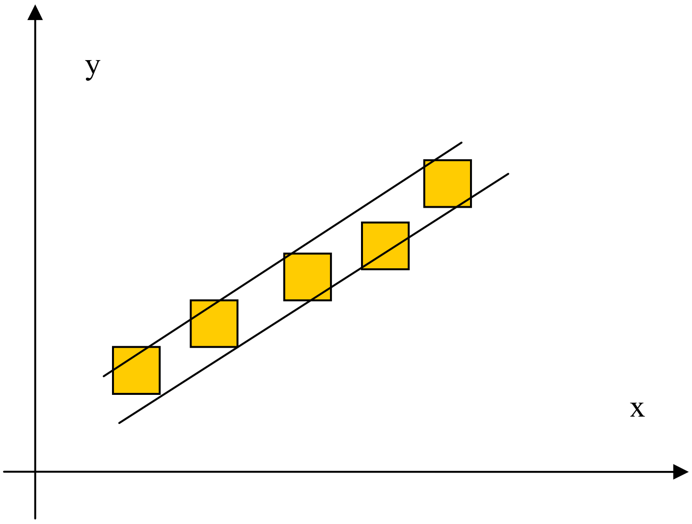

Fig. 2.3 An interval data set

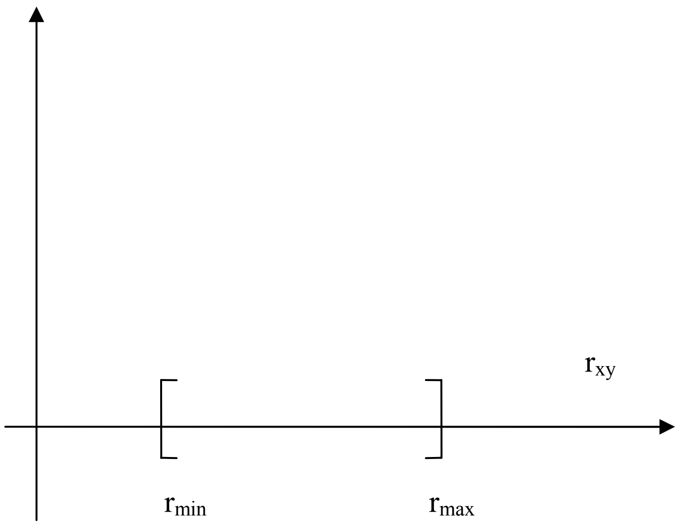

Fig. 2.4 An interval correlation coefficient

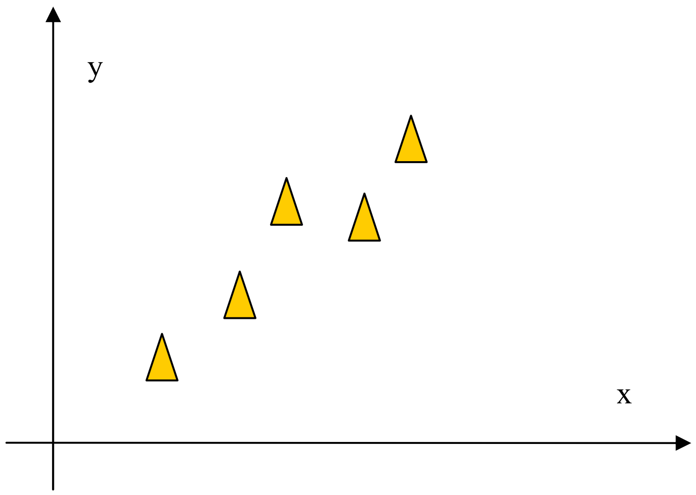

Fig. 2.5 A fuzzy data set with triangular membership function

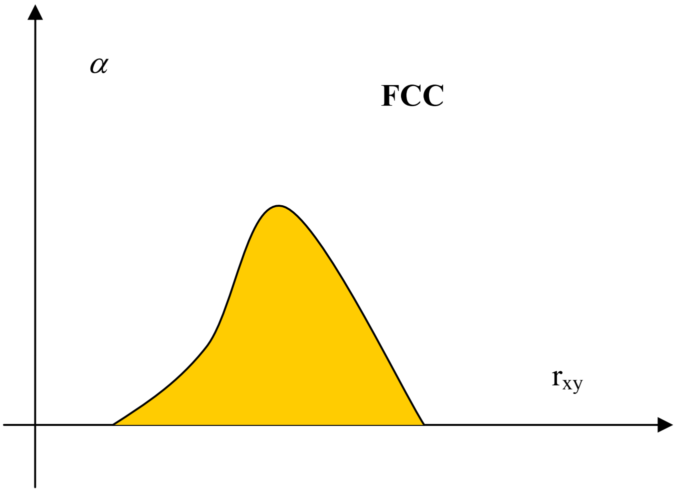

Fig. 2.6 A fuzzy correlation coefficient

Extension principle

The extension principle is one of the most basic ideas of fuzzy set theory. It goes back to Zadeh in 1965 and provides a general method for extending crisp mathematical concepts in order to deal with fuzzy quantities. The extension principle can be described as follows.

For any given function $f: X \to Y$, Zadeh proposed the fuzzy mapping $\tilde{f}$ from the fuzzy set A(x) to the fuzzy set B(y). If we have

$$\tilde{f} : A(x) \to B(y)$$

and the inverse fuzzy mapping $\tilde{f}^{-1} : B(y) \to A(x)$

Based on the extension principle, a mapping from x to y can be extended to the mapping of the fuzzy set A (x) to B(y).

$$B(y) = [\tilde{f}(A)](y) = \sup_{x|y=f(x)} A(x) \quad \text{for all } x \in A(x)$$

The extension principle is illustrated in Figure 2.7. Obviously the output result is a nonlinear mapping of the original fuzzy data according to the mapping function.

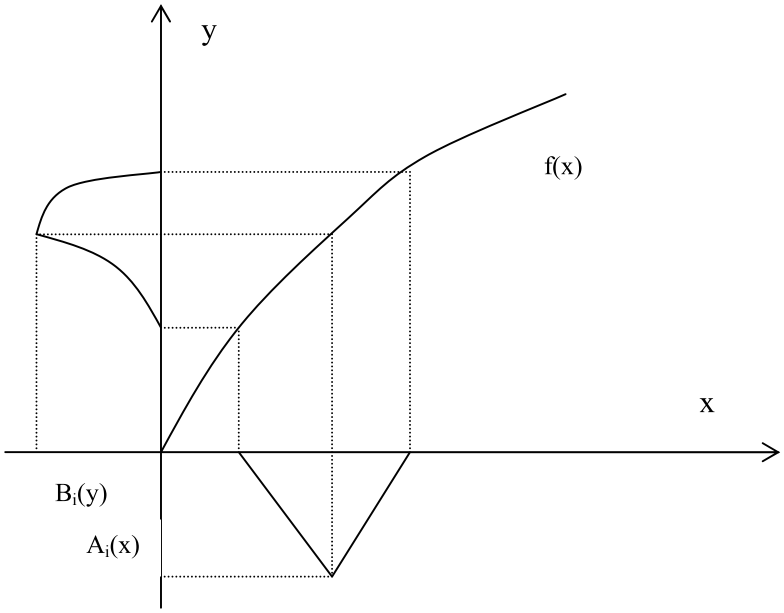

Fig.2.7 Extension principle

Assume f: $(x, y) \rightarrow r_{xy}$, A is the fuzzy data set, and B is the fuzzy correlation coefficient, n is the size of the data set. Let $\overset{\wedge}{r_{xy}}$ be the fuzzy correlation coefficient for a fuzzy data set A(x,y). Then based on the extension principle,

$$\overset{\wedge}{r_{xy}} = [f(x, y)](r_{xy}) = \sup_{x,y|r_{xy}=f(x,y)} \min\ (A(x_1), A(x_2)...)$$

where $$r_{xy} = f(x,y) = \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.5}$$

Hence, the mathematic model of fuzzy correlation coefficient could be summarized as the following constrained optimization problem:

$$r_\alpha^L = \min \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.6}$$

$$r_\alpha^U = \max \sum_{i=1}^{n} \frac{(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.7}$$

s.t. $$(X_i)_\alpha^L \le x_i \le (X_i)_\alpha^U$$

$$(Y_i)_\alpha^L \le y_i \le (Y_i)_\alpha^U$$

This model is a pair of nonlinear programs with bounded constrains. If we have two possibility levels α_1 and α_2 and $0 < \alpha_2 < \alpha_1 \le 1$, the feasible region determined by α_1 is a subset of the region determined by $\alpha_{2.}$ So the resultant interval of FCC determined by α_1 is a subset of the interval determined by α_2. Assume that L(r) and R(r) are the left leg and the right leg of the FCC respectively, the membership function for the FCC can be formulated as

$$\mu(r_{xy}) = \begin{cases} L(r), r_{\alpha=0}^L \le r \le r_{\alpha=1}^L \\ 1, r_{\alpha=1}^L \le r \le r_{\alpha=1}^U \\ R(r),, r_{\alpha=0}^U \le r \le r_{\alpha=1}^U \end{cases} \tag{2.8}$$

Since Pearson's formula is used in the objective function of fuzzy correlation coefficient model, it is not surprising the value of fuzzy correlation coefficient lies in the range of [-1, 1]. More detailed analysis of the fuzzy correlation coefficient and its properties are presented in the next chapter.

2.5 Crisp Data, Fuzzy Correlation

Early fuzzy correlation [6, 25, 27, 28, 32] is defined on the intuitive fuzzy sets and the value of the fuzzy correlation coefficient falls in the interval [0, 1]. This makes it difficult to degenerate the fuzzy data to crisp data which is a highly desirable property.

Pedrycz [2, 3] develops the concept of a granular correlation; the information granules are defined as fuzzy sets in his paper. Suppose the two variables of interest come as pairs of data $\{x(k), y(k)\}, k = 1,2,..., N$, $x(k), y(k) \in R$. The two information granules, fuzzy sets A and B are defined for the value of x and y respectively. Pedrycz's approach abandons a global look at the overall data and concentrates on revealing meaningful relationship on a local level. Since Pedrycz's fuzzy correlation reveals the local dependency by using fuzzy sets, it can help evaluate fuzzy associations from the statistical point of view, and is particularly useful in linguistic-driven data mining applications.

Given two fuzzy sets A(x) and B(y), Pedrycz's method for computing the fuzzy correlation can be summarized in the following steps:

1. Start with the possibility level α=0.0.

2. Obtain all elements P(x, y) for which x belongs to A_α and y belongs to B_α, the subset at each α level is denoted as $P_\alpha = \{(x, y) \mid A(x) \le \alpha, B(y) \le \alpha\}$.

3. Compute the correlation coefficient r (α) for P_α.

4. Determine the number of the elements involved in the computations of the correlation coefficient for the statistical relevance testing.

5. Increase the possibility level α by a fixed increment until it reaches the value 1. Repeat step (2) through (4).

In the above algorithm, as the numeric data are activated by fuzzy sets to a certain degree, the correlation is computed to reflect the data relationship at different possibility level. This leads to the idea of the correlation coefficient that becomes a fuzzy set. The membership function of the fuzzy correlation depends on the location and size of the linguistic granules in the study. Pedrycz's method manages the association of two information granule. This method only applies to crisp data sets.

2.6 Fuzzy Data, Fuzzy Correlation

If the input data is fuzzy, Liu [7] gives us a fuzzy number correlation coefficient with domain from -1 to 1. His approach is in essence to apply the extension principle to compute the correlation coefficient on fuzzy data at different possibility level. The model is based on a pair of nonlinear programs with bounded constraints which are given in equation (2.6) and (2.7),

A commercial nonlinear programming solver is turned to when multiple fuzzy observations are involved. Nonlinear optimization is an intractable problem mathematically; the algorithm usually becomes inefficient when the size of data set is large. Moreover, although a fuzzy correlation coefficient reflects the uncertainty of the data, sometimes a crisp value is desired to have a summarized knowledge of the data association. One must take different action depending on whether this value exceeds the threshold or not.

In chapter 3, a mathematical model is developed to calculate the fuzzy correlation coefficient of fuzzy observation. In order to make this model practical, two computationally efficient algorithms are provided for the model. Simulation results in chapter 4 show that the presented algorithms give better performance than standard commercial software with large data set size.

2.7 Summary and Research Goal

There are four problems based on the input-correlation relationship in fuzzy correlation analysis: 1) Crisp input, crisp correlation. 2) Crisp input, fuzzy correlation, 3) Fuzzy input, crisp correlation, 4) Fuzzy input, fuzzy correlation. Pearson's product-sum formula has been widely accepted to compute the correlation coefficient between two crisp random variables. This chapter starts by discussing the recent research work that had been developed and had been applied in engineering practice. While most of these works concentrate on the analysis of fuzzy attributes, very few works have been done to analyze the correlation of fuzzy data.

CHAPTER 3

Fuzzy Correlation Analysis

This chapter deals with the definition and analysis of the fuzzy correlation coefficient. The mathematical model of the fuzzy correlation coefficient will be discussed, as well as its properties. This chapter presents a number of computationally efficient methods for calculating the FCC.

3.1 Properties of the Crisp Correlation Coefficient

Based on the computational formula given in (2.1), Pearson's crisp correlation coefficient r_{xy} has the following properties.

1 Range of correlation coefficient is between -1 and 1, $-1 \le r_{xy} \le 1$

2 If one random variable x is independent of the other random variable y, then $r_{xy}=0$

3 For linear relationship y=ax+b

$$r_{xy} = \begin{cases} 1 & a > 0 \\ -1 & a < 0 \end{cases}$$

4 If two random variables x and y co-vary in the same direction, then $r_{xy} > 0$; if they co-vary in the opposite direction, then $r_{xy} < 0$.

We have to realize that the crisp correlation coefficient reflects the linear association between the two random variables; it does not mean causality exists between the two random variables.

3.2 Definitions

The uncertainty of data has many sources. If we stress impreciseness and vagueness then it is reasonable to model the data by fuzzy sets. In many cases, it is of interest to have suitable measures of vagueness. The uncertainty can be manifested in different forms and these forms represent distinct types of uncertainty.

Two measures of uncertainty are most widely computed today. The two uncertainty types are: (1) nonspecificity which is related to the cardinalities of sets of relevant alternatives and (2) fuzziness that results from the imprecise boundaries of fuzzy sets.

3.2.1 Nonspecificity of Fuzzy Sets

In crisp set theory, Hartley [1] used (3.1) as a measure of uncertainty in terms of a subset of a universal set.

$$U(A) = \log_2 |A| \tag{3.1}$$

where |A| denotes the cardinality of a countable nonempty set A, this measurement of uncertainty has the unit of bits.

The Hartley function is a measure of uncertainty associated with available alternatives in the set. We can see that full specificity is obtained when all alternatives are eliminated except one. Hence this measure is also called nonspecificity. Hartley function has been generalized from the classical set theory to the fuzzy set theory in the early 1980s. For any nonempty fuzzy set A defined on a countable universal set X, the generalized Hartley function has the form

$$U(A) = \frac{1}{h(A)} \int_0^{h(A)} \log_2 \left| {}^{\alpha}A \right| d\alpha \tag{3.2}$$

where $\left| {}^{\alpha}A \right|$ denotes the cardinality of the α-cut of A and h (A) is the height of A. Fuzzy sets are equal if when normalized, they have the same nonspecificity as measured by the function U. Given a nonempty fuzzy set A defined on R and if the α-cuts are uncountable sets, then (3.2) can be modified as

$$U(A) = \frac{1}{h(A)} \int_{0}^{h(A)} \log[1 + \mu(^{\alpha}A)]d\alpha \tag{3.3}$$

where ${}^{\alpha}A$ is a measurable and Lebesgue-integrable function, $\mu({}^{\alpha}A)$ is defined by the Lebesgue integral of the characteristic function ${}^{\alpha}A$.

3.2.2 Fuzziness of Fuzzy Sets

In general, the measure of fuzziness of a function is a mapping from a fuzzy set to a crisp value [1].

$$f : A(x) \to R^{+} \tag{3.4}$$

where A(x) denotes a fuzzy set. For a given fuzzy set A, this function assigns a nonnegative real number f (A) that expresses the degree to which the boundary of A is not sharp. The function f must satisfy three requirements in order to qualify as a meaningful fuzziness measure.

1. f(A)=0 iff (if and only if) A is a crisp set
2. f(A) reaches its maximum value iff A(x)=0.5 for all $x \in X$
3. $f(A) \le f(B)$ If the membership function of fuzzy set A is obviously sharper than the membership function of fuzzy set B.

Two methods are widely accepted to measure fuzziness that satisfies all the three important requirements. One way is to measure fuzziness of a fuzzy set A by a distance between its membership function and the nearest crisp set. Another more practical method is to consider the fuzziness of a set A comes from the lack of distinction between its membership function and its complement. The less a set differs from its complement, the fuzzier it is.

We assume only the standard fuzzy complement is used, we also choose the Hamming distance, so the local distinction of a given set A and its complement is measured by

$$|A(x)-(1-A(x))|=|2A(x)-1|$$

and the lack of each local distinction is measured by $1-|2A(x)-1|$. If the fuzzy set is defined on a countable set, the measure of fuzziness is then obtained as:

$$f(A)=\sum_{x\in X}(1-|2A(x)-1|)$$

The range of function f is $[0,|X|]$; If the fuzzy set is defined on an uncountable but bounded subsets of R, for example X= [a, b], then the formula needs to be modified as follows:

$$f(A)=\int_a^b(1-|2A(x)-1|)dx=b-a-\int_a^b|2A(x)-1|dx \tag{3.5}$$

The nonspecificity and fuzziness are distinct types of uncertainty and they are totally independent of each other.

3.2.3 Defuzzification

Defuzzification is the process of mapping a fuzzy set onto a crisp value, a number of defuzzification methods are proposed in the literature [1]. The most widely used ones are the center of area method and the center of maxima method.

1 Center of Area Method

Suppose the membership function of a fuzzy data is A(x), the support of A(x) is in the interval [-c, c]. The defuzzified value d_{CA} is defined as the value of variable x for which the area under the curve of A is divided into two equal sub areas.

$$d_{CA}(A) = \frac{\int_{-c}^{c} A(x)xdx}{\int_{-c}^{c} A(x)dx} \tag{3.6}$$

2 Center of Maxima Method

The defuzzified value d_{CM} (A) is defined as the average of the smallest value and the largest value of x for which A(x) equals to the height h(A) of A.

$$d_{CM}(A) = \frac{\inf M + \sup M}{2} \tag{3.7}$$

where $M = \{x \in [-c, c] \mid A(x) = h(A)\}$

3.3 Theorems of Fuzzy Correlation Coefficient

The proposed measurement of FCC satisfies the following theorems.

Theorem1. Fuzzy correlation coefficient depends on input data membership function

According to FCC's definition, suppose $x_{\min}$, $x_{\max}$ are the low and upper bounds of the interval generated by taking α-cut on the input data membership function in x direction, and $y_{\min}$, $y_{\max}$ are the low and upper bounds of the interval generated by taking α-cut on the input data membership function in y direction, then we have

$$f\colon\ [x_{\min}, x_{\max}; y_{\min}, y_{\max}]_\alpha \rightarrow r_{xy}(\alpha)$$

f is a mapping from input data to fuzzy correlation coefficient at possibility level α.

Since different input data membership function will generate different intervals at α level, it is trivial that FCC depends on input data membership function. #

Theorem2. Given a fuzzy data set x, let μ(x) be its membership function. Let X and Y be two fuzzy data sets, x and y are any data in the data sets X and Y. If $\forall \mu(x) \subseteq \mu(y)$ $\forall x \in X, y \in Y$ **, Then** $\mu(FCC(X)) \subseteq \mu(FCC(Y))$

From the condition, we know at each α level, X is a subset of Y, so all the data points which are involved in computing FCC(X) is a subset of the data points involved in computing FCC(Y). The resultant correlation coefficient of X is accordingly the subset of correlation coefficient of Y. #

Theorem3. Fuzzy correlation coefficient $r_{xy}(\alpha)$ ranges from -1 to 1

At each α level, we use Pearson's formula to compute the corresponding measure. Since Pearson's formula always yields a coefficient between -1 and 1, so $-1 \le r_{xy}(\alpha) \le 1$, MAX($r_{xy}(\alpha)$) and MIN ($r_{xy}(\alpha)$) must also be between -1 and +1. #

Theorem4. If the input data are fuzzy numbers, then the fuzzy correlation coefficient is also a fuzzy number.

To be a fuzzy number, a number of requirements must be met [3]:

1. The fuzzy set is a normal set, i.e. the height of the membership function is 1.

2. The α-cut is a closed interval.

3. The support of the membership function must be bounded.

4. ${}^{\alpha_1}A \subseteq {}^{\alpha_2}A$ For $\alpha_1 \ge \alpha_2$ (monotonic)

As for the first condition, since we have restricted in this dissertation to discuss fuzzy number data set, it is straightforward to see that the correlation coefficient which is computed with those fuzzy data is also a normal set.

Since the input data are fuzzy numbers, α-cuts of each fuzzy number are also closed intervals of real numbers. When two fuzzy numbers are cut at α level, the result produces two closed intervals [a b] and [d e]. According to interval analysis theory, we have

[a b] + [d e]=[a+d b+e]

[a b] - [d e]=[a-e b-d]

[a b] * [d e]=[min(ad,ae,bd,be) max(ad, ae, bd, be)]

[a b] / [d e]=[min(a/d, a/e, b/d, b/e), max(a/d, a/e, b/d, b/e)]

Four arithmetic operations on close intervals also produce close intervals. Since the computation of fuzzy correlation coefficient is in essence a combination of four arithmetic operations, the result should also be a close interval. So the second condition for a fuzzy number is satisfied.

According to Theorem 3, the fuzzy correlation coefficient ranges from -1 to 1, hence the value is bounded. Since the input data set A is composed of only fuzzy numbers, then for any two possibility levels $\alpha_1 \geq \alpha_2$, we have ${}^{\alpha_1}A \subseteq {}^{\alpha_2}A$. According to Theorem 2, we have $\mu(FCC({}^{\alpha_1}A)) \subseteq \mu(FCC({}^{\alpha_2}A))$. Hence the fourth condition holds for the fuzzy correlation coefficient.

Since all four conditions are satisfied, our proposed measure for a fuzzy correlation coefficient is a fuzzy number. #

Theorem5. Given a fuzzy data x, let U(x) be its nonspecificity measurement. Let X and Y be two fuzzy data sets, and x and y are any data in data sets X and Y. If $\forall \mu(x) \subseteq \mu(y) \quad x \in X, y \in Y$ **, then U (FCC(X)) < U (FCC(Y))**

Proof: We assume that the input data has a triangle membership function without loss of generality. The support of the membership function is [x_0-x_l, x_0+x_r]

$$U(A)=\frac{1}{h(A)}\int_0^{h(A)}\log_2(A^{\alpha})d\alpha$$

$$=\int_0^1\frac{\ln[(1-\alpha)(x_l+x_r)]d\alpha}{\ln 2}$$

$$=\frac{1}{\ln 2}[\int_0^1\ln(1-\alpha)d\alpha+\int_0^1(x_l+x_r)d\alpha]$$

$$=\frac{1}{\ln 2}[\int_0^1\ln(\alpha)d\alpha+(x_l+x_r)]$$

From the above equation, we can see that if U(x) <U(y), then $(x_l+x_r)<(y_l+y_r)$. In other words, the support of the membership function of x is less than the support of membership function of y. According to Theorem 2, we have $\mu(FCC(X))\subseteq\mu(FCC(Y))$, so the support of FCC(X) at each level is less than the support of FCC(Y).This implies that $U(FCC(X)<U(FCC(Y))$ #

Theorem6. Given a fuzzy data set x, let f(x) denote its fuzziness measurement. Let X and Y be two fuzzy data sets, and x and y be any data in data sets X and Y. If $\forall\mu(x)\subseteq\mu(y)\quad x\in X, y\in Y$**, then f (FCC(X)) < f (FCC(Y))**

Proof: Without loss of generality, let us assume that the input data has triangular membership function. The support of the membership function is [x_0-x_l, x_0+x_r]

$$f(A)=(x_r+x_l)-\int_{x_0-x_l}^{x_0}\left|2\frac{x}{x_l}-2\frac{x_0}{x_l}+1\right|dx-\int_{x_0}^{x_0+x_r}\left|-2\frac{x}{x_r}+2\frac{x_0}{x_r}+1\right|dx$$

$$=x_r+x_l-\frac{x_l}{2}-\frac{x_r}{2}=\frac{x_r+x_l}{2}$$

So if U(x) <U(y), this implies that $(x_l+x_r)<(y_l+y_r)$. In other words, the support of the membership function of x is less than the support of membership function of y.

According to Theorem 2, we have $\mu(FCC(X)) \subseteq \mu(FCC(Y))$, so the support of FCC(X) at each level is less than the support of FCC(Y). This implies that $f(FCC(X) < f(FCC(Y))$. #

3.4 Direct Method

Based on the description of the fuzzy correlation coefficient in (2.3), one can easily see that the computation is in fact an optimization problem with bounded constraints. In theory, the constrained variable metric method and the reduced gradient method are very effective and efficient for solving this type of problems [31]. Those methods are widely used by commercial optimization software. However, the commercial software usually can not take advantage of the characteristics of a specific problem. Furthermore, as the number of unknowns increases according to the size of data set, the search space also becomes very complicated due to the dimension and nonlinearity of the problem. So in this dissertation we intend to develop some approximate methods to obtain computationally efficient solutions. These methods often give fast and effective approximations to complicated problems.

Since general purpose nonlinear optimization software does not take advantage of the characteristic of a specific problem, it is unrealistic to expect them to work efficiently for every kind of nonlinear model. Instead, we try to select a model that fits the specific problem at hand. In this case it is hard to transform the problem to a general Linear Programming or Quadratic Programming format. Hence, general purpose software is not likely to provide an efficient solution.

Widely used commercial nonlinear optimization software packages include

Large-Scale GRG Solver Engine, LINDO systems, MATLAB optimization toolbox etc. If n is the size of the data set, usually the computational complexity of those algorithms is greater than O (n^2).

3.5 Random Search Method

The random search method is a direct simulation approach. Instead of solving the problem analytically, one seeks to find a good solution by randomly sampling the search space.

3.5.1 Monte Carlo Sampling

Monte Carlo methods have been used for a long time; it provides approximate solutions to a number of complex applications by conducting statistical sampling techniques on a computer. Monte Carlo is now used widely in many different fields, from the simulation of complicated prototype physical system to the ordinary life.

It is required by Monte Carlo sampling that the targeted system should be formulated by the probability density functions (pdfs). Once the probability density functions are given, sampling can be taken on those pdfs randomly. The desired result will be obtained by averaging over the number of observations. People usually can compute the statistical error for this result and estimate the number of trials that are needed to achiever a predetermined error.

The typical components included in a Monte Carlo application include: probability distribution functions (pdfs), random number generator, sampling rule design, output computation and error estimation etc.

In the section 2.4 we have presented a constrained optimization model for the computation of fuzzy correlation coefficient. Each fuzzy data could locate on any point in the corresponding interval. Although our concern is about the possibility instead of the probability, the idea of random sampling in Monte Carlo simulation also can be applied to obtain the "possible result".

3.5.2 Algorithm Description

We assume that the actual value of the data follows a uniform distribution in the above defined rectangular region. Hence, at each α level, for every fuzzy observation data points (X_i, Y_i) that are generated randomly (i=1, 2…m), we will get one sample value of r_{xy} (α). If the number of samples is big enough, we will obtain an interval to represent the correlation coefficient at a particular α level.

$$r_{xy}(\alpha) = [\ \min r_{xy}(\alpha),\ \ \max r_{xy}(\alpha)]$$

After the membership function of correlation coefficient is computed, it is not difficult to derive a crisp value which reflects the association between two fuzzy data (this step is usually referred as defuzzification). If a crisp correlation coefficient is desired, then defuzzification should be applied.

In general, the data interval included in computing with a bigger α is a subset of data interval included in the calculation with a smaller α, so the above defined correlation coefficient provides us a complete view of the data characteristics with different uncertainty.

The algorithm can be described in pseudo code as follows:

```
1. Let $\alpha$=0.0
2. Obtain $[X_{\alpha}^{L}, X_{\alpha}^{U}]_i$ $[Y_{\alpha}^{L}, Y_{\alpha}^{U}]_i$. i=1, 2…n (size of the data set)
3. Randomly select a crisp point from each fuzzy region
       $x_i = \lambda X_{\alpha i}^{L} + (1-\lambda) X_{\alpha i}^{U}$
       $y_i = \lambda Y_{\alpha i}^{L} + (1-\lambda) Y_{\alpha i}^{U}$
   i=1, 2…n, $\lambda$ is a random number between 0 and 1
4. Compute the Pearson correlation coefficient (PCC)
5. Store this PCC value in an array
6. Repeat (3), (4), (5) a predetermined number of times
7. Compute $\max r_{xy}(\alpha) = \max_i r_{xyi}(\alpha)$ and $\min r_{xy}(\alpha) = \min_i r_{xyi}(\alpha)$
8. Increment $\alpha = \alpha + \Delta\alpha$
9. Repeat (2) ~ (8) until $\alpha$=1.0
10. Defuzzify using the center of area approach if a crisp value is desired.
```

3.5.3 Computational Complexity

For the sake of simplicity, we consider every addition, subtraction, multiplication, division, and square root as one indistinguishable operation. Suppose the number of fuzzy data in the data set is n. If the FCC is computed with L different α levels, for each level K random samplings are taken. Step 2 needs 2*4*n=8n operations, and step 3 needs 2*3n=6n. Step 4 needs 10n+3 operations, and step 10 takes 3L-1 operations. The total number of operations = (8n+ (6n+10n+3)*K)*L+3L-1= (16K+8) Ln+3KL+3L-1.

So the computational complexity of the above random sampling method is O (LKn). In general L is not small and K cannot be a small number in order to get good results. So this method is not efficient with big data set for real life applications.

3.6 Approximate Bounds Method

In often times, the exact value for the fuzzy correlation coefficient is not needed, only the bound for the FCC is desired. The bound gives the instinct where the FCC could lie.

3.6.1 The Derivation of Approximate Bounds

We can derive an approximate bound for r_{xy} (α).

$$r_{xy} = \frac{\sum_{i=1}^{n}\left(x_i - \overline{x}\right)\left(y_i - \overline{y}\right)}{\sqrt{\sum_{i=1}^{n}\left(x_i - \overline{x}\right)^2}\sqrt{\sum_{i=1}^{n}\left(y_i - \overline{y}\right)^2}}$$

at each α level, x_i, y_i are the i-th random variables distributed in the interval $[X_\alpha^L, X_\alpha^U]_i$ $[Y_\alpha^L, Y_\alpha^U]_i$, here we assume that they follow the uniform distribution without losing generality, and n is the size of data set.

Let (x_{i0}, y_{i0}) be the i-th data point when α equals to 1. For large data set, according to the Central Limit Theorem we have

$$\bar{x} \approx \frac{\sum_{i=1}^{n} x_{i0}}{n}$$

$$\bar{y} \approx \frac{\sum_{i=1}^{n} y_{i0}}{n}$$

This average value for x_{i0} and y_{i0} can be thought of as constants for a given data set. Let $X_i = x_i - \bar{x}$ and $Y_i = y_i - \bar{y}$, also let $\Delta X_i = X_i - X_{i0}$, $\Delta Y_i = Y_i - Y_{i0}$ then Pearson's correlation coefficient can be reformulated as below.

$$r_{xy} = \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n} X_i^{2}} \sqrt{\sum_{i=1}^{n} Y_i^{2}}} = r_{xy}(X_0 + \Delta X, Y_0 + \Delta Y) \qquad (3.8)$$

where (X_0, Y_0) denotes all data points in this data set when α equals to 1. Expanding r_{xy} in a Taylor series around the values (X_0, Y_0) and omitting those terms above the second order in the variation, (3.8) becomes:

$$r_{xy} \approx r_{xy}(X_0, Y_0) + \sum_{i=1}^{n} \left(\frac{\partial r_{xy}}{\partial X_i} \right) \Bigg|_{X_0, Y_0} \Delta X_i + \sum_{i=1}^{n} \left(\frac{\partial r_{xy}}{\partial Y_i} \right) \Bigg|_{X_0, Y_0} \Delta Y_i$$

$$\frac{\partial r_{xy}}{\partial X_i}\Bigg|_{X_0, Y_0} = \frac{Y_i \sqrt{\sum_{i=1}^{n} X_i^2} - \left(\sum_{i=1}^{n} X_i Y_i \right) \times X_i \times \left(\sum_{i=1}^{n} X_i^2 \right)^{-\frac{1}{2}}}{\sqrt{\sum_{i=1}^{n} Y_i^2} \times \left(\sum_{i=1}^{n} X_i^2 \right)} \Bigg|_{X_0, Y_0} = \frac{Y_i (\sum_{i=1}^{n} X_i^2) - X_i \left(\sum_{i=1}^{n} X_i Y_i \right)}{\sqrt{\sum_{i=1}^{n} Y_i^2} \times \left(\sum_{i=1}^{n} X_i^2 \right)^{1.5}} \Bigg|_{X_0, Y_0}$$

$$\frac{\partial r_{xy}}{\partial Y_i}\Bigg|_{X_0, Y_0} = \frac{X_i \sqrt{\sum_{i=1}^{n} Y_i^2} - \left(\sum_{i=1}^{n} X_i Y_i \right) \times Y_i \times \left(\sum_{i=1}^{n} Y_i^2 \right)^{-\frac{1}{2}}}{\sqrt{\sum_{i=1}^{n} X_i^2} \times \left(\sum_{i=1}^{n} Y_i^2 \right)} \Bigg|_{X_0, Y_0} = \frac{X_i (\sum_{i=1}^{n} Y_i^2) - Y_i \left(\sum_{i=1}^{n} X_i Y_i \right)}{\sqrt{\sum_{i=1}^{n} X_i^2} \times \left(\sum_{i=1}^{n} Y_i^2 \right)^{1.5}} \Bigg|_{X_0, Y_0}$$

Let
$$\Delta X_{ir} = X_{iU\alpha} - X_{i0}$$
$$\Delta Y_{ir} = Y_{iU\alpha} - Y_{i0}$$
$$\Delta X_{il} = X_{i0} - X_{iL\alpha}$$
$$\Delta Y_{ir} = Y_{i0} - Y_{iL\alpha}$$

Max:

$r_{xy}(\alpha) \approx$

$$r_{xy}(X_0,Y_0) + \sum_{i=1}^{n} \max(\left(\frac{\partial r_{xy}}{\partial X_i}\right)\Bigg|_{X_0,Y_0} \Delta X_{ir}, \left(\frac{\partial r_{xy}}{\partial X_i}\right)\Bigg|_{X_0,Y_0} \Delta X_{il}) + \sum_{i=1}^{n} \max(\left(\frac{\partial r_{xy}}{\partial Y_i}\right)\Bigg|_{X_0,Y_0} \Delta Y_{ir}, \left(\frac{\partial r_{xy}}{\partial Y_i}\right)\Bigg|_{X_0,Y_0} \Delta Y_{il})$$

Min:

$r_{xy}(\alpha) \approx$

$$r_{xy}(X_0,Y_0) + \sum_{i=1}^{n} \min(\left(\frac{\partial r_{xy}}{\partial X_i}\right)\Bigg|_{X_0,Y_0} \Delta X_{ir}, \left(\frac{\partial r_{xy}}{\partial X_i}\right)\Bigg|_{X_0,Y_0} \Delta X_{il}) + \sum_{i=1}^{n} \min(\left(\frac{\partial r_{xy}}{\partial Y_i}\right)\Bigg|_{X_0,Y_0} \Delta Y_{ir}, \left(\frac{\partial r_{xy}}{\partial Y_i}\right)\Bigg|_{X_0,Y_0} \Delta Y_{il})$$

Assume that $\Delta X_{1l} = \Delta X_{2l} = \ldots = \Delta X_{Nl} = \Delta X_{1r} = \Delta X_{2r} = \ldots = \Delta X_{Nr} = \Delta X$, and $\Delta Y_{1l} = \Delta Y_{2l} = \ldots = \Delta Y_{Nl} = \Delta Y_{1r} = \Delta Y_{2r} = \ldots = \Delta Y_{Nr} = \Delta Y$

We will have

Max:

$$r_{xy}(\alpha) \approx r_{xy}(X_0,Y_0) + \Delta X \frac{\sum_{i=1}^{n} \left| Y_i \sum_{i=1}^{n} X_i^{\ 2} - X_i \sum_{i=1}^{n} X_i Y_i \right|}{\left(\sum_{i=1}^{n} Y_i^2\right)^{0.5} (\sum_{i=1}^{n} X_i^2)^{1.5}} + \Delta Y \frac{\sum_{i=1}^{n} \left| X_i \sum_{i=1}^{n} Y_i^{\ 2} - Y_i \sum_{i=1}^{n} X_i Y_i \right|}{\left(\sum_{i=1}^{n} X_i^2\right)^{0.5} (\sum_{i=1}^{n} Y_i^2)^{1.5}}$$

Min:

$$r_{xy}(\alpha) \approx r_{xy}(X_0,Y_0) - \Delta X \frac{\sum_{i=1}^{n} \left| Y_i \sum_{i=1}^{n} X_i^{\ 2} - X_i \sum_{i=1}^{n} X_i Y_i \right|}{\left(\sum_{i=1}^{n} Y_i^2\right)^{0.5} (\sum_{i=1}^{n} X_i^2)^{1.5}} - \Delta Y \frac{\sum_{i=1}^{n} \left| X_i \sum_{i=1}^{n} Y_i^{\ 2} - Y_i \sum_{i=1}^{n} X_i Y_i \right|}{\left(\sum_{i=1}^{n} X_i^2\right)^{0.5} (\sum_{i=1}^{n} Y_i^2)^{1.5}}$$

(3.9)

3.6.2 Computational Complexity

Using the same assumption as in the random sampling method, the total number of operations of the above formula is 10n+3+(2(4n+4)+2)*L+3L-1= (10+8L) n+13L+2, where L is the number of possibility levels and n is the number of data points. The computational complexity of the approximate bounds is O (Ln).

3.7 Heuristic Method

In this section we will consider a heuristic method. The object of the heuristic method is to use reasoning to quickly determine the location of data points that will contribute to the minimum and maximum values of the fuzzy correlation coefficient.

3.7.1 Algorithm Description

This method is best illustrated in Figure 3.1. First we set α value to 1 and pick the corresponding data points. Then we apply the linear regression model to those data so we will obtain a straight line which represents the linear relationship of the data set (α =1) in the minimum mean square error sense. This is shown as the dotted line in Figure 3.1.

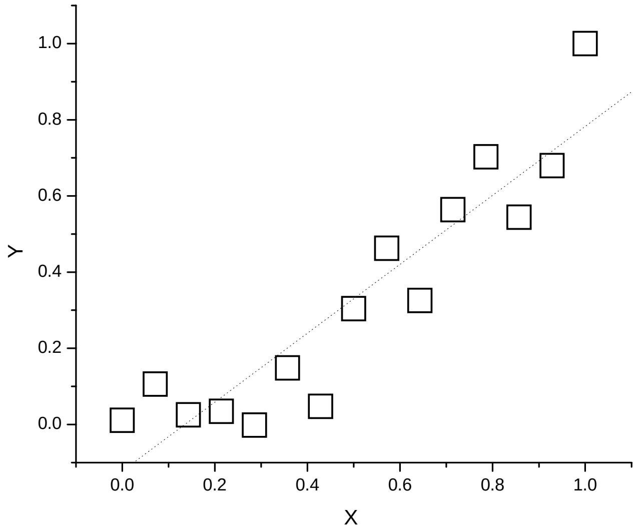

Fig. 3.1 Heuristic method illustration

Next we need to explore the impact of data fuzziness on the regression model. Here we have several cases:

1) The regression line goes through the region composed of $[X_{\alpha}^{L}, X_{\alpha}^{U}]_j$ $[Y_{\alpha}^{L}, Y_{\alpha}^{U}]_j$. Depending on the sign of the regression line slope, the point contributing to the maximum (or minimum) of $r_{xy}(\alpha)$ is any point on the straight line and in the region. Hence the point contributing to the minimum (or maximum) must be the corner point of the region that has the largest deviation from the regression line in y coordinate.

2) If the regression straight line does not pass through the region just like the case for most fuzzy data in Figure 3.1, then one can use the corner points of the region. It is intuitively reasonable that the point which has the smallest or largest deviation from the line contributes to the maximum (or minimum) value of $r_{xy}(\alpha)$, and the corner point

with the largest (or smallest) deviation from the line contributes to minimum (or maximum) value.

This process could be expressed by the following pseudo code:

1. Set α=1.0

2. Compute the regression line $y = a + bx$ based on the crisp data points which can be obtained by setting the possibility level α of the data set to 1.

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

$$b = \frac{\sum_{i=1}^{n} x_i y_i - (\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)/n}{\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x)^2/n}$$

$$a = \bar{y} - b\bar{x}$$

where n is the size of the data set.

3. Set up two sets P and Q, data points in P set contribute to the maximum $r_{xy}(\alpha)$ and data points in Q set contribute to the minimum $r_{xy}(\alpha)$ respectively. Eventually all data points belong to either P or Q depending on their locations. The two sets are initially empty.

4. Find the corner points of each data region and determine if they contribute to the maximum set or the minimum set. For example, if the data region is above the regression line, then the upper corner points contribute to minimum correlation and the lower corner points contribute to the maximum correlation.

5a. If the regression line travels through the data region, we have

$$P = P + \{\frac{(X_\alpha^L + X_\alpha^U)}{2}, \frac{(y_\alpha^L + y_\alpha^U)}{2}\}$$

$$Q = Q + \{(X, Y)\}$$

where corner point (X, Y) distance to regression line=

$MAX\{| y_\alpha^L - Y_\alpha^L |, \left|y_\alpha^L - Y_\alpha^U\right|, \left|y_\alpha^U - Y_\alpha^L\right|, \left|y_\alpha^U - Y_\alpha^U\right|\}$

5b. If the regression lines is above or below the data region, we have

$$P = P + \{(X_1, Y_1)\}$$

where corner point (X_1, Y_1) distance to regression line =

$MIN\{| y_\alpha^L - Y_\alpha^L |, \left|y_\alpha^L - Y_\alpha^U\right|, \left|y_\alpha^U - Y_\alpha^L\right|, \left|y_\alpha^U - Y_\alpha^U\right|\}$

and $Q = Q + \{(X_2, Y_2)\}$

where Corner point (X_2, Y_2) distance to regression line=

$MAX\{| y_\alpha^L - Y_\alpha^L |, \left|y_\alpha^L - Y_\alpha^U\right|, \left|y_\alpha^U - Y_\alpha^L\right|, \left|y_\alpha^U - Y_\alpha^U\right|\}$

6. Compute the maximum and minimum values of the fuzzy correlation coefficient.

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$\bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

$$Max \qquad r_{xy}(\alpha) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2 \sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2}} \qquad (x_i, y_i) \in P$$

$$Min \qquad r_{xy}(\alpha) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2 \sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2}} \qquad (x_i, y_i) \in Q$$

where n is the size of the data set.

7. Increment the α-cut value, $\alpha = \alpha + \Delta\alpha$.

8. Repeat step (4) through (7) until α=1.0

9. Defuzzification if a crisp value is required.

$$d_p(r) \equiv \frac{\sum_{k=1}^{L} r(\alpha_i)\alpha_i}{\sum_{K=1}^{L} \alpha_i}$$

where L is the number of possibility levels involved in the computation.

In theory, the fuzzy correlation coefficient ranges from -1 to 1. When searching for a possible solution for this problem, we can assume that those points with the biggest deviation from the regression line will contribute to the minimum absolute correlation coefficient value and those points with the smallest deviation will contribute to the maximum absolute value.

3.7.2 Computational Complexity

Now let us compute the total operation of this algorithm. Step 2 needs 6n+9 operations, step 4 needs 8n operations, step 6 needs 2*(10n+3)=20n+6 operations, step 7 needs one operation and step 9 needs 3L-1 operations. So the total number of operations =6n+9+L*(8n+20n+6+1) +3L-1= (28L+6) n+10L+8. It is obvious that the time required is only O (Ln); however this solution is only an approximate one.

CHAPTER 4

Simulation Studies of Fuzzy Correlation

In this chapter a number of test data sets are designed to test the proposed methods. Suppose we want to know the correlation coefficient of a set of n fuzzy observations. Assume that each observation has a symmetric triangular membership function for convenience without loss of generality. The algorithms presented in the last chapter were applied to the data sets; the fuzzy correlation coefficients were computed at each possibility level.

4.1 Data Preparation

Figure 4.1 is the data set we will use in this simulation study. This data set is generated by the relationship $y = x^2$ with additive white Gaussian noise $N(0, \sigma^2 = 1)$. If all data are crisp, the correlation coefficient can be computed directly by Pearson's formula. The crisp data points are fuzzified by assuming that a membership function exists for both the x and y coordinates. The membership function is assumed to be triangular. The support of the membership function is set at 10% of the maximum data set value. The points appearing on the curve in Figure 4.1 are the centers of each membership function; i.e. the original crisp value.

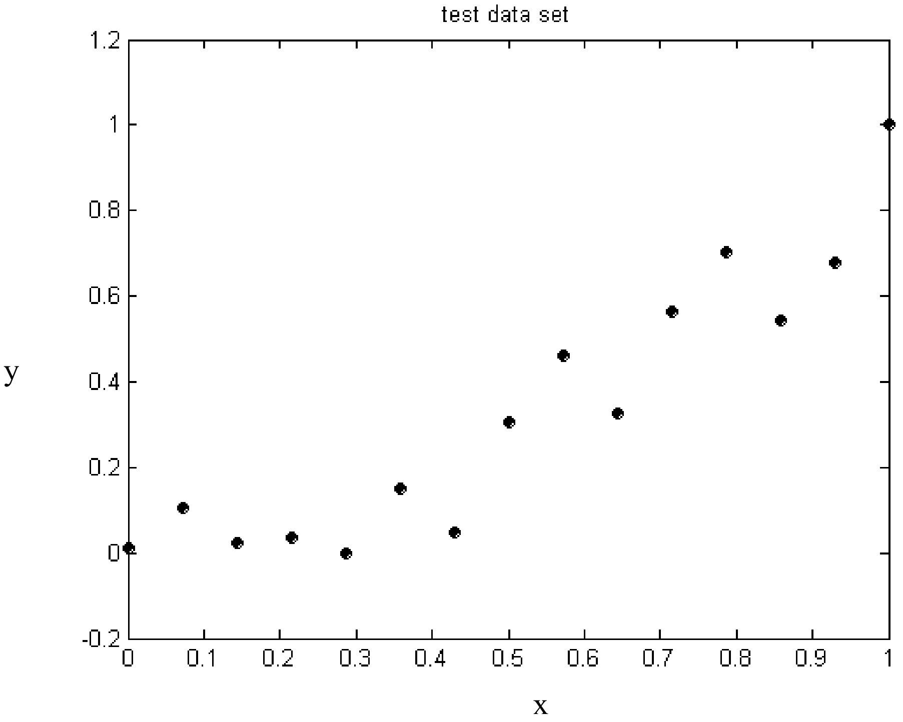

Fig. 4.1 Test data set 1: y=x^2+N (0, $\sigma^2 = 1$)

4.2 Complexity Comparison

In the last chapter, the computational complexity of each method was presented. In summary, the computational complexity of the random sampling method, the heuristic method and the analytical method are O (n) and the direct method is usually bigger than O (n^2). In this section, some empirical results are given via simulation.

The algorithms were executed on an HP pavilion zt1270 notebook at 1.6 G HZ with 256M SRAM. By observing the execution time of algorithms, we may obtain some intuition about the performance of the proposed methods.

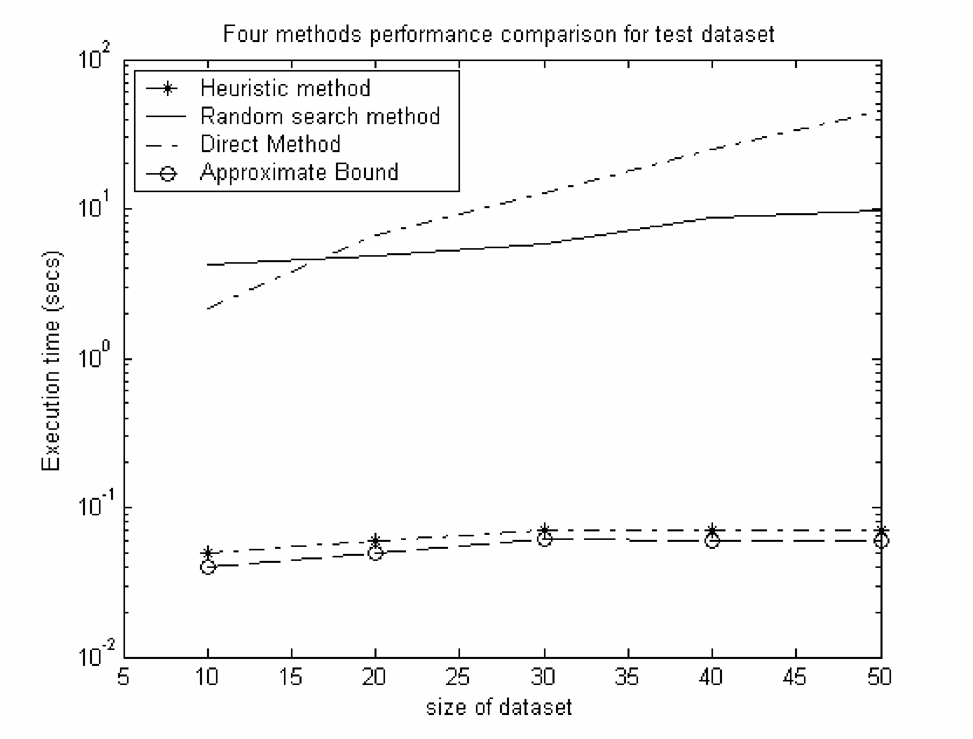

Fig. 4.2 Computational complexity comparison of four methods

Figure 4.2 gives the performance comparison of four methods respectively. The X axis of those plots represent the size of the data set, the y axis represent the execution time of these algorithms. The dash-dot line with star curve, solid line curve, dash-dot line curve and dash line with circle curve describe the performance of the heuristic method, the random search method, the direct method using the MATLAB optimization toolbox, and the approximate bounds method respectively.

In general, the direct method needs more time to finish the task. This is expected as the execution time grows significantly with increasing size of the problem. Heuristic method gives solution that is quite close to the direct method, but with much less time. Approximate bound method is an analytical method, and it requires the least execution time. The random search method has a lackluster performance. When the data set is large, the heuristic method is a powerful tool to estimate the fuzzy correlation coefficient.

4.3 Fuzzy Correlation Coefficient Computed by Four Methods

Figure 4.3 shows the simulation result for the fuzzy correlation coefficient for this fuzzy data set. It is clear from Figure 4.3 that the correlation coefficient is a single value when α=1 which means that there is no uncertainty at all in the data set. In that case the fuzzy data is boiled down to the crisp data, therefore the fuzzy estimation degenerates to the classic Pearson's correlation coefficient. In other words, Pearson's correlation coefficient is a special case of the fuzzy correlation coefficient. When α decreases, the uncertainty of the data increases. So the correlation coefficient becomes more widely distributed. This membership function shape intuitively matches our expectation.

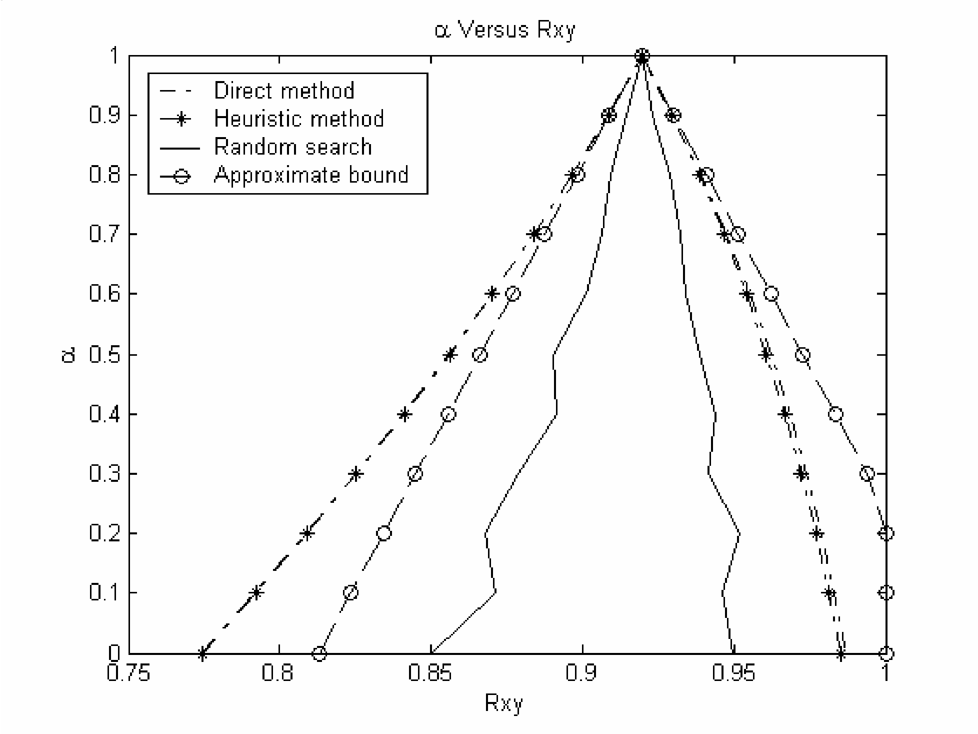

Fig. 4.3 Fuzzy correlation at data membership support = 10% ×max(y)

There are four curves in Figure 4.3, we assume that the solution provided by the standard optimization software gives the correct solution, for comparison the heuristic method yields a solution that is very close to this, the random search method solution is apparently different from the other two. The jagged curves can be attributed to insufficient samples. The approximate bounds described by the dash-circled curve behave worst since all the high order terms are omitted.

4.4 Data Membership Function Impaction: Different Support

In this simulation, we explore the effect of the fuzzy correlation coefficient membership function shape due to different support of the membership function of the data points. The support of the membership functions for the data points were increased from 5% of the maximum measurement of data set to 20%. The computing process was repeated and results were shown in Figure 4.4 and Table 4.1.

Table 4.1 Comparison of the four methods

Method	5%	10%	15%	20%
Direct Method	[0.856, 0.961]	[0.77, 0.985]	[0.68,0.99]	[0.57, 1]
Heuristic Method	[0.856,0.960]	[0.77, 0.98]	[0.68,0.99]	[0.572,1]
Random Search	[0.883,0.940]	[0.847,0.952]	[0.8,0.96]	[0.77,0.96]
Approximate Bounds	[0.865,0.97]	[0.82, 1]	[0.79,0.96]	[0.71, 1]

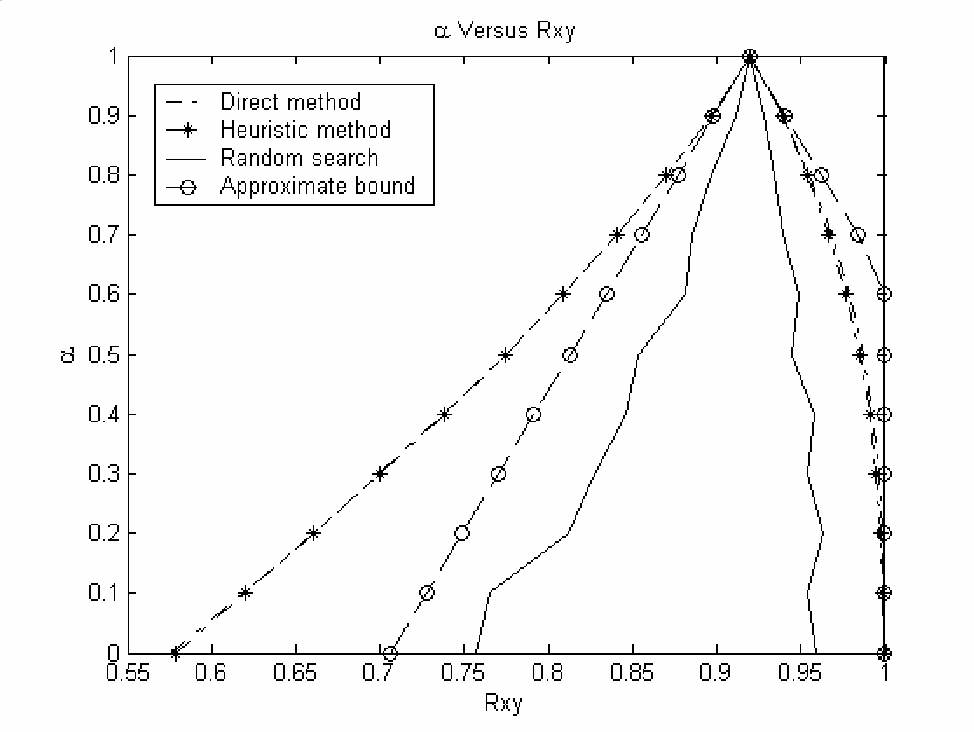

Fig. 4.4 Fuzzy correlation at data membership support = 20% × max(y)

According to Table 4.1, the support of the fuzzy correlation increases with the increasing support of the input data. In Figure 4.4 it is observed that the spread of the fuzzy correlation of 20% support case is bigger than the 10% support case. According to the Theorem 2 in the last chapter, these simulation results are exactly within our expectation. The performance of the four methods of 20% support case is similar to that in the 10% support case. The heuristic method is very close to direct method. The random search method and the approximate bounds method show somehow slightly different curves.

4.5 Data Membership Function Impaction: Different Shape

In this experiment, we present the effect of different membership function shapes of the data set on the membership function shapes of the fuzzy correlation coefficient. Instead of the membership function of the input data being of the triangle shape, we use a trapezoidal shape, we then apply the algorithms proposed in the last section, and the results are shown in Figure 4.5.

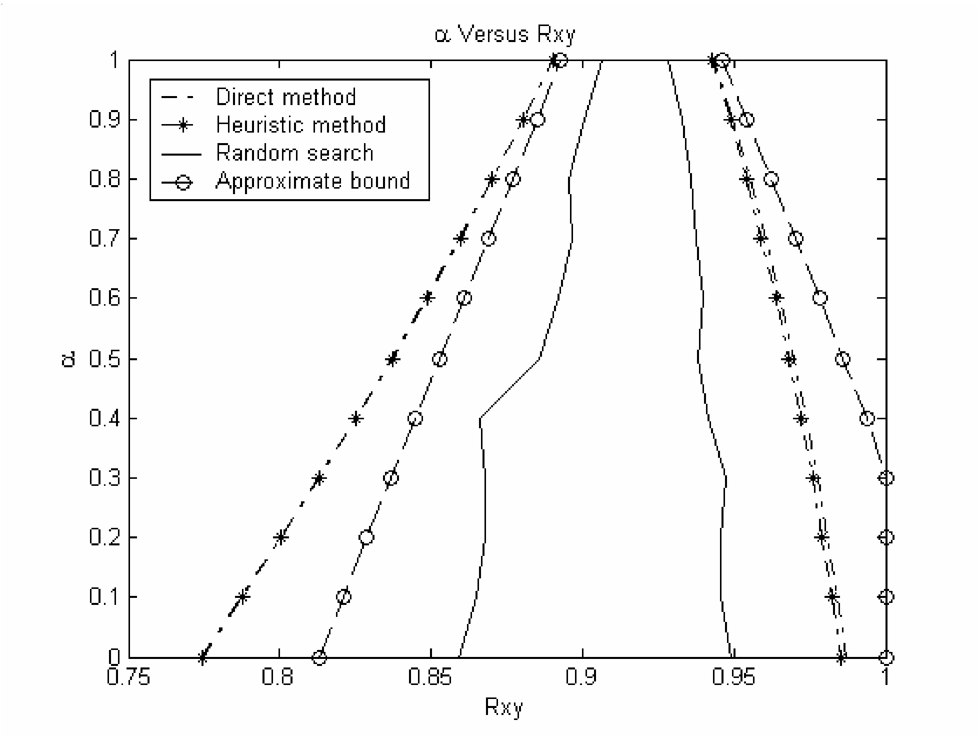

Fig. 4.5 Fuzzy correlation: trapezoidal data membership support =10% ×max(y)

When α equals to 1, since the input data is an interval, accordingly an interval will appear in the fuzzy correlation coefficient. In this case, we can not decide where the exact input data locates at any possibility level, classical correlation theory does not work at all, and fuzzy correlation has to be resorted to. Again from the plot we find that the heuristic method works well when compared to the standard optimal software.

4.6 Nonspecificity & Fuzziness

If we return back to the triangle shape membership function and increase the support of the membership function from 5% to 25%, and repeat the same computational steps we used before. The 20% support case is shown in Figure 4.4. Comparing the Figure 4.3 to Figure 4.4, it is obvious the two legs of FCC become more widely spread in 20% support case than 10% support case which is intuitively reasonable. The nonspecificity measurement of FCC is shown in Table 4.2.

Table 4.2 Nonspecificity of input data and FCC with different membership function support

Support of input data	Nonspecificity of input data	Nonspecificity of FCC
5%	0.027	0.0565
10%	0.0532	0.1092
15%	0.0785	0.1584
20%	0.1030	0.2047
25%	0.1268	0.2491

According to Table 4.2, the nonspecificity measurement of input data increases monotonically with the increasing support of its membership function, so does the nonspecificity measurements of its FCC. This relationship can be plotted in Figure 4.6, the solid line is the regression line that predicts the nonspecificity of FCC based on the nonspecificity of input data, this line is represented by Y= 0.00578+1.92799X.

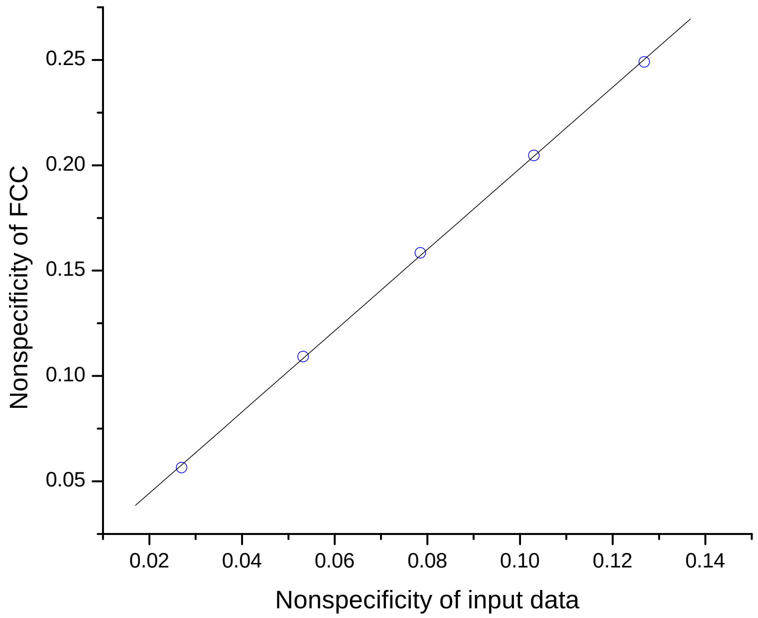

Fig. 4.6 Nonspecificity relationship between input data and FCC

Using the same data set as in nonspecificity computation, the fuzziness measurement of FCC is summarized in Table 4.3.

Table 4.3 Fuzziness of input data and FCC with different membership function support

Support of input data	Fuzziness of input data	Fuzziness of FCC
5%	0.025	0.0532
10%	0.05	0.1063
15%	0.075	0.1590
20%	0.1	0.2113
25%	0.125	0.2648

Observing Table 4.3, when the support of input data increases, both the fuzziness measure of the input data and the FCC increase. Also we can find the fuzziness of fuzzy correlation coefficient is bigger than the fuzziness of the original data. This relationship between the fuzziness of input data and the fuzziness of FCC can be plotted in Figure 4.7, the solid line is the regression line that predicts the fuzziness of FCC based on the fuzziness of input data, this line is represented by Y=0.0046+2.1128X.

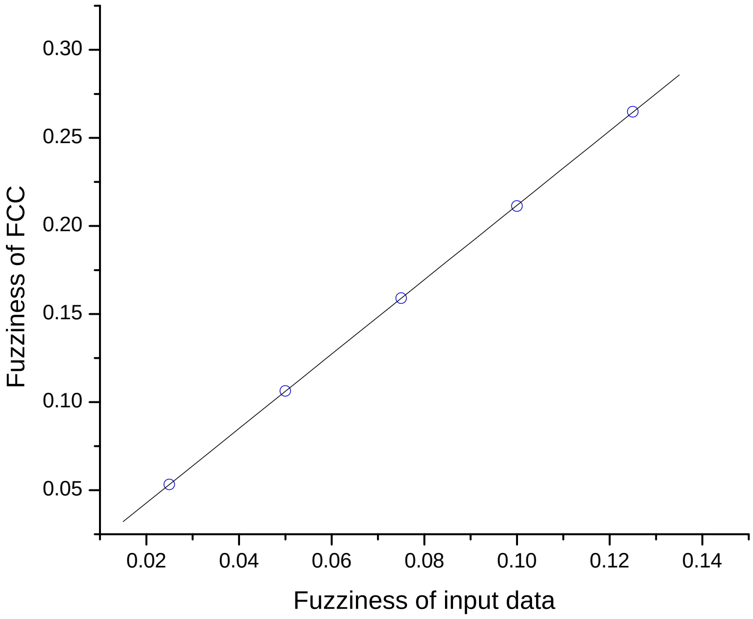

Fig. 4.7 Fuzziness relationship between input data and FCC

4.7 Defuzzified Values

If a crisp correlation coefficient is desired, the fuzzy correlation coefficient can be defuzzified to yield a crisp value. This has been done for this test data set. Some of the defuzzified values are listed in Table 4.4.

Table 4.4 Defuzzified values of different shape membership function

Data Set	Pearson's method	Direct method	heuristic method	random search
10% support(triangle)	0.9195	0.8911	0.8902	0.9008
10% support(trapezoid)	0.9195	0.9203	0.9190	0.9185
10% support (bell curve)	0.9195	0.9040	0.9033	0.9144

All the data sets have membership functions with the same support but different shapes. The first column of the above table lists the different cases we have just tested; the second column is Pearson's correlation coefficient for each data set if they are crisp. The next several columns provide defuzzified values based on the direct method, the heuristic method and the random search method respectively.

Defuzzified correlation coefficient value of a fuzzy data set is surely different from their crisp counterparts. Since it reflects the inherent data characteristic in the context of fuzzy environment, it can provide us a more realistic view about the data set.

Exploring the simulation result, we find that the correlation coefficient membership function is in essence a nonlinear mapping from membership function of fuzzy data according to extension principle. Since our membership function of the data set is a fuzzy number, the correlation coefficient is also a fuzzy number with the domain in the interval [-1, +1]. If the fuzzy correlation coefficient is +1 or -1, the two random variables co-vary perfectly.

4.8 Varied Support Data Set

In the previous simulations we set a fixed support for all data points of a data set. Now let us consider a varied support fuzzy data set. We still use the symmetric triangular membership function for each data point. Suppose all the data points except one have a 10% uncertainty, but one data point has a 20% uncertainty. The resultant fuzzy correlation coefficient is presented in the next four graphs for each of the four methods.

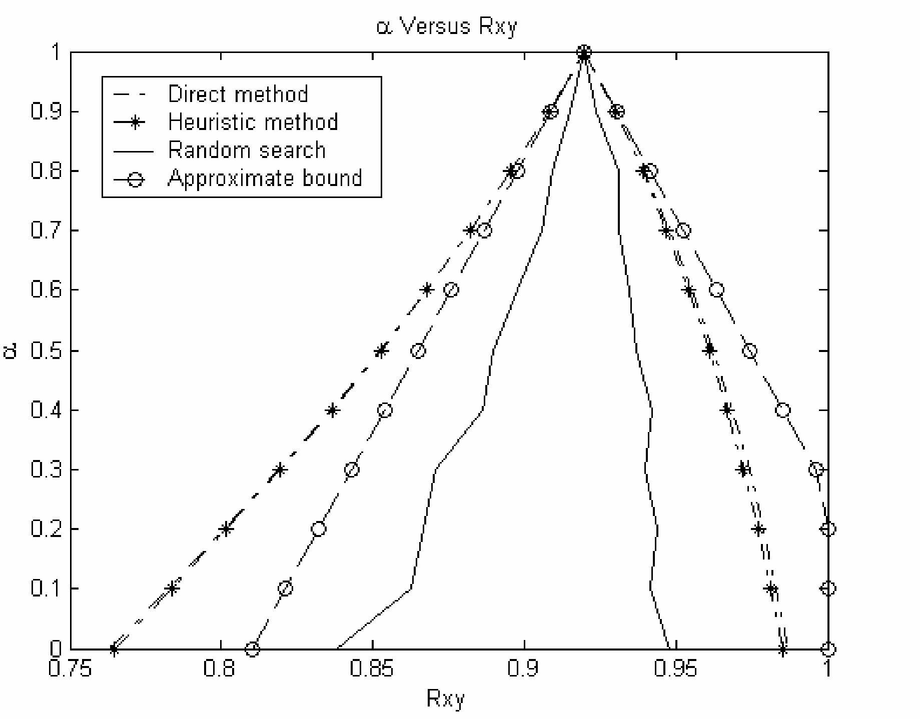

Fig. 4.8 Varied support fuzzy correlation

In each of the four cases, the current data set yields a FCC that is between all data points with 10% uncertainty and the case when all data points have 20% uncertainty.

4.9 Crisp Data and Fuzzy Correlation

Suppose we are measuring the correlation between two variables x and y drawn from a system with measurement errors. Suppose we sample the system with N points and compute the Pearson's correlation coefficient. Now if we draw another pair x_{N+1}, y_{N+1} and include the new pair in the computation of the correlation coefficient, it is quite possible that the computed correlation coefficient will be different from the original one. In fact if we take a random sample of the collected data points and compute the relevant correlation coefficient, we will find that they form some kind of distribution for the correlation coefficient values. Hence a single computation of the correlation coefficient does not actually reflect the real distribution. The actual correlation between the two random variables is best illustrated by the distribution.

The data set can be thought of as a fuzzy system. The correlation coefficient of the data set generated by this system is also fuzzy; we proposed the following method to derive a membership function of the correlation coefficient. Suppose the size of the data set is N, first pick three data points and compute the corresponding correlation coefficient, then continue this process till all the combinations of three data points are finished. Then pick 4,5…,N data points and following the same procedure, finally we will obtain a whole set of r_{xy}. Since this set of correlation coefficients come from the same system, if we consider the r_{xy} as independent variable and normalized the occurrence of each r_{xy} as the dependent variable, the relationship should reflects the characteristic of the system.

As we have known, the probability property is a subset of the possibility property for a system, so the normalized distribution function can also be formulated as the membership function of the correlation. According to the computation procedure, we can see that the distribution is a normal fuzzy set , its α-cut is a closed interval for every $\alpha \in [0, 1]$ and the support is bounded within [-1 1], so it is also a fuzzy number. This measurement is more informative than a single correlation value since it provides us not only with the value but also the occurrence distribution of each value. When α equals to 1 that means all the data information must be available to get the correlation, the range of the data correlation becomes wider with more missing data, possibility level α will decrease in this case. Pedryz's fuzzy correlation coefficient is also based on crisp observations; however, his method measures the correlation of two fuzzy sets so that the result relied heavily on the definition of membership functions of those two fuzzy sets. No systematic methods for this have been developed up to now.

This combinatorial problem is one of the classes of "NLP" problem. It is intractable for large data set size N. Monte Carlo simulation is used to solve this problem.

As an example, assume we have a data set $y=-0.2x^2+5x$ and the samples are corrupted by additive measurement noise which is a zero mean uniform noise with a standard deviation σ of 2, and this additive noise can be denoted as $\mu(0, \sigma^2 = 4)$. Repeatedly select the data points from the original data set and compute the correlation coefficient of the selected data set, and then we can obtain the probability density function of the FCC. We refer to this data set as test data set 2. The data set and the result are shown in Figure 4.9 and Figure 4.10 respectively. It can be observed that the correlation coefficient of this system is an approximately symmetric distribution with the mean FCC value of 0.91.

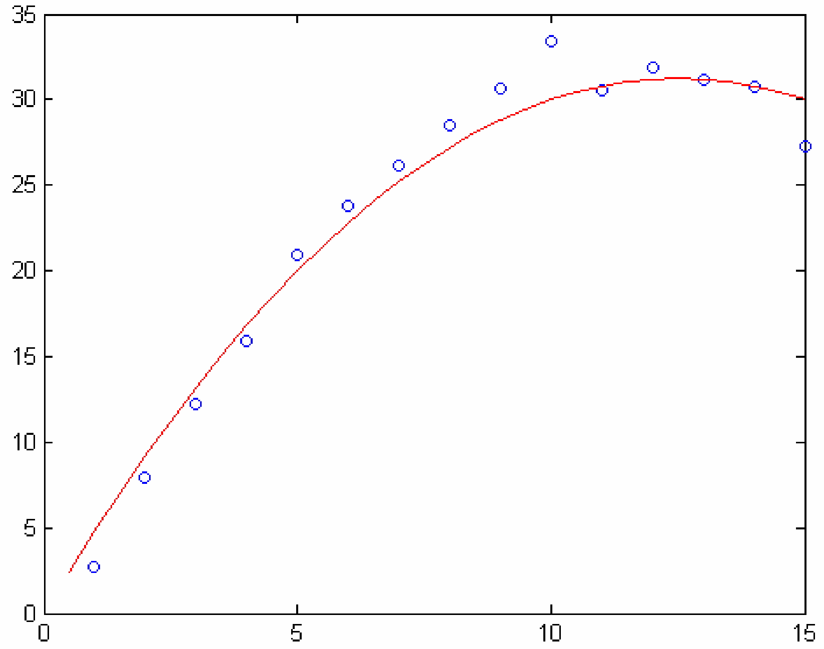

Fig. 4.9 Test data set 2: y=-0.2x^2+5x+µ $(0,\sigma^2 = 4)$

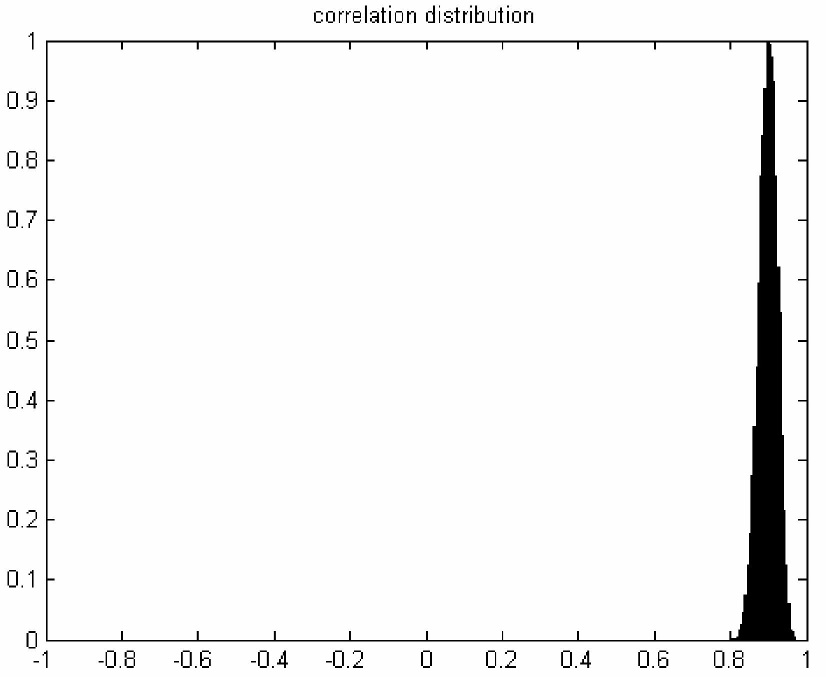

Fig. 4.10 Correlation probability density function of test data set 2

Figure 4.11 is the membership function of the fuzzy correlation coefficient. It is observed that this fuzzy correlation can be considered an approximation of the true probability density function of the true crisp correlation coefficient. What we also can observe is that the correlation coefficient with the presence of the whole data set is around but not necessary actually at the peak of the fuzzy correlation.

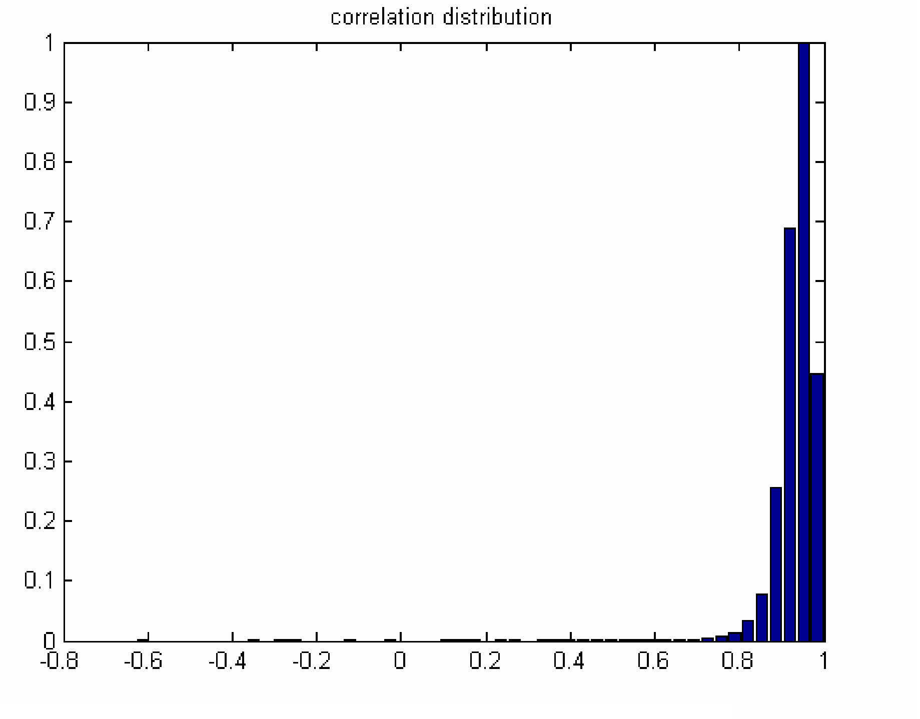

Fig. 4.11 Fuzzy correlation of test data set 2

Consider the system with additive white Gaussian noise N ($\mu = 0, \sigma^2 = 4$). We refer to this data set as test data set 3. The raw data set and the corresponding probabilistic density function of correlation are plotted in Figure 4.12 and Figure 4.13. It can be observed that the correlation probabilistic density function of this data set follows an approximate Gaussian distribution with mean value 0.91 and Figure 4.14 is the fuzzy correlation coefficient of this data set. It is obvious that the fuzzy correlation is very close to the probability density function of the input data set correlation. From the simulation results on test data set 2 and 3, we believe the fuzzy correlation can be used as an approximation of the correlation probabilistic density function of the raw data set. This hypothesis can be experimentally verified by testing more data sets. Note also that as the added measurement noise increases, the pdf of the crisp correlation coefficient (CCC) also gets wider, this of course is expected.

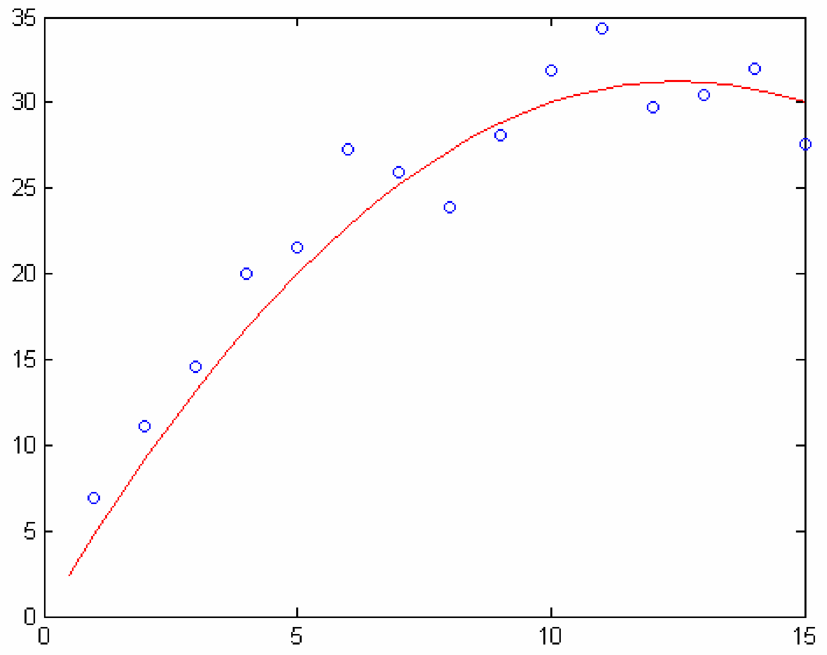

Fig. 4.12 Test data set 3: y=-0.2x^2+5x+N $(0, \sigma^2 = 4)$

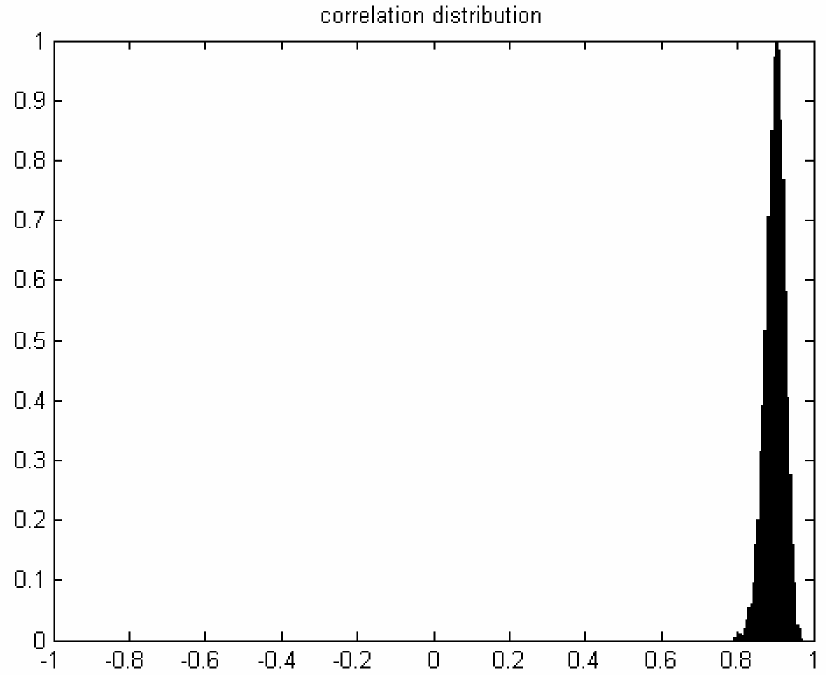

Fig. 4.13 Correlation probability density function of test data set 3

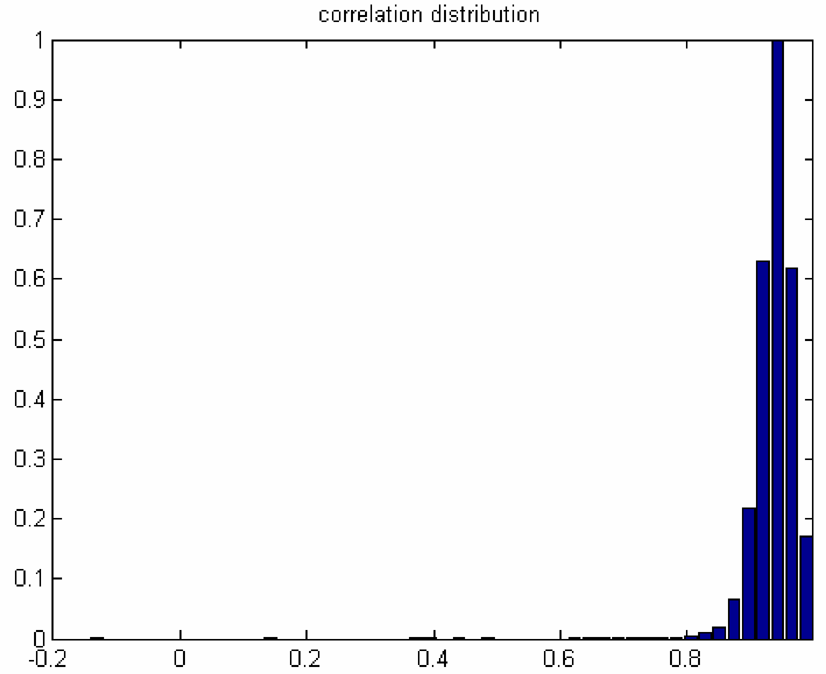

Fig. 4.14 Fuzzy correlation of test data set 3

It can also be observed that the peak of the pdf coincides with the CCC for the noiseless case. In this experiment, the CCC for data set y=-0.2x^2+5x is 0.92, and the peaks of the pdfs in Figure 4.10 and Figure 4.13 are both around 0.92.

4.10 Correlation and Spread

We want to know whether there is any relationship between the support of the fuzzy correlation coefficient and the correlation coefficient of the whole data set. The function we use in this simulation is y=x, with additive zero mean uniform distribution noise. Figure 4.15 and Figure 4.17 are two examples with standard deviation 1 and 2. We refer to those two data sets as test data set 4 and test data set 5 respectively. Figure 4.16 and Figure 4.18 are their corresponding fuzzy correlations.

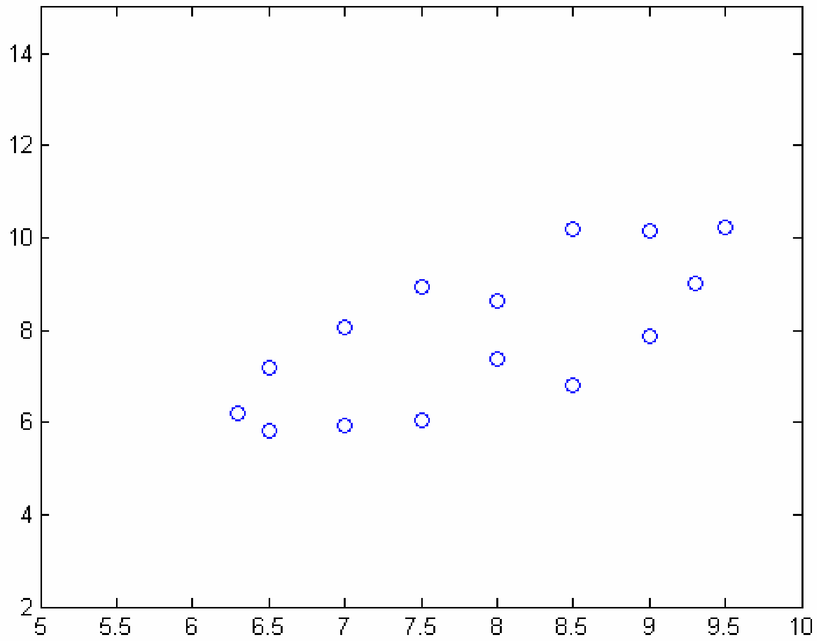

Fig. 4.15 Test data set 4: y=x+μ($0, \sigma^2 = 1$)

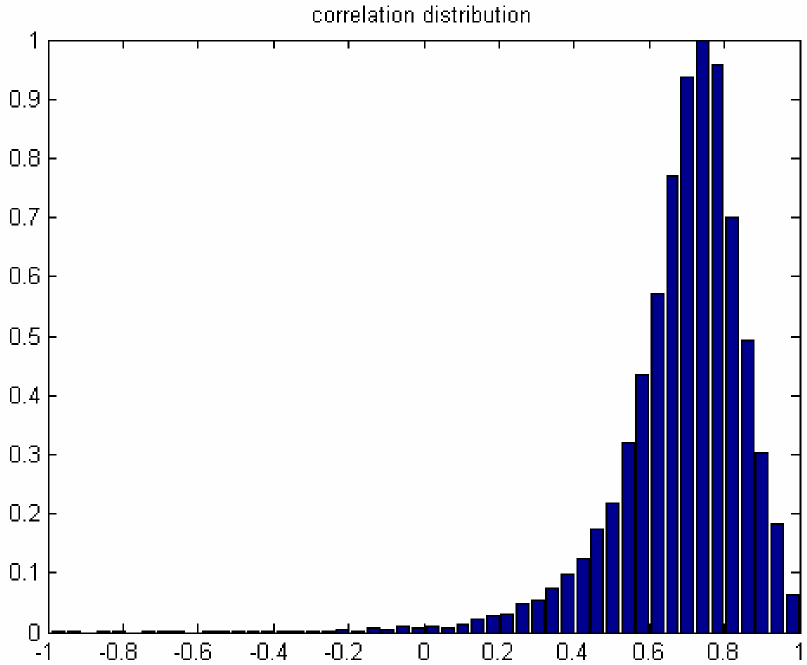

Fig. 4.16 Fuzzy correlation of test data set 4

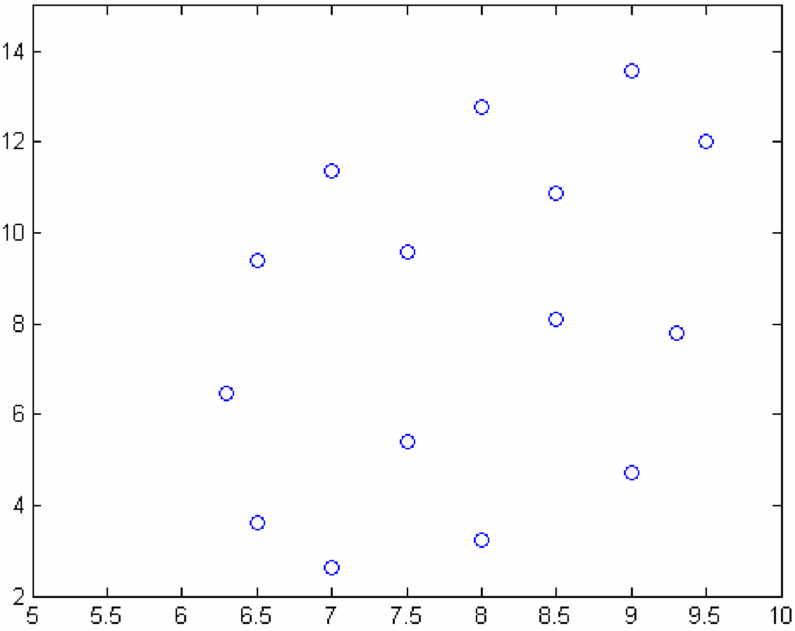

Fig. 4.17 Test data set 5: y=x+μ(0, $\sigma^2 = 4$)

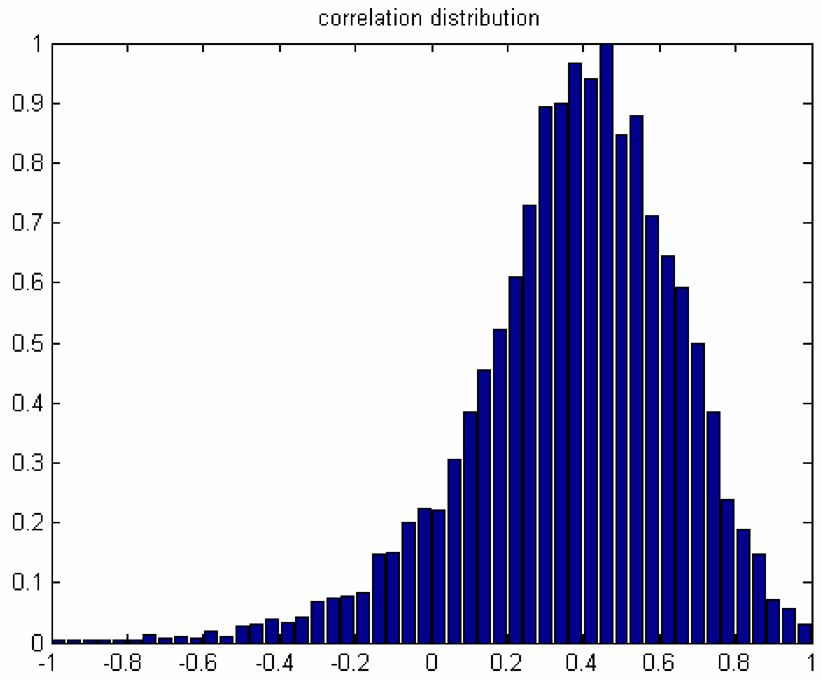

Fig. 4.18 Fuzzy correlation of test data set 5

Seventeen data sets are generated by y=x with additive zero mean uniform distribution noise and the standard deviation of the distribution ranges from 1 to 8. The result is plotted in Figure 4.19. The x axis represents the correlation coefficient of the whole data set and the y axis represents the standard deviation of the probability density function of the correlation coefficient which is computed by crisp data, fuzzy correlation method we proposed in 4.9.

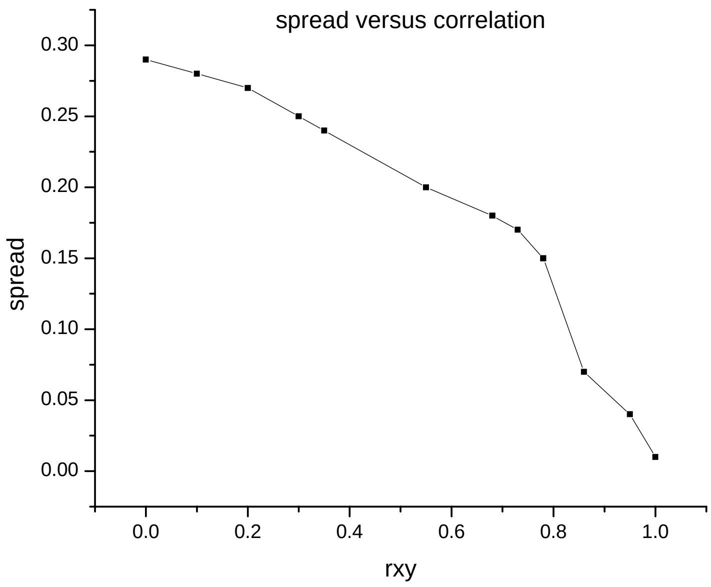

Fig. 4.19 Correlation versus spread curve

We can see that when the correlation increases, the spread of the fuzzy correlation becomes smaller. It is intuitively reasonable to see that this is the case because for a perfect straight line no matter how many data points are missed, its fuzzy correlation is always one. But when the data set becomes more widely spread, more combinations with different correlation values can be obtained which makes the spread of the fuzzy correlation wider.

4.11 Raw Data Set Shape and Fuzzy Correlation Shape

In test data set 4 and 5, the additive measurement noise is roughly symmetric and all data points have the same support in the membership function. As a result, we find that the corresponding fuzzy correlation distributions also present roughly symmetric characteristic.

Figure 4.20 is a data set approximately symmetric to y=x, the shape of this data set looks like a heart, and the wider end of the data set is close to the origin of the coordinate system and the narrow end extends in both the up and the right direction. The data set illustrated in Figure 4.22 has similar orientation and similar shape to the data set in Figure 4.20. However the narrow end of the data set in Figure 4.22 is close to the origin of the coordinate system and the wide end extends to the up and right direction. We refer to these two data sets in Figure 4.20 and Figure 4.22 as test data set 6 and test data set 7. The fuzzy correlations of test data set 6 and 7 are plotted in Figure 4.21 and Figure 4.23 respectively. It can be observed fuzzy correlations for those two data sets are very close; both take on roughly symmetrical distribution around the peak which is between 0.4 and 0.6. The result is within our expectation since both data sets are symmetrically distributed to y=x and the deviations from y=x in both data sets are very close.

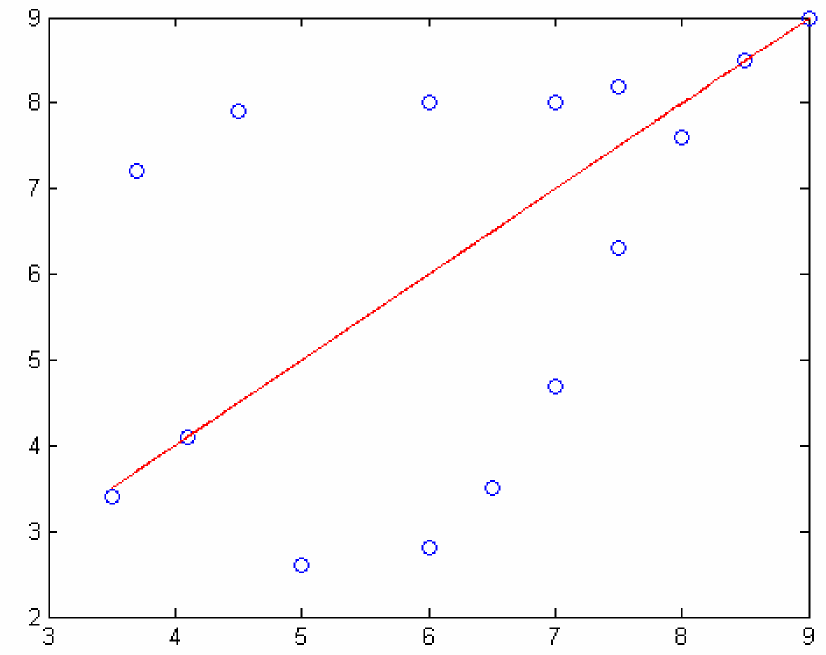

Fig. 4.20 Test data set 6

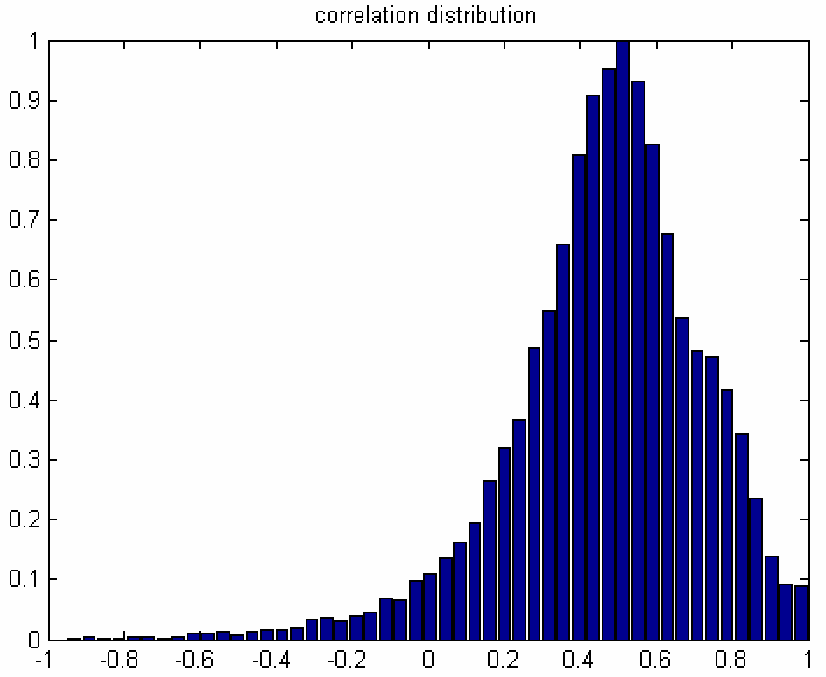

Fig. 4 .21 Fuzzy correlation of test data set 6

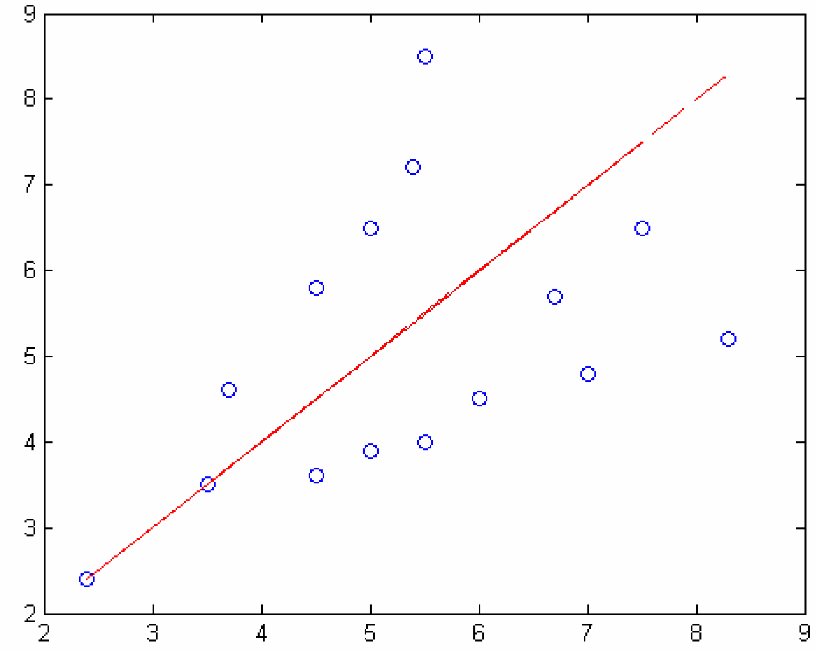

Fig. 4.22 Test data set 7

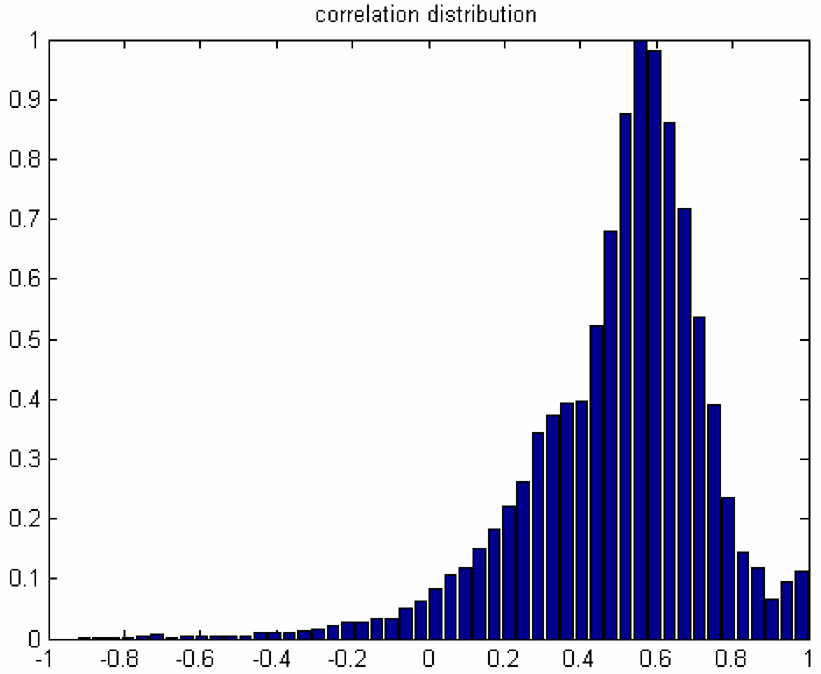

Fig. 4 .23 Fuzzy correlation of test data set 7

Figure 4.24 is a data set with quadratic form; we refer to this data set as test data set 8. Figure 4.25 is its fuzzy correlation plot. It is obvious that the fuzzy correlation of this data set is strongly skewed. In contrast to this data set, the data set in Figure 4.26 shows the skewing in the opposite direction to Figure 4.24, and its corresponding fuzzy correlation also has the opposite skewing direction to Figure 4.25. The reason for this phenomenon is that in Figure 4.24, most of data points are positively correlated and we only get negative correlation cases occasionally, so there is a peak on the high correlation side and a long tail towards the left side of x axis. However in Figure 4.26, the situation is exactly the contrary. From these simulation results, we can summarize that there is a relationship between the shape of the original data set and the shape of the fuzzy correlation. It is therefore positive to estimate the distribution of fuzzy correlation according to its original data set although a lot of research work must be performed in this direction.

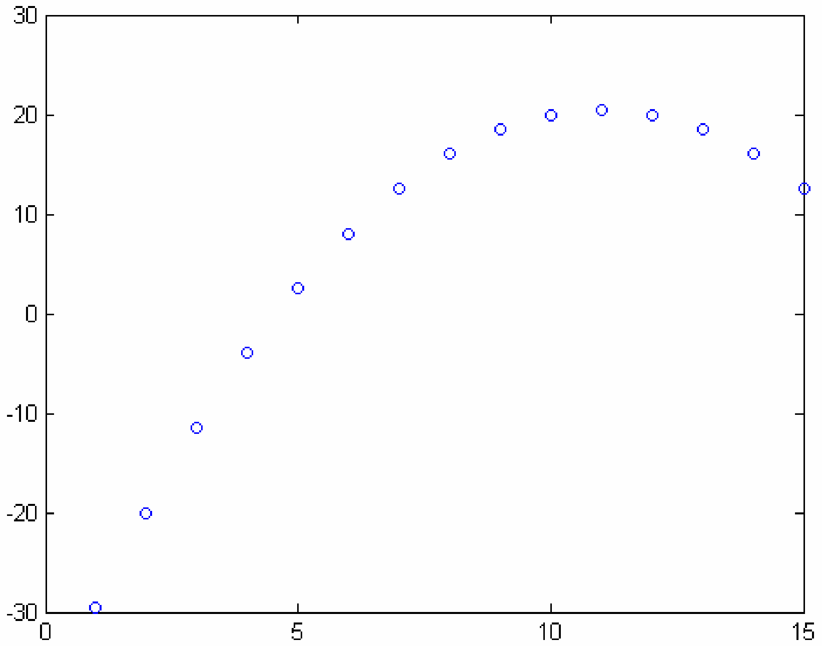

Fig. 4 .24 Test data set 8: y=-0.5 (x-10)^2+10+x

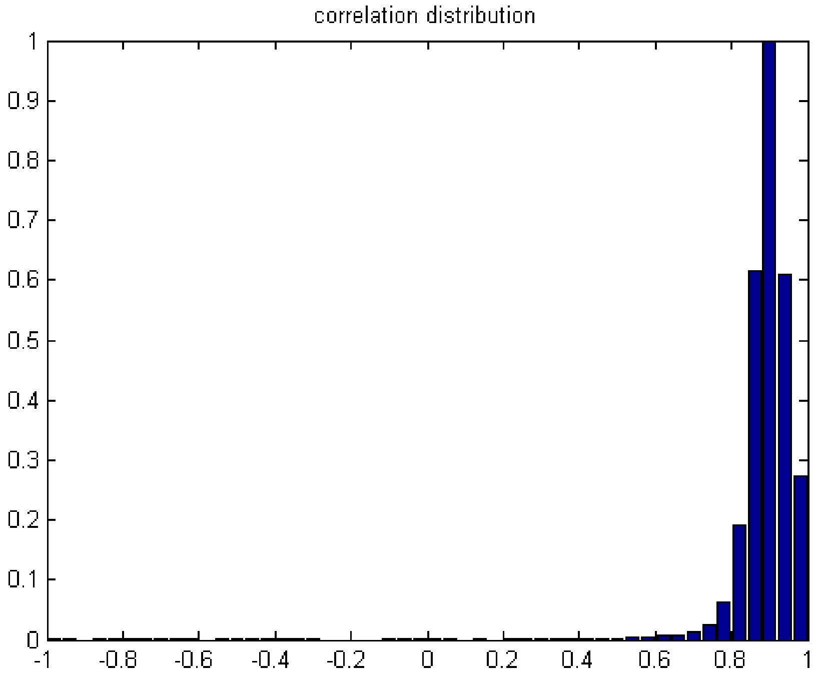

Fig. 4 .25 Fuzzy correlation of test data set 8

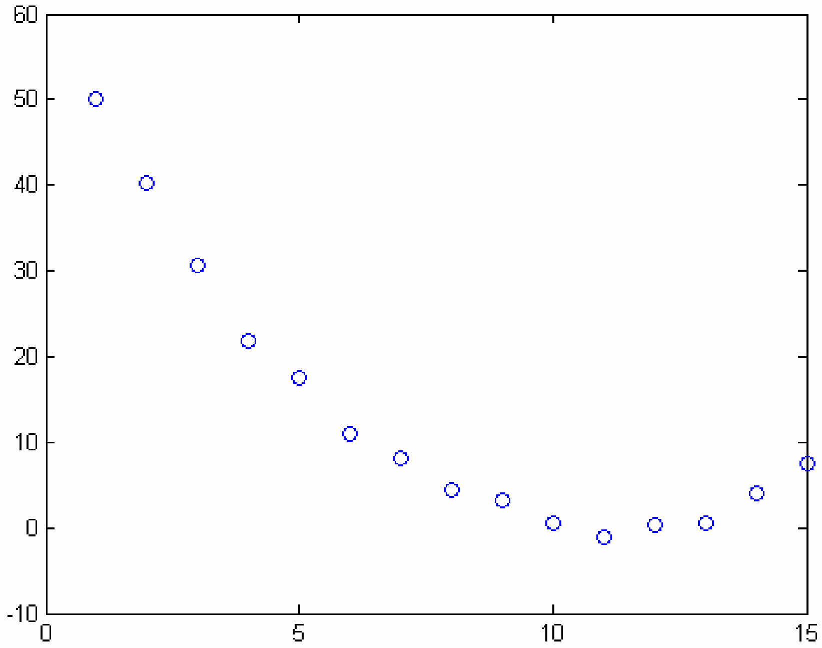

Fig. 4 .26 Test data set 9: y=0.5(x-10)^2+10-x

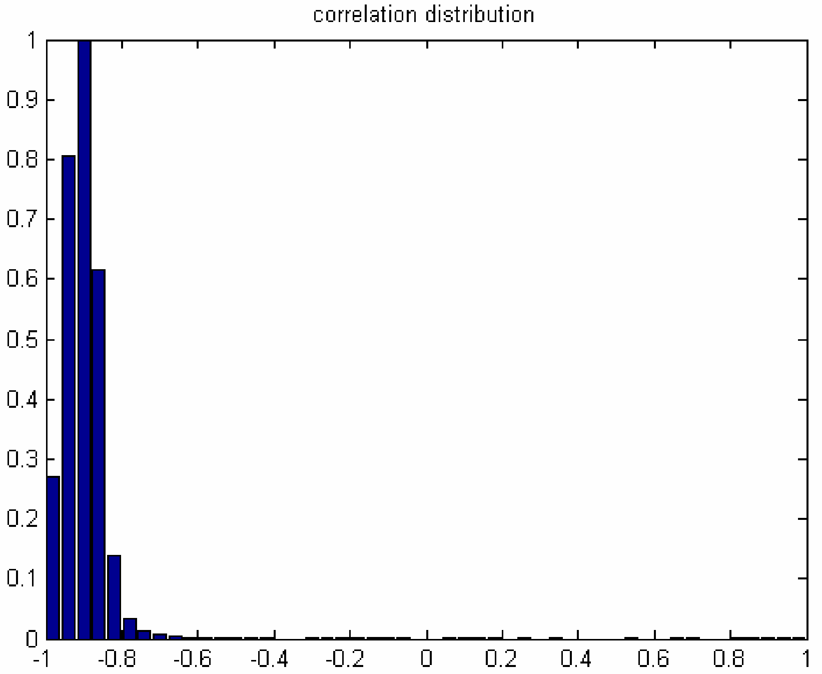

Fig. 4 .27 Fuzzy correlation of test data set 9

4.12 Summary

In this chapter, we designed a number of test data sets, and the fuzzy correlation coefficient is computed on those data sets. Then we perform a series of simulation to compare the four methods which are proposed in chapter 3. Firstly empirical execution time is plotted against the size of data set for four methods; it was found that direct method needs much more time in the case of big size data set. Next we change the memberships with different supports and different shapes, fuzzy correlation coefficients are plotted under all these cases and we found that the fuzzy correlation coefficient reflects the changes in the membership function of the original data set. In order to see if the computed fuzzy correlation coefficient satisfies the mathematical properties presented by chapter 3, nonspecificity measurement, fuzziness measurement and defuzzified value are computed and tabulated. Observing those results, it can be concluded that the simulation results meet the theoretical expectations. The method to generate the fuzzy correlation coefficient from a crisp data set is proposed, the result is a probability density function of the correlation coefficient of the crisp data set when some information is missing. Test data sets are designed with different type random noise and different shape to explore how those factors impact the corresponding fuzzy correlation. It turns out that the support of fuzzy correlation is in inverse proportion to the correlation coefficient of the whole data set. It is also observed that if the original data set is in symmetric shape, then the fuzzy correlation takes on roughly symmetric distribution; otherwise, the membership function of the fuzzy correlation is askew distributed.

CHAPTER 5

Background of Fuzzy Regression

Fuzzy regression provides an alternative modeling approach to evaluate the relationship between the independent variables and the dependent variables in typical data mining applications when the data on hand is vague and uncertain. Such phenomenon is particularly significant for the situations when a large amount of data is required to show the underlying pattern. However, not much research has been done on this issue. Because of its increasing importance in the industries, we investigated the different types of fuzzy regression models based on whether the input parameters, the regression coefficients, or the output parameters are fuzzy or not. The different regression models can be applied to a variety of applications. The purpose of this chapter is to revisit the fuzzy regression models of the ongoing studies and to discuss issues which have yet to be done in this area. This discussion is not meant to be exhaustive but intended to point out some of the most important considerations.

In ordinary regression analysis, the unfitted errors between the regression model and observation are generally assumed to be random error with normal distribution having zero mean and constant variance. In fuzzy regression analysis, the unfitted error is viewed as the error of model structure. A handful of studies addressing regression analysis for fuzzy data have been reported. In the next several sections, I will review some of the landmark work by Tanaka [47-53], who initially developed the idea of fuzzy regression. His result powerfully excites a new application area for fuzzy data mining. Other main contributors in this area include Celmins [66, 67], Diamond [69], Ishibuchi [51, 53, 70, 72, 73], Savic and Pedrycz [63] etc.

5.1 Compatibility of Two Fuzzy Sets

Given two fuzzy sets A and B, the extension principle allows evaluating the compatibility of the fuzzy set A with a fuzzy set B. If com (A, B) denotes this compatibility, we could have

$$com(A, B) = \sup_{y \in R} \min(A, B)$$

The compatibility concept will be used in comparing two fuzzy sets in the definition of fuzzy regression model.

5.2 Crisp Input, Crisp Parameter and Crisp Output

The derivation of the regression equation is based on the principle of minimum mean square error. Given paired samples {(x_i, y_i)}, i=1,…,n, where x_i belongs to the set of independent variable X and y_i belongs to the set of dependent variable Y, n is the size of data set, the linear regression model postulates that [81]

$$Y = a + bX + \varepsilon \qquad (5.1)$$

where ε is often referred as residual and it is a random variable with zero mean. The coefficients a and b are determined by the condition that the sum of the square residuals is as small as possible. The regression model is illustrated in Figure 5.1.

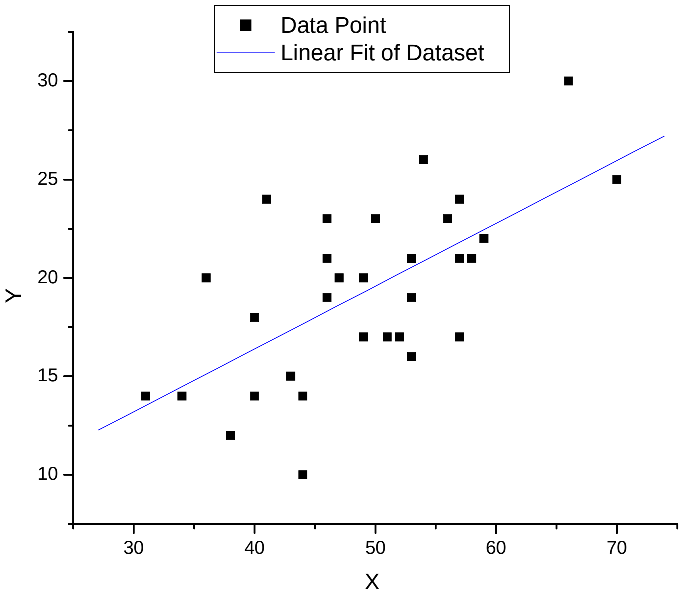

Fig. 5.1 Classical regression model illustration

The following assumptions are widely used in formulating a classical regression problem:

1. The mean value of ε is assumed to be 0. i.e., E (ε) =0.

2. The variance of ε for each sample of X is the same. Namely

$$Var(\varepsilon) = E(\varepsilon^2) - [E(\varepsilon)]^2 = E(\varepsilon^2)$$

3. The samples are independent, so the value of the ε s for two different samples of X are uncorrelated, i.e. $Cov(\varepsilon_i, \varepsilon_j) = 0$

4. For any given independent variable, all of its ε s is normally distributed. That is

$$\varepsilon \approx N(0, \sigma^2)$$

For a particular x_i, the prediction of y_i can be computed as

$$\hat{y}_i = a + bx_i$$

Our goal is to find the regression coefficients a and b that minimize the total squared residue $S = \sum (Y - a - bX)^2$. If we take the partial derivatives and set them equal to 0, the regression coefficients can be computed by solving the resulting equations. The formulas for b and a are shown in (5.2).

$$a = \overline{Y} - b\overline{X}$$

$$b = \frac{S_{XY}}{S_{XX}}$$

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

$$\overline{Y} = \frac{\sum_{i=1}^{n} Y_i}{n} \tag{5.2}$$

where $S_{XX} = \sum_{i=1}^{n} \left(X_i - \overline{X}\right)^2$ and

$$S_{XY} = \sum_{i=1}^{n} (X_i - \overline{X})(Y_i - \overline{Y})$$

In the case of multiple linear regressions, there are several independent variables used to predict a single dependent value with a linear combination. Suppose we have p-1 independent variables ($X_1,\ldots,X_{p-1}$), a single dependent variable Y, and n observations. The samples can be describe as in the form of a vector $\{(x_{11},\ldots,x_{(p-1)1},y_1),\ldots,(x_{1n},\ldots,x_{(p-1)n}, y_n)\}$, then a general form of a multiple linear regression model can be written as

$$Y = X\beta + \varepsilon$$

where Y is an $n \times 1$ vector of observations, X is an $n \times p$ matrix from the given samples where the first column is all 1s and the other columns are $x_{j1}\ldots x_{jn}$ for j=1… p-1. β is a $p \times 1$ vector of regression coefficients, and ε is an $n \times 1$ vector of residual errors. The regression coefficients can be computed based on the least square principle as

$$\beta = (X'X)^{-1}(X'Y)$$

So the residual can be computed as $\varepsilon = \hat{Y} - \overline{Y} = X\beta - \overline{Y}$ for any given X.

One of the most widely used statistics that tells how a regression model can help explain the variance of the model is the coefficient of determination. It is also called the R^2 statistic, and it is the ratio of the total regression variance divided by the total variation. In fact R^2 provides the proportion of the total variation that could be explained by the regression model.

$$R^2 = \frac{SS_{reg}}{S_{YY}} = \frac{\sum(\hat{Y}_i - \overline{Y})^2}{\sum(Y_i - \overline{Y})^2} \tag{5.3}$$

The coefficient of determination equals to the square of the correlation coefficient of the data set. Its value may vary from zero to one. However it has the advantage over the correlation coefficient in that it is interpreted directly as the proportion of variance in the dependent variable that can be explained by the regression equation.

5.3 Crisp Input, Crisp Parameter and Fuzzy Output

In this case, the input data is crisp, and we try to use crisp regression coefficient to figure out the relationship between the input data and output data and to produce a fuzzy output. To my knowledge, this case remains completely uninvestigated up to now. We will explain more about this case in chapter 6.

Figure 5.2 is an illustration of this type of fuzzy regression model. The regression result computed with classical method is denoted by the dotted line, and the pair of solid lines illustrates the interval that regression equations could lie.

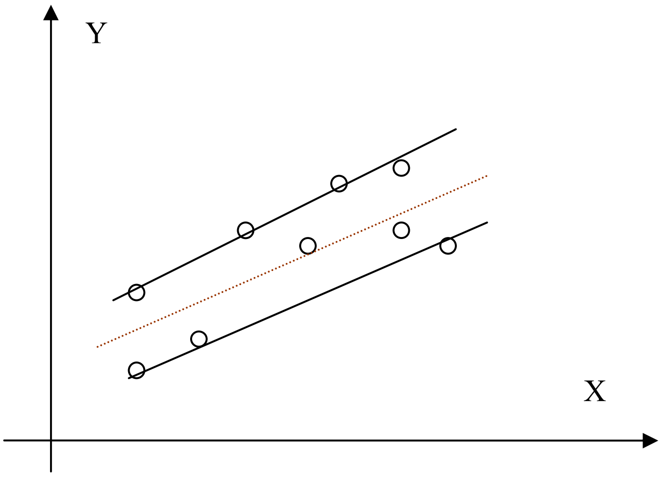

Fig. 5.2 Illustration of crisp linear regression

5.4 Crisp Input, Fuzzy Parameter and Crisp Output

Fuzzy linear regression analysis was introduced by Tanaka [47], unlike the ordinary regression model, the unfitted errors between the fuzzy regression model and the observed data are viewed as the fuzziness of the model structures. The goal of the fuzzy regression analysis is to find a regression model that fits all observed fuzzy data within a specified fitting criterion. Different fuzzy regression models are derived according to different fitting criterion. Figure 5.3 is an illustration of this fuzzy linear regression model. The small circles in Figure 5.3 represent crisp data points, the dotted line is the regression line computed by the classical method, and the pair of solid lines denotes regression lines derived by the method we will present in this section.

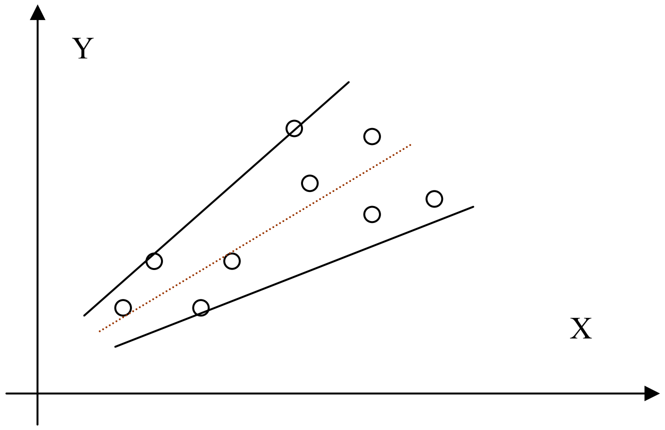

Fig. 5.3 Illustration of crisp input, fuzzy parameter, crisp output linear regression

Tanaka's regression coefficients are fuzzy numbers, so the predicted value is also a fuzzy number.

$$\hat{Y} = \widetilde{A}_0 + \widetilde{A}_1 X$$

where $\widetilde{A}_i = (c_i, s_i)$ is described as a symmetrical triangular membership function which has the center c_i and the fuzzy half-width s_i.

Given the crisp input-output data (x_j, y_j), Tanaka formulated the fuzzy regression problem based on the minimum fuzziness principle which means that the uncertainty within the fuzzy prediction should be minimized; the crisp output and the fuzzy prediction should be compatible at some given possibility level. His method can be formulated as the following linear programming problem:

Minimize $$J = \sum_{j=1}^{m} s^t \left| x_j \right|$$

St. $$c^t x_j + (1-h) s^t \left| x_j \right| \geq y_j$$

$$c^t x_j - (1-h) s^t \left| x_j \right| \leq y_j \qquad \text{j=1, 2… m}$$

$$s_i \geq 0, \qquad \text{i=1, 2…n} \qquad (5.4)$$

where m is the size of data set, n is the number of independent variables, x_j is a vector $\{x_{j1}, x_{j2}, \ldots, x_{jn}\}$which denotes j-th observation. *s* is a vector $\{s_1, s_2, \ldots, s_n\}$which denotes the half spreads of fuzzy regression coefficients, c is a vector $\{c_1, c_2, \ldots, c_n\}$which denotes the center of fuzzy regression coefficients. *h* is a possibility level predetermined by a decision-maker.

Problems connected with this approach are the influences of different trends and the problem of outliers. In order to overcome these limitations, Tanaka and Ishibuchi [51, 53] introduced an interval regression based on the Quadratic Programming (QP) approach. They claimed in [51] that the obtained result by QP has better central tendency than the results of former studies, and this approach can be effectively applied for data set with outliers. The model can be formulated as following:

$$J = k_1 \sum_{j=1}^{m} (y_j - c^t x_j)^2 + k_2 \sum_{j=1}^{m} s^t |x_j| |x_j|^t s$$

St. $$c^t x_j + (1-h) s^t |x_j| \geq y_j$$

$$c^t x_j - (1-h) s^t |x_j| \leq y_j \qquad \text{j=1,2,…,m}$$

$$c_i \geq 0 \qquad \text{i=0,1,…,n} \qquad (5.5)$$

where h is the possibility level on which the crisp output data is included in the predicted interval, k_1 and k_2 are two weighting factors and they are used to balance the least-squares principle and the minimum fuzziness fitting criteria. The meanings of other parameters are the same as in (5.4).

Tanaka's model can be reduced to a crisp least square regression equation when there is no fuzziness in the system. The weights k_1, k_2 take an important role in the regression model. Tanaka proposed two kinds of method to remove an outlier. If a person has enough knowledge on the data set, he or she can give different h-values to each data and determine the acceptable h-level set of estimated fuzzy output. Another way is to divide the data in two groups: reliable and suspicious groups. The dividing criterion is based on the standard deviation of the data set. By dividing into two groups different strategies can be applied for those groups.

To compare fuzzy regression analysis and statistical analysis of conventional regression model, Tanaka and Ishibuchi [53] proposed an exponential possibility regression, in which the result of fuzzy regression corresponds to the probabilistic regression.

The main shortcoming of all these methods is that the concept of least square is not utilized. It is intuitively reasonable that the fuzzy regression model should approach the conventional regression model when the fuzziness of the system tends towards zero.

5.5 Crisp Input, Fuzzy Parameter and Fuzzy Output

Different aspects of fuzzy least-squares regression were investigated by Celmins [66, 67], Diamond [69], Savic and Pedrycz [63] and Chang and Ayyub [45, 46]. Celmins [66] proposed an approach for fuzzy least-squares regression based on the compatibility measurement between the observed data $Y^{(j)}$ and a fitted model Y_j. The objective of data fitting according to this approach is to find a model such that the overall compatibility between the output data and the fitted model prediction is maximized.

Savic and Pedryz [63] formulated the fuzzy regression method by combining the least-squares principle and the minimum fuzziness criterion. Their method is performed in two consecutive steps. The first step uses ordinary least-squares regression to find the fuzzy center values of the fuzzy regression coefficients. The second step uses the minimum fuzziness criterion to find the fuzzy widths of the fuzzy regression coefficients.

Another fuzzy regression method is called the interval regression proposed by Ishibuchi [70]. The fuzzy regression coefficients are determined such that all fuzzy outputs are within a fuzzy regression model. The constraint in this model becomes $Y_j \subset F(x_j)$, where Y_j is the j-th fuzzy number output and F (x_j) is the j-th fuzzy prediction.

In this regression model, we consider the observations {x_j, $Y^{(j)}$} where x_j is the crisp independent data and $Y^{(j)}$ is the fuzzy number output data. Let us assume that it is symmetric triangular fuzzy number coefficients for convenience. The coefficient A_i is denoted by its center c_i and width s_i as $A_i= (c_i,s_i)$ as shown in Figure 5.4. A_i can also be described as $A_i = (a_i^L, c_i, a_i^U)$ where $c_i = \frac{(a_i^L + a_i^U)}{2}$.

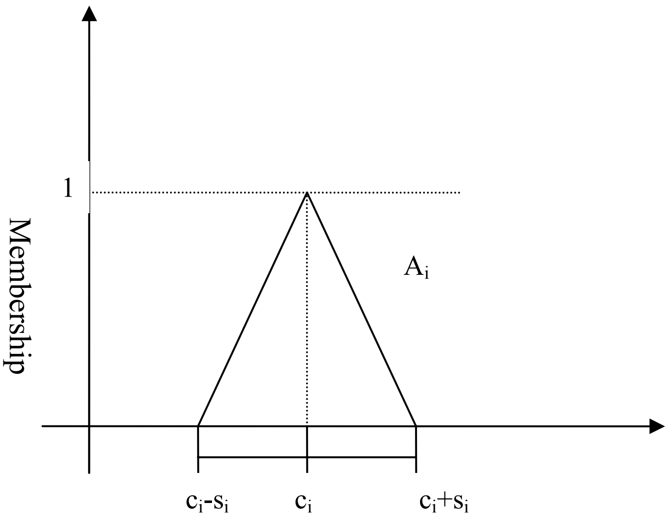

Fig. 5.4 Symmetric triangular fuzzy number coefficient A_i

The aim of this regression problem is to find the regression coefficients such that the fuzzy linear function fits the given fuzzy data as best as possible. Two criteria of goodness are usually employed. According to the first criterion, the total difference between the areas of the actual fuzzy observation $Y^{(j)}$ and the areas of the fuzzy number Y_j obtained from the regression equation should be minimized. According to the second criterion, the fuzzy data $Y^{(j)}$ and Y_j should be compatible at least to some given degree.

$$com(Y^{(j)}, Y_j) = \sup_{y \in R} \min\left[Y^{(j)}(y), Y_j(y)\right]$$

In the case of symmetrical triangular fuzzy number coefficients, the predicted fuzzy output should also be a fuzzy number with symmetrical triangular membership as shown in Figure 5.4. The fuzzy data with the dotted line triangular membership is the fuzzy output. The aim of the regression model is to minimize the difference between the prediction and observation with the compatibility level at certain acceptable level.

According to the fuzzy arithmetic principle for fuzzy numbers, the fuzzy linear regression model can be calculated as follows [47]:

$$Y = F(x) = (f^{c}(x), f^{(s)}(x)) = (c_0 + c_1 x_1 + ... + c_n x_n, s_0 + s_1 |x_1| + ... + s_n |x_n|)$$

where $f^{(c)}(x)$ and $f^{(s)}(x)$ are the center and the spread of the fuzzy linear model F(x) respectively.

If the fuzzy data points (x_j,Y_j) j=1, 2… m, are given, the constraint condition can be formulated as an inclusion relation [53].

$$[Y_j]_h \subset [F(x_j)]_h = [A_0]_h + [A_1]_h x_{j1} + ... + [A_n]_h x_{jn} \qquad \text{j=1, 2… m}$$

This inclusion relationship is illustrated in Figure 5.5. We need to be cautious that this relationship does not hold for all possibility levels. The described fuzzy regression problem can be formulated in terms of the following formula:

$$\text{Min } \sum_{j=1}^{m} f^{(s)}(x_j)$$

$$\text{St. } [y_j]_h \subset [F(x_j)]_h \qquad \text{j=1,2,…,m} \qquad (5.6)$$

where h is the preset threshold for inclusion relationship between the predicted value and the observations. y_j is the j-th observation, and $F(x_j)$ is the j-th prediction.

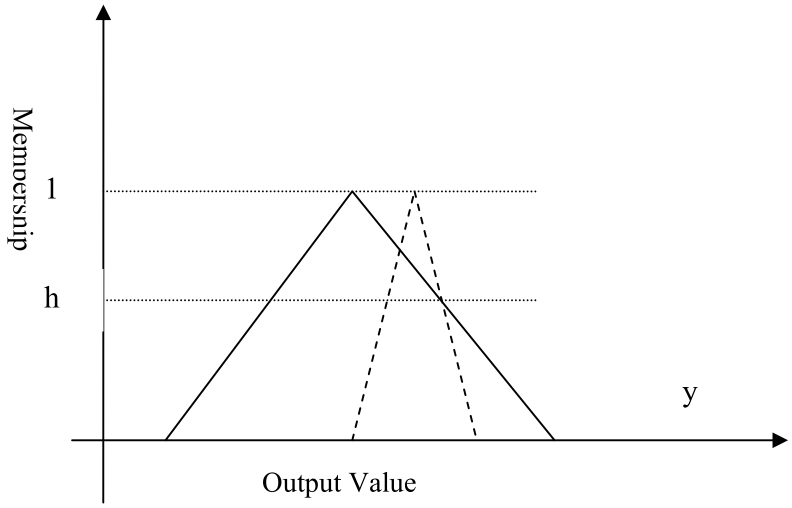

Fig. 5.5. Fuzzy regression model with symmetric triangular coefficients

Non-Symmetric Fuzzy Number Coefficients

If the regression coefficients have nonsymmetrical membership function, the output from the fuzzy linear regression can be calculated as a non-symmetric fuzzy number by the fuzzy arithmetic operations. In order to determine the non-symmetric fuzzy number coefficients $A_i = (a_i^L, c_i, a_i^U)$, i=0, 1… n, the following method is proposed by Diamond [69]:

1. Determine the center of the fuzzy linear model by least square regression.
2. Determine the lower limit and the upper limit of the fuzzy linear model by solving the following linear programming problem.

Minimize $J = \sum_{j=1}^{m} [f^U(x_j) - f^L(x_j)]$

St. $y_j \in [F(x_j)]_h$

$$a_i^L \leq c_i \leq a_i^U \qquad \text{i=0,1,…,n}$$

$$\text{Or} \quad y_j \subset [F(x_j)]_h \qquad \text{j=1,2,…,m} \qquad (5.7)$$

where m is the size of the data set, n is the number of independent variables, h is the preset threshold for inclusion relationship between the predicted values and the observations. y_j is the j-th observation, and $F(x_j)$ is the j-th prediction.

Figure 5.6 is an illustration of the regression model of this category. Since the data point has the crisp input, fuzzy output characteristic, it is represented with a short vertical line which is a crisp value in X axis and an interval in Y axis. The dotted line denotes the regression line computed by the classical method and the pair of solid lines is the regression model derived from the method we presented in this section.

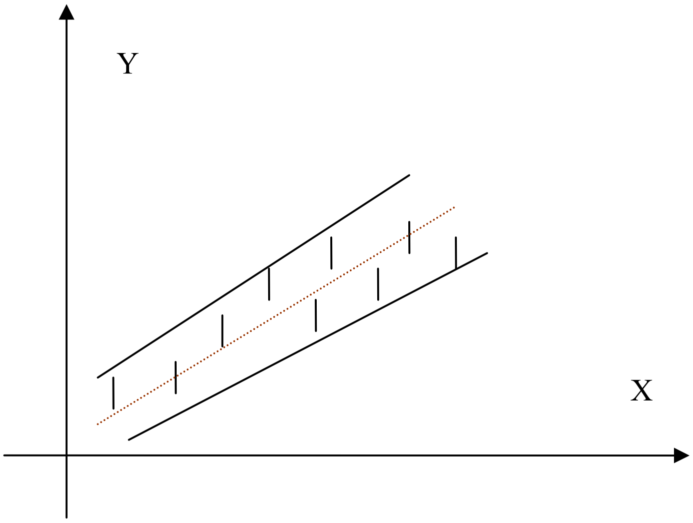

Fig.5.6 Illustration of crisp input, fuzzy parameter, fuzzy output linear regression

5.6 Fuzzy Input, Crisp Parameter and Crisp Output

To the best of my knowledge, nobody has worked on this topic. The situation could be illustrated in Figure 5.7. The horizontal short lines represent the fuzzy input while the output is crisp. The dotted line describes the regression equation we are supposed to get. In some applications, the input data has a certain amount of uncertainty or measurement error, hence the input is fuzzy. But the output is categorical, hence the output is crisp.

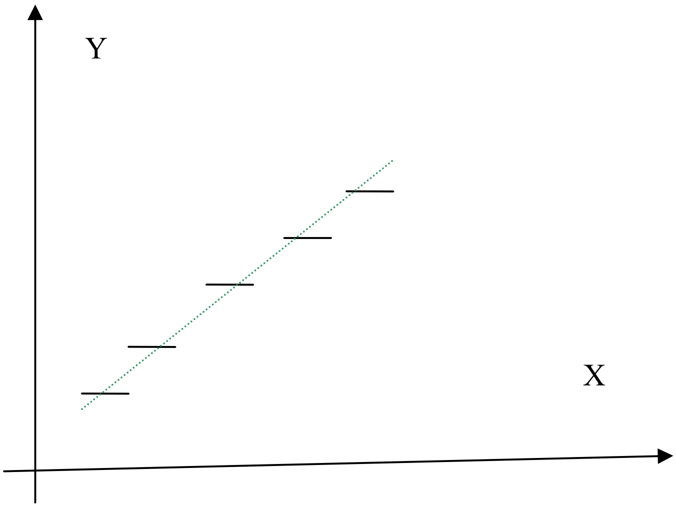

Fig. 5.7 Illustration of fuzzy input, crisp parameter, crisp output linear regression

5.7 Fuzzy input, Crisp parameter and Fuzzy output

This case is rather common as a direct extension of the traditional regression case and is also called the fuzzy least square method [69, 75]. If uncertainty is to be accommodated, then fuzzy input is used in a regression model with crisp parameters, the output of this model is clearly fuzzy.

In the case of fuzzy input data, we need to consider fuzzy least-squares regression model. It could be expressed by the form:

$$Y = a_1 X_1 + a_2 X_2 + ... + a_n X_n$$

where the $\{X_1, X_2, \ldots, X_n\}$ are fuzzy independent variables and $\{a_1, a_2, \ldots, a_n\}$ are real-valued parameters. The difference between the observations and predictions can be formulated as below.

$$J = \sum_{j=1}^{m} \left| \int Y^{(j)}(y)dy - \int Y_j(y)dy \right|$$

where m is the size of the data set, $Y^{(j)}$ and Y_j are j-th observation and prediction respectively. Our goal is to minimize J and at the same time let the fuzzy prediction and the fuzzy observation be compatible to a given degree.

If we assume that the membership function of fuzzy data is in the form of symmetric and triangle shape with center $\{x_1, x_2, \ldots, x_n\}$or $\{y_1, y_2, \ldots, y_n\}$ and half spread $\{s_1, s_2, \ldots, s_n\}$ and the size of the data set is m, j is the index of observations, the regression model can be written as:

Minimize $$J = \sum_{j=1}^{m} \left| y^{(j)} + s^{(j)} - \sum_{i=1}^{n} \left| a_i x_i^{(j)} \right| - \sum_{i=1}^{n} \left| a_i s_i^{(j)} \right| \right|$$

st. $$-\sum_{i=1}^{n} |a_i| s_i^{(j)} + \sum_{i=1}^{n} a_i x_i^{(j)} \leq y^{(j)} + s^{(j)}$$

$$\sum_{i=1}^{n} |a_i| s_i^{(j)} + \sum_{i=1}^{n} a_i x_i^{(j)} \geq y^{(j)} - s^{(j)}$$

j=1, 2…m (5.8)

Another method is to convert both fuzzy input and fuzzy output to crisp values by some kind of defuzzified method, and then apply the classical regression method to obtain the crisp regression coefficients.

Figure 5.8 is an illustration of the regression model of this case. Since the data point has the characteristic of fuzzy input and fuzzy output, it is reasonable to use a small rectangle to represent a data point which is an interval in both X and Y axis. The dotted line is computed by the classical method and the solid line is derived by the regression method we presented in this section.

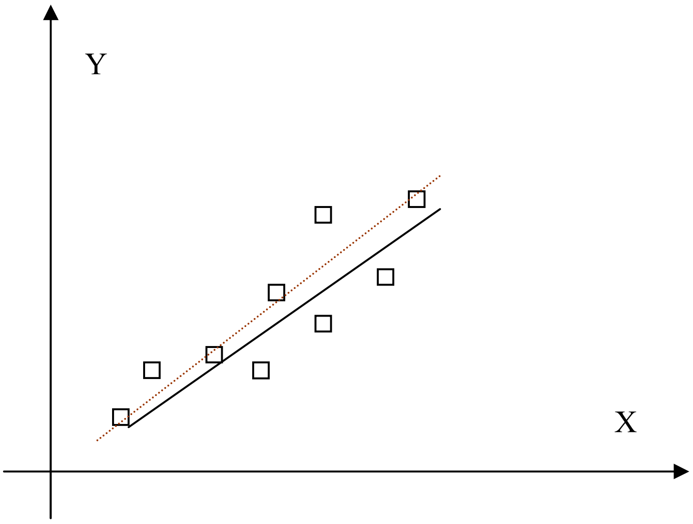

Fig. 5.8 Illustration of fuzzy input, crisp parameter, fuzzy output linear regression

5.8 Fuzzy Input, Fuzzy Parameter and Crisp Output

No literature is available on this topic as far as I know. This situation could be illustrated in Figure 5.9. The horizontal short lines represent the fuzzy input and crisp output data, the dotted line describes the regression equation obtained by applying convention crisp regression method under the condition that we only use the α=1 data point of the fuzzy input data as the independent variable. The pair of solid lines is the fuzzy regression equations we want to get. This case is similar to the fuzzy input, fuzzy parameter and fuzzy output case except the output is defuzzified data.

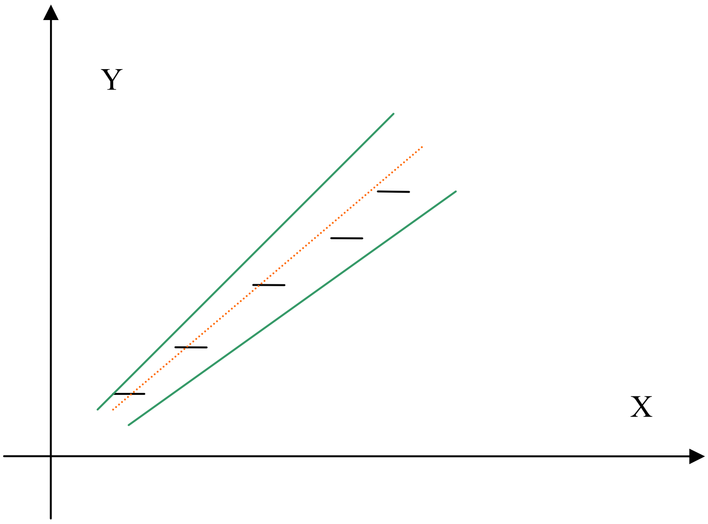

Fig. 5.9 Illustration of fuzzy input, fuzzy parameter, crisp output linear regression

5.9 Fuzzy Input, Fuzzy Parameter and Fuzzy Output

In 1992, Sakawa and Yano [78] considered the fuzzy linear regression models with fuzzy outputs, fuzzy parameters and also fuzzy inputs. They formulated the multiobjective programming methods for the model estimation along with a linear-programming-based approach. Figure 5.10 is an illustration of this model; the pair of solid lines is derived regression equations.

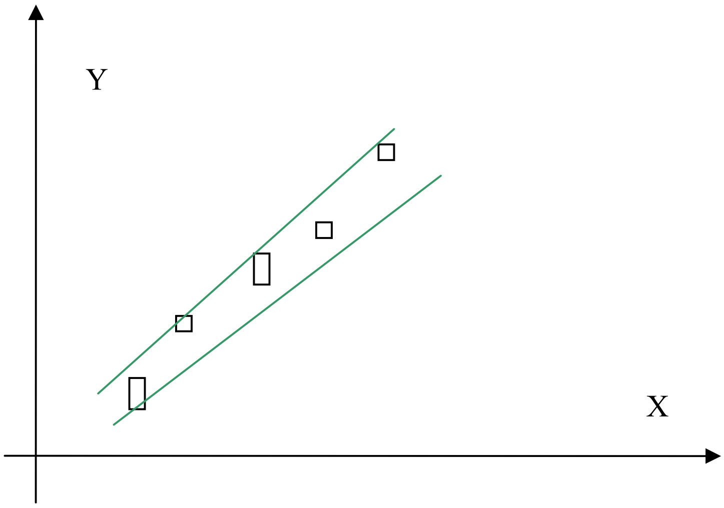

Fig. 5.10 Illustration of fuzzy input, fuzzy parameter, fuzzy output linear regression

Assume the fuzzy input data, fuzzy output and fuzzy parameter all have symmetric triangular membership functions. The regression equation can be formulated as

$$Y = A_0 + A_1X_1 + A_2X_2 + \ldots + A_nX_n$$

where $A_i(m_i,c_i)$, $X_i(a_i,s_i)$ and Y are fuzzy numbers.

The objective is based on the minimum fuzziness principle, we want to make the uncertainty of the fuzzy predictions as small as possible, and the constraints are generated by considering the set inclusion relationship between the fuzzy output and the fuzzy prediction to be compatible at least to a given degree. The following mathematical model is proposed by Sakawa and Yano.

$$\text{Min} \qquad J = \sum_{j=1}^{m} (Y_j{}^R - Y_j{}^L)$$

$$\text{St.} \qquad [y_j]_h \subset [F(x_j)]_h \qquad \text{j=1,2,...m} \tag{5.9}$$

where m is the size of data set, Y_j^L, Y_j^c, Y_j^R are the left leg, the center and the right leg of the j-th fuzzy prediction; h is the preset threshold for inclusion relationship between the predicted value and the observation. y_i is the j-th observation, and $F(x_j)$ is the j-th prediction., j is the index of observations.

5.10 Fuzzy Nonlinear Regression

Fuzzy nonlinear regression (FNR) modeling differs from classical nonlinear regression in that the output of a FNR model is a fuzzy number. The assumption of a linear relationship between the variables is not always appropriate; instead nonlinearity may be the proper relationship. This can be detected by the lack of fit by the model or the data set scatter plot.

Fuzzified neural networks which have real number inputs and fuzzy number connection weights are usually used to extend the fuzzy linear regression methods to fuzzy nonlinear regression analysis [71-73]. The network relationship can be formulated as follows:

Input units: $O_{pi} = x_{pi}$, i=1,2,…,n

Hidden units: $O_{pj} = f(Net_{pj})$, j=1,2,…,n$_H$

$$Net_{pj} = \sum_{i=1}^{n} o_{pi} \cdot W_{ji} + \theta_j$$

Output units: $O_p = f(Net_p)$

$$Net_p = \sum_{j=1}^{n_H} O_{pj} \cdot W_j + \theta \tag{5.10}$$

where real numbers and fuzzy numbers are denoted by lowercase and uppercase respectively. The connection W and biases θ are fuzzy numbers, n is the number of input unit, n$_H$ is the number of hidden unit.

5.11 Summary

The fuzzy regression model can be used to evaluate the functional relationship between the dependent and independent variables in a fuzzy environment. Fuzzy regression models can be categorized into eight classes based on the fuzzy (or crisp) characteristic of the input, fuzzy (or crisp) parameters and fuzzy (or crisp) output data. In general, there are two approaches in the analysis of fuzzy regression models: minimum fuzziness methods and the fuzzy least-squares methods. Those approaches are used to model fuzzy regression equations for a variety of cases.

CHAPTER 6

Fuzzy Linear Regression

Although statistical regression has many applications, problems with regression can occur in many situations. For example, the number of observations is inadequate, there is difficulty verifying distribution assumptions, there is vagueness in the relationship between input and output variables, or there is ambiguity associated with the event etc. These are the situations that a fuzzy regression has to address. Based on the components of the regression equations, there are eight categories of regression models. We have discussed the previous work already in the previous chapter. In this chapter we propose a method to solve the fuzzy linear regression problem, the presented approach is more efficient and the result is more informative than the classical model. We have also extended the minimum fuzziness principle to the fuzzy input, crisp parameter, crisp output case and the fuzzy input, fuzzy parameter, crisp output case which are rarely discussed in the literature.

6.1 Crisp Input, Crisp Parameter and Fuzzy Output

6.1.1 Introduction

Given a crisp data set, it is a common practice to try to apply a regression model to explore the relationship between the independent variables and the dependent variables. The classical regression model assumes that the observation errors follow the normal distribution and the fitting criterion is the minimum mean square error principle. In this section we will present a new regression method which generates a fuzzy prediction.

Assume that we have a fuzzy system with N crisp input and fuzzy output observations (X, Y). Firstly let us consider that all data information is available, so there is no uncertainty in this system. We can set the possibility level α of the prediction to be 1. A regression model can be derived with classical method and the corresponding output data can be provided with this model. Next consider the case when one of the N data information is missing, a new regression equation will be obtained based on those remainder N-1 data points. There could be N new regression equations for this case and N output data are generated with those models. Next, assume two data points are missing, in which case N*(N-1)/2 new regression equations and new predictions will be generated…, repeat this process till only three data points are left. Summarize the output results and put them in histogram format by setting the output value as the x-axis and the number of occurrence of each value as the y-axis. Finally, one can obtain the probability density function of the prediction data which can be seen as the membership function of the output fuzzy data after normalization. It is obvious that this output fuzzy data is more informative than the traditional crisp prediction. In addition, in classical regression theory, we have to collect many data samples to satisfy the statistical condition; however, only a single data set realization is needed in this method.

Figure 6.1 shows an illustration of the fuzzy number prediction computed by this method. When α equals to 1, that is the case when all the data information are available. There is no vagueness in the system. So the prediction value is the same as the one generated by the classical regression model. Then when more and more data information becomes unavailable, the uncertainty of this system becomes bigger and bigger. We can see that the prediction value is no longer a single value but an interval. This fuzzy number reflects the inherent data set characteristic which is determined by the system and it gives us more information than the classical regression models.

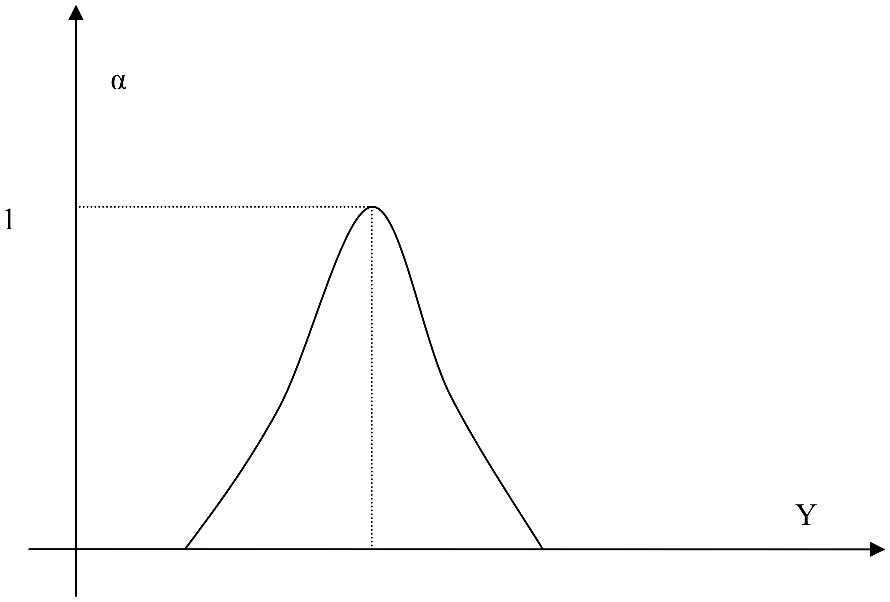

Fig. 6.1 Illustration of fuzzy number prediction

6.1.2 Algorithm Description

From the above description, our goal is to explore the correlation of a data set under the condition that some data points are missing. Assume the size of the data set is n, and we consider only those subsets of the original data set which have at least three data points, then the number of subsets we could obtain should be:

$$C_{n-1}^{n} + C_{n-2}^{n} + ... + C_{3}^{n} = \sum_{i=1}^{n-3} C_{i}^{n}$$

where C_i^n denotes the number of combinations of n things taken i at a time. We need to compute the correlation coefficient for each subset; the results can be plotted as the probability density function (pdf) of the correlation coefficient of the original data set.

We can see that it can be classified as a combinatorial problem. It has intractable characteristic for large data set. We will use the Monte Carlo based method to obtain an approximate solution for this problem.

The number of possible combinations for a subset is determined by the number of data points which are missing; for example, we can get $C_{n-1}^{n} = n$ possible combinations for subsets which miss one data point from the original data set. If we want to consider subsets which miss two data points, the possible combinations equals to $C_{n-2}^{n} = \frac{n(n-1)}{2}$. In order to let Monte Carlo sampling reflect the possible combinations of a subset, our strategy is to let the number of subsets that are taken into the computation is related to the number of missing data points. The size of the subset under consideration is denoted by m, and the number of subsets (with the same size) we use in the computation is referred as K.

In summary, assume that the original data set is P= $\{(x_1, y_1), (x_2, y_2)\ldots (x_n,y_n)\}$, n is the size of the original data set, we have the following steps:

1. Initialize the size of the subset: let m=n
2. Initialize the number of loops: count=1
3. Randomly pick up m data points from the original data set P to form a subset Q
4. Generate a regression model with the classical method on the current data set Q
5. Increment count, record the regression model computed in (4)
6. Repeat step (3) through (5) until count=K
7. Decrement m
8. Repeat step (2) through (7) until m=3
9. Compute the fuzzy number prediction with the recorded regression coefficients

10. The output result is normalized and plotted in a histogram format.

Notice that when the number of missing data points increases, there are more possible combinations available to pick out a subset Q from the original data set P. It is desired to consider K as a variable that is in proportional to the theoretic possible combinations.

6.1.3 Computational Complexity

Let us count the number of additions, subtractions, multiplications, divisions and square root operations as individual operations. Assume that the current data set size is m, then according to (5.2), in order to obtain the regression coefficients, we need (m+1)+ (m+1) + (2m+m+1) + (2m+m+1) +3=8m+7 operations for computing the classical regression. Step (5) and step (7) need 2 additional operations. So the total number of operations is

$$\sum_{m=3}^{n}(8m+7)K = K\sum_{m=3}^{n}(8m+7) = K(4n^2+11n-38)$$

Hence the computational complexity of the method presented in 6.1.2 is O (Kn^2).

6.2 Fuzzy Input, Crisp Parameter and Crisp Output

6.2.1 Introduction

This combination of fuzzy regression model has not been addressed by the present literature. In fact, this problem can be solved by extending the minimum fuzziness fitting criterion to this specific case. Assume that the fuzzy input data has the symmetric triangular membership function with c_i as the central point and s_i as the half spread. The regression equation can be formulated as

$$Y = a_0 + a_1 X_1 + a_2 X_2 + ... + a_n X_n \tag{6.1}$$

where n is the number of independent variables, $\{a_{0,} a_1 ... a_n\}$ are the crisp regression coefficients, $\{X_1,X_2,...,X_n\}$ are the fuzzy predictor variables and Y is the predicted variable.

If we assume that the membership function of the fuzzy input data is in the form of symmetric and triangle shape with center $\{c_1,c_2,...,c_n\}$ and half spread $\{s_1,s_2,...,s_n\}$, m is the size of the data set, j is the index of the observations. Based on the minimum fuzziness principle, we can minimize the uncertainty in the prediction data. The following mathematical model is proposed.

$$\text{Minimize} \quad J = \sum_{j=1}^{m} a^t \left| s_j \right|$$

$$\text{St.} \qquad a^t c_j + (1-h) a^t \left| s_j \right| \geq y_j$$

$$a^t c_j - (1-h) a^t \left| s_j \right| \leq y_j$$

$$s_j \geq 0 \qquad \text{j=1,2,...,m} \tag{6.2}$$

where J is the summation over spreads of all fuzzy prediction, y_j is j-th crisp output data. The constraints indicate the inclusion relationship between the crisp output and fuzzy prediction, in this case, the crisp output should be located in the interval which is generated by taking a cut on the membership function of the fuzzy prediction at the possibility level h. Figure 6.2 is an illustration of this inclusion relationship, the triangle denotes a fuzzy prediction, after taking a cut at the possibility level h, we get an interval $[a^t c_j - (1-h)a^t|s_j|, a^t c_j + (1-h)a^t|s_j|]$. The crisp output observation represented by a small circle should locate at this interval.

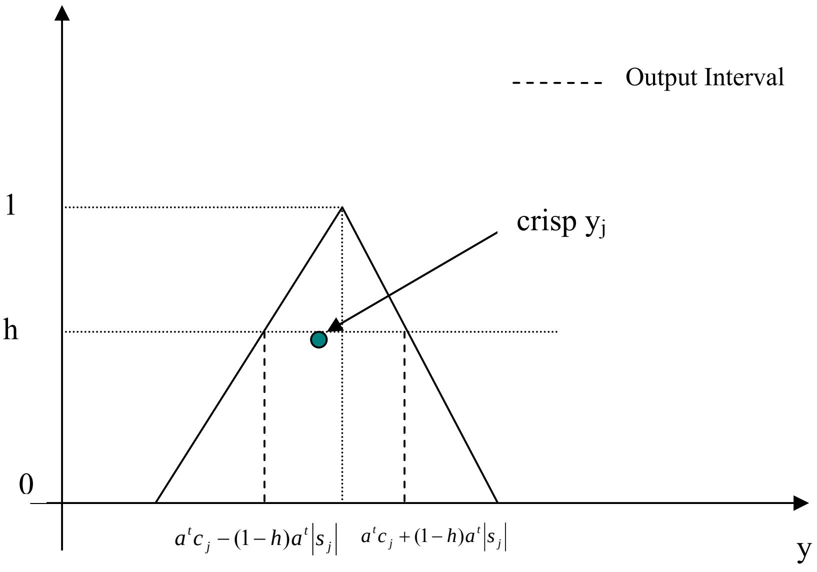

Fig. 6.2 Illustration of inclusion relationship

The selection of h is critical in fuzzy regression modeling. If h is too big, it will be hard to generate a precise model for the decision maker, but if h is too small, some observations could not be covered in this interval and the decision maker will be too optimistic about the estimation. Thus an appropriate value of h is desired.

Of course the membership functions of input data could be given in another shape, but we can use the same regression model described by (6.2). We want to minimize the overall system fuzziness under the condition that the observations have to be within intervals which are generated by fuzzy linear regression models at some possibility level.

6.2.2 Computational Complexity

According to the above discussion, the fuzzy input, crisp parameter, crisp output type of fuzzy regression problem can be formulated as (6.2). Observing (6.2), both the objective function and the constraints are in linear forms. So this is a linear programming (LP) model. The Simplex method is widely used to solve LP problem.

Assume we have a standard form LP problem which has m equality constraints and n variables, the computational needed by the Simplex method can grow as fast as 2^m. People also have found some laboratory cases where the Simplex method exhibits its exponential growth with the size of the data set. However in practice, the Simplex method for linear programming has usually worked pretty well (see Appendices), its computational complexity seems to grow linearly with the size of the data set.

6.3 Fuzzy Input, Fuzzy Parameter and Crisp Output

6.3.1 Introduction

Assume the fuzzy input data has a symmetric triangular membership function with c_i as the central point and s_i as the half spread. The regression equation in this case can be formulated as

$$Y = A_0 + A_1 X_1 + A_2 X_2 + ... + A_n X_n \tag{6.3}$$

where $A_i(c_i,m_i)$, $X_i(x_i,s_i)$ i=1,2…,n and Y are fuzzy numbers.

The approach for this method is based on the minimum fuzziness principle, our goal is to minimize the uncertainty of the fuzzy prediction, the crisp output and the fuzzy prediction should be compatible to some degree. The following mathematical model is proposed.

Min $$J = \sum_{j=1}^{m} (Y_j^{\ R} - Y_j^{\ L})$$

St. $$Y_j^c + (1-h)(Y_j^R - Y_j^c) \geq y_j$$

$$Y_j^c - (1-h)(Y_j^R - Y_j^c) \leq y_j \quad \text{j=1,2…,m} \tag{6.4}$$

where Y_j^L, Y_j^c, Y_j^R are the left, the center and the right leg of the j-th fuzzy prediction, m is the size of the data set, the parameter h is the value determined by the decision-maker, and y_j is the j-th crisp output.

The fitting criterion of this problem is to minimize the fuzziness of the predicted data. Since both the input and the parameters are fuzzy numbers, the prediction is no longer in a triangular shape. We have

$$Y_j^{\ L} = \sum_{i=0}^{n} (c_i - m_i)(x_{ji} - s_i) \tag{6.5}$$

$$Y_j^{\ R} = \sum_{i=0}^{n} (c_i + m_i)(x_{ji} + s_i) \tag{6.6}$$

$$Y_j^c = \sum_{i=0}^{n} \ c_i \cdot x_{ji} \qquad\qquad \text{j=1, 2…, m} \tag{6.7}$$

where x_{ji} is the i-th component of j-th input data.

We can see that when both regression coefficients and input data become crisp, i.e. m_i=0 and s_i=0, i=1,2…,n. the fuzzy prediction degenerates to a crisp value which can be computed by the classical regression method.

In order to satisfy the constraints in (6.4), the crisp output has to be included in the interval by taking a cut on the membership function of the fuzzy prediction. Figure 6.3 is an illustration of this inclusion relationship; we can see that the membership function of the fuzzy prediction is not in a triangular shape. Crisp output y_i which is represented by a small circle should locate at the interval$[Y_j^L, Y_j^R]$ and this interval is generated by taking a cut on the membership function of the fuzzy prediction at the possibility level h.

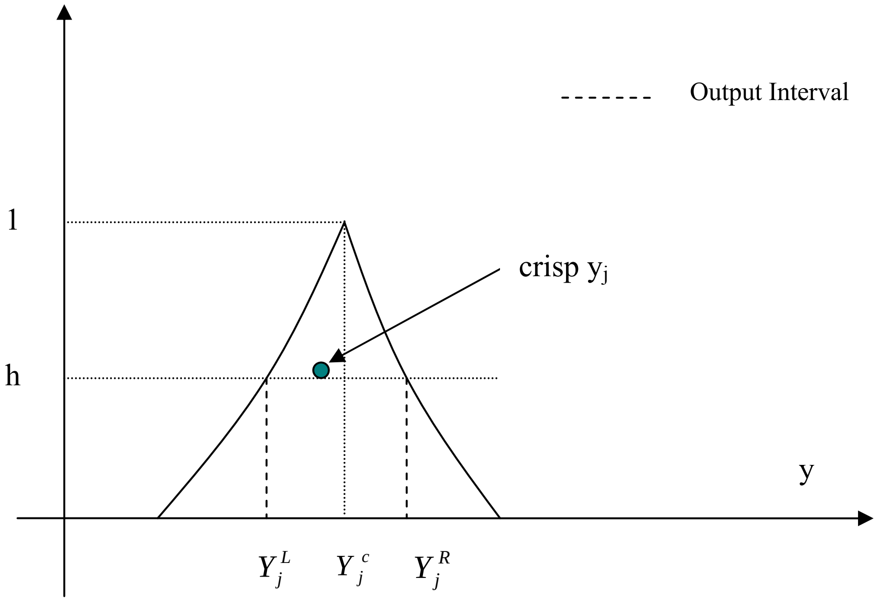

Fig. 6.3 Constraints of fuzzy input, fuzzy parameter and crisp output case

6.3.2 Computational Complexity

The model in (6.4) is also a linear programming model. We can use the Simplex method to solve the problem. If the size of data set is m, the computational complexity of the Simplex method could be 2^m . However, this method works pretty well in practice, the computation seems to grow linearly with the size of the data set.

.

CHAPTER 7

Simulation of Fuzzy Regression

Numerical examples are used in this chapter to illustrate the fuzzy regression models that are discussed in chapter 5 and chapter 6. For convenience, we assume all membership functions of the fuzzy data are in the symmetric triangular shape.

7.1 Crisp input, Crisp parameter and Crisp output

This type is the classical linear regression case. Assume we have a data set Y=1+2X and the samples are corrupted by additive measurement noise which is a zero mean Gaussian noise with a standard deviation σ of 2, and this additive noise can be denoted as μ(0, $\sigma^2 = 4$), we modify this data set to introduce abnormal values. This data set is shown in Table 7.1; the 6th data point is an abnormal value.

Table 7.1 Crisp input, crisp parameter and crisp output test data set

x	1	2	3	4	5	6	7	8	9	10
y	1.2	6.2	8.0	12.1	12.3	25.2*	15.7	14.9	18.9	20.9

where * indicates an outlier which is an the abnormal value

Based on the classical linear regression method, the regression line can be computed from the data set and is shown in Figure 7.1. In the figure, the small circles denote data points and the dash-dotted line is the regression model. The computed regression line is Y = a + b X where a= 2.72 and b= 1.97. The determination of coefficient R^2 = 0.83 which means the regression line only can explain around 83% data variation which is not very satisfactory.

In Figure 7.1, the dashed line pairs represent the lower limit and the upper limit of the

95% confidence bands; a 95% confidence band implies a 95% chance that the true regression line fits within the confidence bands. Observe that in Figure 7.1, it is apparent that the 6th data point is an outlier and it has a big impact on the value of regression coefficients.

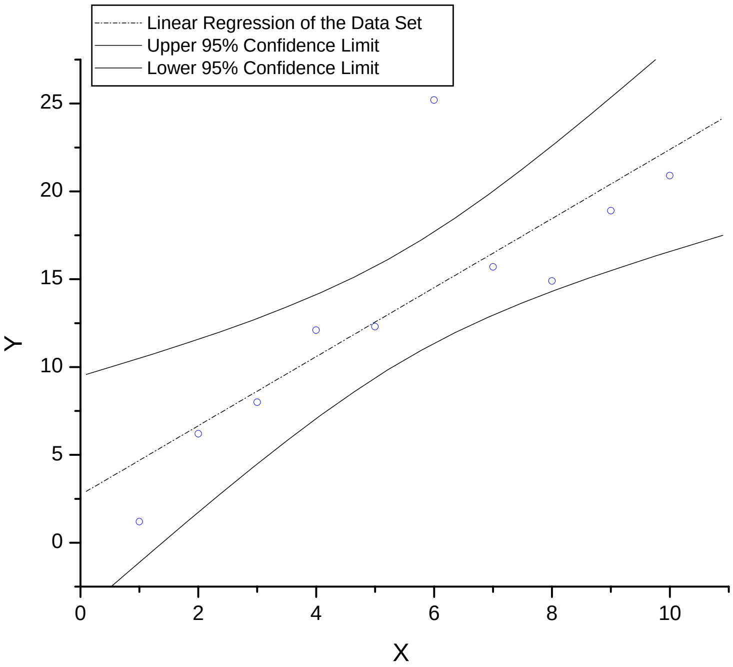

Fig. 7.1 Crisp input, crisp parameter, crisp output linear regression

Now suppose we removed the outlier and recomputed the regression line. The regression coefficients for the new regression line is a= 1.93 and b= 1.89, the regression equation is Y=a+bX. The determination of coefficient R^2 = 0.97, the regression line now reflects the data variation more precisely. The new result is shown in Figure 7.2. We can see that the 95% confidence interval of this case is much narrower after the outlier is removed. This is reasonable since now the regression line is very close to the true relationship and there is little variation in the estimation of regression coefficients.

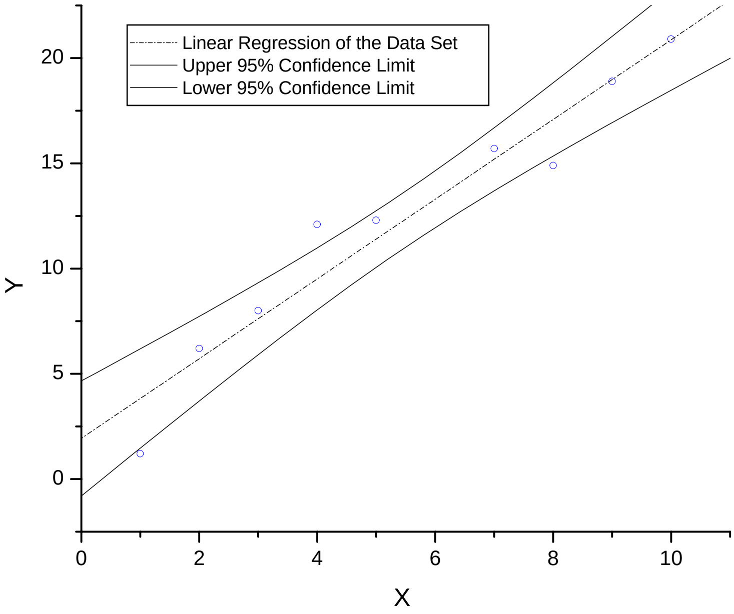

Fig. 7.2 Crisp input, crisp parameter, crisp output regression-outlier removed

The comparison of the observed output y with the predicted output $\hat{y}$ with this model is shown in Table 7.2. The first row of Table 7.2 is the input data, the second row is the observed output values, and the third row is the predicted output values. We can see that the predicted values are very close to the observations.

Table 7.2 Comparison between observations and predictions (CCC)

x	1	2	3	4	5	6	7	8	9	10
y	1.2	6.2	8.0	12.1	12.3	25.2	15.7	14.9	18.9	20.9
$\hat{y}$	3.8	5.7	7.6	9.5	11.4	13.3	15.2	17.0	18.9	20.8

7.2 Crisp Input, Crisp Parameter and Fuzzy Output

Suppose we have a crisp input, fuzzy output data set generated by a system. The data set can be described by the equation y= $1.5 \times x$ with additive white Gaussian noise N (0, 4). We want to know the distribution of the prediction. The raw data set is shown in Figure 7.3. Let us set x=16 and see the prediction provided by the model. The result is plotted in Figure 7.4.

If we apply the classical regression model to this data set, our prediction for x=16 is 23.77, and the 95% confidence level prediction is at the interval [18.86 28.67]. The fuzzy regression prediction gives us the whole distribution of the prediction instead of a single value so it is more informative than the classical statistical method. The defuzzified prediction value of this case is 23.72 and this value is very close to the classical regression prediction however the distribution has provided more information. In addition to that, in classical regression theory, it is assumed that the observation errors follow zero mean Gaussian distribution; we can see that the distribution shape in Figure 7.4 gives a good approximation to the distribution derived by the classical method in Figure 7.5. We have to collect many groups of data to get the distribution with the classical method; however, only a single data set realization is needed when this proposed method is used.

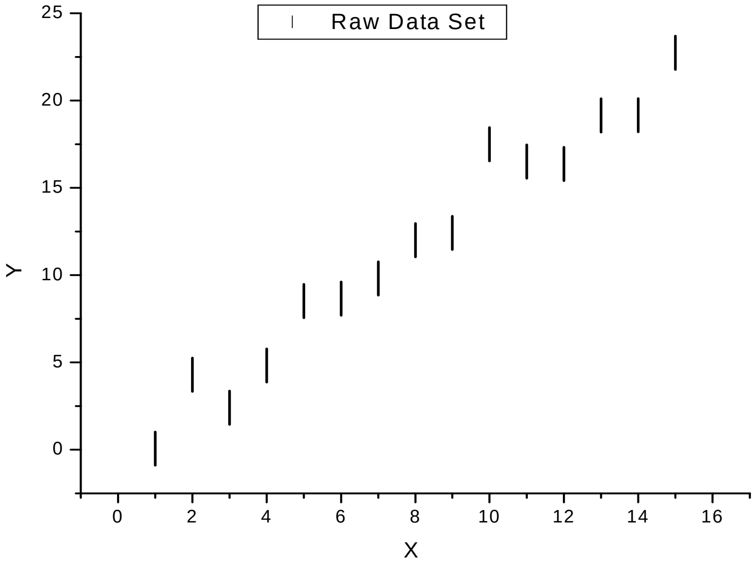

Fig. 7.3 Crisp input, crisp parameter, fuzzy output test data set

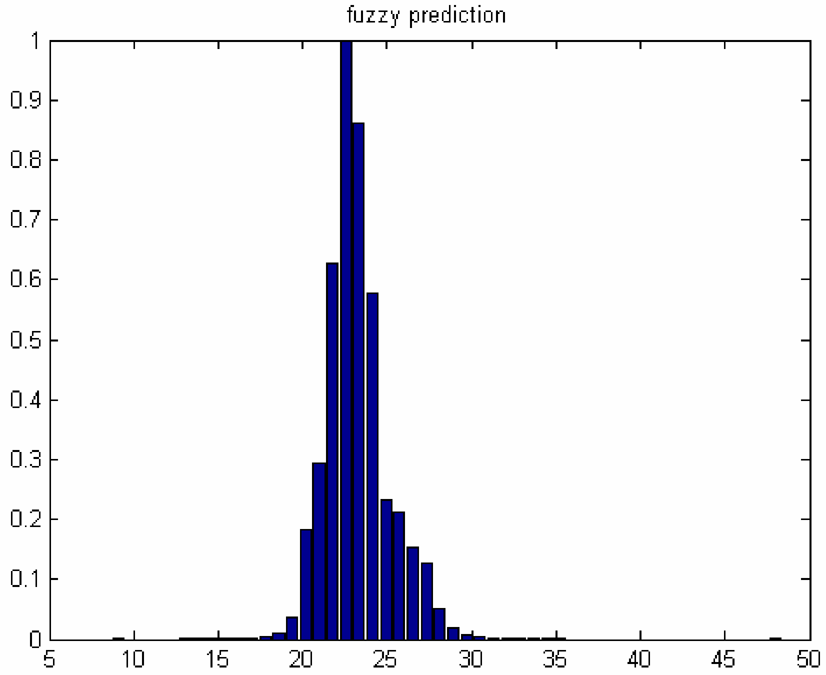

Fig. 7.4 Fuzzy regression prediction at X=16

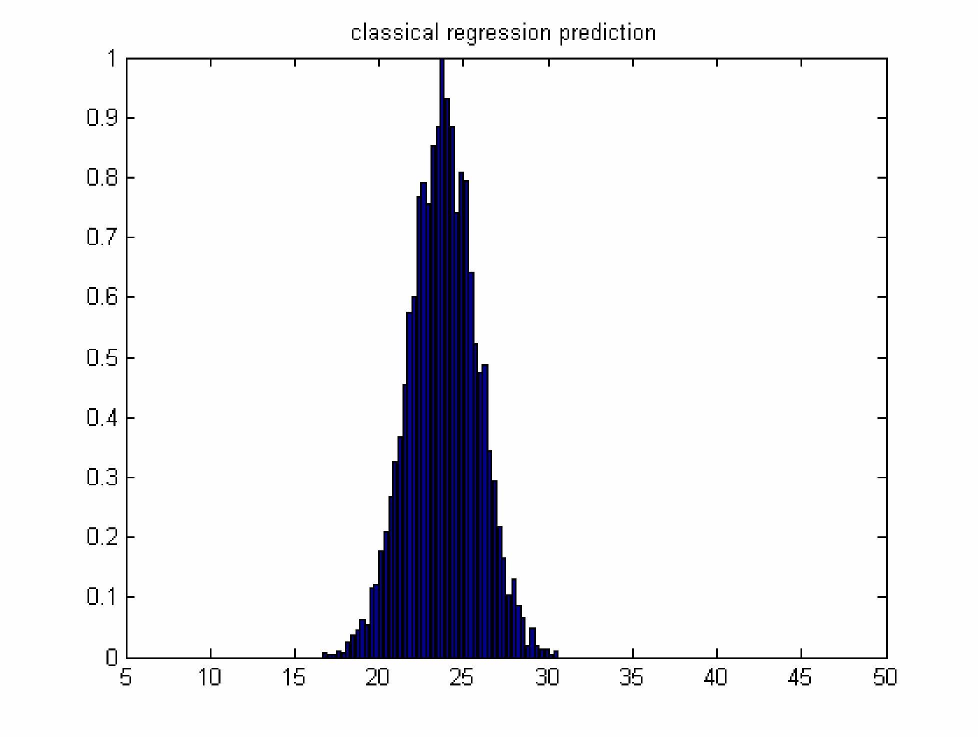

Fig. 7.5 Classical regression prediction at X=16

7.3 Crisp Input, Fuzzy parameter and Crisp output

In this experiment we will use the data set given by Table 7.1. If we consider the data variation as a result from the fuzziness of the system instead of the randomness of the process, we may need to consider a fuzzy regression coefficient model. Crisp input, fuzzy parameter, crisp output model we presented in (5.4) is applied in this case, let h=0, the fuzzy regression coefficients with triangular shape membership functions are computed as $\widetilde{A}_0 = (2.72,0)$ and $\widetilde{A}_1 = (1.97,1.77)$, the regression equation is $Y = \widetilde{A}_0 + \widetilde{A}_1 X$.

The result is shown in Figure 7.6. In Figure 7.6, small circles are original data points, the dash-dotted line denotes the regression line computed by the classical method and the pair of dashed lines represents the fuzzy regression model. We can see that the fuzzy regression lines include most of the observations. It is not surprising outliers have more impaction on the fuzzy regression model than the crisp regression model since it is the common practice that the decision-maker wants most observations are included in the estimated intervals.

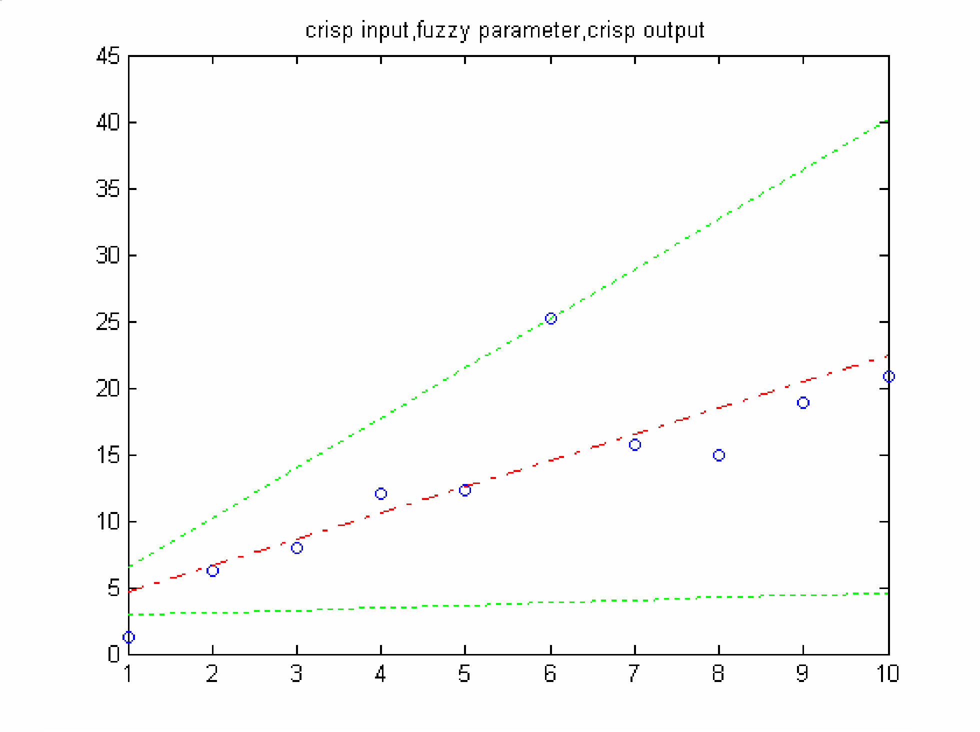

Fig. 7.6 Crisp input, fuzzy parameter, crisp output linear regression

Again let us remove the outlier and the resultant regression lines are shown in Figure 7.7. The fuzzy regression coefficients with triangular shape membership functions are computed as $\widetilde{A}_0 = (1.93,2)$ and $\widetilde{A}_1 = (1.89,0.15)$, the regression model is $Y = \widetilde{A}_0 + \widetilde{A}_1 X$. Now we can see that the interval between regression lines becomes much narrower, if the interval is too wide, the regression model is not very meaningful in explaining data variations, so the outlier removal is a big concern in formulating the fuzzy regression model.

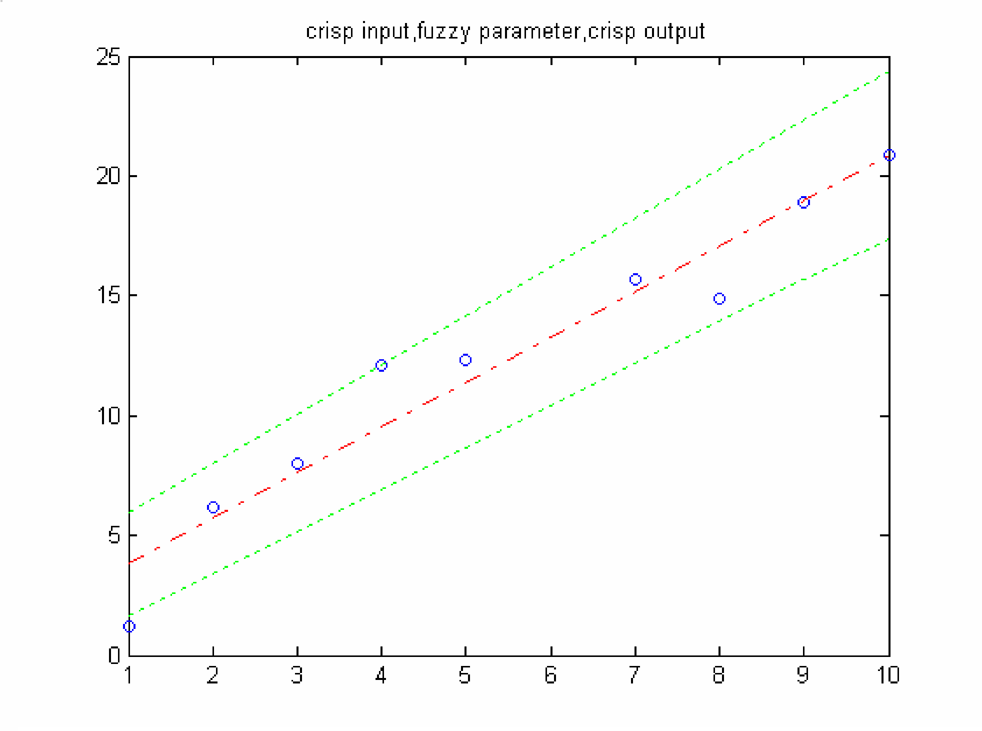

Fig. 7.7 Crisp input, fuzzy parameter, crisp output regression- outlier removed

Compare Figure 7.1, 7.2 with Figure 7.6, 7.7, It is obvious that fuzzy regression lines are different from 95% confidence intervals. Although both input and output data are crisp in the two cases, the assumptions and fitting criterion to solve the problems are totally different. In classical regression case, we consider the difference between observations and predictions to come from the observation errors or the random noise. In fuzzy regression we think the system has the characteristic of uncertainty and it is unrealistic to figure out the true relationship between the dependent variable and the independent variables. Observation errors are not considered in this case.

The regression coefficients in this case are fuzzy data, so the predictions are also supposed to be fuzzy. In order to obtain a crisp predicted value, defuzzification methods should be applied.

We have assumed that the membership function of the fuzzy data is in a triangular shape, if the fuzzy prediction is defined as (Y_{il},Y_{im},Y_{ir}), where Y_{il}, Y_{im}, Y_{ir} are the left, the middle and the right corner point of the triangle respectively, i is the index of the predicted data, then we can defuzzify Y by the centroid method [61]. The defuzzified value Y_{ic} is:

$$Y_{ic} = \frac{1}{3}(Y_{il} + Y_{im} + Y_{ir})$$

The comparison of the observed output y with the predicted defuzzified output $\hat{y}$ with this model is shown in Table 7.3. The first row of Table 7.3 represents the crisp input data, the second row is the observed output, and the third row is the predicted output values. We can see that predicted values are very close to the observations, they are even more close to the “true relationship” Y=1+2X than the corrupted observations.

Table 7.3 Comparison between observations and predictions (CFC)

X	1	2	3	4	5	6	7	8	9	10
y	1.2	6.2	8.0	12.1	12.3	25.2	15.7	14.9	18.9	20.9
$\hat{y}$	3.8	5.7	7.6	9.5	11.4	12.8	15.2	17.1	19.0	20.9

7.4 Crisp Input, Fuzzy Parameter and Fuzzy Output

In this case, the input data is crisp; the regression coefficients are fuzzy so it yields a fuzzy output, i.e. $\hat{Y} = \widetilde{A}_0 + \widetilde{A}_1 X$

where A and Y are fuzzy numbers and X is a crisp input variable.

Suppose we have a data set Y=1+2X, by adding ± 0.3 to Y_i i=1,2…,n (n is the size of the data set) the crisp data set is fuzzified to produce the case of crisp X and fuzzy Y data set as in Table 7.4.

Table 7.4 Crisp input, fuzzy parameter and fuzzy output test data set

Obs.	x	Y
1	1	(2.7, 3.0, 3.3)
2	2	(4.7, 5.0, 5.3)
3	3	(6.7, 7.0, 7.3)
4	4	(8.7, 9.0, 9.3)
5	5	(10.7, 11.0, 11.3)

First we have to check the existence of outliers in this data set; there are many ways to solve this problem. For example, outliers can be removed based on abnormal residuals to a simple fitted model. If the outlier is outside of a particular probability limit (95 or 99), then we need to locate if there is something missing from the model. If not, just remove it. In this example no outlier is found. The regression model presented in (5.6) can be applied to this case, let h=0 and the resultant regression line is Y= (1, 0.3) + (2, 0) X. The result is shown in Figure 7.8, the pair of dashed lines represents the regression model and the input-output pairs are plotted as a short vertical line which is a crisp value in x direction and an interval in y direction.

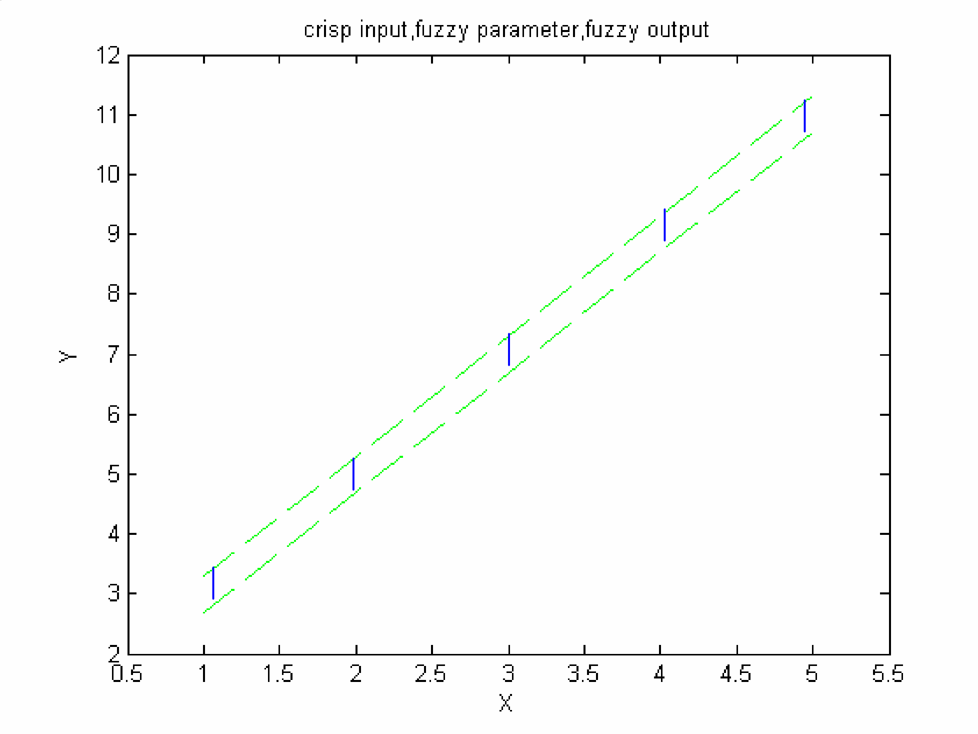

Fig. 7.8 Crisp input, fuzzy parameter, fuzzy output linear regression

The comparison of the observed output y with the predicted output $\hat{y}$ with this model is shown in Table 7.5. We can see that the predicted intervals and the fuzzy outputs are exactly the same.

Table 7.5 Comparison between observations and predictions (CFF)

Obs.	x	y	$\hat{y}$
1	1	(2.7, 3,3.3)	(2.7, 3,3.3)
2	2	(4.7, 5,5.3)	(4.7, 5,5.3)
3	3	(6.7, 7,7.3)	(6.7, 7,7.3)
4	4	(8.7, 9, 9.3)	(8.7, 9, 9.3)
5	5	(10.7, 11,11.3)	(10.7, 11,11.3)

7.5 Fuzzy Input, Crisp Parameter and Crisp Output

Suppose we have a data set Y=1+2X, by adding ± 0.3 to X_i i=1,2…,n (n is the size of the data set) the crisp data set is fuzzified to produce the case of fuzzy X and crisp Y data set as in Table 7.6.

Table 7.6 Fuzzy input, crisp parameter and crisp output test data set

Obs.	X	Y
1	(0.7,1,1.3)	3
2	(1.7, 2, 2.3)	5
3	(2.7, 3, 3.3)	7
4	(3.7, 4, 4.3)	9
5	(4.7, 5, 5.3)	11
6	(5.7, 6, 6.3)	13

First we need to check if there is any outlier existing in this data set, we consider using only those center points in X, and follow the same checking process which is discussed in 7.2. We find there is no outlier in this data set. Next, we apply (6.2) to this data set and let h=0, the result is shown in Figure 7.9. The regression model denoted by the dashed line is Y= 2+ 1.746X, the short horizontal lines represent the fuzzy input data which is a crisp value in y direction and an interval in x direction.

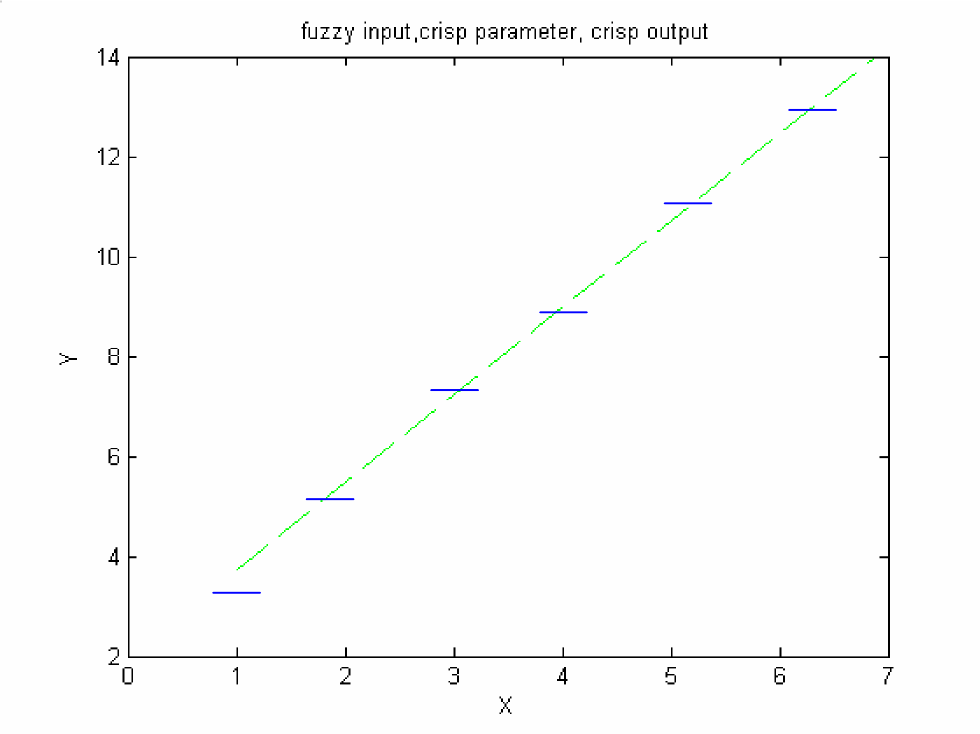

Fig. 7.9 Fuzzy input, crisp parameter, crisp output linear regression

The comparison of the observed output y with the predicted output $\hat{y}$ with this model is shown in Table 7.7. The fifth column is the defuzzification value of the predicted output.

Table 7.7 Comparison between observations and predictions (FCC)

Obs.	x	y	$\hat{y}$	$\hat{y}$ (defuzzification)
1	(0.7,1,1.3)	3	(3.2,3.7,4.2)	3.7
2	(1.7, 2, 2.3)	5	(5.0,5.5,6.0)	5.5
3	(2.7, 3, 3.3)	7	(6.7,7.2,7.7)	7.2
4	(3.7, 4, 4.3)	9	(8.5,9.0,9.5)	9.0
5	(4.7, 5, 5.3)	11	(10.2,10.7,11.2)	10.7
6	(5.7, 6, 6.3)	13	(12.0,12.5,13)	12.5

7.6 Fuzzy Input, Crisp Parameter and Fuzzy Output

The regression model of this case can be described as:

$$Y = a_0 + a_1 X_1 + a_2 X_2 + ... + a_n X_n$$

where {a_i, i=0,1…n}are the crisp regression coefficients. {X_i, i=1,2…n} are the fuzzy input and Y is the fuzzy output. The problem is to find the estimations for the regression coefficients that can provide the best explanation for the relationship between the predictor variables and the dependent variables.

For convenience, assume the membership function of the fuzzy data is in the triangular shape. (X_{il},X_{im},X_{ir}) and (Y_{il},Y_{im},Y_{ir}) denote the independent variable X_i and dependent variable Y respectively; X_{il}, X_{im}, X_{ir} are the left, the middle and the right vertex of the triangular membership function of the fuzzy input data X_i, Y_{il}, Y_{im}, Y_{ir} are the left, the middle and the right vertex of the triangular membership function of the fuzzy output data Y_i. We consider applying the classical least squares method, first we need to defuzzify the fuzzy data to crisp value, many defuzzification methods have been discussed in the literature. Here we use the centroid method, let X_{ic} and Y_{ic} be the defuzzified values of X_i and Y_i, and then we have [61]

$$X_{ic} = \frac{1}{3}(X_{il} + X_{im} + X_{ir})$$

$$Y_{ic} = \frac{1}{3}(Y_{il} + Y_{im} + Y_{ir})$$

After both fuzzy input and fuzzy output are converted to crisp values, the classical regression method can be applied. Suppose we have a data set Y=1+2X, by adding ± 0.1

to X_i and ± 0.3 to Y_{i}, i=1,2…,n (n is the size of the data set), the crisp data set is fuzzified to produce the case of fuzzy X and fuzzy Y data set as in Table 7.8.

Table 7.8 Fuzzy input, crisp parameter and fuzzy output test data set

Obs.	X	Y
1	(0.9, 1.0, 1.1)	(2.7,3.0,3.3)
2	(1.9, 2.0,2.1)	(4.7,5.0,5.3)
3	(2.9,3.0,3.1)	(6.7,7.0,7.3)
4	(3.9,4.0,4.1)	(8.7,9.0,9.3)
5	(4.9, 5.0,5.1)	(10.7,11.0,11.3)
6	(5.9,6.0,6.1)	(12.7,13.0,13.3)
7	(6.9,7.0,7.1)	(14.7,15.0,15.3)
8	(7.9,8.0,8.1)	(16.7,17.0,17.3)

In order to check if there is any outlier existing in this data set, we consider using only those center points in both X and Y, i.e. the points with possibility level as 1, just follows the checking process we discussed in 7.3, finally we find there is no outlier in this data set. The regression model is computed as Y=1+2X. The result is plotted in Figure 7.10; the dashed line denotes the regression model, the fuzzy input- fuzzy output data is denoted by a solid square which is an interval in both x and y direction . Because of the uncertainty in the data set is symmetric; the resultant regression model is the same as the model computed with the classical method.

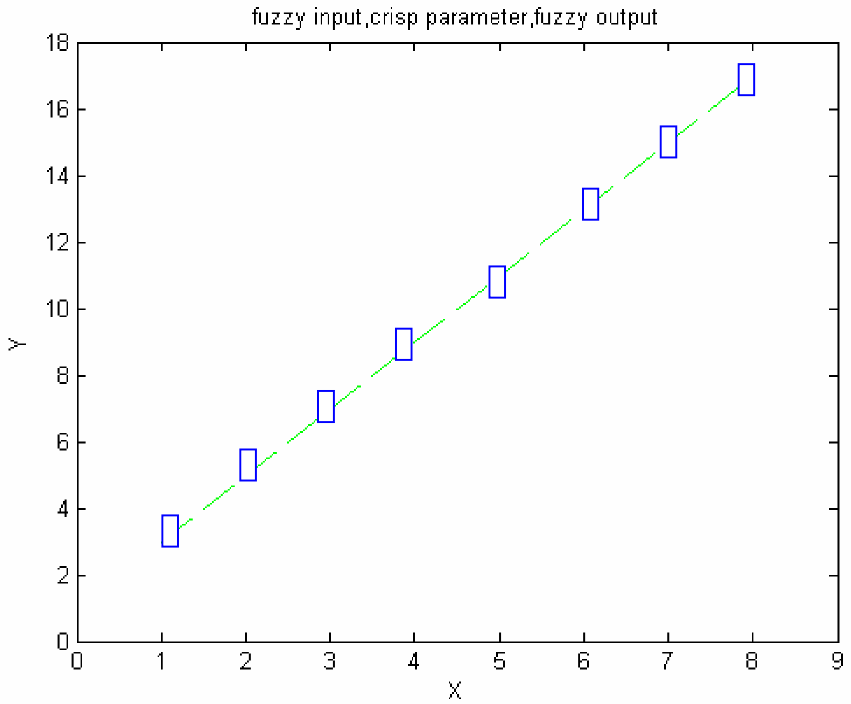

Fig. 7.10 Fuzzy input, crisp parameter, fuzzy output linear regression

The comparison of the observed output y with the predicted output $\hat{y}$ with this model is shown in Table 7.9. Since the membership function of the fuzzy input is in a symmetric triangular shape, and regression coefficients are crisp values, the uncertainty in the predicted value is scaled by the regression coefficients with regard to the uncertainty in the fuzzy input. In this example, since the half spread of the fuzzy input is 0.1, so the half spread of the fuzzy prediction is 0.2 since the regression model is Y=1+2X.

Table 7.9 Comparison between observations and predictions (FCF)

Obs.	x	y	$\hat{y}$
1	(0.9,1,1.1)	(2.7,3.0,3.3)	(2.8, 3.0, 3.2)
2	(1.9, 2,2.1)	(4.7,5.0,5.3)	(4.8, 5.0,5.2)
3	(2.9, 3,3.1)	(6.7,7.0,7.3)	(6.8,7.0,7.2)
4	(3.9, 4,4.1)	(8.7,9.0,9.3)	(8.8, 9.0,9.2)
5	(4.9, 5,5.1)	(10.7,11.0,11.3)	(10.8,11.0,11.2)
6	(5.9, 6,6.1)	(12.7,13.0,13.3)	(12.8, 13.0,13.2)
7	(6.9,7.0,7.1)	(14.7,15.0,15.3)	(14.8, 15.0,15.2)
8	(7.9,8.0,8.1)	(16.7,17.0,17.3)	(16.8, 17.0,17.2)

7.7 Fuzzy Input, Fuzzy Parameter and Crisp Output

Let us work on the test data set in Table 7.6. Apply (6.4) to this data set, let h=0 and we can obtain the following regression equation Y= (1, 0.01) + (2, 0.09) X. The result is shown in Figure (7.11), the pair of dashed lines denotes the regression model, and the horizontal short line denotes the fuzzy input-crisp output data which is an interval in x direction and a crisp value in y direction. Since the regression coefficient is a fuzzy data, according to equations (6.5), (6.6), the j-th prediction should locate at the interval

$$[\sum_{i=0}^{n}(c_i - m_i)x_{ji}, \sum_{i=0}^{n}(c_i + m_i)x_{ji}]$$

where (c_i,m_i) is the center and the half spread of the regression coefficient, x_{ji} is the defuzzified value of the j-th fuzzy input data, and n is the number of independent variables.

The estimated output equals to $\sum_{i=0}^{n} c_i x_{ji}$ at the possibility level 1. We can defuzzify the fuzzy prediction value to obtain a crisp estimation.

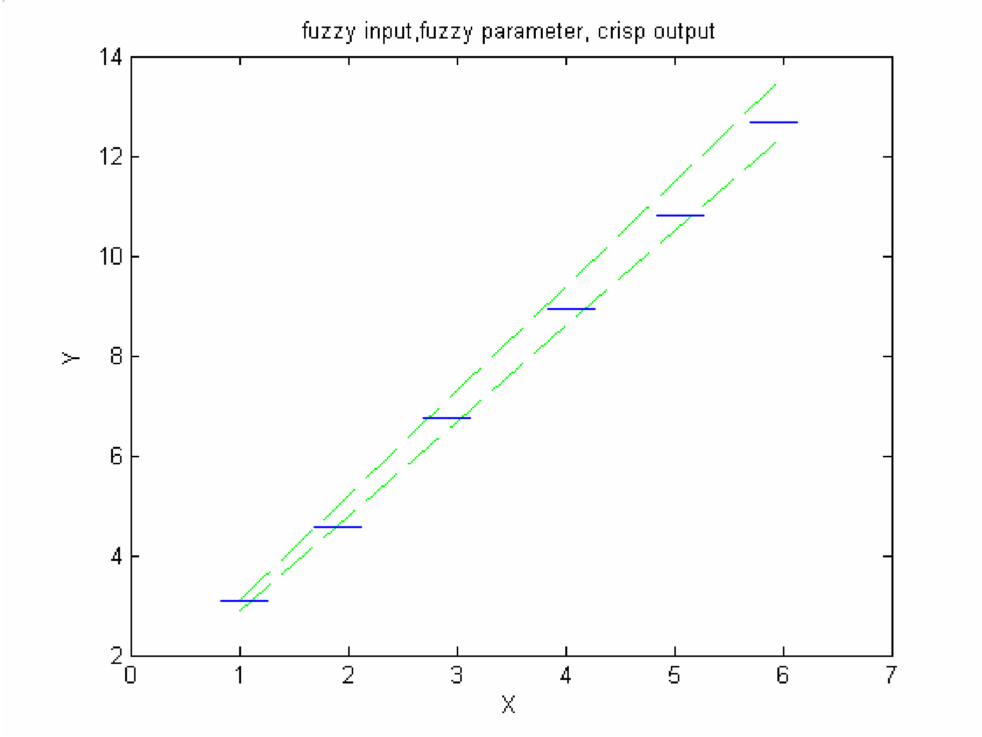

Fig. 7.11 Fuzzy input, fuzzy parameter, crisp output linear regression

The comparison of the observed y with the predicted output $\hat{y}$ with this model is shown in Table 7.10. The fourth column denotes the fuzzy output derived from the regression model, and the fifth column denotes the defuzzification value of the fuzzy output. It can be observed the defuzzified values are exactly the same as the observed values. Since only fuzzy errors are considered in this example, and those fuzzy errors are symmetrically distributed and they just cancel out each other, it is not surprising the observations and the estimations are the same.

Table 7.10 Comparison between observations and predictions (FFC)

Obs.	x	y	$\hat{y}$	$\hat{y}$ (defuzzification)
1	(0.7,1,1.3)	3	(2.9,3,3.1)	3
2	(1.7, 2, 2.3)	5	(4.8,5,5.2)	5
3	(2.7, 3, 3.3)	7	(6.7,7,7.3)	7
4	(3.7, 4, 4.3)	9	(8.6,9,9.4)	9
5	(4.7, 5, 5.3)	11	(10.5,11,11.5)	11
6	(5.7, 6, 6.3)	13	(12.4,13,13.6)	13

7.8 Fuzzy Input, Fuzzy Parameter and Fuzzy Output

The data set in Table 7.8 is used for this simulation, we apply (5.9) to this data set. Let h=0 and we obtain Y= (1, 0.1) + (2, 0) X. The result is shown in Figure (7.12), the pair of dashed lines is the regression model, and fuzzy input-output data is denoted by a solid square which is an interval in both x and y direction. It can be observed that the fuzzy data locate around the regression lines and they are compatible to some degree.

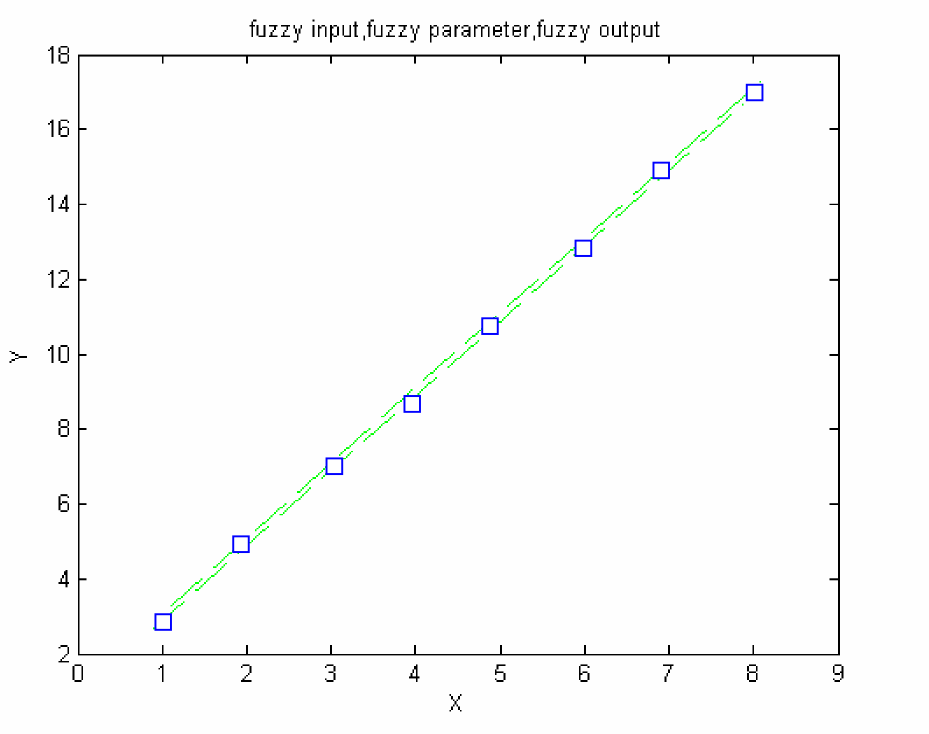

Fig. 7.12 Fuzzy input, fuzzy parameter, fuzzy output linear regression

In order to predict the output data with the regression model, we can defuzzify the input data, and then multiply the defuzzified input data by fuzzy regression coefficients.

The comparison of the observed output y with the predicted output $\hat{y}$ with this model is shown in Table 7.11. The third and the fourth column are observed outputs and the predicted fuzzy output data respectively. It is observed that this model does not change the center vertex of the fuzzy observations, however, the uncertainty in the input data is taken into account, that makes the spread of the predicted fuzzy output data is somewhat different from the spread of the observed output.

Table 7.11 Comparison between observations and predictions (FFF)

Obs.	x	y	$\hat{y}$
1	(0.9,1,1.1)	(2.7, 3, 3.3)	(2.9,3,3.1)
2	(1.9, 2,2.1)	(4.7, 5,5.3)	(4.9,5,5.1)
3	(2.9, 3,3.1)	(6.7, 7,7.3)	(6.9,7,7.1)
4	(3.9, 4,4.1)	(8.7, 9,9.3)	(8.9,9,9.1)
5	(4.9, 5,5.1)	(10.7, 11,11.3)	(10.9,11,11.1)
6	(5.9, 6,6.1)	(12.7, 13,13.3)	(12.9,13,13.1)
7	(6.9,7.0,7.1)	(14.7, 15,15.3)	(14.9,15,15.1)
8	(7.9,8.0,8.1)	(16.7, 17,17.3)	(16.9,17,17.1)

7.9 Summary

In this chapter, eight categories of fuzzy regression models are simulated, the classification is made based on the characteristic of the input, the parameter and the output data. Any of the input, the parameter and the output could be a crisp value or a fuzzy value. In general, minimum fuzziness criterion is applied to the fuzzy regression models, and we also need to consider the prediction and the observation to be compatible to some given degree. Numerical examples in this chapter are designed to test the methods we discussed in chapter 5 and chapter 6. In the crisp input, crisp parameter and crisp output case, only random errors are considered, and then we focus on exploring the fuzzy errors for the other seven cases. Simulation results show our methods work very well on the test data sets by comparing the observations and the predicted values.

CHAPTER 8

Applications of Fuzzy Regression

In this chapter, two applications are given showing how a fuzzy regression can be used.

8.1 Stock Price Forecast

The fuzzy regression models developed in the preceding section can be used in real applications. To illustrate, we will apply a fuzzy regression model to predict the short-term stock market price.

Fuzzy regression model is an alternative to evaluate the relation between the independent variables and the dependent variables among the forecasting models when the relationship is not obvious. Such phenomenon is significant especially for seasonal variation data when large amount of data are required to show the pattern. Because of its increasing importance in industries, in this study, we propose a method of applying fuzzy regression model for this purpose. By using two independent variables of the historical periodical data and the time index, the developed model shows the pattern of the short term stock price variation.

Forecasts can be generated in many different ways using many different approaches. Some forecasts are purely based on intuition and human judgment, while others require complex mathematical and computer based models.

Though approaches and techniques of forecasting come from many different disciplines including economics, mathematics, engineering, psychology and statistics, it has only been reality that fuzzy forecasting has become an identifiable and serious area for study.

Common forecast techniques include regression, pattern recognition, and time series analysis etc. Time series analysis includes moving average (MA), exponential smoothing and double exponential smoothing etc. Other forecasting method includes autoregressive method (AR), autoregressive and moving average method (ARMA), autoregressive and integrated moving average method (ARIMA) etc.

8.1.1 Experimental Setup and Data Preparation

In this section, we apply fuzzy regression model to predict the short term stock price based on the last few days' historical data. The stock price used in this study is selected from Intel Corporation (INTC) in the year of 2003. The historical stock price data is shown in Figure 8.1, our concern is to predict the closing price by taking advantage of both the low price and the high price from historical data.

The Table 8.1 shows two of the weekly reports of Intel Corporation in the year 2003, as shown by Figure 8.2. Observing the price curve, it is intuitively reasonable to assume a linear regression model to solve the problem.

Table 8.1 Weekly price report of Aug. 2003 [9]

Date	Time	Open	High	Low	Close	Volume
20030818	1600	25.0137	26.1399	24.9639	26.1	59081000
20030819	1600	26.2803	26.4498	25.8319	26.38	55966300
20030820	1600	26.0508	26.6487	26.0408	26.27	47210300
20030821	1600	26.599	26.6887	25.9213	26.3	66434900
20030822	1600	28.0675	28.9446	27.2302	27.3	1.21E+08
20030825	1600	27.4689	27.6683	26.9806	27.15	52037500
20030826	1600	26.8724	27.6499	26.5933	27.62	65213400
20030827	1600	27.5213	27.9898	27.3319	27.93	58217200
20030828	1600	28.0007	28.2498	27.7516	28.2	48631600
20030829	1600	28.0814	28.5498	27.9419	28.49	41986600

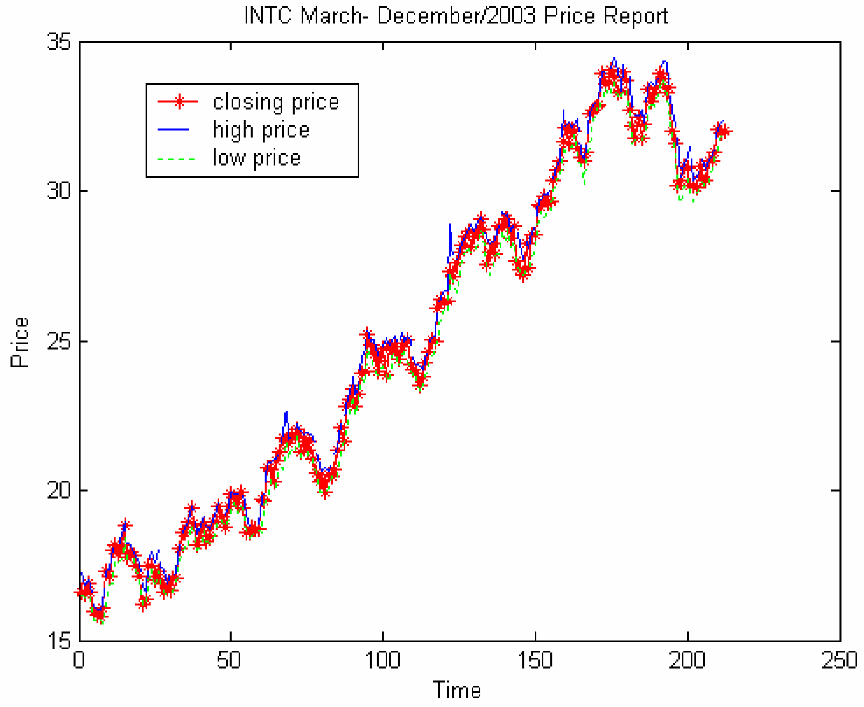

Fig. 8.1 INTC price report (03/2003 – 12/2003)

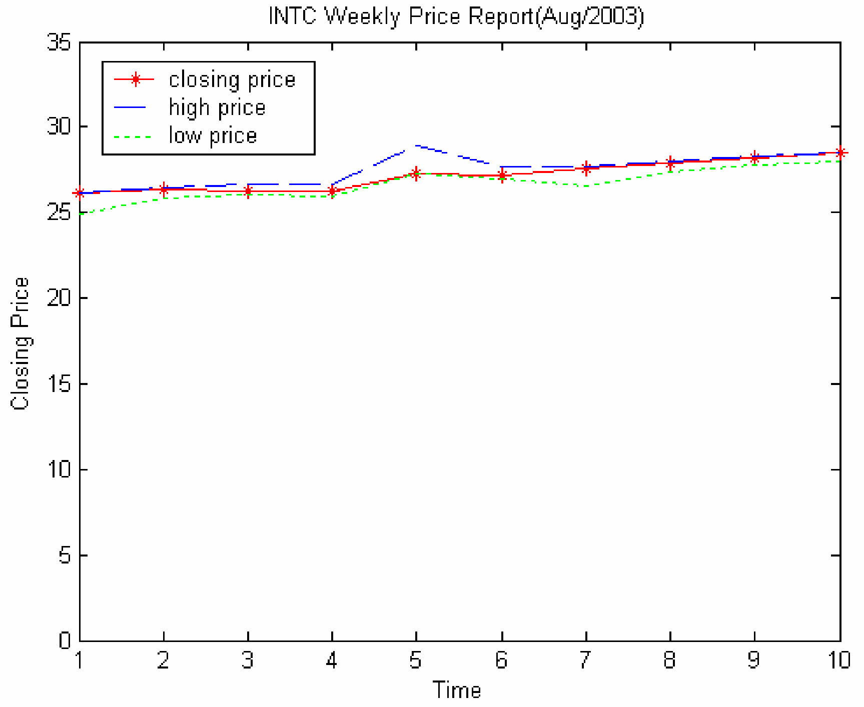

Fig. 8.2 INTC weekly price report (Aug. 18-29, 2003)

Since the daily stock price is an interval data which fluctuates between the low price and high price. It is more realistic to estimate an interval for closing price rather than an exact number. So the problem can be processed as a fuzzy input, crisp parameter, fuzzy output regression model which is defined in (5.8). The support of both fuzzy input and output data are an interval which is determined by the high price and the low price.

8.1.2 Procedure & Results

The procedure is described as follows.

1. The goal is to use the past three days' price to predict tomorrow's price, hence the objective function becomes $Y=m_0\times x_0+m_1\times x_1+m_2\times x_2+m_3$. where Y is tomorrow's price, x_0,x_1 and x_2 are past three days' price respectively, m_0,m_1,m_2 and m_3 are crisp regression coefficients.

2. Put all the week's data (or longer time) into the constraints in (5.8)

3. Minimize the sum of errors which come from the difference between the predictions and the observations and obtain the optimized solution.

Apply the above process to the data set given in Table (8.1). The goal is to predict the closing price of Aug.29 which is the dependent variable, the price data of Aug. 26, Aug. 27, and Aug. 28 are used as predictor variables and the whole two weeks' data are put into the constraints. We can obtain the simulation result:

$$Y=0.6139x_0+0.0382x_1+0.0874x_2+7.7119$$

and the predicted closing price is (27.51, 28.19, 28.22) which represents a fuzzy data with triangle membership function, the support of the data is in the interval [27.51, 28.22] and the defuzzified value is 27.98.

According to Table 8.1, the price of Aug. 29 fluctuates in the interval [27.94, 28.55], and the closing price is 28.49.

Define the prediction error $e = \frac{observation - prediction}{observation} * 100\%$

We have closing price prediction error = $\frac{(28.49 - 27.98)}{28.49} = 1.8\%$

8.2 Dosage-Film Response Analysis

Increasing concern for the potentially high radiation dose in interventional radiological procedures has led to the use of radiochromic films in the imaging modality. The GafChromic XR-Type R Dosimetry film has been used in this study. In this section, the whole experiment procedure will be described in detail.

8.2.1 Experimental Setup and Data Preparation

The x-ray photons incident on the film carries a statistical variation of fluctuations in the photon arrival rate at a given pixel point. This phenomenon is known as photon noise and follows Poisson distribution. Additionally, other inherent noise sources resulting from imaging system, film, and scanner also need to be taken into account. The goal of this application is to compute the dosage with given X-Ray images. In order to do that, first we need to know the relationship between the image intensity and the dosage. We create a series of calibration patterns and each of them corresponds to a specific dosage. In fact, pixel values in each calibration pattern are not homogeneous; it is common practice to take the average intensity value as the reference for the corresponding dosage.

To study the film response, an x-ray machine (Philips Optimus V5000, Philips Medical Sytems, Andover, MA) was used to create a calibration tablet. Pieces of the GafChromic film were exposed to different amounts of radiation with a maximum air kerma of 13.88Gy. This tablet was stored in a dark room under normal room temperature and humidity.

This created tablet was then scanned by a flatbed reflective-type scanner (Microtek ScanMaker 4800), as well as images with different dosage levels. The calibration patterns are shown in Figure 8.3. The goal of this application is to estimate the dosage based on the image and calibration patterns.

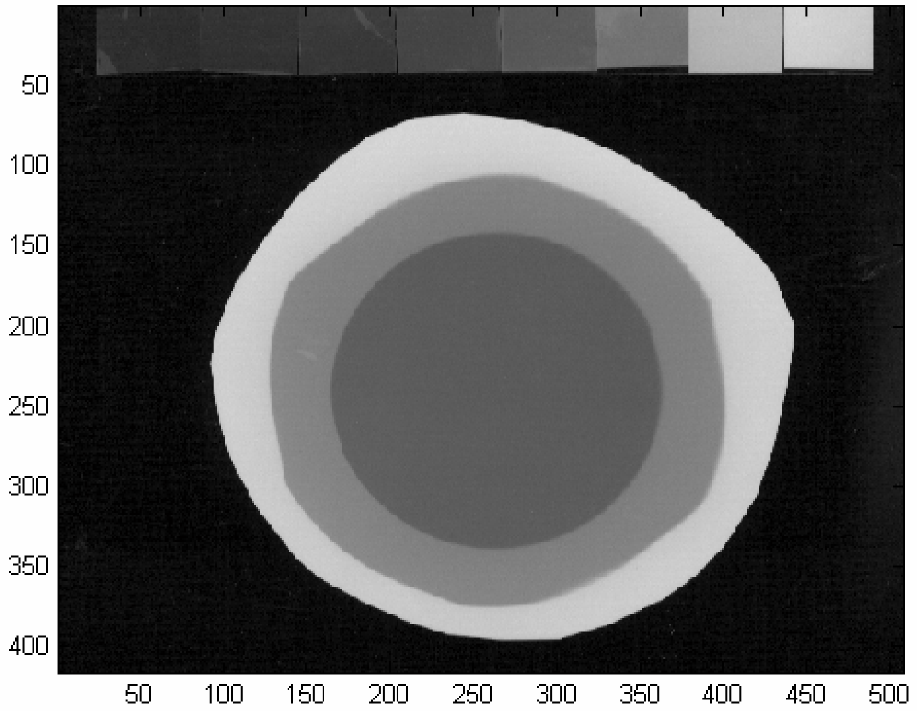

Fig. 8.3 Calibration patterns and film image

8.2.2 Scatter Plot of Dosage-film Response

From Figure 8.4, it is obvious the relationship between film response intensity and dosage is a nonlinear relationship. In the low intensity area, a small amount change in intensity could result in a big variation in the dosage computation. We have to be very careful about those low intensity pixel values since they are sensitive to our method.

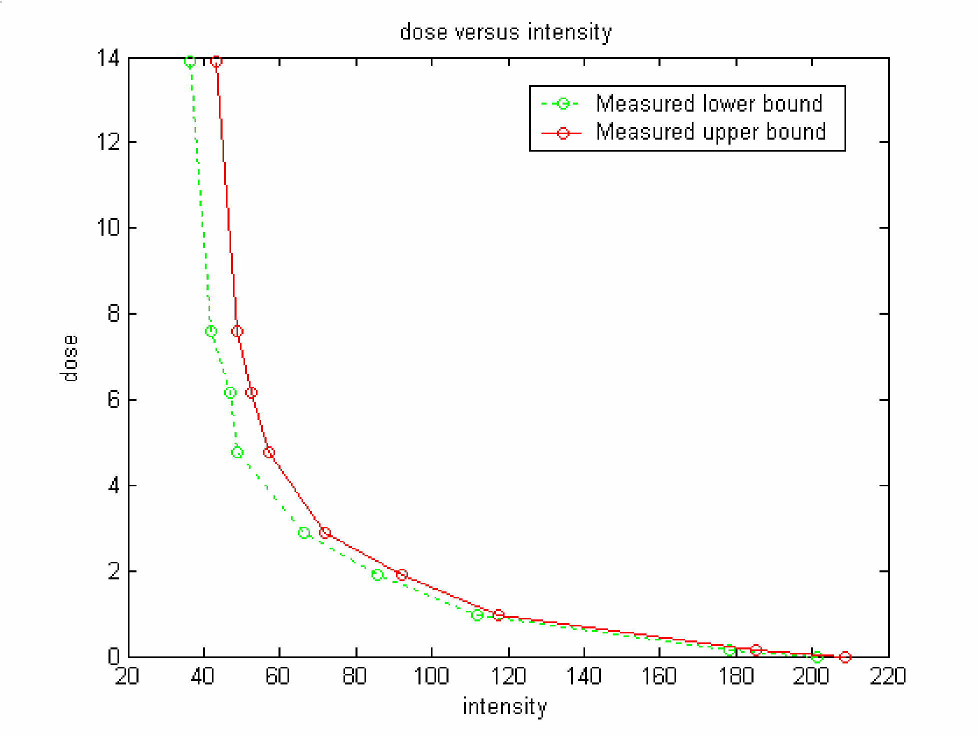

Fig. 8.4 Measured dosage-film response

8.2.3 Curve Fitting

Measured data is often accompanied by noise. A process of quantitatively estimating the trend of the output is known as curve fitting, and it is widely used. The curve fitting process fits equations of approximate curves to the data. The fitting curves are not unique for a given set of data. A curve with a minimal deviation from all data points is usually desired. The best-fitting curve can be obtained by the method of least squares.

In our study we will use least squares polynomials fitting. Polynomials are one of the most commonly used types of curves in regression. Suppose the least-squares m-th degree polynomials are used in curve fitting, it can be formulated as $y = f(x) = a_0 + a_1 x + a_2 x^2 + ... + a_m x^m$ to approximate the given set of data (x_1, y_1), (x_2, y_2)… (x_n, y_n) where $n \geq m+1$

The best fitting curve $f(x)$ should have the least square error, i.e.,

$$\text{Min} \quad \sum_{i=1}^{n}[y_i - f(x_i)]^2 = \sum_{i=1}^{n}\left[y_i - (a_0 + a_1 x_i + a_2 x_i^2 + ... + a_m x_i^m)\right]^2 \qquad (8.1)$$

To obtain the least square error, we take the first derivative of (8.1), with respect to unknown coefficients a_0, a_1,…,a_m, we set the gradients to zero.

$$\sum y_i = a_0 n + a_1 \sum x_i + a_2 \sum x_i^2 + ... + a_m \sum x_i^m$$

$$\sum x_i y_i = a_0 \sum x_i + a_1 \sum x_i^2 + a_2 \sum x_i^3 + ... + a_m \sum x_i^{m+1}$$

……

$$\sum x_i^m y_i = a_0 \sum x_i^m + a_1 \sum x_i^{m+1} + a_2 \sum x_i^{m+2} + ... + a_m \sum x_i^{2m}$$

By solving the above linear equations, the unknown coefficients can be obtained. In this study, we reform the above curve fitting on the dosage-film response with a third order polynomials.

8.2.4. Interpolation

In many real applications, data collected from the field are usually discrete and the physical meanings of the data are not always well known; the process of estimating the outcomes between sampled data points is called interpolation; most popular interpolation techniques include polynomial interpolation, rational function interpolation and cubic spline interpolation etc.

Given a set of data (x_1,y_1),(x_2,y_2),…,(x_n,y_n) where $x_1 \neq x_2 \neq ... \neq x_n$ .The Lagrange's formula of polynomial interpolation is

$$y \approx \frac{(x-x_2)(x-x_3)...(x-x_n)}{(x_1-x_2)(x_1-x_3)...(x_1-x_n)}y_1 + \frac{(x-x_1)(x-x_3)...(x-x_n)}{(x_2-x_1)(x_2-x_3)...(x_2-x_n)}y_2 + ... + \frac{(x-x_1)(x-x_2)...(x-x_{n-1})}{(x_n-x_1)(x_n-x_2)...(x-x_{n-1})}y_n$$

The cubic spline interpolation technique is used for the interpolation of the dosage-film response curve. Given a data set (x_0, f(x_0)), (x_1, f(x_1))… (x_n, f (x_n)), assume that $f''(x_0) = f''(x_n) = 0$, then the other second derivatives can be estimated as follows:

$$(x_i - x_{i-1})f''(x_{i-1}) + 2(x_{i+1} - x_{i-1})f''(x_i) + (x_{i+1} - x_i)f''(x_{i+1})$$

The cubic function for each interval is then modeled as:

$$f_i(x) = \frac{f''(x_{i-1})}{6(x_i - x_{i-1})}(x_i - x)^3 + \frac{f''(x_i)}{6(x_i - x_{i-1})}(x - x_{i-1})^3 + \left[\frac{f(x_{i-1})}{x_i - x_{i-1}} - \frac{f''(x_{i-1})(x_i - x_{i-1})}{6}\right](x_i - x) + \left[\frac{f(x_i)}{x_i - x_{i-1}} - \frac{f''(x_i)(x_i - x_{i-1})}{6}\right](x - x_{i-1})$$

(8.2)

8.2.5 Consider Both Errors in Regression Model

As we have discussed above, random errors can be seen as observations errors, and fuzzy errors come from the inherent uncertainty of the system. Both randomness and fuzziness have to be considered in this process. In order to compute the dosage, first we need to find the relationship between the pixel intensity and dosage. Let dosage be the independent variable and intensity be the dependent variable, since the pixel intensity in each calibration pattern has some variation; it can be processed as a fuzzy data. Here we use triangle membership function, the vertex is the average intensity value, and the support is determined by the minimum and maximum intensity value. So this problem becomes a fuzzy input, fuzzy parameter, crisp output regression problem.

We combine both the probability theory and the possibility theory together in this study, and propose the following algorithm:

Step1: preprocessing the calibration pattern

Since calibration pattern is scanned with black reference, it is possible that some lower pixel readings in the pattern is the black reference itself instead of the film. One must deal with those pixels or it will severely distort the estimation since the darker area means a high dosage level.

Outliers have significant impact on the regression equation, it is important to remove them before we do a proper regression analysis. Based on the intensity histogram for each calibration pattern, we remove those points whose frequency count is less than 5% of the total events.

Step2: Consider random errors

Determine the center of the fuzzy regression coefficient by the least square method. In this study a 4th order polynomial curve fitting technique is used.

Step3: Consider fuzzy errors

Since there is only one independent variable, and all the other X^n terms are highly correlated with the X term, we set spreads of those high order regression coefficients as zero.

By solving the linear programming problem, the lower limit and the upper limit dosage-film response curves can be computed. The true dosage value will be within this estimated interval.

Step 4: Compute the dosage for each image using dose- film response curve

Now we have obtained the dosage-intensity relationship curves. It is straightforward to compute the dosage by interpolating or extrapolating this curve if X-Ray images are available.

Step 5: Defuzzification

We use the centroid method in this case. This method weighs all the values with different possibility levels to form a single value. In our case, let X_{ic} and Y_{ic} be the defuzzified values of X_i and Y_i, the following formulas define the defuzzification values [61]:

$$X_{ic} = \frac{\int_{-\infty}^{+\infty} x\mu(x)dx}{\int_{-\infty}^{+\infty} \mu(x)dx} = \frac{1}{3}(X_{il} + X_{im} + X_{ir})$$

$$Y_{ic} = \frac{\int_{-\infty}^{+\infty} y\mu(y)dy}{\int_{-\infty}^{+\infty} \mu(y)dy} = \frac{1}{3}(Y_{il} + Y_{im} + Y_{ir}) \tag{8.3}$$

8.2.6 Results & Discussion

To test the validity of the proposed method, we used some scanned film images obtained from University of Oklahoma Health Science Center along with the calibration patterns. Four images were randomly picked from those films and the dosage level of each image was computed. Finally we compared those results with the values given by the X-ray machine.

Figure 8.5 and Figure 8.6 describe the resultant dosage-film response for h equals to 0 and 0.9 respectively. As we can see, the estimation becomes fuzzier with the increasing h. If the estimated interval is too wide, it will be hard to provide precise estimation for the decision maker, but if it is too narrow, some observations could not be covered in this interval, the decision maker will be too optimistic about the estimation. Thus an appropriate value is desired. In our case, even when h equals to 0, the estimation has covered all the observations (after data preprocessing), it is still reasonably informative when h increases to 0.9.

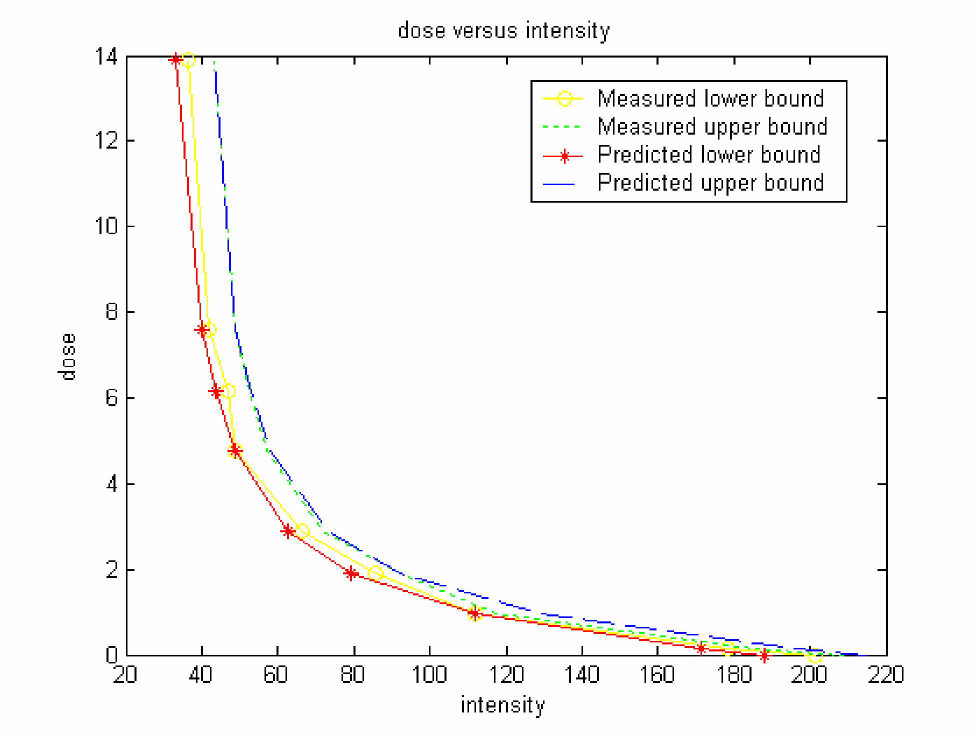

Fig. 8.5 Dosage- film response at h=0

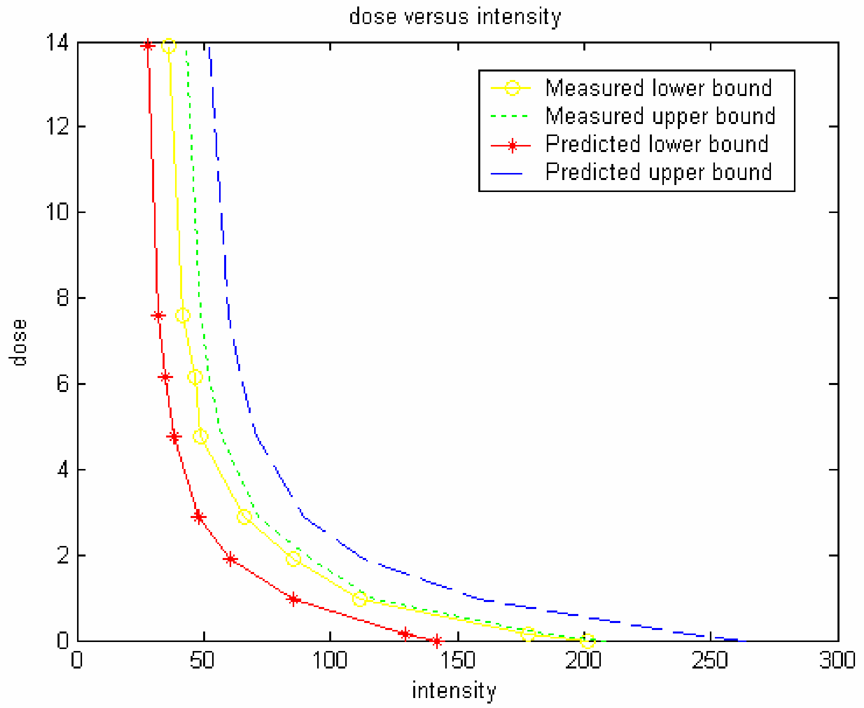

Fig. 8.6 Dosage- film response at h=0.9

Table 8.2 Performance comparison between classical regression model and fuzzy regression model

No.	Machine value	CLR	Error	FLR(h)	Error
1	79.3	86.20	8.7%	87.44 (0)	10.26%
2	584	763.83	30.79%	681.69 (0.9)	16.73%
3	748	929.95	24.32%	777.32(0.9)	3.92%
4	1124	1007.7	-10.35%	948.12 (0.5)	-15.65%

Finally we list the experiment result as the Table 8.2. It is the results obtained by applying the method we proposed in this paper to the four test film images. If we only consider the random errors of this process, we will get results which are listed in column 3.

Column 5 gives results which have considered both the random errors and the model errors. From Figure 8.4, we find that most of the data in the calibration patterns is focused on the dosage range 0~8 Gy; while there is almost no reference data in the range between 8Gy and14 Gy. So those pixels with value between 30 and 40 are unreliable in the computation and it is a major error source especially when those data correspond to a high dosage level. Note that we have assumed those pixels with value less than 30 are black background and they have nothing to do with dosage. If the image has a large amount of unreliable data, then the model errors can not be ignored. So it is intuitively reasonable that we should set a relatively bigger h for the data set whose histogram has a large part of low pixels. Figure 8.7, 8.8, 8.9, 8.10 are histograms of the four test images in this study. We can see that the first image has almost no low value pixels except those black backgrounds. And the other three images have more or less amount of low value pixels that is the reason why they use high h value.

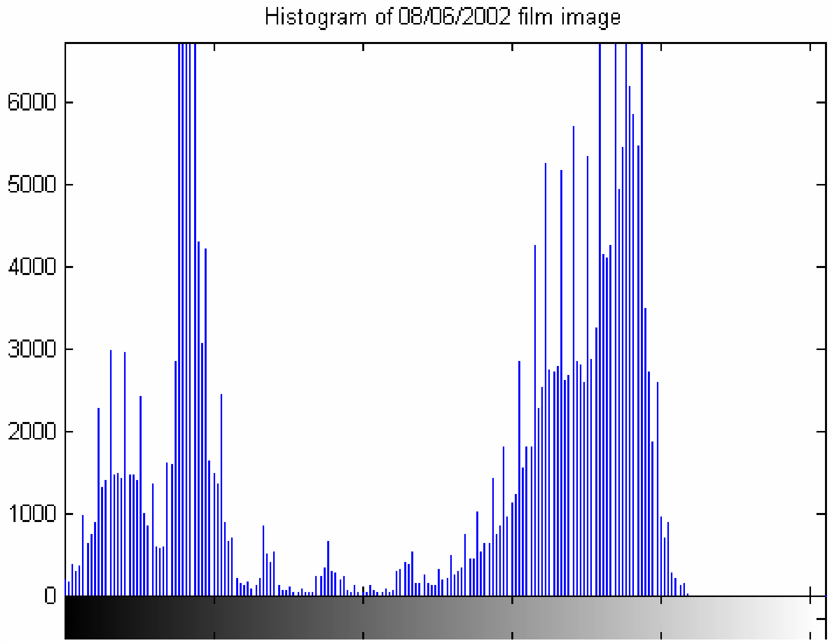

Fig. 8.7 Histogram of 08/06 image

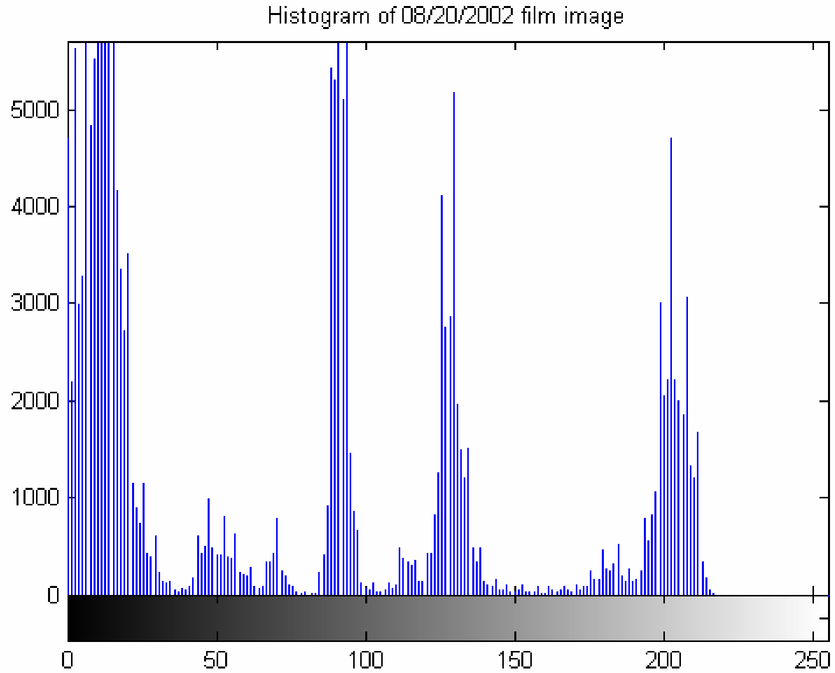

Fig. 8.8 Histogram of 08/20 image

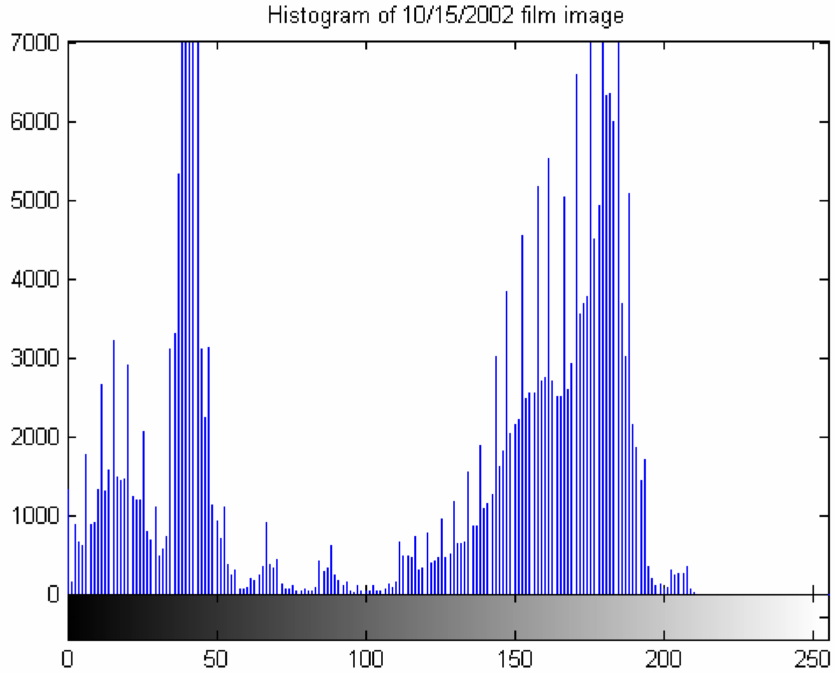

Fig. 8.9 Histogram of 10/15 image

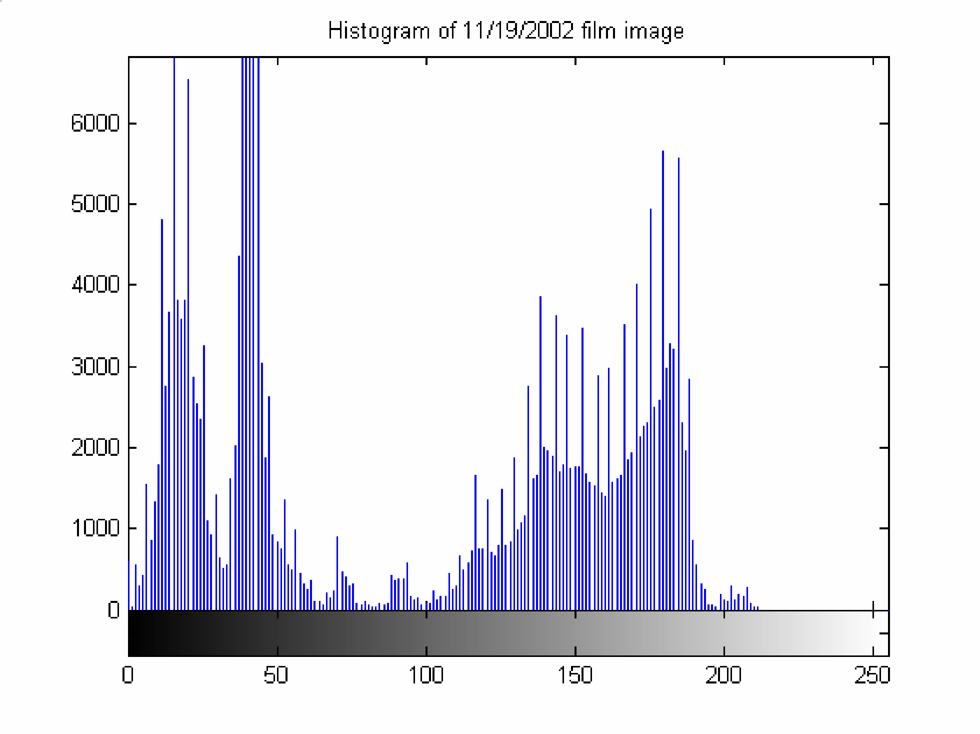

Fig. 8.10 Histogram of 11/19 image

Of course, besides the distribution the accuracy of the scanned data also plays an important role in the final result. From Table 8.2, it is obvious that the 8/20 and 11/19 images have relatively good image quality. Although the 11/19 image has some low pixel part, the estimation error is still acceptable. The 8/20 image has the lowest random error. In those last two images, the randomness phenomenon dominates this process. Degradation in the fuzzy regression error is observed; in the 8/6 and 10/15 images, if only random errors are considered, unbearable errors were observed. In this case fuzzy regression method improves the performance significantly. In general, the fuzzy regression method generates a more consistent result than the classical method.

Both opportunity and challenge exist in applying fuzzy regression model to data analysis. It can take advantage of expert knowledge to improve the model which is hard to be implemented by classic probability theory, however, the decision about the possibility of a data set is very flexible. If a bad decision has been made, the result would not improve much even gets worse in some cases. So it is still a research topic about how to make a good support system which can eliminate the random factor in decision making process as much as possible. It is now under development for our future research.

CHAPTER 9

Summary

Fuzzy set theory is widely applied to a variety of data mining applications recently [95-105]. Those are data sets with imperfect knowledge. For example human language, the data from imprecise measuring instruments etc. This dissertation contributes to the application of fuzzy set theory on classical correlation and regression analysis. I presented two types of fuzzy correlation models. In the first type of fuzzy model we consider the data points are fuzzy, our method works more efficiently than the methods recently reported in the literature. The influence of the fuzziness and the shape of data membership functions are investigated, simulation results verify our theoretic derivation.

I developed the concept and the model of the second type of fuzzy correlation for the first time and presented simulation results for a variety of data sets with different parameters. The results show the relationship between the raw data sets and the distribution of fuzzy correlations, and also show the good possibility that fuzzy correlations can be a good approximation to probabilistic density functions of the correlation coefficient for data sets generated by the same system. The proposed model saves people a lot of work to collect data samples in order to get statistical information.

A family of fuzzy regression models was developed based on different combinations of input, output and regression parameters. Those models are formulated as the linear programming problems. I complemented three categories of models which have not been filled in the previous work. In crisp input, crisp parameter, fuzzy output case, experimental results show our method will give more informative result than the classical regression estimation. Fuzzy regression is based on the possibility instead of the probability theory; however, it can give us a good approximation to the estimation derived from the statistical theory.

Two application cases are given in the last section of this dissertation. The first is short term stock price prediction. A fuzzy regression model is built to predict next day's stock price based on the last two weeks' price trend. The result generated by the fuzzy regression model is a range instead of a single value, and it gives the decision maker a more realistic view about his or her investment. In the dosage-film response estimation example, fuzzy regression model also provides us a more realistic and consistent result than the traditional methods since the model has formulated the uncertainty characteristic of the system in a mathematical way.

The research in applying fuzzy set theory to correlation and regression analysis is ongoing. The relationship between the statistical probability density function and the fuzzy correlation, the relationship between the classical regression prediction and the fuzzy regression prediction has yet to be addressed seriously. More theoretic work is still under development at this time.

It is a common starting point to assume linear relationship between variables, for some cases, a nonlinear function can be expressed as a straight line by appropriate transformation. However, not all data sets can be or transformed into linear relationship. Nonlinear regression models have to be developed to fit nonlinear data sets. To my knowledge not much work has been done on this issue, future research will explore the modeling and usage of fuzzy nonlinear regression models in many applications.

REFERENCES

[1] George J. Klir, Bo Yuan, Fuzzy Sets and Fuzzy Logic, Englewood Cliffs, NJ: Prentice-Hall, 1995

[2] Witold Pedrycz, Michael H. Smith, Granular correlation analysis in data mining, IEEE International Fuzzy Systems Conference Proceedings, pp.1235-1240, 1999

[3] W.Pedrycz, F.Gomide, Fuzzy Sets: An Introduction, Analysis and Design, pp.129-148, MIT Press, Cambridge, MA, 1998

[4] L.A. Zadeh, Probability measures of fuzzy events, Journal of Math. Analysis Application, Vol.23, pp. 421-427, 1968

[5] Peter Y. Chen, Paula M. Popovich, Correlation: Parametric and Nonparametric Measures, pp.9-15, John Wiley & Sons, 1982

[6] C.Yu, Correlation of fuzzy numbers, Fuzzy Sets & Systems 55, pp.303-307, 1993

[7] Shiang-Tai Liu, Chiang Kao, Fuzzy measures for correlation coefficient of fuzzy numbers, Fuzzy Sets and Systems 128, pp. 267-275, 2002

[8] Ding-An Chiang, Nancy P. Lin, Correlation of fuzzy sets, Fuzzy Sets and Systems 102, pp. 221-226, 1999

[9] HQuotes.com company, HQuote Pro Downloader software v6.51, http://www.hquotes.com/

[10] K. Atanassov, Intuitionistic fuzzy sets, Fuzzy Sets and Systems 20, pp. 87-96, 1986

[11] Ding-An Chiang, Nancy P. Lin, Partial correlation of fuzzy sets, Fuzzy Sets and Systems 110, pp.209-215, 2000

[12] B.B. Chaudhuri, A. Bhattacharya, On correlation between two fuzzy sets, Fuzzy Sets and Systems 118, pp.447-456, 2001

[13] A.M. Goon, M.K. Gupta, D. Dasgupta, Fundamentals of Statistics, Vol. 1, the World Press Private Limited, India, 1985

[14] A. Kendal, Fuzzy Mathematical Techniques with Applications, Addison-Wesley, Reading, MA, 1986

[15] C.A. Murthy, S.K. Pal and D. Dutta Majumder, Correlation between two fuzzy membership functions. Fuzzy Sets and Systems 17, pp. 23–38, 1985

[16] Norio Watanabe and Tadashi Imaizumi, A fuzzy correlation coefficient for fuzzy random variables, IEEE International Fuzzy Systems Conference Proceedings, pp.1035-1038, 1999

[17] Daniel Ramot, Ron Milo, Menahem Friedman and Abraham Kandel, On fuzzy correlations, IEEE Transactions on Systems, Man, and Cybernetics-part B: Cybernetics, Vol. 31, No. 3, 2001

[18] A. Kandel, Fuzzy Mathematical Techniques and Applications, Reading, MA: Addison-Wesley, 1986

[19] D.Dubois and H. Prade, Fuzzy Sets and Systems: Theory and Applications. New York: Academic, 1980

[20] M. Sugeno, Fuzzy measures and fuzzy integrals: a survey, fuzzy automata and decision processes, M.M. Gupta, G.N. Saridis and B.R. Gains, Eds Amsterdam, The Netherlands: North-Holland, pp. 89-102, 1977

[21] A. Kaufmann and M.M. Gupta, Introduction to Fuzzy Arithmetic: Theory and Applications. New York: Van-Nostrand, 1985

[22] D.Dubois and H.Prade, Fuzzy numbers: an overview, Analysis of Fuzzy Information –Vol. 1: Mathematics and Logic, J.C. Bezdek, Ed. Boca Raton, FL: CRC, pp.3-39, 1987

[23] R.E. Moore, Interval Analysis, Englewood Cliffs, JF: Prentice-Hall, 1966

[24] R.E. Moore, Methods and Applications of Interval Analysis, SIAM, Philadelphia, 1979

[25] H. Bustince, P.Burillo, Correlation of interval-valued intuitionistic fuzzy sets, Fuzzy Sets and Systems 74, pp. 237-244, 1995

[26] D.H.Hong, S.Y.Hwang, A note on the correlation of fuzzy numbers, Fuzzy Sets and Systems 75, pp. 77-81, 1995

[27] D.H. Hong, S.Y. Hwang, Correlation of intuitionistic fuzzy sets in probability spaces, Fuzzy Sets and Systems 75, pp. 77-81, 1995

[28] T.Gerstenkorn, J.Manko, Correlation of intuitionistic fuzzy sets, Fuzzy Sets and Systems 44, pp. 39-43, 1991

[29] LINDO Systems Inc. LINGO User's Guide, LINDO Systems Inc., Chicago, 1999

[30] H.T.Nguyen, A note on the extension principle for fuzzy sets, Journal of Mathematical Analysis Application, Vol.64, pp. 369-380, 1978

[31] G.V.Reklaitis, A. Ravindran, K.M. Ragsdell, Engineering Optimization, Wiley, New York, 1983

[32] G.Wang, X. Li, Correlation and information energy of interval-valued fuzzy numbers, Fuzzy Sets and Systems 103, pp.169-175, 1999

[33] R.R.Yager, A characterization of the extension principle, Fuzzy Sets and Systems 18, pp. 205-217, 1986

[34] L.A.Zadeh, Fuzzy sets as a basis for a theory of possibility, Fuzzy Sets and Systems 1, pp.3-28, 1978

[35] L.A. Zadeh, J. Kacprzyk, Fuzzy Logic for the Management of Uncertainty, Wiley,

New York, 1992

[36] L.A. Zadeh, Fuzzy sets, Information and Control 8, pp. 338-353, 1965

[37] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning I, Information Science 8, pp.199-249, 1975

[38] H.J. Zimmermann, Fuzzy Set Theory and its Applications, 3rd Edition, Kluwer-Nijhoff, Boston, 1996

[39] D.C.Montgomery, E.A.Peck, Introduction to Linear Regression Analysis, Wiley, New York, 1982

[40] H.Bandemer, Evaluating explicit functional relationship from fuzzy observations, Fuzzy Sets and Systems 16, pp. 41-52, 1985

[41] D. Dubois, H. Prade, Possibility Theory, Plenum Press, New York, 1988

[42] D.Redden, W.Woodall, Properties of certain fuzzy regression methods, Fuzzy Sets and Systems 64, pp.361-375, 1994

[43] D.Redden, W.Woodall, Further examination of fuzzy linear regression, Fuzzy Sets and Systems 79, pp. 203-211, 1996

[44] Yun-His O. Chang, Bilal M. Ayyub, Fuzzy regression methods- a comparative assessment, Fuzzy Sets and Systems 119, pp.187-203, 2001

[45] Yun-His O. Chang, Bilal M. Ayyub, Reliability analysis in fuzzy regression, Proc. Annual Conf. of the North American Fuzzy Information Processing Society, Allentown, PA, USA, pp. 93-97, 1993

[46] Yun-His O. Chang, P. Johnson, S.Tokar, B.M.Ayyub, Least squares in fuzzy regression: limits and extensions, Proc. Annual Conf. of the North American Fuzzy Information Processing Society, Allentown, PA, USA, pp. 98-102, 1993

[47] H.Tanaka, S.Uejima, Kazak, Fuzzy linear regression analysis with fuzzy model, IEEE Systems, Trans. Systems Man, Cybernet. SMC-2, pp. 903-907, 1982

[48] H. Tanaka, Fuzzy data analysis by possibilistic linear models, Fuzzy Sets and Systems 24, pp.363-375, 1987

[49] H. Tanaka, J.Watada, Possibilistic linear systems and their applications to the linear regression model, Fuzzy Sets and Systems 27, pp. 275-289, 1988

[50] Hideo Tanaka, Haekwan Lee, Fuzzy linear regression combining central tendency and possibilistic properties, IEEE Trans. on Fuzzy Systems, pp. 63-68, 1997

[51] H.Tanaka, H. Ishibuchi, Identification of possibilistic linear systems by quadratic membership functions of fuzzy parameters, Fuzzy Sets and Systems 41, pp.145-160, 1991

[52] Hideo Tanaka, Interval regression analysis by quadratic programming approach, IEEE Trans. on Fuzzy Systems, 1998

[53] H.Tanaka, H. Ishibuchi, S.Yoshikawa, Exponential possibility regression analysis, Fuzzy Sets and Systems 69, pp.305-318, 1995

[54] J.Kacprzyk, M.Fedrizzi, Fuzzy regression analysis, Physica-Verlag, Heidelberg, 1992

[55] K.J.Kim, H. Moskowitz, M.Koksalan, Fuzzy versus statistical linear regression, European J. Oper. Res. 92, pp.417-434, 1996

[56] H. Moskowitz, K.J.Kim, On assessing the h value in fuzzy linear regression, Fuzzy Sets and Systems 58, pp. 303-327,1993

[57] B.Kim, R.R. Bishu, Evaluation of fuzzy linear regression models by comparing membership functions, Fuzzy Sets and Systems 100, pp. 343-352, 1998

[58] K.K. Yen, S.Ghoshray, G.Roig, A linear regression model using triangular fuzzy

number coefficients, Fuzzy Sets and Systems 106, pp.167-177, 1999

[59] Rajan Alex, P.Z.Wang, A new resolution of fuzzy regression analysis, IEEE Trans. On Fuzzy Systems, 1998

[60] Yun-Shiow Chen, Outliers detection and confidence interval modification in fuzzy regression, Fuzzy Sets and Systems 119, pp.259-272, 2001

[61] Chiang Kao, Chin-Lu Chyu, A fuzzy linear regression model with better explanatory power, Fuzzy Sets and Systems 126, pp. 401-409, 2002

[62] Yun-His O.Chang, Hybrid fuzzy least-squares regression analysis and its reliability measures, Fuzzy Sets and Systems 119, pp.225-246, 2001

[63] D.Savic and W.Pedrycz, Evaluation of fuzzy linear regression models, Fuzzy Sets and Systems, vol.39, pp.51-63, 1991

[64] E.S.Lee, P.T. Chang, Fuzzy linear regression analysis with spread unconstrained in sign, Comp. Math. Appl. 28(4), pp.61-70, 1994

[65] G. Peters, Fuzzy linear regression with fuzzy intervals, Fuzzy Sets and Systems 63, pp. 45-55, 1994

[66] A.Celmins, Least squares model fitting to fuzzy vector data, Fuzzy Sets and Systems 22, pp 245-269, 1987

[67] A.Celmins, Multidimensional least-squares model fitting of fuzzy models, Mathematical Modeling 9, pp. 669-690, 1987

[68] P.T. Chang, E S. Lee, A generalized fuzzy weighted least-squares regression, Fuzzy Sets and Systems 82, pp. 289-298, 1996

[69] P.Diamond, Fuzzy least squares, Inform. Sci., vol.46, pp. 141-157, 1988

[70] H. Ishibuchi, Fuzzy regression analysis, Japan. J. Fuzzy Theory and Systems 4, pp.

137-148, 1992

[71] James J. Buckley, Yoichi Hayashi, Fuzzy neural networks: a survey, Fuzzy Sets and Systems 66, pp. 1-13, 1994

[72] Hisao Ishibuchi, Kitack Kwon, Hideo Tanaka, A learning algorithm of fuzzy neural networks with triangular fuzzy weights, Fuzzy Sets and Systems 71, pp. 277-293, 1995

[73] Hisao Ishibuchi, Hideo Tanaka, Fuzzy regression analysis using neural networks, Fuzzy Sets and Systems 50, pp. 257-265, 1992

[74] Andras Bardossy, Note on fuzzy regression, Fuzzy sets and systems 37, pp.65-75, 1990

[75] Andras Bardossy, Roberta Hagaman, Lucien Duckstein, Istvan Bogardi, Fuzzy least squares regression: theory and application, Fuzzy regression analysis, Physical Verlag, Heidelberg, pp.181-193, 1992

[76] Arnold Kaufmann, Madan M. Gputa, Van Nostrand Reinhold, Introduction to fuzzy arithmetic theory and applications, New York, NY, 1991

[77] Pierpaolo D'Urso, Tommaso Gastaldi, An "orderwise" polynomial regression procedure for fuzzy data, Fuzzy Sets and Systems 130, pp.1-19, 2002

[78] M.Sakawa, H.Yano, Multiobjective fuzzy linear regression analysis for fuzzy input-output data, Fuzzy Sets and Systems 47, pp.173-181, 1992

[79] G. Thomas, R.Y.L.Chu, and Frank Rabe, A study of GafChromic XR Type R film response with reflective-type densitometers and economical flatbed scanners, Journal of applied clinical medical physics, Vol. 4, No. 4, 2003

[80] Willi Meier, Richard Weber, Hans-J. Zimmermann, Fuzzy data analysis: methods and industrial applications, Fuzzy Sets and Systems 61, pp. 19-28, 1994

[81] Norman R. Draper, Harry Smith, Applied Regression Analysis, pp15-45, Wiley Series in Probability and Statistics, 1998

[82] Albin, M.A., Fuzzy sets and their application to medical diagnosis and patterns recognition. (Ph.D. Dissertation), University of California, Berkeley, 1975

[83] Su, C.Y. and Y. Stepanenko, Adaptive control of a class of nonlinear systems with fuzzy logic, IEEE Trans. on Fuzzy Systems 2, pp.285-294, 1994

[84] Smithson. M., Applications of fuzzy set concepts to behavioral sciences. J. of Mathematical Social Sciences, Vol. 2, pp.257-274, 1982

[85] Lee, E.T. and L.A. Zadeh, Note on fuzzy languages, Information Sciences, 1(4), pp.421-434, 1969

[86] Wang, L.X., Stable adaptive fuzzy control of nonlinear systems, IEEE Trans on Fuzzy Systems, 1(2), pp.146-155, 1993

[87] Yager, R.R. and L.A.Zadeh, An introduction to fuzzy logic applications in intelligent systems, Kluwer, Boston, 1992

[88] Zadeh, L.A., K.S.Fu, K.Tanaka and M.Shimura, Fuzzy sets and their applications to cognitive and decision processes. Academic Press, New York, 1975

[89] Salski, A. Fuzzy knowledge-based models in ecological research, Ecological Modelling 63(1-4), pp. 103-112, 1992

[90] Kandel, A., Fuzzy statistics and forecast evaluation, IEEE Trans. on Systems, Man and Cybernetics, 8(5), pp.396-400, 1978

[91] Eick, C.F. and N.N.Mehta, Decision making involving imperfect knowledge, IEEE Trans. on Systems, Man, and Cybernetics, 23(3), pp. 840-851, 1993

[92] Onisawa, T., An application of fuzzy concepts to modeling of reliability analysis,

Fuzzy Sets and Systems, 37(3), pp.267-286, 1990

[93] Sun, Y.L., Er, M.J**.,** Hybrid Fuzzy Control of Robotics Systems, IEEE Trans. on Fuzzy Systems, pp 755- 765, 2004

[94] Pedrycz W., Why triangular membership functions? , Fuzzy Sets and Systems 64(1), pp. 21-30, 1994

[95] Jung-Hsien Chiang, Shihong Yue, Zong-Xian Yin, A new fuzzy cover approach to clustering, IEEE Trans. on Fuzzy Systems, pp199- 208, 2004

[96] Gaweda A.E., Zurada J.M., Data-driven linguistic modeling using relational fuzzy rules, IEEE Trans. on Fuzzy Systems, pp121- 134, 2003

[97] Eschrich S., Jingwei Ke, Hall, L.O. and Goldgof, D.B., Fast accurate fuzzy clustering through data reduction, IEEE Trans. on Fuzzy Systems, pp 262- 270, 2003

[98] Van De Ville D., Nachtegael M., Van der Weken D., Kerre E.E., Philips, W., Lemahieu, I., Noise reduction by fuzzy image filtering, IEEE Trans. on Fuzzy Systems, pp 429- 436, 2003

[99] Govindaraju, V., Ianakiev, K., Potential improvement of classifier accuracy by using fuzzy measures, IEEE Trans. on Fuzzy Systems, pp.679-690, 2000

[100] Auephanwiriyakul, S., Keller, J.M., Analysis and efficient implementation of a linguistic fuzzy c-means, IEEE Trans. on Fuzzy Systems, pp563- 582, 2002

[101] Kolen, J.F.; Hutcheson, T., Reducing the time complexity of the fuzzy c-means algorithm, IEEE Trans. On Fuzzy Systems, pp263-267, 2002

[102] Kaymak U., Setnes M., Fuzzy clustering with volume prototypes and adaptive cluster merging, IEEE Trans. on Fuzzy Systems, pp705- 712, 2002

[103] Takagi, T., Kawase, K., A trial for data retrieval using conceptual fuzzy sets, IEEE

Trans. on Fuzzy Systems, pp 497-505, 2001

[104] Runkler, T.A., Bezdek, J.C., Alternating cluster estimation: a new tool for clustering and function approximation, IEEE Trans. on Fuzzy Systems, pp 377-393, 1999

[105] Emami, M.R., Turksen, I.B., Goldenberg, A.A., Development of a systematic methodology of fuzzy logic modeling, IEEE Trans. on Fuzzy Systems, pp346-361, 1998

[106] Ayyub, B.M., M.M.Gupta and L.N. Kanal, Analysis and Management of Uncertainty: Theory and Applications. North-Holland, New York, 1992

[107] G. Dantzig, R. Fulkerson, and S. Johnson, Solution of a large-scale traveling-salesman problem, Operations Research 2, 1954

[108] G. Dantzig and P. Wolfe, Decomposition principle for linear programs, Operations Research 8, 1960

[109] Fiacco and McCormick, Sequential unconstrained minimization techniques, SIAM Books, 1968

[110] Yongshen Ni, John Y. Cheung, Correlation coefficient estimate for fuzzy data, Proceeding of Intelligent Systems Design and Applications, pp 105-114, 2003

[111] Yongshen Ni, John Y. Cheung, Fuzzy correlation coefficient approximate estimation, Proceeding of Artificial Neural Networks in Engineering, pp105-116, 2003

Appendices

A. Linear Programming Introduction

As we have described, fuzzy linear regression model is based on linear programming. A Linear Program (LP) is a problem that can be expressed as following standard form:

$$\text{Minimize} \quad c^t x$$

$$\text{Subject to } Ax = b$$

$$x \geq 0$$

where x is the vector of variables to be solved for, A is a matrix of known coefficients, and c and b are vectors of known coefficients. The matrix A is generally not square, hence you do not solve an LP by just inverting A. Usually A has more columns than rows, and Ax=b is therefore quite likely to be under-determined, leaving great latitude in the choice of x with which to minimize $c^t x$.

The importance of linear programming derives in part from its many applications and in part from the existence of good general-purpose techniques for finding optimal solutions.

Two families of solution techniques are in wide use today. Both visits a progressively improving series of trial solutions, until a solution is reached that satisfies the conditions for an optimum. Simplex methods, introduced by Dantzig [107,108] about 50 years ago, visit "basic" solutions computed by fixing enough of the variables at their bounds to reduce the constraints Ax = b to a square system, which can be solved for unique values of the remaining variables. Basic solutions represent extreme boundary points of the feasible region defined by Ax = b, $x \geq 0$, and the simplex method can be viewed as moving from one such point to another along the edges of the boundary. Barrier or interior-point methods, by contrast, visit points within the interior of the feasible region. These methods derive from techniques for nonlinear programming that were developed and popularized in the 1960s by Fiacco and McCormick [109], but their application to linear programming dates back only to Karmarkar's innovative analysis in 1984.

Simplex-based LP efficiently detects when no feasible solution is possible; some early interior-point codes could not detect an infeasible situation as reliably, but remedies for this flaw have been introduced. The source of infeasibility is often difficult to track down. It may stem from an error in specifying some of the constraints in your model, or from some wrong numbers in your data. A useful approach is to forestall meaningless infeasibilities by explicitly modeling those sources of infeasibility.

The importance of linear programming derives from its many applications and existence of good techniques for finding optimal solutions. Those techniques take as input an LP in the above standard form, and they are fast and reliable over a substantial range of problem sizes and applications.

B. MATLAB Codes

Crisp data, fuzzy correlation

```
close all;
clear all;

% Pdffuzzy.m  is used to compute the fuzzy correlation
% data set 1    small circle
%   x=[6.3 6.5 7.0 7.5  8.0 8.5 9.0  9.5 6.5    7.0    7.5    8.0    8.5    9.0    9.3];
%   y=[6.2  7.18  8.07  8.95  8.62  10.18   10.14   10.24 5.82  5.93  6.04  7.38  6.82  7.86  9];
% big circle
% x=[6.3 6.5 7.0 7.5  8.0 8.5 9.0  9.5 6.5    7.0    7.5    8.0    8.5    9.0    9.3];
%  y=[6.47 9.38 11.38  9.59  12.78   10.89 13.57 12 3.62  2.62  5.41 3.22  8.10 4.7 7.8];
% data set3   ------ quadratic form
%   x2=0.5:0.5:15;
%   x=zeros(1,15);
%   y=zeros(1,15);
%   count=1;
%   for i=1:30
%       y2(i)=-0.5*(x2(i)-8)^2+10+x2(i);
%       if mod(i,2)==0
%           x(count)=x2(i);
%           y(count)=y2(i)+(-1+2*rand)*5;
%           count=count+1;
%       end
%   end
%  y=[ -11.6 -4.1 0.8 5.1 9.6 13.2 15.8   22.4  20.2 14.3  16.5 18.8 5.9   9.6  4.2];
%  figure(1);
%  plot(x,y,x2,y2)

% ------another trend quadratic form
  x2=0.5:0.5:15;
  x=zeros(1,15);
  y=zeros(1,15);
  count=1;
  for i=1:30
      y2(i)=-0.2*x2(i)^2+5*x2(i);
      if mod(i,2)==0
          x(count)=x2(i);
          y(count)=y2(i)+(-1+2*rand)*2;
          count=count+1;
      end
  end
 figure(1);
 plot(x,y,x2,y2)
```

```
% data set 4   -- down big-up small
%   x=[3.7 4.5  6.0  3.5  7.0  7.5   9.0  8.5  5.0  4.1  6.0 6.5 7.0 7.5 8.0];
%   y=[7.2 7.9  8.0  3.4  8.0  8.2   9.0   8.5  2.6  4.1  2.8 3.5 4.7  6.3 7.6];
%  figure(1)
%   plot(x,y,x,x,'r--')

% down small-up big
%  x=[3.5 3.7  4.5  5.0  5.4   6.7  7.5  5.5  2.4  4.5  5.0  5.5  6.0  7    8.3];
%  y=[3.5 4.6 5.8  6.5   7.2   5.7  6.5    8.5  2.4  3.6  3.9  4.0  4.5  4.8  5.2];
%  figure(1)
%  plot(x,y,x,x,'r--')

% test r and spread
%  x2=0.5:0.5:15;
%  y2=x2;
%  x=zeros(1,15);
%  y=zeros(1,15);
%  count=1;
%  sigma=10;
%  for i=1:30
%     if mod(i,2)==0
%         x(count)=x2(i);
%         y(count)=y2(i)+(-1+2*rand)*sigma;
%         count=count+1;
%     end
%  end
%  plot(x,y)

%  x=[9.5677    8.3457    6.5451    4.4774    2.5   0.9549    0.1093    0.1093    0.9549   2.5
4.4774    6.5451    8.3457    9.5677   10];
%  y=[2.0337    3.7157    4.7553    4.9726    4.3301    2.9389    1.0396   -1.0396   -2.9389
-4.3301   -4.9726   -4.7553   -3.7157   -2.0337    0];

peason=xcov(x,y,0,'coeff')
n=length(x);
testnum=n;

total=0;
loopnum=zeros(1,1);
amp=1;
for num=3:n
if num==8 || num==7
     loopnum=6435*amp;
  elseif num==9|| num==6
     loopnum=5005*amp;
```

```
 elseif num==10 || num==5
     loopnum=3003*amp;
 elseif num==11 || num==4
     loopnum=1365*amp;
 elseif num==12 || num==3
     loopnum=455*amp;
 elseif num==13
     loopnum=105*amp;
 elseif num==14
     loopnum=15*amp;
 elseif num==15
     loopnum=1*amp;
  end
rxy=zeros(1,loopnum);

 for i=1:10000
   if i>loopnum
        break;
    end
  count=0;
  index=zeros(1,num);
   for k=1:150
       if num==n
           index=1:n;
           break;
       end
       i1=floor(1+rand*(n-1));
       flag=0;
       for j=1:count
           if i1 == index(j)
               flag=1;
               break;
           end
       end
       if flag==0
           count=count+1;
           index(count)=i1;
       end
       if count==num
           break;
       end
   end
   for k=1:num
       temp=index(k);
       x1(k)=x(temp);
       y1(k)=y(temp);
   end
   rxy(i)=xcov(x1,y1,0,'coeff');
 end
 rxy1(total+1:total+loopnum)=rxy;
 total=total+loopnum;
end

[pe,xout]=hist(rxy1,50);
pe=pe./max(pe);
spread=std(xout)
figure(2)
bar(xout,pe);
title('correlation distribution')
```

Fuzzy Correlation - Approximate Bound

```
close all;
clear all;

%  bounds.m is used to compute the approximate bounds of fuzzy correlation
%input data set
x=[0  0.0714  0.1429  0.2143  0.2857  0.3571  0.4286  0.5000  0.5714  0.6429  0.7143
0.7857    0.8571    0.9286    1.0000];
y =[0.0108 0.1057 0.0249  0.0344  -0.0015  0.1485 0.0474  0.3041 0.4625  0.3255  0.5638
0.7026 0.5442  0.6793 1.0000];
N=length(x);
variation=0.1;
diav= variation*max(abs(y(:)));
xmean=mean(x);
ymean=mean(y);
x1=zeros(N,1);
y1=zeros(N,1);
x1(:)=x(:)-xmean;
y1(:)=y(:)-ymean;


x2sum=0;
y2sum=0;
xysum=0;
for i=1:N
   x2sum=x2sum+x1(i)^2;
   y2sum=y2sum+y1(i)^2;
   xysum=xysum+x1(i)*y1(i);
end

rxy0=xcov(x,y,0,'coeff');

alphanum=11;
rxyb1_b=zeros(alphanum,1);
rxya1_b=zeros(alphanum,1);
alphaa=zeros(alphanum,1);
alpha=0;
for j=1:alphanum
   sx0=0;
   sy0=0;
   for i=1:N
      sx0=sx0+abs(y1(i)*x2sum-x1(i)*xysum)/((x2sum^1.5)*(y2sum^0.5));
      sy0=sy0+abs(x1(i)*y2sum-y1(i)*xysum)/((y2sum^1.5)*(x2sum^0.5));
   end
      % triangle membership function
      rxyb1_b(j)=rxy0+sx0*(1-alpha)*diav+sy0*(1-alpha)*diav;
      rxyb1_b(j)=min(rxyb1_b(j),1);
      rxya1_b(j)=rxy0-sx0*(1-alpha)*diav-sy0*(1-alpha)*diav;
      rxya1_b(j)=max(rxya1_b(j),-1);
       % trapezoid membership function
%
rxyb1_b(j)=rxy0+abs(sx0*(diav-0.75*diav*alpha))+abs(sy0*(diav-0.75*alpha*diav));
%         rxyb1_b(j)=min(rxyb1_b(j),1);
%
rxya1_b(j)=rxy0-abs(sx0*(diav-0.75*alpha*diav))-abs(sy0*(diav-0.75*alpha*diav));
%         rxya1_b(j)=max(rxya1_b(j),-1);

    alpha=j/(alphanum-1);
end

for i=1:alphanum
    alphaa(i)=(i-1)/(alphanum-1);
end

plot(rxya1_b,alphaa,'b-',rxyb1_b,alphaa,'r-')
```

Fuzzy Correlation -Heuristic Method

```
close all;
clear all;

% heuristicmethod.m is used to compute fuzzy correlation with heuristic method
% input data set
x=[0  0.0714  0.1429  0.2143  0.2857  0.3571  0.4286  0.5000  0.5714  0.6429  0.7143
0.7857    0.8571    0.9286    1.0000];
y =[0.0108 0.1057 0.0249  0.0344  -0.0015  0.1485 0.0474  0.3041 0.4625  0.3255  0.5638
0.7026 0.5442  0.6793 1.0000];
N=length(x);
ymax=zeros(1,N);
xmax=zeros(1,N);
ymin=zeros(1,N);
xmin=zeros(1,N);

variation=0.1;
diav= variation*max(abs(y(:)));
gamma=diav;

X=[ones(size(x')) x'];
a=X\y';
N=length(x);
alphanum=11;
% Nonspecificity of original data
nonspecificity0=0;
for alpha=0:1/(alphanum-1):1
   nonspecificity0=nonspecificity0+1/(alphanum-1)*log(1+2*diav*(1-alpha));
end
nonspecificity0

rxy=zeros(2, alphanum);
alpha=0;
tic
for i=1:alphanum
    for k=1:N
  % triangle membership function
     yl=y(k)-(1-alpha)*diav;
     yu=y(k)+(1-alpha)*diav;
     xl=x(k)-(1-alpha)*diav;
     xu=x(k)+(1-alpha)*diav;
     % trapezoidal membership function
%       if(abs(alpha-1)<1e-5)
%          yl=y(k)-0.25*diav;
%          yu=y(k)+0.25*diav;
%          xl=x(k)-0.25*diav;
%          xu=x(k)+0.25*diav;
%       else
%          yl=y(k)-(diav-0.75*alpha*diav);
%          yu=y(k)+(diav-0.75*alpha*diav);
%          xl=x(k)-(diav-0.75*alpha*diav);
%          xu=x(k)+(diav-0.75*alpha*diav);
%       end
        % bell curve membership function
    %   if alpha==0
    %      alpha=0.05;
    %   end
    %   xl=x(k)-diav*sqrt((log(1/alpha))*sqrt(2))/3;
    %   xu=x(k)+diav*sqrt((log(1/alpha))*sqrt(2))/3;
    %   yl=y(k)-diav*sqrt((log(1/alpha))*sqrt(2))/3;
    %   yu=y(k)+diav*sqrt((log(1/alpha))*sqrt(2))/3;
    %%%%%%
        t=a(1)+a(2)*[xl xu];
        tmean=(t(1)+t(2))/2;
        if  t(1)<t(2)
           if t(2)<yl
              xmax(k)=xu;
              ymax(k)=yl;
              xmin(k)=xl;
              ymin(k)=yu;
           elseif t(1)>yu
              xmax(k)=xl;
              ymax(k)=yu;
              xmin(k)=xu;
              ymin(k)=yl;
           elseif t(1)>=yl & t(2)<=yu
              xmax(k)=(xl+xu)/2;
              ymax(k)=tmean;
              if (yl+yu)/2>tmean
                 xmin(k)=xl;
                 ymin(k)=yu;
               else
                 xmin(k)=xu;
                 ymin(k)=yl;
               end
            elseif t(1)>=yl & t(2)>yu
                 xmax(k)=xl;
                 ymax(k)=t(1);
                 xmin(k)=xu;
                 ymin(k)=yl;
            elseif t(2)<=yu & t(1)<yl
```

```
          xmax(k)=xu;
          ymax(k)=t(2);
          xmin(k)=xl;
          ymin(k)=yu;
   elseif t(1)<yl & t(2)>yu
       xmax(k)=(yl+yu-2*a(1))/(2*a(2));
       ymax(k)=(yl+yu)/2;
       if abs(t(1)-yu)>abs(t(2)-yl)
           xmin(k)=xl;
           ymin(k)=yu;
       else
           xmin(k)=xu;
           ymin(k)=yl;
       end
   else
      xmin(k)=x(k);
      ymin(k)=y(k);
      xmax(k)=x(k);
      ymax(k)=y(k);
   end
elseif t(1)>t(2)
  if t(2)>yu
      xmin(k)=xu;
      ymin(k)=yu;
      xmax(k)=xl;
      ymax(k)=yl;
  elseif t(1)<yl
      xmin(k)=xl;
      ymin(k)=yl;
      xmax(k)=xu;
      ymax(k)=yu;
  elseif t(1)<=yu & t(2)>=yl
      xmin(k)=(xl+xu)/2;
      ymin(k)=tmean;
      if (yl+yu)/2>tmean
          xmax(k)=xu;
          ymax(k)=yu;
       else
          xmax(k)=xl;
          ymax(k)=yl;
       end
   elseif t(1)<=yu & t(2)<yl
          xmin(k)=xl;
          ymin(k)=t(1);
          xmax(k)=xu;
          ymax(k)=yu;
```

```
             elseif t(2)>=yl & t(1)>yu
                    xmin(k)=xu;
                    ymin(k)=t(2);
                    xmax(k)=xl;
                    ymax(k)=yl;
             elseif t(1)>yu & t(2)<yl
                    xmin(k)=(yl+yu-2*a(1))/(2*a(2));
                    ymin(k)=(yl+yu)/2;
                    if abs(t(1)-yl)<=abs(t(2)-yu)
                        xmax(k)=xu;
                        ymax(k)=yu;
                    else
                        xmax(k)=xl;
                        ymax(k)=yl;
                    end
             else
                xmin(k)=x(k);
                ymin(k)=y(k);
                xmax(k)=x(k);
                ymax(k)=y(k);
            end
         else
             xmin(k)=x(k);
             ymin(k)=y(k);
             xmax(k)=x(k);
             ymax(k)=y(k);
         end
    end
        rxy(1,i)=xcov(xmax,ymax,0,'coeff');
        rxy(2,i)=xcov(xmin,ymin,0,'coeff');
        xmin=zeros(1,N);
        xmax=zeros(1,N);
        ymin=zeros(1,N);
        ymax=zeros(1,N);
        alpha=i/(alphanum-1);
 end

theuristic=toc
alphaa=zeros(1,alphanum);
rxya=zeros(1, alphanum);
rxyb=zeros(1, alphanum);

alpha=0;
for i=1:alphanum
    alphaa(i)=alpha;
    alpha=alpha+1/(alphanum-1);
```

```
   rxya(i)=min(rxy(:,i));
   rxyb(i)=max(rxy(:,i));
end
%plot(rxya,alphaa,'-b',rxyb,alphaa,'-b')
rxya_h=rxya;
rxyb_h=rxyb;
%xlabel('Rxy')
%ylabel('\alpha')
%Grid;
%title('\alpha Versus Rxy')

%Nonspecificity
h=1;
nonspecificity=0;
for i=1:alphanum
   nonspecificity=nonspecificity+1/(alphanum-1)*log(1+rxyb(i)-rxya(i));
end
nonspecificity_h=nonspecificity/h
%  defuzzification
sum1=0;
sum2=0;
for i=1:alphanum
   sum1=(rxyb(i)-rxya(i))*1/(alphanum-1)+sum1;
end

dr_h=sum1+min(rxya(:))
```

Fuzzy Correlation – Direct Method

```
close all;
clear all;

x=[0  0.0714  0.1429  0.2143  0.2857  0.3571  0.4286  0.5000  0.5714  0.6429  0.7143
0.7857    0.8571    0.9286    1.0000];
y =[0.0108  0.1057  0.0249  0.0344  -0.0015 0.1485  0.0474 0.3041 0.4625  0.3255 0.5638
0.7026 0.5442 0.6793  1.0000];
N=length(x);

variation=0.1;
diav= variation*max(abs(y(:)));
alphanum=11;

% Nonspecificity of original data
nonspecificity0=0;
for alpha=0:1/(alphanum-1):1
   nonspecificity0=nonspecificity0+1/(alphanum-1)*log(1+2*diav*(1-alpha));
end
nonspecificity0

rxya_m=zeros(alphanum,1);
rxyb_m=zeros(alphanum,1);
xl=zeros(N,1);
xu=zeros(N,1);
yl=zeros(N,1);
yu=zeros(N,1);
xx0(1:N)=x;
xx0(N+1:2*N)=y;
alpha=0;
tic
for i=1:alphanum
    % triangle membership function
   xl=x-(1-alpha)*diav;
   xu=x+(1-alpha)*diav;
   yl=y-(1-alpha)*diav;
   yu=y+(1-alpha)*diav;
    % trapezoidal membership function
%    if (abs(alpha-1)<1e-5)
%        yl=y(:)-0.25*diav;
%        yu=y(:)+0.25*diav;
%        xl=x(:)-0.25*diav;
%        xu=x(:)+0.25*diav;
%    else
%        yl=y(:)-(diav-0.75*alpha*diav);
%         yu=y(:)+(diav-0.75*alpha*diav);
%         xl=x(:)-(diav-0.75*alpha*diav);
%         xu=x(:)+(diav-0.75*alpha*diav);
%     end
   % bell curve membership function
   %if alpha==0
   %    alpha=0.05;
   %end
   %xl=x(:)-diav*sqrt((log(1/alpha))*sqrt(2))/3;
   %xu=x(:)+diav*sqrt((log(1/alpha))*sqrt(2))/3;
   %yl=y(:)-diav*sqrt((log(1/alpha))*sqrt(2))/3;
   %yu=y(:)+diav*sqrt((log(1/alpha))*sqrt(2))/3;
   %%%%%%
   lb(1:N)=xl;
   lb(N+1:2*N)=yl;
   ub(1:N)=xu;
   ub(N+1:2*N)=yu;
   [xx1,rxya_m(i)]=fmincon(@corrfun,xx0,[],[],[],[],lb,ub);
   [xx2,rxyb_m(i)]=fmincon(@maxcorr,xx0,[],[],[],[],lb,ub);
   rxyb_m(i)=-rxyb_m(i);
   alpha=i/(alphanum-1);
end
rxya_m(alphanum)=corrfun(xx0);
rxyb_m(alphanum)=rxya_m(alphanum);
toptimal=toc
alphaa=zeros(1,alphanum);
alpha=0;
for i=1:alphanum
   alphaa(i)=alpha;
   alpha=alpha+1/(alphanum-1);
end

%plot(rxya_m,alphaa,'b',rxyb_m,alphaa,'b')
%xlabel('Rxy')
%ylabel('alpha')
%Grid;
%title('Alpha Versus Rxy')

%Nonspecificity
h=1;
nonspecificity=0;
for i=1:alphanum
   nonspecificity=nonspecificity+1/(alphanum-1)*log(1+rxyb_m(i)-rxya_m(i));
end
nonspecificity_t=nonspecificity/h
% Fuzziness
```

```
fuzziness=0;
alpha=0;
for i=1:alphanum-1

fuzziness=fuzziness+(1-abs(2*alpha-1))*abs(rxya_m(i+1)-rxya_m(i))+(1-abs(2*alpha-1)
)*abs(rxyb_m(i)-rxyb_m(i+1));
   alpha=alpha+1/(alphanum-1);
end
fuzziness
%  defuzzification
sum1=0;
sum2=0;
%sum1=(rxyb_m(1)-rxya_m(1))*0.05;
for i=1:alphanum
   sum1=(rxyb_m(i)-rxya_m(i))*1/(alphanum-1)+sum1;
end
dr_t=sum1+min(rxya_m(:))
```

Fuzzy Correlation – Random Search Method

```
close all;
clear all;
x=[0  0.0714  0.1429  0.2143  0.2857  0.3571  0.4286  0.5000  0.5714  0.6429  0.7143
0.7857  0.8571  0.9286  1.0000];
y =[0.0108 0.1057  0.0249  0.0344  -0.0015  0.1485  0.0474  0.3041  0.4625  0.3255  0.5638
0.7026 0.5442  0.6793 1.0000];
N=length(x);
variation=0.1;
diav= variation*max(abs(y(:)));

lemdanum=200;
alphanum=11;

rxy=zeros(lemdanum, alphanum);
alpha=0;
tic
for i=1:alphanum
   for j=1:lemdanum
       for k=1:N
            lemda1=randn;
            if lemda1<-1
               lemda1=-1;
            elseif lemda1>1
               lemda1=1;
            end

            lemda1=(lemda1+1)/2;
            lemda2=randn;
            if lemda2<-1
               lemda2=-1;
            elseif lemda2>1
               lemda2=1;
            end
            lemda2=(lemda2+1)/2;
            % triangle
           yl=y(k)-(1-alpha)*diav;
           yu=y(k)+(1-alpha)*diav;
           xl=x(k)-(1-alpha)*diav;
           xu=x(k)+(1-alpha)*diav;
          % trapezoidal membership function
%           if(abs(alpha-1)<1e-5)
%              yl=y(k)-0.25*diav;
%              yu=y(k)+0.25*diav;
%              xl=x(k)-0.25*diav;
%          xu=x(k)+0.25*diav;
%        else
%          yl=y(k)-(diav-0.75*alpha*diav);
%          yu=y(k)+(diav-0.75*alpha*diav);
%          xl=x(k)-(diav-0.75*alpha*diav);
%          xu=x(k)+(diav-0.75*alpha*diav);
%        end
      % bell curve membership function
 %       if alpha==0
 %          alpha=0.05;
 %       end
 %       xl=x(k)-diav*sqrt((log(1/alpha))*sqrt(2))/3;
 %       xu=x(k)+diav*sqrt((log(1/alpha))*sqrt(2))/3;
 %       yl=y(k)-diav*sqrt((log(1/alpha))*sqrt(2))/3;
 %       yu=y(k)+diav*sqrt((log(1/alpha))*sqrt(2))/3;
  %%%%%%
      yy(k)=(1-lemda1)*yl+lemda1*yu;
      xx(k)=(1-lemda2)*xl+lemda2*xu;
    end
    rxy(j,i)=xcov(xx,yy,0,'coeff');
  end
   alpha=i/(alphanum-1);
end

trandsearch=toc;
rxya_r=zeros(1, alphanum);
rxyb_r=zeros(1, alphanum);
alphaa=zeros(1,alphanum);
alpha=0;
for i=1:alphanum
  rxya_r(i)=min(rxy(:,i));
  rxyb_r(i)=max(rxy(:,i));
  alphaa(i)=alpha;
  alpha=alpha+0.1;
end
alpha_plot(1:11)=alphaa;
alpha_plot(12:22)=alphaa;
rxy_plot(1:11)=rxya_r;
rxy_plot(12:22)=rxyb_r;
%plot(rxy_plot(1:11),alpha_plot(1:11))
%hold on
%plot(rxy_plot(12:22),alpha_plot(12:22))
%xlabel('Rxy')
%ylabel('\alpha')
%Grid;
%title('\alpha Versus Rxy')
```

```
%hold off

%Nonspecificity
h=1;
nonspecificity=0;
for i=1:10
   nonspecificity=nonspecificity+0.1*log(1+rxyb_r(i)-rxya_r(i));
end
nonspecificity=nonspecificity/h

%  defuzzification
sum1=0;
sum2=0;
sum1=(rxyb_r(1)-rxya_r(1))*0.05;
for i=2:alphanum
   sum1=(rxyb_r(i)-rxya_r(i))*0.1+sum1;
end

dr=sum1/0.95+min(rxya_r(:))
```

Plot

```
xplot1=[ 0  0.1  0.2  0.3  0.4  0.5    0.6    0.7    0.8    0.9    1.0];
xplot2=[1 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0];
alphaa=[xplot1 xplot2];
len=length(rxyb_m);
temp_m=rxyb_m;
temp_h=rxyb_h;
temp_b=rxyb1_b;
temp_r=rxyb_r;
for i=1:len
   rxyb_m(i)=temp_m(len-i+1);
   rxyb_h(i)=temp_h(len-i+1);
   rxyb1_b(i)=temp_b(len-i+1);
   rxyb_r(i)=temp_r(len-i+1);
end

rxy_m=[rxya_m' rxyb_m'];
rxy_h=[rxya_h rxyb_h];
rxy_b=[rxya1_b' rxyb1_b'];
rxy_r=[rxya_r  rxyb_r];
plot(rxy_m,alphaa,'-.b',rxy_h,alphaa,'-.r*',rxy_r,alphaa,'g-',rxy_b,alphaa,'--yo')
title('\alpha Versus Rxy')
h=legend('Direct method','Heuristic method','Random search','Approximate bound',2);
```

Regression Application – Dosage Estimation

```
close all;
clear all;

Q=imread('Aug20','bmp');
[m,n]=size(Q);
y=[205.1317  181.6161  114.7314   88.8306   68.9454   52.8702   49.5915  45.3762
39.8489];
y_low=[201.4537  178.2575  111.9650   85.5300   66.2175   48.7200   46.7625  42.0225
36.3533];
y_high=[208.7263  185.0025  117.4650   91.9375   71.7325   57.0000   52.4600  48.7300
43.2400];
%y=[v1 v2 v3 v4 v5 v6 v7 v8];
%y_low=[v1_low v2_low v3_low v4_low v5_low v6_low v7_low v8_low];
%y_high=[v1_high v2_high v3_high v4_high v5_high v6_high v7_high v8_high];
% preprocessing
y_low=log10(y_low);
y_high=log10(y_high);
y=log10(y);
x=[0 0.1543 0.9651 1.9242 2.8879 4.7537 6.1744 7.6131 13.88];
p=4;
datalen=length(x);
c=polyfit(x,y,p);
ypredict=polyval(c,x);

% fuzzy curve fitting
 yp_low=zeros(datalen,1);
 yp_high=zeros(datalen,1);

  deltal=(y-y_low);
  deltar=(y_high-y);
   h=0;
   A=zeros(2*datalen,4);
   b=zeros(2*datalen,1);
   m=1;
   for i=1:datalen
       A(m,1)=-(1-h);
       A(m,2)=-(1-h)*x(i);
       A(m+1,3)=-(1-h);
       A(m+1,4)=-(1-h)*x(i);
       b(m)=-(1-h)*deltal(i)+y(i)-ypredict(i);
       b(m+1)=-(1-h)*deltar(i)+ypredict(i)-y(i);
       m=m+2;
  end
    lb=0.001;    ub=5;
    initial=zeros(4,1);
    for i=1:4
       initial(i)=lb*rand;
    end
    s=fmincon(@newfun, initial,A,b,[],[],lb,ub);

   for i=1:datalen
       yp_low(i)=ypredict(i)-s(1)-x(i)*s(2);
       yp_high(i)=ypredict(i)+s(3)+x(i)*s(4);
    end
    temp=0;
    for i=1:datalen
       if yp_low(i)>yp_high(i)
          temp=yp_low(i);
          yp_low(i)=yp_high(i);
          yp_high(i)=temp;
       end
    end
    y_low=10.^(y_low);
    y_high=10.^(y_high);
    y=10.^(y);
    yp_low=10.^(yp_low);
    yp_high=10.^(yp_high);
    ypredict=10.^(ypredict);
figure(1);
%plot(x,y_low,'g-',x,y_high,'r')
%h=legend('Measured low bound','Measured upper bound',2);

plot(x,y,'g+',x,ypredict,'r--')
title('Intensity versus dose')
xlabel('dose')
ylabel('intensity')
h=legend('Expected value','Predicted value',2);
 figure(2)
plot(x,y_low,'y-',x,y_high,'g-',x,yp_low,'r--',x,yp_high,'b--')
title('Intensity versus dose')
xlabel('dose')
ylabel('intensity')
h=legend('Measured lower bound','Measured upper bound','Predicted lower
bound','Predicted up bound',2);

minvalue=floor(y_low(datalen));
maxvalue=floor(y_high(1));
dose_low=zeros(maxvalue,1);
dose_high=zeros(maxvalue,1);
dose=zeros(maxvalue,1);
```

```
for i=minvalue:maxvalue
   dose_low(i)=interp1(yp_low,x,i,'spline','extrap'); % calculate dose of pixel_value
belongs to [37,255]
   if dose_low(i)<0
      dose_low(i)=0;
   elseif dose_low(i)>x(datalen)
      dose_low(i)=x(datalen);
   end
   dose_high(i)=interp1(yp_high,x,i,'spline','extrap'); % calculate dose of pixel_value
belongs to [37,255]

   if dose_high(i)<0
      dose_high(i)=0;
   elseif dose_high(i)>x(datalen)
      dose_high(i)=x(datalen);
   end
  dose(i)=interp1(y,x,i,'spline','extrap'); % calculate dose of pixel_value belongs to
[37,255]
   if dose(i)<0
      dose(i)=0;
   elseif dose(i)>x(datalen)
      dose(i)=x(datalen);
   end
end

sum1_low=0;
sum1_high=0;
sum1=0;
for i=100:380
   for j=100:430
        if Q(i,j)>=minvalue&Q(i,j)<=maxvalue
           sum1_low=sum1_low+dose_low(Q(i,j));
            sum1_high=sum1_high+dose_high(Q(i,j));
             sum1=sum1+dose(Q(i,j));
         elseif Q(i,j)<minvalue & Q(i,j)>30
          sum1_low=sum1_low+dose_low(minvalue);
          sum1_high=sum1_high+dose_high(minvalue);
           sum1=sum1+dose(minvalue);
       end
   end
end

sum2_low=sum1_low*0.00114
sum2_high=sum1_high*0.00114
sum2=sum1*0.00114
finaldose=(sum2+sum2_low+sum2_high)/3
```

Fuzzy Regression Application - Stock Price Prediction

```
close all;
clear all;
load INTC_1.txt;
s=INTC_1;
y=s(:,3:6);
closeprice=y(:,4);
highprice=y(:,2);
lowprice=y(:,3);
M=length(closeprice);
x=1:M;
plot(x,closeprice,'r-*',x,highprice,'b--',x,lowprice,'g:')
xlabel('Time')
ylabel('Closing Price')
title('INTC Weekly Price Report(Aug/2003)')
axis([1 10 0 35])
h=legend('closing price','high price','low price',2);

x0=[0 0 0 0];
m=6;
A=zeros(2*m,4);
b=zeros(2*m,1);
for i=4:9
   A(i-3,1)=lowprice(i-3);
   A(i-3,2)=lowprice(i-2);
   A(i-3,3)=lowprice(i-1);
   A(i-3,4)=1;
   b(i-3)=closeprice(i);
end
for i=4:9
   A(i+3,1)=-highprice(i-3);
   A(i+3,2)=-highprice(i-2);
   A(i+3,3)=-highprice(i-1);
   A(i+3,4)=-1;
   b(i+3)=-closeprice(i);
end
a=fmincon(@myfun,x0,A,b,[],[],0.1,10)
% prediction
close_pre=a(1)*closeprice(7)+a(2)*closeprice(8)+a(3)*closeprice(9)+a(4)
low_pre=a(1)*lowprice(7)+a(2)*lowprice(8)+a(3)*lowprice(9)+a(4)
high_pre=a(1)*highprice(7)+a(2)*highprice(8)+a(3)*highprice(9)+a(4)

function f=myfun(a, highprice, lowprice)
f=abs(a(1)*(highprice(1)-lowprice(1))+a(2)*(highprice(2)-lowprice(2))+a(3)*(highprice(
3)-lowprice(3)));
```

Crisp Input, Fuzzy Parameter, Fuzzy Output Linear Regression

```
Close all;
Clear all;
x=[1 2 3 4 5];
y=[6.2 8.0  9.5 11.5 13.0];
y_low=[5.9 7.7  9.2 11.2 12.7];
y_high=[6.5 8.3 9.8 11.8 13.3];
n=length(x);
x1=[ones(n,1)  x'];
[c,cint,r,rint,stats] = regress(y',x1);

% fuzzy regression
% con: condition
A=zeros(2*n,2);
h=0;
b=zeros(2*n,1);
for i=1:n
   b(i)=y_low(i)-c(1)-c(2)*x(i);
   b(2*i)=-y_high(i)+c(1)+c(2)*x(i);
   A(i,1)=-(1-h);
   A(i,2)=-(1-h)*x(i);
   A(2*i,1)=-(1-h);
   A(2*i,2)=-(1-h)*x(i);
end

lb=0;
ub=2;
sumx=sum(x);
f=[n  sumx];
xx= linprog(f,A,b,[],[],lb,ub);
plot(x,y,x,y_low,'r--',x,y_high,'r--',
x,c(1)-xx(1)+(c(2)-xx(2))*x,'g--',x,c(1)+xx(1)+(c(2)+xx(2))*x,'g--');
title('crisp input-fuzzy output, fuzzy parameter')
%h=legend('central value','observed low bound','observed upper bound','predicted low
bound','predicted upper bound',3);
```

Fuzzy Input, Crisp Parameter, Crisp Output Linear Regression

```
Close all;
Clear all;

x=[1 2.2 3.1 4.5 6 7.6 ];
x_low=[0.8 2 2.9 4.4 5.7  7.3 ];
x_high=[1.2 2.4 3.3 4.6 6.3 7.9 ];

y=[1 2 3 4 5 6];
n=length(x);
x1=[ones(n,1)  x'];
c = regress(y',x1);
A=zeros(2*n,2);
h=0;
b=zeros(2*n,1);
for i=1:n
   b(i)=-y(i);
   b(2*i)=y(i);
   A(i,1)=-1;
   A(i,2)=-x(i)-(1-h)*(x(i)-x_low(i));
   A(2*i,1)=1;
   A(2*i,2)=x(i)-(1-h)*(x(i)-x_low(i));
end
lb=0;
ub=2;
deltax=x-x_low;
sumx=sum(x);
sumx1=sum(deltax);
f=[1  2*sumx1];
xx= linprog(f,A,b,[],[],lb,ub);
plot(x_low,y,'y',x_high,y,'b',x,c(1)+c(2)*x,'r--',x,xx(1)+xx(2)*x,'g--');
title('fuzzy input, crisp output, crisp parameter')
```

Fuzzy Input, Fuzzy Parameter, Crisp Output Linear Regression

```
Close all;
Clear all;
x=[1 2.2 3.1 4.5 6 7.6 ];
x_low=[0.8 2 2.9 4.4 5.7  7.3 ];
x_high=[1.2 2.4 3.3 4.6 6.3 7.9 ];
s=x-x_low;

y=[1 2 3 4 5 6];
n=length(x);
x1=[ones(n,1)  x'];
m = regress(y',x1);
A=zeros(2*n,2);
h=0.2;
b=zeros(2*n,1);
for i=1:n
   b(i)=-y(i)+m(1)+m(2)*x(i)+(1-h)*m(2)*s(i);
   b(2*i)=y(i)-m(1)-m(2)*x(i)+(1-h)*m(2)*s(i);
   A(i,1)=-(1-h);
   A(i,2)=-(1-h)*x_high(i);
   A(2*i,1)=-(1-h);
   A(2*i,2)=-(1-h)*x_low(i);
end

lb=0;
ub=2;
sumx=sum(x);
f=[2*n  2*sumx];f
xx= linprog(f,A,b,[],[],lb,ub);
plot(x_low,y,'y',x_high,y,'b',x,m(1)-xx(1)+(m(2)-xx(2))*x_low,'g--',x,m(1)+xx(1)+(m(2)
+xx(2))*x_high,'g--');
title('fuzzy input, crisp output, fuzzy parameter')
```

Fuzzy Input, Fuzzy Parameter, Fuzzy Output Linear Regression

```
Close all;
Clear all;
%
x_low= [1.5 3   4.5 6.5 8.0 9.5 10.5  12.0];
x=     [2.0 3.5 5.5 7.0 8.5 10.5 11.0 12.5];
x_high=[2.5 4.0 6.5 7.5 9.0 11.5 11.5 13.0];
%
y_low= [3.5 5   6.5 6.0 8.0 7.0 10.0 9.0];
y=     [4.0 5.5 7.5 6.5 8.5 8.0 10.5 9.5];
y_high=[4.5 6.0 8.5 7.0 9.0 9.0 11.0 10.0];
%
sx=x_high-x;
sy=y_high-y;
n=length(x);
x1=[ones(n,1)  x'];
m = regress(y',x1);
A=zeros(2*n,2);
h=0.3;
b=zeros(2*n,1);
for i=1:n
   b(i)=-y(i)+m(1)+m(2)*x(i)+(1-h)*m(2)*sx(i)-(1-h)*sy(i);
   b(2*i)=y(i)-m(1)-m(2)*x(i)+(1-h)*m(2)*sx(i)-(1-h)*sy(i);
   A(i,1)=-(1-h);
   A(i,2)=-(1-h)*x_high(i);
   A(2*i,1)=-(1-h);
   A(2*i,2)=-(1-h)*x_low(i);
end

lb=0;
ub=2;
sumx=sum(x);
f=[2*n  2*sumx];
xx= linprog(f,A,b,[],[],lb,ub);
plot(x_low,y_low,'y',x_high,y_high,'b',x,m(1)-xx(1)+(m(2)-xx(2))*x_low,'g--',x,m(1)+xx
(1)+(m(2)+xx(2))*x_high,'g--');
title('fuzzy input, fuzzy output, fuzzy parameter')
```

Crisp Input, Fuzzy Parameter, Crisp Output Linear Regression

```
Close all;
Clear all;
x=[1 2 3 4 5 6 7 8 9 10];
y=[1.5 2.3 2.7 4.4 9.4 6.3 6.5 7.8 8.5 10.5];
n=length(x);
x1=[ones(n,1)  x'];
[b,bint,r,rint,stats] = regress(y',x1);

% fuzzy regression
% con: condition
A=zeros(2*n,2);
h=0.4;
con=zeros(2*n,1);
for i=1:n
   con(i)=-y(i)+b(1)+b(2)*x(i);
   con(2*i)=y(i)-b(1)-b(2)*x(i);
   A(i,1)=-(1-h);
   A(i,2)=-(1-h)*x(i);
   A(2*i,1)=-(1-h);
   A(2*i,2)=-(1-h)*x(i);
end

lb=0;
ub=2;
sumx=sum(x);
f=[n  sumx];
xx= linprog(f,A,con,[],[],lb,ub);
plot(x,y,x,b(1)+b(2)*x,'r--',x,b(1)-xx(1)+(b(2)-xx(2))*x,'g--',x,b(1)+xx(1)+(b(2)+xx(2))*
x,'g--');
title('crisp input-output, fuzzy parameter')
```

Fuzzy Input, Crisp Parameter, Fuzzy Output Linear Regression

```
Close all;
Clear all;
x=[2 3.5 5.5 7 8.5 10.5 11 12.5];
x_low=[1.5 3.0 4.5 6.5 8.0 9.5 10.5 12.0];
x_high=[2.5 4.0 6.5 7.5 9.0 11.5 11.5 13.0];
y=[4 5.5 7.5 6.5 8.5 8.0 10.5 9.5];
y_low=[3.5 5.0 6.5 6.0 8.0 7.0 10.0 9.0];
y_high=[4.5 6.0 8.5 7.0 9.0 9.0 11.0 10.0];

n=length(x);
xc=(x_low+x+x_high)/3;
yc=(y_low+y+y_high)/3;
x1=[ones(n,1)  xc'];
[c,cint,r,rint,stats] = regress(yc',x1);
```

Crisp Input, Crisp Parameter, Fuzzy Output Linear Regression

```
close all;
clear all;
%%  pdfregrssion.m is used to compute the fuzzy prediction
x=1:15;
for i=1:15
   y(i)=1.5*x(i)+(-1+2*rand)*2;
end

figure(1)
plot(x,y)
predictvalue=8;
% classical predition
xtest=[ones(15,1) x'];
bb=regress(y',xtest);
prediction=bb(1)+bb(2)*predictvalue
n=length(x);
total=0;
testnum=n;
total=0;
amp=1;
loopnum=zeros(1,1);
for num=3:n
if num==8 || num==7
     loopnum=6435*amp;
  elseif num==9|| num==6
     loopnum=5005*amp;
  elseif num==10 || num==5
     loopnum=3003*amp;
  elseif num==11 || num==4
     loopnum=1365*amp;
  elseif num==12 || num==3
     loopnum=455*amp;
  elseif num==13
     loopnum=105*amp;
  elseif num==14
     loopnum=15*amp;
  elseif num==15
     loopnum=1*amp;
   end

 bcoeff=zeros(2,loopnum);

 for i=1:10000
    if i>loopnum
```

```
         break;
     end
   count=0;
   index=zeros(1,num);
    for k=1:150
        if num==n
            index=1:n;
            break;
        end
        i1=floor(1+rand*(n-1));
        flag=0;
        for j=1:count
            if i1 == index(j)
                flag=1;
                break;
            end
        end
        if flag==0
            count=count+1;
            index(count)=i1;
        end
        if count==num
            break;
        end
    end
    x1=zeros(1,num);
    y1=zeros(1,num);
    for k=1:num
        temp=index(k);
        x1(k)=x(temp);
        y1(k)=y(temp);
    end
    x2=[ones(k,1)  x1'];
    bcoeff(:,i)=regress(y1',x2);
 end
 bcoeff1(:,total+1:total+loopnum)=bcoeff;
 total=total+loopnum;
end

ypredict=bcoeff1(1,:)+bcoeff1(2,:)*predictvalue;
figure(2)
[pe,xout]=hist(ypredict,50);
pe=pe./max(pe);
bar(xout,pe);
% axis([0.96 1 0 1])
title('fuzzy prediction')
```