UNIVERSITY OF SOUTHAMPTON

Imputation by Neural Networks and Related Methods

By

Xinqiang Zhao

Thesis submitted for the degree of Doctor of Philosophy

Department of Social Statistics

Faculty of Social Sciences

July 2002

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF SOCIAL SCIENCES

SOCIAL STATISTICS

Doctor of Philosophy

IMPUTATION BY NEURAL NETWORKS AND RELATED METHODS

By Xinqiang Zhao

Neural network imputation and regression imputation are compared theoretically and numerically. In the theoretical comparison, we introduce the concept of predictive bias (pbias), which is used to measure the difference between the estimator based on full observations and the estimator based on imputed values. Let y be a continuous response variable, and x be the covariate of y, θ be the parameter of interest. The estimator of θ based on the full observations is denoted $\hat{\theta}$. The estimator based on the observed and the imputed values is denoted $\hat{\theta}_I$. Here the imputation is single imputation. The imputed data set has the same size of the full data set. Then pbias is defined as $E(\hat{\theta}_I - \hat{\theta})$. Due to mathematical difficulty, in the theoretical study we only consider the imputation based on the RBF neural network. We show that the performance of an imputation method depends on how the corresponding model fits the underlying model of y. We also show that the RBF model can be equivalent to a regression model in terms of pbias if the RBF model is properly defined and the underlying model is a linear regression model.

A variant of nearest neighbour imputation (NNI) based on weighted distance is also proposed. This method can represent a wide range of NNIs such as Euclidean based NNI and Mahalanobis based NNI. The asymptotic form of this method and the circumstance where it outperforms other imputation methods are investigated.

In the simulation study, we create several situations to compare neural network imputation with regression imputation and other imputation methods such as tree based imputation and NNI. The results show when a competing imputation method outperforms others.

In the numerical study, we use a subset of 1991 household census data to compare the performance of neural network imputation with the performances of logistic regression imputation, nearest neighbour imputation weighted distance-based nearest neighbour imputation and classification tree imputation. We show that the imputation based on MLP neural network outperforms others for some variables such as "Number of Room" variable. The weighted distance-based NNI also performs better than the Euclidean-based NNI.

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Professor Chris Skinner, who introduced me to the subject of imputation and nonresponse. Without his encouragement and enlightenment I could have never finished my study. I have taken up many of his ideas impliedly. His patience and kindness really helped me to start my academic life. I have learnt a lot not only in the subject of my thesis but also the way to do academic research.

I am especially grateful to Professor Ray Chambers, whose encouragement and positive attitude really helped me to dispel the fear of studying neural networks in the early stage. I also have taken some of his ideas in the neural network experiment.

I am also indebted to my wife Wenting, who quitted her accounting job and came to stay with me.

Finally, I would like to thank the financial support from the Office for National Statistics of the United Kingdom and the ORS award from the Committee of the Vice Chancellor and Principals of the Universities of the United Kingdom.

Contents

1 Imputation and Item Nonresponse

1.1 Missing Values and Non-response

The last decades have witnessed the increasing use of surveys and censuses in government and business decisions. The incompleteness of data has become a big concern (Dillman, Eltinge, Groves, Little, Mason, Lesser and Traugott, Smith, 2002). One main source of incompleteness is non-response, which is the failure to obtain complete measurements on the all members of the sample (or population in the case of a census). It is normally divided into unit non-response – the failure of a selected sample member to respond – and item non-response – failure to obtain some of the desired items of information for individual sample members.

Unit non-response occurs mainly because some individuals are unable to be contacted or refuse to participate in an interview. In the case of administrative data whole units may be missing for a variety of reasons. Lessler and Kalsbeek (1992) summarised the various factors that could lead to unit nonresponse and item nonresponse, such as incorrect contact information in the sampling frame and the timing of attempts. Unit non-response is sometimes ignored in analyses, thereby assuming implicitly that results are not biased by differences between the responding and non-responding people, a generally unwarranted assumption. Otherwise it is usually dealt with by weighting to known characteristics.

Item non-response occurs when interviewers fail to ask the question, respondents are unable or unwilling to answer the question, or interviewers fail to record the answer. The incidence of item nonresponse may relate to the sensitivity of questions asked. For example, respondents often decline to answer questions about income and savings. The improper wording of questions may also be a source of item nonresponse such as long questions, which may confuse respondents. The length of the questionnaire may also cause item nonresponse. Long questionnaires may make some respondents feel tired or bored, and refuse to answer the rest of the questions. Item nonresponse may also relate to the method of data collection. Mail surveys and telephone interviews are likely to produce

more item nonresponses than in person interviews (Groves and Kahn, 1979). The personality of the interviewer may also affect item nonresponse. An interviewer who can make respondents feel comfortable and relaxed may get more answers from respondents (Rogers, 1976). Administrative data may suffer from item non-response because the information was not collected or was not recorded. A further source of missing data is the consequences of edit check failures. For example, some items may be found unusable because they are contradicting to other items that are unlikely to be errors. Outliers might also be treated as item nonresponse and replaced by more acceptable values. Item non-response leads to missing values in the data set. Although the missing value problem can be dealt with by post-data-collection treatments such as imputation, it is equally important to put emphasis on the effort of reducing the chance of generating missing values in the data collection and processing process. For example if adequate training is given to all interviewers involved in a survey, it could reduce the amount of missing values caused by the misconduct of interviewers. In the special case of small sample size survey, if the quality of data is very crucial, it might be necessary to validate all data using a different team of interviewers. For most surveys, it is impossible to check all data collected, especially in the census situation. If sensitive questions are inevitable, the randomised response technique can be used to deal with it (Warner, 1965). For some interview surveys, female interviewers are preferred, because they are more likely accepted by respondent (Lessler and Kalsbeek, 1992).

Item nonresponse is very common in census data, because the size of census puts restrictions on every aspect of data collection. It is impossible to ensure all interviewers are well trained. Meanwhile, the data processing work is also sophisticated, it is another source of generating missing values. For example, some of the questions are open-ended, which result in manual coding and data entry. It inevitably generates errors and missing values. Table 1.1 lists the pattern of the missing values of four variables (BLDTYPE, ROOMS, CARS, HHPERCNT) from a subset of 1991 household census data for Great Britain, made available by the Office for National Statistics (ONS). One can find overall nearly 18% (17.67%) of cases are partly or complete missing in the four variables. The variable ROOMS is missing for 12.92% of the total cases. Another interesting phenomenon is that the majority of missing cases in BLDTYPE are also missing in ROOMS, which may have special reasons. If the data are divided into subsets of small areas, the missing value problem may lead to severe difficulties in estimating totals in

some areas. This problem may be addressed by imputing for missing values before estimation starts.

Table 1.1 Pattern of Missing Values in BLDTYPE, ROOMS, CARS and HHPERCNT

BLDTYPE	ROOMS	CARS	HHPERCNT	Percentage	Frequency
				82.33%	282780
X				0.47%	1611
	X			5.35%	18364
		X		4.00%	13732
X	X			4.77%	16396
X		X		0.02%	74
	X	X		0.25%	867
		X	X	0.26%	893
X	X	X		0.28%	968
	X	X	X	1.94%	6673
X	X	X	X	0.33%	1119
5.87%	12.92%	7.08%	2.53%		
20168	44387	24326	8685	100.00%	343477

Note: The above results are based on a subset of 1991 household census data created by ONS. "X" indicates missing. The first row contains no "X" which indicates all complete cases in the four variables. "BLDTYPE" indicates "Building type". "ROOMS" indicates "Number of rooms". "CARS" indicates "number of cars". "HHPERCNT" indicates "Number of persons in the house". The column under "Percentage" gives the percentage of each missing pattern, while the column under "Frequency" gives the frequency of occurrence of each pattern. The percentages at the line above the bottom line are the percentages of missing cases for individual variables. The bottom line gives the total number of missing values for each variable.

A common method of dealing with missing values is to neglect them and to base estimates on complete cases. As aforementioned this is based on the assumption that the complete cases form a representative sample. The validity of this assumption is never guaranteed. Possible biases exist because the respondents are often systematically different from the non-respondents; of particular concern, these biases are difficult to eliminate since the precise reasons for non-response are usually not known (Rubin, 1987). Moreover excluding missing values will result in less efficient estimates because of the reduced size of the data base.

Methods of dealing with item non-response and missing values have a long history. Back in 1937, Bartlett (1937) reported the missing value problem in analysing agriculture data. In his study Bartlett showed how to obtain appropriate analyses of variance in experiments in which a missing value occurs in the dependent or independent variate (or in both). Hansen and Hurwitz (1946) addressed the non-response problem in a mail survey which was popular at that time (Clausen, Ford, 1947, Scott, 1961). For cost concerns, Hansen and Hurwitz (1946) suggested a call-back strategy by which only a fraction of non-respondents are re-interviewed to achieve an unbiased sample design. Clausen and Ford (1947) argue how to maximise response and correct bias incurred from incomplete returns. At this stage much of the interest of research was in how to prevent non-response, although dealing with missing values was also addressed by several authors (Lury, 1946; Anderson, 1946). Gradually more and more people paid attention to analysis in the presence of missing values, especially those who analysed experiment data. Research in this area has been extensive, such as the analysis of contingency table with missing frequencies (Watson, 1956), the analysis of variance with incomplete data (Wilkinson, 1958), the treatment of missing values in discriminant analysis (Chan and Dunn, 1972), handling missing data in regression analysis (Haitovsky, 1968. Little, 1992. Reilly and Pepe, 1995. Skinner, and Coker, 1996) and Bayesian analysis of nonresponse (Kaufman and King, 1973). The great contribution came from Rubin's paper titled *Inference and Missing Data* (Rubin, 1976), in which the missing mechanism was considered, and likelihood-based inference explored. Since then missing data analysis based on likelihood is widely used by practitioners and academic researchers. The EM method is also brought in to deal with missing values based on the likelihood function (David and Skene, 1979). A good reference on likelihood based inference is Little and Rubin's book on statistical analysis with missing data (Little and Rubin, 1986).

An alternative approach to dealing with missing values is imputation. This involves 'filling in' the missing value by an imputed value, determined in some way. Early adoption of imputation can date back to 1950s. For example, Jaszi (1951) discussed the use of imputation for variables such as wages and salaries in calculating national consumption. Phillips (1956) used imputation to deal with the missing values in the component variables of wholesale price indices. Phillips (1956) argued that the consultation with related sources is crucial to obtaining sensible imputation. Clifton and Wharton (1960) used imputation for family labour in calculating the contribution of labour

in their study on undeveloped data from an undeveloped area. The merit of imputation was recognised by more and more authors (Rockwell, 1975), (Fellegi and Holt, 1976), (Little, 1982). The major reason why imputation was used was to achieve completeness of data. Users of the data could make use of standard complete data methods. Imputation was also used in conjunction with data editing (Freund and Hartley, 1967), where imputed values were used to replace errors identified in editing process. Early imputation approaches failed to reflect the uncertainty regarding the missing values. Rubin (1987) proposed the multiple imputation method to handle the uncertainty in imputation. Analysis method based on multiple data sets generated by the multiple imputation was also discussed by Rubin. Other approaches to handle the uncertainty due to imputation were proposed by authors like Rao and Shao (1992), Chen and Shao (2001). They explored the properties of variance of the nearest neighbour imputation methods (hot deck imputation) using Jackknife approach.

In general, imputation is based on two ideas. One is that a case with a missing value may be very close, in terms of covariates, to a neighbouring case with an observed value. This neighbour may be used as a "donor" to impute the missing value using the donor's observed value. Nearest neighbour imputation is an example of this idea. The other idea is that the variable with missing values may have a functional relationship with observed covariates, which can be used to predict the missing values. Regression imputation is one of the several widely used methods based on this idea. Details of imputation methods will be discussed in section 1.4.

1.2 Notation and Basic Models

It is not an easy task to make the notation both consistent with conventions and distinguishable among different situations. On one hand we need to distinguish scalars vectors and matrix, on the other hand conventional notation in statistics is not always consistent with that from the neural network literature. If some unusual symbols are encountered, they are the result of the balance made between the two restrictions. Through out this thesis we use the following notation and symbols:

- We use italic lowercase letters for scalar variables. The values of a scalar variable are denoted by the variable and its subscripts. For example y is a scalar variable, and y_i is the i^{th} value of y.
- Vectors and vector variables are denoted by boldface lowercase letters. The values of a vector variable are denoted by the variable letter and its subscripts. For example, $\mathbf{y}$ is a vector variable, and $\mathbf{y}_i$ is the ith value of $\mathbf{y}$.
- The covariate variable is denoted by $\mathbf{x}$. Without clarification it is assumed to be a vector. The ith value of this covariate vector is $\mathbf{x}_i$.
- Capital letters are used to denote matrices with two exceptions. The first exception is N, which denotes the size of the finite population. The other exception is R_i ,which indicates the response status of i^{th} observation. Meanwhile the italic capital R indicates the vector composed by R_i, namely, $R=(R_1,\ldots,R_n)$, where n is the number of observations.
- The primary model considered is the scalar superpopulation model for the random variable y. The multivariate case is taken as supplementary. The n realisations of the model are assumed independent, and identically distributed (*iid.*). The census situation is taken as a special case of the primary model with $n=N$. The parameters of interest are the mean and variance. We use μ to denote the mean of y, τ to denote the variance of y. For multivariate variable $\mathbf{y}$ we use $\boldsymbol{\mu}$ to denote the mean, $\boldsymbol{\Sigma}$ to denote the variance matrix. If covariates are considered, μ and $\boldsymbol{\mu}$ are assumed to be the functions of the covariates $\mathbf{x}$, and denoted by $\mu(\mathbf{x})$ and $\boldsymbol{\mu}(\mathbf{x})$ respectively.
- We assume the first m observations are observed with $R_1=\ldots=R_m=1$, and denoted by $Y_{obs}=(y_1,\ldots, y_m)$. The observations from y_{m+1} to y_n are assumed missing. $\hat{y}_i$ indicates the imputation for the i^{th} value of y, where $i=m+1,\ldots,n$. The usual estimator of μ based

on the sample if fully observed is denoted by $\hat{\mu} = n^{-1}\sum_{i=1}^{n} y_i$. The estimator of μ based on m observed values and n-m imputed values is denoted by $\hat{\mu}_I = n^{-1}\sum_{i=1}^{m} y_i + n^{-1}\sum_{i=m+1}^{n} \hat{y}_i$. Similarly the estimators of τ based on n observed values of y and the combination of the observed and the imputed values are denoted by $\hat{\tau} = n^{-1}\sum_{i=1}^{n}(y_i - \hat{\mu})^2$ and $\hat{\tau}_1 = n^{-1}\sum_{i=1}^{n}(\tilde{y}_i - \hat{\mu}_I)^2$ respectively, where $\tilde{y}_i = y_i$ for i=1,…,m and $\tilde{y}_i = \hat{y}_i$ for i=m+1,…,n.

1.3 Missing Mechanism

When missing values are encountered in data analysis, some of them may be determined by the values of other variables, which are the covariates. For example, if the age of respondent is under twelve, the number of children must be zero. In many cases there is no deterministic relation between them. Some assumptions about the characteristics of missingness have to be made, either simple and intuitive ones or formal ones like parametric models. Practitioners may prefer ones with simple intuition, such as the assumptions underlying a donor imputation method, because usually they are easy to understand and consistent with common sense. For donor imputation the underlying assumption is that the missing values are close to the observed ones that have close values of covariates. For formal approaches, one can use stochastic models to describe missing mechanism. Both situations are considered in this thesis. We refer the donor like methods as non-parametric approaches, and the regression like methods and distribution based methods as parametric approaches. For convenience, we make the following notation for further discussion.

Let y_i denote the value of variable y for unit i, and suppose that for a sample of size n, $y_1,\ldots, y_m$ are observed and $y_{m+1},\ldots, y_n$ are missing. Then $Y=(Y_{obs}, Y_{mis})^T$, where $Y_{obs}=(y_1,\ldots, y_m)$, $Y_{mis}=(y_{m+1},\ldots, y_n)$. Let x_{ij} be the value of j^{th} covariate for the i^{th} unit, covariate matrix $X=(x_{ij})_{i=1,\ldots,n;\, j=1,\ldots,q}$ is assumed to be complete, where q is the number of covariates. Let R_i indicate the presence of y_i (Lessler and Kalsbeek, 1992).

$$R_i = \begin{cases} 1, & y_i \ \ observed \\ 0, & y_i \ \ mis\sin g \end{cases}, \ i=1,\ldots,n. \qquad (1.3.1)$$

Without loss of generality we assume $R_1=\ldots=R_m=1$, and $R_{1+m}=\ldots=R_n=0$. From the deterministic point of view, $P(R_i=1)$ is either 1 or 0, in the sense that all units sampled are completely determined to respond or not respond under any circumstances. In this extreme situation, any attempts to improve response rate are destined to fail. A more practical assumption is that the willingness of answering questions from a respondent may change depending on the situation of interview. Therefore R_i can be taken as a random

variable. When the stochastic model of R_i is considered, the first assumption to be made is missing mechanism. That is the foundation of setting the relationship between the observed and the non-observed. Missing mechanisms fall in two broad categories, missing at random mechanism (MAR) and missing not at random (MNAR) mechanism. Basically MAR mechanism means the missing mechanism doesn't depend on the value of the variable of interest, y, itself. It may depend on its covariates $\mathbf{x}$. In a special case when the missing mechanism doesn't depend on y and $\mathbf{x}$, it is called missing completely at random (MCAR). Otherwise the missing mechanism is termed as MNAR. We start with MCAR in the next section.

1.3.1 Missing Completely at Random (MCAR)

If the probability of response (R_i) doesn't depend on y_i and its covariate $\mathbf{x}_i$, namely $P(R_i \mid y_i, \mathbf{x}_i, \xi) = P(R_i \mid \xi)$, the observed cases consist of a random sub-sample of the sampled cases. Here ξ is unknown parameter. In this case the missing mechanism is termed as missing completely at random (MCAR). For convenience, let $R=(R_1,\ldots,R_n)$. The property of MCAR can be described as

$$\begin{aligned} & P(y_i \mid R_i = 0, Y_{obs}, X, \xi) \\ & = P(y_i \mid Y_{obs}, X, \xi) \\ & = P(y_i \mid R_i = 1, Y_{obs}, X, \xi). \end{aligned} \qquad (1.3.2)$$

Therefore the imputation for missing values can be based on the model with parameters estimated using the complete cases. In this special case the unbiased estimators of model parameters may still be unbiased, but the variances are likely to be increased. That is the loss resulted from missing values.

1.3.2 Missing at Random (MAR)

The mechanism is MAR if the probability that an observation is missing doesn't depend on Y_{mis} but on Y_{obs} and/or its covariates. This mechanism can be described by

$P(R \mid Y, X, \xi) = P(R \mid Y_{obs}, X, \xi)$. The name 'MAR' doesn't mean that the missing values are a random sample of all data values. For example, the missing values may be generated according to the values of a covariate. If the missing values do constitute a random sample then the mechanism is missing completely at random (MCAR). MCAR is a special case of MAR. Suppose **x** is the covariate of y, and is completely observed. Then, MAR allows the probability that a datum is missing to depend on the datum itself, but only indirectly through quantities that are observed.

1.4 Imputation Methods

Imputation is a technique used to address the problem of item nonresponse by replacing the missing values by proxy values. The aims of imputation include:

- Reduce non-response bias. Imputation attempts to reduce bias based on assumptions, which specify the missing mechanisms and the relationships between the response and non-response.
- Minor missingness of individual variables can cause heavy aggregate missing problem. When analysis involves multiple variables, the aggregate missingness in these variables may leads to unstable result. In the extreme situation the number of parameters of interest may exceed the number of the number of cases available for analysis. By filling the holes, imputation makes multivariate analysis stable and even possible in some situations.
- Provide suitable data sources for third parties to use.

There are a large number of methods of imputation available which are appropriate in different circumstances. In general, imputation is carried out by assigning a value to a missing item based on some similarity measurement in terms of covariates. One dimension to classify imputation methods is based on whether they depend on some model assumptions. If the imputation method is based on a specific model, it is parametric method. Otherwise if the imputation method is not based on a specific model or distribution, it is non-parametric. It is not an easy task to clearly draw a line between the two, since some models may contain the characteristics of both, and sometimes termed

semi-parametric method. For simplicity we classify imputation methods by whether they are based on a parametric model. There are pros and cons about parametric imputation and non-parametric imputation. Either one could outperform the other in some circumstances. But in general, parametric methods are less time-consuming in terms of computing effort. Compared to parametric models, non-parametric methods normally are computer intensive, either in searching for candidate donors like in donor imputation or for the training model such as in neural network imputations, although some non-parametric methods can be optimised by using better algorithms. The limitation of parametric imputations is in the validity of the models employed.

Another way to classify imputation methods is between single and multiple imputation methods. Single imputation involves assigning a single value to each of the missing values under one or more assumptions. The completed data set is the original one with holes filled. Unlike single imputation, multiple imputation imputes each of the missing values by two or more values to reflect the uncertainty about the missing values (Rubin, 1987). Multiple imputation produces more than one dataset. The data analysis should combine the results from each component data set. All methods are based on some assumptions about the missing mechanism. Without the assumptions, the validity of an imputation cannot be justified. Multiple imputation has the potential of enabling certain kinds of statistical inference, but it increases the complexity of analysis. Therefore single imputation probably remains the most widely used approach.

Some of the widely used single imputation methods are:

a) Deductive methods
b) Imputing mean (continuous) or mode (categorical)
c) Random draw from marginal distribution
d) Sequential hot-deck method
e) Hierarchical hot-deck methods
f) Predictive linear regression imputation
g) Nearest neighbour imputation (NNI)

Before making a comparison, a brief description of these methods is given below.

a) Deductive Methods

Missing values are deduced with near certainty from combinations of non-missing items from the same case. The deductions used will depend on the pool of knowledge of the data set. Only a deterministic missing mechanism is needed. The imputation is based on the deterministic relation of the variable containing missing values and one of the complete variables. For example, if the age of a respondent is under seventeen, the value of the variable "current driving licence" must be "NO".

b) Imputing Mean or Mode

For a continuous variable with a missing value, either the overall mean or class mean calculated from respondent data is chosen for imputation (Lessler and Kalsbeek, 1992, p. 220). Here the class is normally defined by the categorical covariate of the continuous variable. The class could be an individual category, or the combination of several categories depending on some practical concerns. Simply put, the original data set is divided into small sub groups called classes defined by covariates. For a categorical variable a missing value can be replaced by the mode. The simplicity of mean imputation makes it widely used by practitioners. It may be changed when the more sophisticated imputation approaches are implemented in popular statistical software.

c) Random Draw from Marginal Distribution

With this method, a missing value is imputed by a random draw from the respondent values or its marginal distribution. The former is actually a donor imputation (Lessler and Kalsbeek, 1992, p.213). Donor imputation in general is a method of selecting a case (a donor) randomly or by the distances between the covariates of the missing value and the covariates of members within the class. The cases containing missing values are also allocated to the classes they belong to. The donor imputation is then carried out by randomly selecting values from the respondent values with in the class or by finding the case within the class that is least distant in terms of some distance measurement.

d) Sequential Hot-deck Method

This method initially classifies nonrespondents into several groups, called classes, based on the combination of covariates (David, Little, Samuhel, Triest, 1986). Prior to processing, an imputation value is assigned to each class, possibly at random, or from a file relating to a previous survey period or a different area. Each case is processed sequentially. If the case has a missing item, it is replaced by the imputation value from the relevant class. If the item is not missing, it replaces the stored imputation value for its class, and can be used for imputation of subsequent missing items. Cases are often held in geographical order, and, as the donor is selected from the most recently processed valid value, this introduces implicit geographical effects as an additional matching variable. This imputation assumes the homogeneity of the data used as donor and the data to be imputed.

e) Hierarchical Hot-deck Methods

Similar to sequential hot deck method, the data file is sorted into a much larger number of imputation classes in a hierarchical structure (Lessler and Kalsbeek, 1992, page 213). It is possible to include more auxiliary variables and to have a greater number of imputation classes, if no suitable donor is found at the finest level of the classification, classes are collapsed into broader groups until a donor is found. A pattern of 'hard' and 'soft' class boundaries can be programmed into hierarchical structure, e.g. to ensure that an item is always imputed from a donor of the same group, even though the area of residence classes may be collapsed. The method is based on the assumption that the missing mechanism in each sub-class is completely at random. Therefore the imputation is just a random selection of the available values.

f) Predictive Linear Regression Imputation

This method is to impute the missing value by a prediction from the linear model built on the respondent data. The variable with missing values is regarded as the dependent

variable; and the covariates are the independent variables. Here MAR is also implicitly assumed, otherwise the prediction may leads to biased estimates.

One of the predictive linear regression imputation methods is to impute the missing values by their predictive means (regression means) (Lessler and Kalsbeek, 1992, page 220). Assume the model is

$$y_i = x_i\beta + \varepsilon_i,\ i=1,\ldots,n$$

where $\varepsilon_i \sim N(0, \sigma^2)$. The imputation of y_i $(i=m+1,\ldots,n)$ is

$$\hat{y}_i = \mathbf{x}_i\hat{\beta},\ i=m+1,\ldots,n, \qquad (1.4.1)$$

where $\hat{\beta}$ is the estimator of β based on the respondent data. This method has a tendency to deflate the variance of y. To preserve the variance, a correction term, which is the estimator of the residual term ε_i, is added to the regression mean. This is termed random regression imputation.

$$\hat{y}_i = \mathbf{x}_i\hat{\beta} + \hat{\varepsilon}_i,\ i=m+1,\ldots,n. \qquad (1.4.2)$$

where $\hat{\varepsilon}_i$ is the estimator of the residual of y_i. The residual term can be obtained from a random draw from the residuals of complete cases or the distribution of the residual term. The former is more realistic, because it is a direct result from the complete cases rather than derived from the estimated distribution of the residual.

g) Nearest Neighbour Imputation (NNI)

Instead of imputing missing values based on an assumed explicit relationship between y and its covariates, nearest neighbour imputation (NNI) imputes the missing value by the corresponding value of its nearest responding neighbour, where the closeness is measured

by a distance function of covariates (Lessler and Kalsbeek, 1992, p. 218). This method also assumes all covariates are fully observed.

The performance of NNI is determined by the definition of distance. The measurement of distance between two units (or observations) depends on the nature of the variables taken into account. We start with one covariate x. Let x_i and x_j be the values of x for two units i and j. The distance between the two units is denoted $d(x_i, x_j)$. For scalar continuous x, one natural option is Euclidean distance, which is a special case of Minkowski distance.

$$d(x_i, x_j) = |x_i - x_j|^r, \; r>0, \; i \neq j, \; i=1 \ldots n, j=1 \ldots n. \tag{1.4.3}$$

The case r=1 is called the $\boldsymbol{L}_1$ distance and is more robust to outliers than the Euclidean $\boldsymbol{L}_2$ distance with r=2. For nominal variable, the following distance is suggested.

$$d(x_i, x_j) = 1 - \delta_{(x_i = x_j)}, \tag{1.4.4}$$

where $\delta_{(x_i=x_j)} = \begin{cases} 1, x_i = x_j \\ 0, x_i \neq x_j \end{cases}$. For an ordinal variable taking integer values the absolute value of the difference could be adopted.

$$d(x_i, x_j) = |x_i - x_j|, \tag{1.4.5}$$

An extension of NNI is developed in chapter 3 based on the idea of assigning a weight to the component of the distance for each covariate, under the consideration that covariates may give different contributions to the overall distance.

1.5 General Parametric Approach to Imputation

As explained in section 1.4, single imputation methods can be put into two categories, parametric and non-parametric. The donor-based imputation methods such as hot-deck imputation and nearest neighbour imputation are non-parametric. The model-based

imputation methods such as linear regression imputation and random regression imputation are parametric. Here we concentrate on parametric imputation methods, while in chapter 3 we give an extension to the distance based imputation method. Some assumptions have to be made. The first and probably the most important assumption is missing at random (MAR). We denote the probability distribution of y_i given $\mathbf{x}_i$ and R_i by $P(y_i \mid \mathbf{x}_i, R_i, \theta)$, where θ is the parameter specifying the distribution. With MAR assumption, $P(y_i \mid \mathbf{x}_i, R_i = 0, \theta)$ has the following property

$$\begin{aligned} & P(y_i \mid \mathbf{x}_i, R_i = 0, \theta) \\ & = P(y_i \mid \mathbf{x}_i, R_i = 1, \theta) \\ & = P(y_i \mid \mathbf{x}_i, \theta). \end{aligned} \tag{1.5.1}$$

The model $P(y_i \mid \mathbf{x}_i, R_i = 0, \theta)$ can be used for imputation for y_i, i=m+1,...,n, where θ is estimated using $P(y_i \mid \mathbf{x}_i, R_i = 1, \theta)$, i=1,...,m. In other words the estimate of θ based on the observed data can be used as the basis for imputation. Assuming $(y_i, \mathbf{x}_i, R_i)$, i=1,...,m are independently identically distributed (iid), the likelihood of the observed data can be then denoted as

$$L(Y_{obs} \mid X, \theta) = \prod_{i=1}^{m} P(y_i \mid \mathbf{x}_i, \theta).$$

For continuous variable y, the imputation based on this assumption follows this procedure:

- Estimate model parameter θ. From the frequentist perspective, model parameter θ is estimated by MLE $\hat{\theta}$. From Bayesian perspective, the posterior of θ given a prior is obtained, and $\hat{\theta}$ might be the mode of this posterior distribution.
- Based on the conclusion of (1.5.1), the imputation is obtained from the distribution specified by $\hat{\theta}$, namely $P(y_i \mid \mathbf{x}_i, R_i = 0, \hat{\theta})$.

The performance of an imputation method in the parametric category relies on the validity of model assumptions. Normally the best performance can only be achieved when the

assumption is justified. Therefore it is necessary to test the validity of model assumptions before applying it to a data set.

1.5.1 Linear Regression

Let's revisit the random regression imputation method in previous section. Under MAR assumption $\hat{\theta}=(\hat{\beta},\hat{\sigma}^2)$ are obtained from the likelihood function based on $Y_{obs}=(y_1,\ldots,y_m)^T$. The imputations of $Y_{mis}=(y_{m+1},\ldots,y_n)^T$ are

$$\hat{y}_i = \mathbf{x}_i\hat{\beta} + \hat{\varepsilon}_i,\ i=m+1\ldots n, \qquad (1.5.2)$$

where $\hat{\varepsilon}_i$ may be defined in different ways. One can obtain $\hat{\varepsilon}_i$s without estimating σ^2. One way is randomly drawing a sample from the residuals of the complete cases. The other way is to choose the donor that has the nearest predictive value $\mathbf{x}_i\hat{\beta}$ (Lessler and Kalsbeek, 1992, p. 221).

1.5.2 Logistic Regression Imputation

If y_i is a categorical variable with probability function $P(y_i|\mathrm{x}_i, \theta)$, the estimator of θ may be obtained by MLE from observed data as $\hat{\theta}$. The imputation is produced from the estimated distribution $P(y_i \mid \mathbf{x}_i, \hat{\theta})$. One possible approach is to choose the category with highest probability

$$\hat{y}_i = \arg\max_k P(y_i = k \mid \mathbf{x}_i, \hat{\theta}),\ i=m+1\ldots n. \qquad (1.5.3)$$

An other possible method is randomly drawing a value from the distribution. In a special case there is no covariate associated with y, the imputation could just be a random draw from $P(\hat{\boldsymbol{\theta}}\,|Y_{\mathrm{obs}})$, where $\theta=(\theta_j)_{j=1,\ldots,p+1}$, and $\theta_j=P(y=j)$. When y has covariate $\mathbf{x}$, a logistic model or multinomial model can be employed to model the probability of membership in

terms of **x**. Here the assumption is that the relationship between the logit of the probability and variate **x** is linear.

In the special case when y is a binary variable, logistic regression can be used to model the dependence of y on **x**. The relationship between y and **x** is as follows

$$\log it(\pi_{i1}) = \log(\frac{\pi_{i1}}{1-\pi_{i1}}) = \mathbf{x}_i\boldsymbol{\beta}, \; i=1...n, \tag{1.5.4}$$

where $\pi_{i1} = P(y_i=1|\mathrm{x}_i)$ (Agresti, 1990). The estimate of π_{i1} is given by

$$\hat{\pi}_{i1} = \frac{\exp(\mathbf{x}_i\hat{\boldsymbol{\beta}})}{1+\exp(\mathbf{x}_i\hat{\boldsymbol{\beta}})}, \; i=1...n. \tag{1.5.5}$$

Under the MAR assumption $\hat{\boldsymbol{\beta}}$, which is obtained based on the observed data, can be used to construct the posterior probability function of missing values. The imputation is then carried on based on the posterior distribution either by a random draw or the category with the highest probability.

1.5.3 Multinomial Logit Regression

When y has three or more levels, a multinomial logit model can be used to form imputations. The strategy is same as that in (1.5.3) with the probability of membership assumed to be the exponential of a linear function of covariate **x**. Suppose there are $p+1$ categories, and the $p+1^{\text{th}}$ level is selected as base level, the logit regression equation can be expressed as

$$\log(\frac{\pi_{ik}}{\pi_{i,p+1}}) = \mathbf{x}_i\beta_k, \; i=1,...,n, \; k=1,...,p+1, \tag{1.5.6}$$

where $\pi_{ik} = P(y_i=k|\mathrm{x}_i)$. It follows that

$$\log(\frac{\pi_{ik}}{\pi_{ij}}) = \mathbf{x}_i(\beta_{ik} - \beta_j),\ i=1,...,n,\ k,j=1,...,p+1. \tag{1.5.7}$$

With the constraint $\Sigma\pi_{ik}=1$, the estimator of π_{ik} is given by

$$\hat{\pi}_{ik} = \frac{\exp(\mathbf{x}_i\hat{\beta}_k)}{1+\sum_{j=1}^{p}\exp(\mathbf{x}_i\hat{\beta}_j)},\ i=1...n,\ k=1...p. \tag{1.5.8}$$

$$\hat{\pi}_{i,p+1} = \frac{1}{1+\sum_{j=1}^{p}\exp(\mathbf{x}_i\hat{\beta}_j)}, \tag{1.5.9}$$

where $\hat{\beta}_j$ is a MLE.

As in logistic regression, $\hat{\pi}_j s, j=1,p$ are used to predict the posterior probability of each missing value. The imputation could be either the category with highest probability or a random draw from the distribution.

For an ordinal variable y with a unit increment between two adjacent categories, the design matrix may be adjusted. Suppose the ordinal categories follow this pattern, $1<2<...<p+1$. The model may be adjusted by taking in to account of the order (Agresti, 1990). For example it might assumed that

$$\log(\frac{\pi_{ik}}{\pi_{i,p+1}}) = (p+1-k)\mathbf{x}_i\beta_k,\ i=1...n,\ k=1...p. \tag{1.5.10}$$

1.6 Imputing Categorical Data by Classification Methods

When the variable containing missing values is nominal, the imputation problem can be regarded as a classification problem. Instead of filling the holes, one can treat imputation

as finding the class that the unit belongs to. There are many classification methods that can be borrowed for imputation purpose including the latest ones such as ARCING method (Mojirsheibani, 1999) and partitioning method (Breiman, 1998). The partitioning method is based on mapping idea, which regards classification as a mapping from the covariate space to class space. The individual classification method is just a way of partitioning the covariate space in to areas of corresponding classes. We shall describe discriminant classification methods, tree-based classification method and partition based method respectively.

1.6.1 Discriminant Classification

Linear Discriminant Analysis

Suppose the continuous covariates $\mathbf{x}=(x_1 \ldots x_q)$ are completely observed, and have a multivariate normal distribution $N(\mu_j,\Sigma)$ for $y=j$, the j^{th} class, where μ_j is the mean of $\mathbf{x}$ in j^{th} class, and $\Sigma_{q\times q}$ is the variance of $\mathbf{x}$. Here it is assumed that $\Sigma_{q\times q}$ the variance of $\mathbf{x}$ given $y=j$ is the same for all classes (levels) of y. Suppose the prior of y equals to j^{th} level is π_j. Then the posterior probability that y equals to j^{th} level is

$$P(y=j \mid \mathrm{x},\theta) \propto \frac{1}{\sqrt{(2\pi)^q|\Sigma|}} \exp(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_j)\Sigma^{-1}\ (\mathbf{x}-\boldsymbol{\mu}_j)^T)\pi_j, \qquad (1.6.1)$$

where $\theta=(\mu_1,\ldots,\mu_p,\Sigma)$. Maximising $Pr(y=j|\mathbf{x})$ is equivalent to minimising $-2\log Pr(j|\mathbf{x})$. Here some notations are dropped for simplicity. $-2\log Pr(j|\mathbf{x})$ is given by

$$\begin{aligned} -2\log \Pr(j \mid \mathbf{x}_i) &= (\mathbf{x}_i-\boldsymbol{\mu}_j)\Sigma^{-1}(\mathbf{x}_i-\boldsymbol{\mu}_j)^T - 2\log\pi_j + \log((2\pi)^q|\Sigma|) \\ &= -2\mathbf{x}_i\Sigma^{-1}\boldsymbol{\mu}_j^T + \boldsymbol{\mu}_j\Sigma^{-1}\boldsymbol{\mu}_j^T - 2\log\pi_j + \mathbf{x}_i\Sigma^{-1}\mathbf{x}_j^T + \log((2\pi)^q|\Sigma|). \qquad (1.6.2) \end{aligned}$$

If we drop the last two terms in above equation that can be treated as constants and minus the equation, the linear discriminant analysis equation is obtained

$$LDA_j(\mathbf{x}_i) = 2\mathbf{x}_i\Sigma^{-1}\boldsymbol{\mu}_j^T - \boldsymbol{\mu}_j\Sigma^{-1}\boldsymbol{\mu}_j^T + 2\log\pi_j. \quad (1.6.3)$$

The estimates of μ_i and Σ are given by group means and overall sample variance. The imputation for a unit with covariate $\mathbf{x}_i$ is the category with highest LDA score (Ripley, 1996).

Quadratic Discriminant Analysis (QDA)

If there is clear evidence that the variances vary among groups, the variance of j^{th} class Σ_j needs to be specified separately (Ripley, 1996). This leads to quadratic discriminant analysis (QDA) equation

$$QDA_j(\mathbf{x}_i) = 2\mathbf{x}_i\Sigma_j^{-1}\boldsymbol{\mu}_j^T - \boldsymbol{\mu}_j\Sigma_j^{-1}\boldsymbol{\mu}_j^T + 2\log\pi_j. \quad (1.6.4)$$

In this case the variance of each group need to be estimated separately. The choice of LDA and QDA could be made according to the homogeneity of group variances. If there is strong evidence of heterogeneity, the QDA should be used; otherwise LDA may be good enough.

1.6.2 Classification and Regression Tree Imputation

When the normality assumption in discriminant classification doesn't hold, one can consider the non-parametric techniques, such as classification and regression tree (CART). Originally tree model doesn't need parametric assumptions, although more and more parametric CART models are developed (Peng, 1996).

A classification tree divides the whole data into two subgroups (binary splitting) by making the subgroups more homogenous with respect to target categorical variable. More precisely, a tree partitions the space of the observations into more pure leaves by binary splitting with regard to some impurity measures such as Gini index and entropy (Ripley, 1996). The original data (top node) is called root, the splitting continues until some

impurity requirement is satisfied, where impurity measures the heterogeneity of a subgroup.

The performance of a tree model relies on how the data is partitioned. The most commonly used impurity measures are the Gini index

$$i(p) = \sum_{i \neq j} p_i p_j = 1 - \sum_j p_j^2 \qquad (1.6.5)$$

and entropy

$$i(p) = -\sum_j p_j \ln p_j , \qquad (1.6.6)$$

where p_j is the percentage of jth category. The two measures share many common properties that make it safe to use either of them. The Gini index is also equivalent to the association measures for categorical variables given by Nelsen (1998). This property implies that choosing the highly associated variable as partitioning variable is consistent with an impurity measure. Breiman (1998) preferred the Gini index. On the other hand, others (Zhu, 1999) like entropy.

Practical implementation inevitably involves approximations, because the number of possible ways of partitioning is huge. The search for the optimal tree cannot be completed in a standard computer. That gives the slight discrepancies among different software implementations such as CART SAS tree, SPLUS tree and SPSS tree (Wilson, 1998). Caution is needed when one chooses the software programmes.

When the optimal tree is built, classification may be implemented by locating an individual case in one of the terminal nodes (the nodes that have no child node), and assigning the class with highest percentage in the terminal node to this case. For the purpose of imputation, a tree model links a data point with missing item which is the target variable (or dependent variable) to one terminal node by matching the covariates (independent variables). The category with biggest posterior probability in the matched node is given to the missing item. The philosophy is same as that of donor imputation and

nearest neighbour imputation. A tree model gives a different way to find the closest neighbour of missing items based on covariates.

1.6.3 ARCING Method

Breiman (1998) proposes a method that is adaptive re-sampling and combining classification method (ARCING). The motivation is to reduce the uncertainty in classification, since the classes in each terminal node have a distribution which can not be reflected in single classification. Unlike multiple imputation, where multiple values are kept in several data sets, ARCING method repeatedly classifies a case in to a sequence of classes by re-sampling approach. The final class is the mode of this class sequence.

Suppose we have l imputations from l distinct methods. Let $y^{(l)}{}_i$ denote the l^{th} imputation. Voting method takes the mode of the l imputations as the final imputation. Namely the value with highest occurrence rate is chosen as imputation.

$$\hat{y}_i = \text{mod}(y_i^{(k)}, k = 1, l), \tag{1.6.7}$$

where *mod* is the function that returns the mode of the argument. The original data set is first divided into training data and test data in the process of data preparation. The ARCING procedure is as follows.

- Bootstrap the training data
- Train the CART or neural networks using the bootstrapped data.
- Adjust the re-sampling weight, give more weight to cases with higher rate of misclassification.
- Combine the multiple classification of test data and choose the mode as imputation
- Repeat the above process until the expected accuracy is achieved.

Breiman showed that the misclassification rate keeps decreasing even when the model is over-fitted. He gives a Bias-Variance explanation and showed that ARCING can reduce the variance, therefore overcomes the instability of one-shot training.

1.6.4 Combining Classification Methods

Mojirsheibani (1999) proposed a partitioning based classification procedure. Suppose we have M classification methods (classifiers) $C_{m,1}(x),\ldots,C_{m,M}(x)$. Each one is a map $\mathbf{x} \rightarrow \{1\ldots p+1\}$. The new classifier imputes y by k given $\mathbf{x}$ if k has the following property:

$$\sum_{i:y_i=k} I\{C_{m,1}(\mathbf{x}_i)=C_{m,1}(\mathbf{x}),\ldots,C_{m,M}(\mathbf{x}_i)=C_{m,M}(\mathbf{x})\}$$
$$> \sum_{i:y_i=l,l\neq k} I\{C_{m,1}(\mathbf{x}_i)=C_{m,1}(\mathbf{x}),\ldots,C_{m,M}(\mathbf{x}_i)=C_{m,M}(\mathbf{x})\}. \qquad (1.6.8)$$

The motivation behind this formula is that if the individual classifiers $C_1\ldots C_M$ are all non-random, then it simply finds the best match of $(C_{m,1}(x),\ldots,C_{m,M}(x))$ in the iid discretized "data",

$$\{(C_{m,1}(x_i),\ldots,C_{m,M}(x_i)), y_i\}, \text{ for } i=1,\ldots,m.$$

The combined classifier Ψ_m can be rewritten as

$$\Psi_m(C_{m,1}(\mathbf{x}),\ldots,C_{m,M}(\mathbf{x})) = \arg\max_{1\le k\le p+1} \sum_{i=1}^{m}\prod_{j=1}^{M} I\{C_{m,j}(\mathbf{x}_i)=C_{m,j}(\mathbf{x})\}\times I\{y_i=k\}. \qquad (1.6.9)$$

If ties happen, the smallest category is selected for simplicity.

1.7 Imputation for Multivariate Missing Data Patterns

In section 1.5.1 we introduced linear regression imputation, where the dependent variable y is scalar and has a covariate x. Here the covariate x is assumed to have equal role as y.

Both y and x are taken as a component of a multidimensional variable **y**. The missingness is assumed to happen in more than one components of **y**. The multivariate missingness in matrix format data is very common in survey and census situations. Let a sample of n units be drawn from the superpopulation $\mathbf{y} \sim N(\mu, \Sigma)$, whose density function is given by

$$f(\mathbf{y}) = \left|(2\pi)^q \Sigma\right|^{-1/2} \exp(-\frac{1}{2}(\mathbf{y}-\boldsymbol{\mu})' \Sigma^{-1}(\mathbf{y}-\boldsymbol{\mu})). \tag{1.7.1}$$

For convenience, we assume **y** is a 3×1 vector with elements y_1, y_2 and y_3, where y_1 is observed for all cases.

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix}, \tag{1.7.2}$$

$$\Sigma = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \sigma_{13}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \sigma_{23}^2 \\ \sigma_{31}^2 & \sigma_{32}^2 & \sigma_{33}^2 \end{bmatrix}. \tag{1.7.3}$$

Let ω_{123} be the subsample of cases with all three variables available, ω_{12} be the subsample of cases with y_1 y_2 available, y_3 missing, ω_{13} be the subsample of cases with y_1, y_3 available, y_2 missing, and ω_1 be the subsample of cases with only y_1 available. The four subsets have m_1, m_2, m_3, m_4 cases respectively ($\sum_i m_i = n$).

1.7.1 Parameter Estimation

The unknown parameter is denoted $\theta = (\boldsymbol{\mu}, \Sigma)$. The estimation of θ is based on the likelihood function. Under the MAR assumption, using the factorisation corresponding to each parameter, The likelihood can be written as the product of marginal and the conditional probability (Little and Rubin, 1987).

$$f(Y_{obs}\mid\theta)=\prod_{i=1}^{m_1} f(y_{1i},y_{2i},y_{3i}\mid\theta)\prod_{i=m_1+1}^{m_1+m_2} f(y_{1i},y_{2i}\mid\theta)\prod_{i=m_1+m_2+1}^{m_1+m_2+m_3} f(y_{1i},y_{3i}\mid\theta)\prod_{i=m_1+m_2+m_3+1}^{n} f(y_{1i}\mid\theta)$$

(1.7.4)

If we arrange θ in matrix $\theta^*=\begin{bmatrix}-1 & \boldsymbol{\mu}^T\\ \boldsymbol{\mu} & \Sigma\end{bmatrix}$, and apply the sweeping operator (SWP) to the first column of θ^*, the parameters of the regression models of y_2 on y_1 and y_3 on y_1 are obtained as follows.

$$\text{SWP}[1]\begin{bmatrix}-1 & \boldsymbol{\mu}^T\\ \boldsymbol{\mu} & \Sigma\end{bmatrix}=\begin{bmatrix}-1 & \mu_1 & \beta_{20.1} & \beta_{30.1}\\ \mu_1 & \sigma_{11}^2 & \beta_{21.1} & \beta_{31.1}\\ \beta_{20.1} & \beta_{21.1} & \sigma_{22.1}^2 & \sigma_{32.1}^2\\ \beta_{30.1} & \beta_{31.1} & \sigma_{32.1}^2 & \sigma_{33.1}^2\end{bmatrix}. \qquad (1.7.5)$$

The resulting parameters $\beta_{20.1}$ $\beta_{21.1}$ and $\sigma_{22.1}^2$ are the intercept slope and residual variance of the regression of y_2 on y_1 respectively. Similarly $\beta_{30.1}$ $\beta_{31.1}$ and $\sigma_{33.1}^2$ are the corresponding parameters of regression y_3 on y_1. The maximum likelihood estimator (MLE) of θ can be obtained by reverse sweeping (RSW) the matrix of the estimators of parameters of conditional models (Rubin,1987, p. 119),

$$\begin{bmatrix}-1 & \hat{\boldsymbol{\mu}}^T\\ \hat{\boldsymbol{\mu}} & \hat{\Sigma}\end{bmatrix}=\text{RSW}[1]\begin{bmatrix}-1 & \hat{\mu}_1 & \hat{\beta}_{20.1} & \hat{\beta}_{30.1}\\ \hat{\mu}_1 & \hat{\sigma}_{11}^2 & \hat{\beta}_{21.1} & \hat{\beta}_{31.1}\\ \hat{\beta}_{20.1} & \hat{\beta}_{21.1} & \hat{\sigma}_{22.1}^2 & \hat{\sigma}_{32.1}^2\\ \hat{\beta}_{30.1} & \hat{\beta}_{31.1} & \hat{\sigma}_{32.1}^2 & \hat{\sigma}_{33.1}^2\end{bmatrix}, \qquad (1.7.6)$$

where RSW is the reverse sweep operator. The coefficient estimators are obtained based on the following properties of multinormal distribution,

$$E(y_2 \mid y_1) = \beta_{20.1} + \beta_{21.1} y_1,\ \text{var}(y_2 \mid y_1) = \sigma^2_{22.1}, \tag{1.7.7}$$

$$E(y_3 \mid y_1) = \beta_{30.1} + \beta_{31.1} y_1,\ \text{var}(y_3 \mid y_1) = \sigma^2_{33.1}, \tag{1.7.8}$$

$$\text{cov}(y_2, y_3 \mid y_1) = \sigma^2_{32.1}. \tag{1.7.9}$$

These above results pave the way of using plug-in estimators to reduce the variances of the estimators of population parameters.

1.7.2 Imputation

Regression Imputation

For an individual case with y_{2j} missing y_{3j} available or vice versa, the imputation can be the estimated conditional mean of the missing variable given the observed variables or the conditional mean plus a residual term (random regression imputation). Let's consider mean imputation first. It can be described as follows.

$$\hat{y}_{2j} = \hat{\beta}_{20.13} + \hat{\beta}_{21.13} y_{1j} + \hat{\beta}_{23.13} y_{3j}, j \in \omega_{13} \tag{1.7.10}$$

$$\hat{y}_{3j} = \hat{\beta}_{30.12} + \hat{\beta}_{31.12} y_{1j} + \hat{\beta}_{32.12} y_{2j}, j \in \omega_{12} \tag{1.7.11}$$

where $\hat{\beta}_{20.13}, \hat{\beta}_{21.13}, \hat{\beta}_{23.13}, \hat{\beta}_{30.12}, \hat{\beta}_{31.12}, \hat{\beta}_{32.12}$, can be obtained by sweeping matrix (1.7.6) by column index [1,3] and [1,2] respectively.

For the sample with both y_2 and y_3 missing, we can impute one variable with the conditional mean given y_1, then impute the other one with the mean conditioning on both y_1 and the imputed variable. For convenience, Let's start with y_2,

$$\hat{y}_{2j} = \hat{\beta}_{20.1} + \hat{\beta}_{21.1} y_{1j}, j \in \omega_1 \tag{1.7.12}$$

$$\hat{y}_{3j} = \hat{\beta}_{30.12} + \hat{\beta}_{31.12} y_{1j} + \hat{\beta}_{32.12} \hat{y}_{2j}, j \in \omega_{12} \tag{1.7.13}$$

As aforementioned, the coefficients are obtained by sweeping matrix (1.7.6) at [1] and [1,2] respectively.

Random Regression Imputation

The random regression imputation is simply the regression imputation plus a residual term, which can be generated from the estimated conditional variance or a random draw from the residuals of the complete cases.

For single variable missingness the imputation is

$$\hat{y}_{2j} = \hat{\beta}_{20.13} + \hat{\beta}_{21.13} y_{1j} + \hat{\beta}_{23.13} y_{3j} + e_{22.13}, j \in \omega_{13} \quad (1.7.14)$$

$$\hat{y}_{3j} = \hat{\beta}_{30.12} + \hat{\beta}_{31.12} y_{1j} + \hat{\beta}_{32.12} y_{2j} + e_{33.12}, j \in \omega_{12} \quad (1.7.15)$$

where $e_{22.13} \sim N(0, \hat{\sigma}^2_{22.13})$, $e_{33.12} \sim N(0, \hat{\sigma}^2_{33.12})$.

For the individual sample with both y_2 and y_3 missing, the imputation can be

$$\hat{y}_{2j} = \hat{\beta}_{20.1} + \hat{\beta}_{21.1} y_{1j} + e_{22.1}, j \in \omega_1 \quad (1.7.16)$$

$$\hat{y}_{3j} = \hat{\beta}_{30.12} + \hat{\beta}_{31.12} y_{1j} + \hat{\beta}_{32.12} \hat{y}_{2j} + e_{33.12}, j \in \omega_1 \quad (1.7.17)$$

where $e_{22.1} \sim N(0, \hat{\sigma}^2_{22.1})$, $e_{33.12} \sim N(0, \hat{\sigma}^2_{33.12})$.

1.8 Multiple Imputation

In the previous section, various methods of single imputation were reviewed. This section will address multiple imputation. Multiple imputation is the imputation approach that replaces each missing value with two or more imputed values representing a distribution of the possibilities; this approach was originally proposed by Rubin (1977, 1978).

As we already learnt in the previous sections, the single imputation method is focused on how to produce a representative single value to replace the missing value in terms of preserving marginal distribution and true values as well. Once the missing values are filled by imputations, it looks like a complete data set. Therefore standard complete-data methods of analysis can be used with extra effort. This could be the biggest advantage that attracts many practitioners to stick to it. For example in the census situation, single imputation provides a simple complete data. Third party users will take it as true complete data. The disadvantage of single imputation is that it is unable to reflect the uncertainty arisen from sample variability and the cause of nonresponse.

Since multiple imputation is constructed with the aim of reflecting the uncertainties associated with sampling and nonresponse, the advantages are obvious. It is equally important to address the disadvantages arising from the complexity of multiple imputation in terms of analysing and combining multiple data sets. If the contribution is minor, the complexity will become an apparent disadvantage.

The foundation of multiple imputation is Bayesian theory (Rubin, 1987). Suppose the quantity of interest is the population mean μ. Assume μ is a k-dimensional row vector. With complete data, inferences for μ would be based on the statement

$$(\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}) \sim N(0, \hat{\Sigma}) \qquad (1.8.1)$$

where $\hat{\boldsymbol{\mu}}$ is the estimator of μ, $\hat{\Sigma}$ is the estimated variance of $\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}$. Under a specified Bayesian model, l sets of repeated imputations have been drawn and used to construct l complete data sets, where $\hat{\boldsymbol{\mu}}_{*1}, \ldots, \hat{\boldsymbol{\mu}}_{*l}$ and $\Sigma_{*1}, \ldots, \Sigma_{*l}$ are the values of statistics, $\hat{\boldsymbol{\mu}}$ and Σ for each of these data sets.

The population estimates need to combine the l repeated complete data estimates under one model for nonresponse. Let

$$\bar{\boldsymbol{\mu}}_l = \sum_{i=1}^{l} \hat{\boldsymbol{\mu}}_{*i} / l \tag{1.8.2}$$

be the average of the l complete-data estimates, and

$$\bar{\Sigma}_l = \sum_{i=1}^{l} \Sigma_{*i} / l \tag{1.8.3}$$

be the average of the l complete-data variances, and

$$B_l = \sum_{i=1}^{l} (\hat{\boldsymbol{\mu}}_{*i} - \bar{\boldsymbol{\mu}}_l)^T (\hat{\boldsymbol{\mu}}_{*i} - \bar{\boldsymbol{\mu}}_l) / (l-1) \tag{1.8.4}$$

be the variance between (among) the l complete-data estimates. The quantity

$$T_l = \bar{\Sigma}_l + (1 + l^{-1}) B_l \tag{1.8.5}$$

is the estimated variance of $\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}$ using the multiple imputation method.

1.9 Aim of Thesis and Overview of the Chapters

The aim of this thesis is to investigate the possibility of using neural networks and related methods for imputation. Basically, we compare neural network imputation with other imputation methods such as regression imputation, and explore the advantages and disadvantages of neural network imputation. The comparison is based on how an imputation method can preserve the properties of population or original data such as mean, variance and marginal distribution. Meanwhile the practical concerns such as computing time are also considered. Based on the results of the comparison, a new imputation method, the weighted distance-based nearest neighbour imputation method (WD), is proposed. Later on the performances of the new method and other imputation methods are tested in the simulation study and numerical study.

The current chapter (Chapter 1) has reviewed the non-response problem in surveys and censuses as well as the main imputation methods. The topic of data analysis with incomplete data has not been covered, since the main objective of this thesis has been to consider the use of imputation in the estimation of descriptive population parameters such as mean (or total) and variance. Multiple imputation has also been mentioned in this review. This chapter provides a foundation for the comparison in the later chapters.

Chapter 2 is a review of neural networks from a statistical point of view. Only the radial basis function neural networks (RBF) and the multilayer perceptron neural networks (MLP) are considered. Other neural networks like Kohonen (1982) unsupervised model can also be used to find donors in nearest neighbour imputations. But it is not the objective of this study. Therefore it is excluded. The review starts with an introduction to the MLP and RBF and how they can be used for imputation. Thereafter the criteria of neural network training are discussed. Then the Wald criterion is provided with the motivation of preserving the marginal distribution. The relationship between MLP and the polynomial regression is also explored. The aim is to aid understanding of the structure of MLP. In the end, the relationship between RBF and non-parametric regression is reviewed.

In Chapter 3, a new imputation method is proposed, the weighted distance nearest neighbour imputation method (WD). This chapter describes the motivation and the formulation of this method as well as some asymptotic results. Comparisons with the Euclidean distance based nearest neighbour imputation method and predictive mean match imputation based on a linear regression model are also provided. Finally, situations where WD can outperform other methods are considered.

Chapter 4 discusses the theoretical properties of imputation methods. For continuous variables, the focus is on how an imputation method can preserve the population mean and variance. The main results are based on the asymptotic properties of linear regression imputation and RBF imputation. Several special cases are discussed to aid understanding of the theoretical results. Imputation for multiple variables is also discussed under the assumption of a multinormal distribution. This chapter also reviews evaluation criteria for the imputation of categorical variables.

Chapter 5 provides simulation studies based on real data and simulated data. The simulations are designed to assess the properties of the different imputation methods empirically and to test the theoretical results in Chapter 3 and Chapter 4.

Chapter 6 summarises the conclusions and provides some ideas for future research.

2 Introduction to Neural Networks

The first neural network was invented by McCullock and Pitts (1943) who used a simple calculus method to describe nervous activity. Since then it has evolved rapidly into a major area of several fields such as computer science and cognitive research. But it was in the last two decades that neural networks has become a practical tool for practitioners in many field, thanks to the revolution of the computing industry. In general, neural networks fall into two broad categories: supervised and unsupervised. Unsupervised neural networks can be used to find patterns in data without a target variable, or are to be used to confirm that the original classes are suitable (Ripley, 1997). Unsupervised neural networks such as Kohonen self-organising map (Kohonen, 1995) can also be used to defined classes, which then can be used to locate donors in donor-based imputations. On the contrary, supervised neural networks such as multi-layer perceptron (MLP) and radial basis function (RBF) (Bishop, 1996) have a target variable, which is always referred as the output variable, and several input variables, which are the counterparts of independent variables in regression. Supervised neural networks can be used to predict the individual values of the target variable with known input variables when the dependent variable is continuous, or be used to predict the probability of falling into one of the possible categories (Cheng, Titterington, 1994). Thus supervised neural networks can be viewed as a form of nonparametric regression. Some neural networks can be regarded as semi-parametric models since they contain parametric features or are based upon some model specifications. For imputation purpose, we focus on supervised neural networks. Supervised neural networks can either predict the missing values directly when the variable containing the missing values is continuous, or predict the posterior probability of a category when the variable containing missing values is categorical.

For further discussion we denote (X, Y) as the data matrix of n observations, where $X=(\mathbf{x}_1,\ldots,\mathbf{x}_n)^T$ is a $n\times q$ design matrix, and $Y=(y_1,\ldots,y_n)^T$ is the vector of n observations of the response variable.

Let $y_i \sim N(f(\mathbf{x}_i), \sigma^2)$, and

$$y_i = f(\mathbf{x}_i) + \varepsilon_i, \qquad \varepsilon_i \sim N(0, \sigma^2), \tag{2.0.1}$$

where f is the mean function of y_i given $\mathbf{x}_i$, and σ^2 is the variance of residual term. For convenience we denote $f(X)=(f(\mathbf{x}_1),\ldots,f(\mathbf{x}_n))^T$. In practice the form of f is unknown. MLP and RBF are two alternative approximations to more common choices such as linear model.

The process of estimating the parameters of a given neural networks is termed neural network training. One common training strategy is using the cross-validation method, by which the original data set is divided into two subsets. One data set (bigger one) is used for estimating the parameters, normally neural network weights (2.0.1) and is termed training data. The other one (smaller one) is used to adjust the estimates, and is termed test data. The optimal value of an estimator is the one that minimises the mean square error of the test data.

Details of the two supervised neural networks: Radial Basis function (RBF) and Multi-layer perceptron (MLP) are discussed in the next two sections.

2.1 Radial Basis Function Neural Network (RBF)

The RBF neural network approximates the expectation function f given in (2.0.1) by a weighted sum of the transformed observations of X, where the transformation is a mapping by kernel-like basis functions from the original data X to the pre-determined data points in the space spanned by X. For convenience we term radial basis function neural network RBF. The data points are normally termed data centres. They could be a sample of the original observations or the multivariate quantiles of X. The mapping is depicted in the following picture.

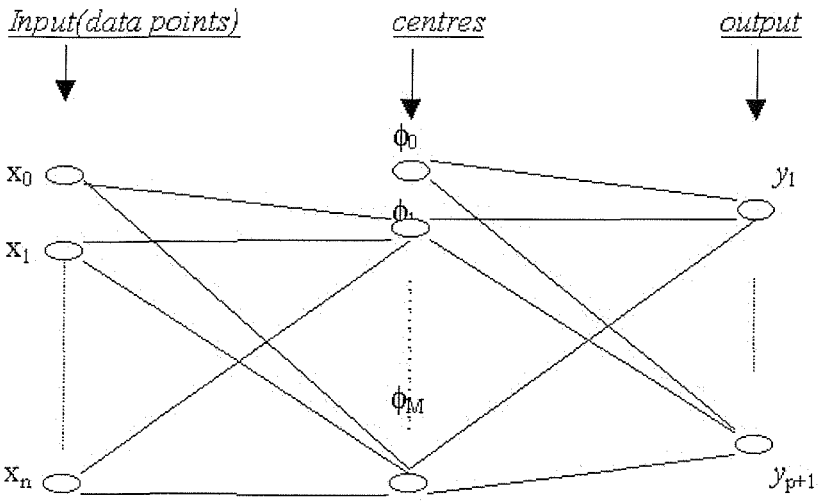

Figure 2.1 RBF neural network projects the original data points to the hidden layer by the basis function $\phi(x)$. The prediction is based on the weighted sum of the hidden nodes.

RBF approximates $f(\mathbf{x})$ by:

$$f(\mathbf{x},W) = w_0 + \sum_{j=1}^{M} w_j \phi_j \left(\left\| \mathbf{x} - \boldsymbol{\mu}_j \right\| \right), \qquad (2.1.1)$$

where ϕ_j is a basis function, which is similar to a kernel function in non-parametric regression (Hardle,1989), M is the number of basis functions, μ_j is the jth centre, w_j is the weight of j^{th} basis function, $W=(w_0,\ldots, w_M)^T$. The commonly used basis function is the Gaussian function

$$\phi(x)=exp(-x^2/2\lambda), \qquad (2.1.2)$$

where λ is a parameter that controls the smoothness of the function. This is similar to the bandwidth in kernel smoothing.

The value μ_j cab be pre-determined. Some RBFs treat μ_j as part of the training parameter. This will slow down the training process. The smoothness parameter λ can be determined in the training process by cross-validation method. Once the two parameters of the basis function are determined, the RBF network can be regarded as linear regression:

$$E(y_i)=f(\mathbf{x}_i,W)=\Phi(\mathbf{x}_i)^{\mathrm{T}}W, \qquad (2.1.3)$$

where $\Phi(\mathbf{x}_i)=(1,\phi_1(\mathbf{x}_i)\ldots\phi_M(\mathbf{x}_i))^{\mathrm{T}}$. Once a RBF is built, the diagnostic methods of linear regression can be used to assess model. For example, if we denote $\Phi=(\Phi(\mathbf{x}_1),\ldots,\Phi(\mathbf{x}_n))^{\mathrm{T}}$, the hat matrix H becomes

$$H=\Phi(\Phi^T\Phi)^{-1}\Phi^T.$$

Where the hat matrix is a non-negative idempotent matrix in linear regression model, which is composed of the covariate X ($X(X^TX)^{-1}X^T$). The vector of the predicted values of y is therefore as follows

$$\hat{Y} = HY, \qquad (2.1.4)$$

and

$$\mathrm{var}(\hat{y}_i) = h_{ii}\sigma^2. \qquad (2.1.5)$$

The deviance residuals are

$$\hat{\varepsilon}_i = \frac{y_i - \hat{\mu}_i}{\hat{\sigma}\sqrt{1-h_{ii}}}, \qquad (2.1.6)$$

which provide evidence of goodness of fit. If the residual plot displays some patterns, re-defining centres is required. A practical way to define data centres is to use classification methods such as classification and regression tree models. Zhu, Yao and Liu (1999) give an application of this approach.

2.2 RBF Networks and Non-parametric Regression

Nonparametric regression is a technique of estimating the regression curve by a weighted average of response values in a neighbourhood defined by the covariate of the response variable (Hardle, 1989). The weights are defined by a density-like function of the

covariate. The weight function is normally termed kernel function. The neighbourhood of the covariate is defined by bandwidth, the radius of neighbourhood. Once the kernel function is chosen, the weight value can be adjusted by using different bandwidth to achieve goodness of fit. Unlike nonparametric regression, RBF method projects the original values of covariates to data centres, then averages the transformed values to produce predictions. The transformation function in RBF model is also a density-like function. The weights in RBF model are estimated by linear regression model. In some sense RBF neural network can be regarded as a mixture of kernel method and dictionary method. The critical step in building RBF model is to define the centres. Overall both methods form prediction by weighted average of response values. There is a possibility to construct a RBF model to be equivalent to a nonparametric regression. Kay and Titterington (1999) gave an extensive comparison of RBF with statistical methods. One of their results is the equivalence of RBF and non-parametric regression. For convenience we assume the covariate is a scalar variable and denote x. For a given covariate value, the RBF approximates the expected response value by

$$f_{RBF}(x,\hat{W}) = \hat{w}_0 + \sum_{j=1}^{M} \hat{w}_j \phi(\frac{x-\mu_j}{\sigma_j})$$

$$= \Phi(x)(\Phi(X)^T \Phi(X))^{-1} \Phi(X)^T Y, \qquad (2.2.1)$$

where $\Phi(x)^{\mathrm{T}}=(1, \phi(\frac{x-\mu_1}{\sigma_1}) \ldots \phi(\frac{x-\mu_M}{\sigma_M}))$.

Similarly, a non-parametric regression approach represents the mean of y by

$$E(y \mid x) = \sum_{j=1}^{n} y_j K(x_j), \qquad (2.2.2)$$

where $K(x_j)$ is the kernel function corresponding to jth observation (Hardle, 1989).

If we let $\mu_i=x_i$, $i=1,\ldots,n$, and define the kernel function in non-parametric regression as

$$K^*(x_j) = \phi_j(x)[(\Phi(X)^T\Phi(X))^{-1}\Phi(X)^T]_j, \qquad (2.2.3)$$

the RBF model can be expressed in the non-parametric regression form.

$$f_{RBF}(x,\hat{W}) = \sum_{j=1}^{n} y_j K^*(x_j). \qquad (2.2.4)$$

We need to demonstrate that $K^*(x_j)$ is a kernel function for nonparametric regression. Since $\phi_j(x)$ is a density function (see 2.12), and the second term in the right side of (2.2.3) $[(\Phi(X)^T\Phi(X))^{-1}\Phi(X)^T]_j$ is a constant given X, therefore $K^*(x_j)$ can be taken as a kernel function, which leads to the equivalence of the two methods.

2.3 Multi-layer Perceptron Neural Network (MLP)

Similar to projection pursuit regression, MLP projects the independent variables to hidden nodes by a non-linear function, normally a logistic sigmoid function. The predicted value of a continuous dependent variable is the weighted sum of the transformed values of the nodes in the last hidden layer. For a categorical dependent variable, the predicted membership probability is the transformed value of the weighted sum by the sigmoid function. The MLP neural networks can be depicted in the following figure.

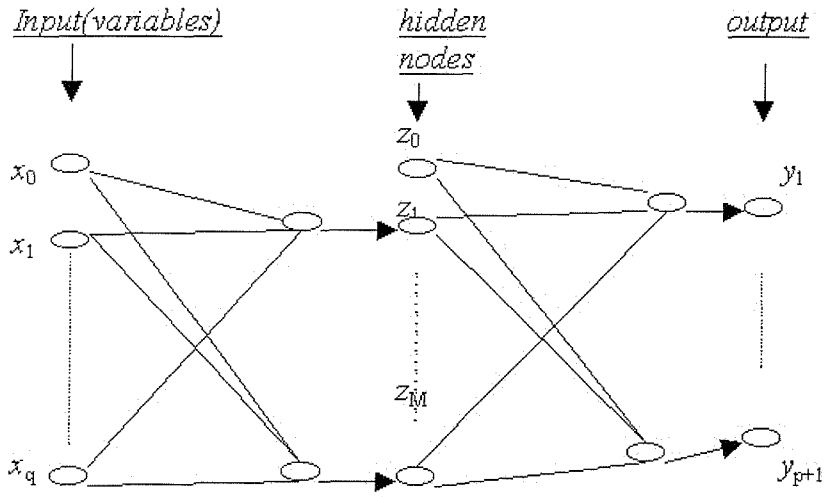

Figure 2.2 MLP neural networks project the independent variables to the hidden layer by the logistic sigmoid function $g_1(x)$. The prediction is based on the weighted sum of the hidden nodes.

The more nodes it has the more complex the MLP model is. The widely used non-linear function is the logistic sigmoid function

$$g_1(x) = \frac{1}{1+e^{-x}}. \tag{2.3.1}$$

Another widely used transformation function is *tanh*, which has advantages in some circumstances.

$$\tanh(x) \equiv \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}. \tag{2.3.2}$$

The MLP with one hidden layer has the following explicit expression

$$f(\mathbf{x},W) = g_2(w_0^{(2)} + \sum_{j=0}^{M} w_j^{(2)} g_1(\sum_{i=1}^{q} w_{ji}^{(1)} x_i)), \tag{2.3.3}$$

where g_1 is the sigmoid logistic function, g_2 is a linear or identity function, $\mathbf{x}=(x_1,\ldots,x_q)$ is the design variable whose values are organised in $X_{n\times q}$, $w_{ji}^{(1)}$ is the weight of x_i to j^{th} node, $w_j^{(2)}$ is the weight of j^{th} node to the final output. If we assume that $Y \sim N(f(X,W), \sigma^2 I_n)$ as in the previous section, the model can be described as:

$$Y = f(X,W) + \varepsilon. \tag{2.3.4}$$

where $f(X,W)=(f(\mathbf{x}_1,W),\ldots, f(\mathbf{x}_n,W))^{\mathrm{T}}$. The MLE (maximum likelihood estimator) of W is just the LSE (least square estimator) $\hat{W}$ obtained by solving the following equation

$$\frac{\partial \|Y - f(X,W)\|^2}{\partial W} = 0. \tag{2.3.5}$$

Neural networks such as MLP use the back-propagation method to iteratively estimate $\hat{W}$, because the explicit solution of (2.3.5) is normally too complex to be obtained. As we see, MLP is just a non-linear regression with complex expression. For the multinomial model, a softmax transformation is used at final layers to predict class probability $\pi_i=(\pi_{i1} \ldots \pi_{ip})$, where π_{ij} is the probability of y_i taking value j.

$$\pi_{ij} = \frac{\exp(f(\mathbf{x}_i, W_j)}{\sum_j \exp(f(\mathbf{x}_j, W_j)}. \tag{2.3.6}$$

The log-value of probability ratio can be obtained as

$$ln(\pi_{il}/\pi_{ik}) = f(\mathbf{x}_i, W_l) - f(\mathbf{x}_j, W_k). \tag{2.3.7}$$

Both RBF and MLP are very flexible. They can be configured to be any continuous functions by computer intensive training. On the other hand, the flexibility usually causes time-consuming training process. Fortunately computer power is growing rapidly; this will not be a severe inconvenience. In light of this statement, investigating neural networks has long-term benefits.

RBF has its origins in techniques for performing exact interpolation of a set of data points in a multi-dimensional space. A set of basis functions (Powell, 1987) is used to project original data to basis function space. The weighted basis function is then taken as the estimate of response values. Unlike RBF, a MLP approximates the complexity by setting more hidden nodes. The more nodes it has the more flexible it is.

2.4 Criteria of Training Neural Networks

Historically, neural networks were developed by computer scientists. Their statistical properties have only been investigated more recently. Fortunately, more and more statisticians are paying attention to these methods, such as Ripley (1997) and Titterington

(1994) who have already given great contributions in this field. Meanwhile, computer scientists also give their views in statistical way (Bishop, 1995).

When the structure of a neural network, namely the number of layers and the number of nodes in each layer, is determined, the performance will depend on the estimator of the weight W. Sum-square-error (SSE) is widely employed. A more generalised error criteria, Minkowski error ($|\varepsilon|^R$, $R>0$), is also suggested in some circumstances. The rest of this section is dedicated to reviewing the error functions from the statistical perspective.

Sum-of-Squares Error (SSE)

As in section 1.3, Y_{obs} is the observed data matrix. We assume variable $\mathbf{y}$ is a vector, which can be the vector notation of a categorical variable. The corresponding expectations of Y_{obs} are $(f(\mathbf{x}_1,W),\ldots,f(\mathbf{x}_m,W))^T$. Then the sum-squares- error is

$$E_{sse} = \sum_{l=1}^{m} \left\| \mathbf{y}_l - \mathrm{f}(\mathbf{x}_l, W) \right\|^2$$

$$= \sum_{l=1}^{m} \sum_{k=1}^{p+1} (y_{lk} - f(x_{lk}, W))^2 . \qquad (2.4.1)$$

W can be estimated by minimising E_{sse}.

If we assume $\mathbf{y}_1 \ldots \mathbf{y}_m$ are independent, and follow the normal distribution, $N(\mathrm{f}(\mathbf{x}_1, W), \Sigma) \ldots N(\mathrm{f}(\mathbf{x}_m, W), \Sigma)$ respectively. For simplicity we start with $\Sigma=\sigma^2 \boldsymbol{I}_{p+1}$. The general case will be discussed later on. Then $y_{11} \ldots y_{m,p+1}$ are independently distributed with the distributions $N(f(x_{11}, W), \sigma^2) \ldots N(f(x_{m,p+1}, W), \sigma^2)$ respectively. The log-likelihood function is

$$L \propto -\frac{1}{2\sigma^2} \sum_{l=1}^{m} \sum_{k=1}^{p+1} (y_{lk} - f(x_{lk}, W))^2 - \frac{m(p+1)}{2} \log \sigma , \qquad (2.4.2)$$

Then the maximum likelihood estimator of σ^2 is

$$\hat{\sigma}^2 = \frac{1}{m(p+1)} \sum_{l=1}^{m} \sum_{k=1}^{p+1} (y_{lk} - f(x_{lk}, W))^2 . \qquad (2.4.3)$$

Plugging (2.4.3) into (2.4.2), we obtain

$$\hat{L} \propto -\frac{1}{2} \log \hat{\sigma} . \qquad (2.4.4)$$

Therefore maximising $\hat{L}$ is equivalent to minimising $\hat{\sigma}^2$. The OLS rule is obtained.

Maximum Likelihood Criterion

In this section we deduce the training criterion from the likelihood function under the normality assumption given above but with general variance Σ. The log-likelihood for W and Σ is

$$L(W, \Sigma^{-1}) = cons + \frac{m}{2} \ln\left|\Sigma^{-1}\right| - \frac{tr(Z\Sigma^{-1}Z^T)}{2}, \qquad (2.4.5)$$

where *cons* is a constant, | | is matrix determinant, Z is a m×(p+1) matrix,

$$Z = \mathbf{Y}_{obs} - \mathrm{f}(X, W) = (y_{ik} - f(x_{ik}, W))_{\substack{i=1,m \\ k=1,p+1}} . \qquad (2.4.6)$$

The last term of the log-likelihood can be written as $tr(\mathrm{Z}^{\mathrm{T}}\mathrm{Z}\Sigma^{-1})$ by matrix manipulation rule. The maximum likelihood estimate (MLE) of Σ^{-1} is composed of element wise estimation. Using the result (Bard, 1974)

$$\frac{\partial \ln\left|\Sigma^{-1}\right|}{\partial \sigma^{ij}} = \{\Sigma\}_{ij}, \qquad (2.4.7)$$

Allows us to write

$$\frac{\partial L(W,\Sigma^{-1})}{\partial \sigma^{ij}} = \frac{m}{2}\{\Sigma\}_{ij} - \frac{1}{2}\left[Z^T Z\right]_{ij}, \tag{2.4.8}$$

Setting the derivatives to zero provides the conditional estimates

$$\hat{\Sigma}(W) = \frac{Z^T Z}{m}. \tag{2.4.9}$$

Substituting this estimate into log-likelihood function gives the following conditional likelihood function

$$L(W,\hat{\Sigma}^{-1}) = cons - \frac{m}{2}\ln\left|Z^T Z\right|. \tag{2.4.10}$$

The MLE of W is obtained by minimising the determinant of Z^TZ with respect to W. If the off-diagonal elements are ignored, the determinant is the product of diagonal elements.

$$|Z^T Z| \approx \prod_{k=1}^{p+1}\left[Z^T Z\right]_{k,k}, \tag{2.4.11}$$

where

$$\left[Z^T Z\right]_{k,k} = \sum_{i=1}^{m}(y_{ik} - f(x_{ik}, W_k))^2. \tag{2.4.12}$$

Notice the W_k's are distinct; the solution of setting derivatives of equation (2.4.11) to zeros equals that of E_{sse}. This provides the evidence of equivalence to OLS.

Entropy Error – Log-Likelihood Ratio Criterion

In the previous section, the response variable y is assumed to follow a multivariate normal distribution. For categorical response, an alternative error function is the entropy error

(Bishop, 1996). If we denote $\pi_{ik}=P(y_{ik}=1|\mathbf{x}_i)$, the likelihood of the observed data Y_{obs} can be written as

$$L(Y_{obs} \mid X,\theta) = \prod_{i=1}^{m} \prod_{k=1}^{p+1} \pi_{ik}{}^{y_{ik}} ,$$

where $\theta = (\pi_{i,k})_{\substack{i=1,\ldots,m \\ k=1,\ldots,p+1}}$. The log-likelihood is the log-value of the above expression

$$\ell(Y_{obs} \mid X,\theta) = \sum_{i=1}^{m} \sum_{k=1}^{p+1} y_{ik} \ln \pi_{ik} , \qquad (2.4.13)$$

The neural networks method estimates the conditional probability $\pi_i = (\pi_{i1},\ldots,\pi_{i,p+1})$ with the help of the softmax function (2.3.6) in the last layer. The log-likelihood based on the estimators $\hat{\boldsymbol{\pi}}_i = (\hat{\pi}_{i1},\ldots,\hat{\pi}_{i,p+1})$ from neural networks is

$$\ell(Y_{obs} \mid X,\hat{\theta}) = \sum_{i=1}^{m} \sum_{k=1}^{p+1} y_{ik} \ln \hat{\pi}_{ik} . \qquad (2.4.14)$$

Then the log-likelihood ratio is obtained as

$$\ell(Y_{obs} \mid X,\hat{\theta}) - \ell(Y_{obs} \mid X,\theta) = \sum_{i=1}^{m} \sum_{k=1}^{p+1} y_{ik} \ln \frac{\hat{\pi}_{ik}}{\pi_{ik}} . \qquad (2.4.15)$$

If we use the approximation $\pi_{ik}=P(y_{ik}=1|\mathbf{x}_i)\approx y_{ik}$, and let $y_{ik} \ln \frac{\hat{\pi}_{ik}}{y_{ik}}$ equal zero if y_{ik} is zero, then the log-likelihood ratio approximately becomes

$$\ell(Y_{obs} \mid X,\hat{\theta}) - \ell(Y_{obs} \mid X,\theta) = \sum_{i=1}^{m} \sum_{k=1}^{p+1} y_{ik} \ln \frac{\hat{\pi}_{ik}}{y_{ik}} . \qquad (2.4.16)$$

The validity of the above approximation may be justified by assuming that $y_{ik}=1$ or $y_{ik}=0$ is fully determined by $\mathbf{x}_i$, for example $\mathbf{y}$ and $\mathbf{x}$ have a deterministic relationship. If we let

the neural networks maximising the log likelihood ratio, we obtain the entropy error function E_{entropy}.

$$E_{entropy} = -\sum_{i=1}^{m}\sum_{k=1}^{p+1} y_{ik} \ln \frac{\hat{\pi}_{ik}}{y_{ik}}$$

$$= \sum_{i=1}^{m}\sum_{k=1}^{p+1} y_{ik} \ln \frac{y_{ik}}{\hat{\pi}_{ik}} \quad . \tag{2.4.17}$$

As the training iterations increase, $\hat{\pi}_{ik}$ become more and more close to y_{ik}, because of the effect of minimising E_{entropy} in the process of neural network training. Notice $\hat{\pi}_{ik} \le y_{ik}$ for y_{ik} =1. If we assume the training converges to an optimal point although it diverges with improper initial weights. Then $\left| \frac{\hat{\pi}_{ik} - y_{ik}}{y_{ik}} \right| < 1$ holds eventually. Substitute this term with first three terms of its Taylor expansion

$$\ln \frac{\hat{\pi}_{ik}}{y_{ik}} = \ln(\frac{y_{ik} - (y_{ik} - \hat{\pi}_{ik})}{y_{ik}})$$

$$= \ln(1 - \frac{y_{ik} - \hat{\pi}_{ik}}{y_{ik}})$$

$$= 1 + \frac{y_{ik} - \hat{\pi}_{ik}}{y_{ik}} + (\frac{y_{ik} - \hat{\pi}_{ik}}{y_{ik}})^2 + (\frac{y_{ik} - \hat{\pi}_{ik}}{y_{ik}})^3 + \cdots . \tag{2.4.18}$$

The entropy error becomes

$$E_{entropy} \approx -\sum_{i=1}^{m}\sum_{k=1}^{p+1} y_{ik}(1 + \frac{y_{ik} - \hat{\pi}_{ik}}{y_{ik}} + (\frac{y_{ik} - \hat{\pi}_{ik}}{y_{ik}})^2)$$

$$= -\sum_{i=1}^{m}\sum_{k=1}^{p+1} (y_{ik} + y_{ik} - \hat{\pi}_{ik} + \frac{(y_{ik} - \hat{\pi}_{ik})^2}{y_{ik}})$$

$$= -\sum_{i=1}^{m}\sum_{k=1}^{p+1} y_{ik} - \sum_{i=1}^{m}\sum_{k=1}^{p+1} \frac{(y_{ik} - \hat{\pi}_{ik})^2}{y_{ik}}$$

$$= const - \sum_{i=1}^{m}\sum_{k=1}^{p+1} (y_{ik} - \hat{\pi}_{ik})^2 . \tag{2.4.19}$$

The last equation is based on the fact that y_{ik}=1 or 0, where *const* is the constant term. If 0 is taken, we let $\frac{\hat{\pi}_{ik} - y_{ik}}{y_{ik}} = 0$, which originates from the fact that $y_{ik} \ln \frac{\hat{\pi}_{ik}}{y_{ik}}$ equals zero (when y_{ik}=0). This gives the evidence of the equivalence between entropy error and sum-square-error.

2.5 Neural Networks Imputation for Missing Values in a Single Variable

We start with the missing values in a single variable. The assumption is that all covariates are completely observed. The application of neural networks for imputation is similar to the use of regression model for continuous data or logistic regression for categorical data (see section 1.4). The application of neural networks model for imputation is emerging gradually. The earliest application may be the work in the US Census Bureau (Creezy, Masand, Smith and Waltz, 1992). Neural networks have also been employed to edit statistical records (Nordbotten, 1995). One recent application of MLP neural network is carried out by Nordbotten (1996) to deal with the small area problem. The data he studied is the combination of the register data obtained from administrative registers and a survey data compiled by mail from a sample of the registered population. The derived data set contains blanks in the survey variables due to exclusion in the survey. This leads to inadequate cases for estimating the totals of some small areas based on the survey data. Nordbotten used a MLP model to impute the values in the survey variables for non-sampled cases. The estimators of totals for small areas can be re-constructed using the survey data and the imputed data. The MLP model in his study has one hidden layer without feedback connections among its units. The optimal number of hidden nodes is determined by experimenting five options at 10,15,25,40 and 60. The number corresponding the best imputation result is taken as optimal value. In this case it is 25. Nordbotten also experimented a single imputation model to impute all survey variables using the registered variables. He pointed that such a large neural network model in terms of the number of weights needs more training cycles than the models for individual variables.

Under the MAR assumption, the model estimated from the observed data can be used to impute the missing cases. The rationale of neural networks imputation also relies on this assumption.

With continuous y, the imputation can be the conditional mean predicted by neural networks.

$$\begin{aligned}\hat{y}_j &= E(y_j \mid Y_{obs}, \mathbf{x}_j, W), j=m+1,\ldots,n, \\ &\approx E(y_j \mid \mathbf{x}_j, \hat{W}(Y_{obs})) \\ &= f(\mathbf{x}_j, \hat{W}(Y_{obs})),\end{aligned} \tag{2.5.1}$$

$\hat{W}(Y_{obs})$ is the estimator of W by minimising the sum-of-squares error, which is equivalent to maximum likelihood estimate under the normal distribution assumption. Alternatively the imputation can be the conditional mean plus a residual term to preserve the variance

$$\hat{y}_j = f(\mathbf{x}_j, \hat{W}(Y_{obs})) + \hat{\varepsilon}_j, j=m+1,\ldots,n, \tag{2.5.2}$$

where

$$\hat{\varepsilon}_j \sim N(0, \hat{\sigma}^2). \tag{2.5.3}$$

If instead $\mathbf{y}$ is a categorical variable with $p+1$ categories, let the last be a reference category. If $\mathrm{y_i}$ denotes the i^{th} observation of $\mathbf{y}$, then $\mathbf{y}_\mathrm{i}$ is a $p+1$ vector with y_{ik}=1 indicating i^{th} observation falls into category k and y_{ik}=0 otherwise. For convenience we denote $\pi_{ik} = \Pr(y_{ik} = 1 \mid Y_{obs}, \mathbf{x}_j, W)$ and $\boldsymbol{\pi}_i = (\pi_{i1},\ldots,\pi_{i,p+1})^T, i = 1,\ldots,n$. In this notation, Y_{obs} becomes a multi-response matrix with $m\times(p+1)$ dimensions. Neural networks implicitly assume $\mathrm{y_i}$ follows a multivariate normal distribution $N(f(\mathrm{x}_i, W), \sigma^2\mathrm{I}_{\mathrm{p+1}})$ (Bishop, 1995). Then the expectation of j^{th} missing observation is

$$\hat{\boldsymbol{\pi}}_j = E(\mathbf{y}_j \mid Y_{obs}, \mathbf{x}_j, W), j=m+1,\ldots,n,$$

$$\approx E(\mathbf{y}_j \mid \mathbf{x}_j, \hat{W}(Y_{obs}))$$

$$= f(\mathbf{x}_j, \hat{W}(Y_{obs})), \qquad (2.5.4)$$

where

$$f(\mathbf{x}_j, \hat{W}(Y_{obs})) = (f(x_{j,1}, \hat{W}(Y_{obs})) \cdots f(x_{j,p+1}, \hat{W}(Y_{obs})))^T.$$

The softmax transformation in the final layer makes the summation of the elements of $f(\mathbf{x}_j, \hat{W}(Y_{obs}))$ equal one. Like in the logistic regression situation, the imputation is the category with the largest expectation that is actually the posterior probability or a random draw from the estimated distribution. One thing needs to be mentioned here, for ordinal variable **y**, the adjustment in logistic model (Agresti, 1990) for dealing with the order of categories is not necessary with the neural networks model. Because the adjustment can be achieved by changing the weights, which is actually done in training process. Therefore neural networks models have a unifying form for nominal and ordinal variables.

2.6 Neural Networks Imputation for Multivariate Missing Data

When the missing values occur in multiple variables, a simultaneous approach is suggested using regression model under the normality assumption (see section 1.5). We are interested in how the neural networks model can deal with multivariate missing values simultaneously. Let us use the neural networks model to approximate the conditional means. We assume the conditional means are the neural networks functions of given conditions.

$$E(y_2 \mid y_1) = f^N(y_1, w_{2.1}), \qquad (2.6.1)$$

$$E(y_3 \mid y_1) = f^N(y_1, w_{3.1}), \qquad (2.6.2)$$

$$E(y_2 \mid y_1, y_3) = f^N(y_1, y_3, w_{2.13}), \qquad (2.6.3)$$

$$E(y_3 \mid y_1, y_2) = f^N(y_1, y_2, w_{3.12}), \qquad (2.6.4)$$

where f^N denotes the neural networks function, and there is no missing values in y_1. The imputation can be carried out accordingly.

$$\hat{y}_2 \mid y_1 = f^N(y_1, \hat{w}_{2.1}), \tag{2.6.5}$$

$$\hat{y}_3 \mid y_1 = f^N(y_1, \hat{w}_{3.1}), \tag{2.6.6}$$

$$\hat{y}_2 \mid y_1, y_3 = f^N(y_1, y_3, \hat{w}_{2.13}), \tag{2.6.7}$$

$$\hat{y}_3 \mid y_1, y_2 = f^N(y_1, y_2, \hat{w}_{3.12}). \tag{2.6.8}$$

The neural networks models in (2.6.5) and (2.6.6) can be used to impute missing values in y_2 and y_3 when y_2 and y_3 are simultaneously missing. The imputation given in (2.6.7) is used to deal with the situation that missing values are only present in y_2. Similarly imputation in (2.6.8) is for the situation of missing values in y_3 while y_1 and y_2 are all observed. The random imputation of neural networks can be implemented by adding a residual term to the conditional mean.

$$\hat{y}_2 \mid y_1 = f^N(y_1, \hat{w}_{2.1}) + e_2, \tag{2.6.9}$$

$$\hat{y}_3 \mid y_1 = f^N(y_1, \hat{w}_{3.1}) + e_3, \tag{2.6.10}$$

$$\hat{y}_2 \mid y_1, y_3 = f^N(y_1, y_3, \hat{w}_{2.13}) + e_2, \tag{2.6.11}$$

$$\hat{y}_3 \mid y_1, y_2 = f^N(y_1, y_2, \hat{w}_{3.12}) + e_3, \tag{2.6.12}$$

e_2 and e_3 are random draws from the derived distribution. For example, they can be the draws from $N(0, \hat{\sigma}_2^2)$ and $N(0, \hat{\sigma}_3^2)$.

As shown in the above results, without auxiliary variables, it is not an easy task to specify a neural networks model that can impute the missing values in multiple variables simultaneously. Several independent neural networks corresponding different variables have to be build to impute the missing values respectively.

2.7 Wald Error Neural Networks

Suppose $\mathbf{y}_i$ follows a normal distribution $N(f(\mathbf{x}_i,W),\Sigma)$, then the Wald statistic for testing the goodness of fit can be obtained as (Wald, 1948, Lindsey, 1996):

$$\sum_i (\mathbf{y}_i - f(X_i,\hat{W}))^T \hat{\Sigma}^{-1} \sum_i (\mathbf{y}_i - f(X_i,\hat{W})) \sim \chi^2_p, \tag{2.7.1}$$

where

$$\hat{\Sigma} = \frac{1}{m}\sum_{i=1}^{m} (\mathbf{y}_i - f(X_i,W))(\mathbf{y}_i - f(X_i,W))^T. \tag{2.7.2}$$

Here $\mathbf{y}_i$ is the vector notation of the ith observation of a categorical variable. The chi-square distribution in (2.7.1) is obtained asymptotically based on log-likelihood of $\mathbf{y}_i$s. This result is valid if the normal distribution approximation is justified. The expression in (2.7.1) can be used to evaluate the consistency of the marginal distributions of the true values and the imputed values. If we train the neural network by minimising the *Wald* statistics, the *Wald Statistic Criterion* is obtained. Namely the weights of neural networks can be estimated by minimising

$$E_W = \sum_{i=1}^{m} (\mathbf{y}_i - f(X_i,W))\hat{\Sigma}^{-1} \sum_{i=1}^{m} (\mathbf{y}_i - f(X_i,W))^T, \tag{2.7.3}$$

For convenience, we denote the *Wald* statistics in matrix format.

Let $\mathbf{G} = \begin{bmatrix} y_{11} - f_1(X_1, W) & \cdots & y_{1,p+1} - f_{p+1}(X_1, W) \\ \vdots & \vdots & \vdots \\ y_{m1} - f_1(X_m, W) & \cdots & y_{m,p+1} - f_{p+1}(X_m, W) \end{bmatrix}_{m \times (p+1)}$

$$= \begin{bmatrix} (\mathbf{y}_1 - \mathbf{f}(X_1, W))^T \\ \vdots \\ (\mathbf{y}_m - \mathbf{f}(X_m, W))^T \end{bmatrix} = \begin{bmatrix} \mathbf{y}_1^T \\ \vdots \\ \mathbf{y}_m \end{bmatrix} - \begin{bmatrix} \mathbf{f}(X_1, W)^T \\ \vdots \\ \mathbf{f}(X_m, W)^T \end{bmatrix}.$$

Denote $\mathbf{G}_1 = \begin{bmatrix} \mathbf{y}_1^T \\ \vdots \\ \mathbf{y}_m \end{bmatrix}$, and $\mathbf{G}_2 = \begin{bmatrix} \mathbf{f}(X_1, W)^T \\ \vdots \\ \mathbf{f}(X_m, W)^T \end{bmatrix}$, then $\mathbf{G}=\mathbf{G}_1-\mathbf{G}_2$.

Then,

$$E_W \propto \mathbf{1}^T \mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{1}. \tag{2.7.4}$$

According to Newton-Raphson algorithm, the parameter W can be estimated by the following iterative equation.

$$W^{t+1} = W^t - \mathbf{H}^{-1}(W^t)\mathbf{J}(W^t), \tag{2.7.5}$$

where $\mathbf{H}^{-1}(W^t)$ is the Hessian matrix (second order derivative matrix) of E_W, $\mathbf{J}(W^t)$ is the first order derivative vector. Using matrix derivative rules, the first and second derivatives can be obtained as follows.

$$\frac{\partial \mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T}{\partial w_{ij}} = \frac{\partial \mathbf{G}}{\partial w_{ij}}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T + \mathbf{G}\frac{\partial (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T}{\partial w_{ij}}$$

$$= \frac{\partial \mathbf{G}}{\partial w_{ij}}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T + \mathbf{G}\frac{\partial (\mathbf{G}^T\mathbf{G})^{-1}}{\partial w_{ij}}\mathbf{G}^T + \mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{ij}}. \tag{2.7.6}$$

$$\frac{\partial (\mathbf{G}^T\mathbf{G})^{-1}}{\partial w_{ij}} = -(\mathbf{G}^T\mathbf{G})^{-1}(\frac{\partial \mathbf{G}^T}{\partial w_{ij}}\mathbf{G} + \mathbf{G}^T\frac{\partial \mathbf{G}}{\partial w_{ij}})(\mathbf{G}^T\mathbf{G})^{-1}. \tag{2.7.7}$$

Plugging (2.7.7) into (2.7.6), we obtain,

$$\frac{\partial E_W}{\partial w_{ij}} \propto 2\mathbf{1}^{\mathbf{T}}\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{ij}}(\mathbf{I}-\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T)\mathbf{1}, \tag{2.7.8}$$

$$\frac{\partial^2 E_W}{\partial w_{ij}\partial w_{kl}} \propto 2\mathbf{1}^{\mathbf{T}}(\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial^2 \mathbf{G}^T}{\partial w_{ij}\partial w_{kl}}+\frac{\partial \mathbf{G}}{\partial w_{kl}}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{ij}}$$

$$-2\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{kl}}\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{ij}}-2\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{ij}}\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\frac{\partial \mathbf{G}^T}{\partial w_{kl}})\bullet$$

$$(\mathbf{I}-\mathbf{G}(\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T)\mathbf{1}, \tag{2.7.9}$$

where

$$\frac{\partial \mathbf{G}}{\partial w_{ij}} = -\frac{\partial \mathbf{G}_2}{\partial w_{ij}},$$

$$\frac{\partial^2 \mathbf{G}}{\partial w_{ij}\partial w_{lk}} = -\frac{\partial^2 \mathbf{G}_2}{\partial w_{ij}\partial w_{lk}}.$$

The main task is to calculate the derivatives of $f(X,W)$ with respect to W. The RBF and MLP neural networks are considered. The transformation function (or activation function) in the terminal layer is a softmax function (see 2.3.6). Suppose there are M centres in RBF model and M hidden nodes in MLP model respectively.

RBF

$$\frac{\partial f_{ik}}{\partial w_{l1,l2}} = \begin{cases} -\phi_{l1}(\mathbf{x}_i)f_{ik}f_{il2}, k \neq l_2 \\ \phi_{l1}(\mathbf{x}_i)f_{ik}(1-f_{ik}), k = l_2 \end{cases}, \; i=1...m,\; k,l_2=1...p+1,\; l_1=1...M. \tag{2.7.10}$$

$$\frac{\partial^2 f_{ik}}{\partial w_{l1,l2}\partial w_{l3,l4}} = \phi_{l1}(\mathbf{x}_i)\phi_{l3}(\mathbf{x}_i)\begin{cases} f_{ik}(1-f_{ik})(1-2f_{ik}), k = l_2 = l_4 \\ -f_{ik}f_{il4}(1-2f_{ik}), k = l_2, l_2 \neq l_4 \\ -f_{ik}f_{il2}(1-2f_{ik}), k \neq l_2, k = l_4 \\ -f_{ik}f_{il2}(1-2f_{il4}), k \neq l_2, l_2 = l_4 \\ 2f_{ik}f_{il2}f_{il4}), k \neq l_2, l_2 \neq l_4, k \neq l_4 \end{cases},$$

$$i=1...m,\; k,l_2=1...p+1,\; l_1=1...M. \tag{2.7.11}$$

MLP

$$\frac{\partial f_{ik}}{\partial w^{(2)}_{l1,l2}} = \begin{cases} -g^{(1)}_{l1}(\mathbf{x}_i, W) f_{ik} f_{il2}, k \neq l_2 \\ g^{(1)}_{l1}(\mathbf{x}_i, W) f_{ik}(1 - f_{ik}), k = l_2 \end{cases}, i=1...m,\ k,l_2=1...p+1,\ l_1=1...M. \quad (2.7.12)$$

$$\frac{\partial^2 f_{ik}}{\partial w^{(2)}_{l1,l2} \partial w^{(2)}_{l3,l4}} = g^{(1)}_{l1}(\mathbf{x}_i, W) g^{(1)}_{l3}(\mathbf{x}_i, W) \begin{cases} f_{ik}(1 - f_{ik})(1 - 2f_{ik}), k = l_2 = l_4 \\ -f_{ik} f_{il4}(1 - 2f_{ik}), k = l_2, l_2 \neq l_4 \\ -f_{ik} f_{il2}(1 - 2f_{ik}), k \neq l_2, k = l_4 \\ -f_{ik} f_{il2}(1 - 2f_{il4}), k \neq l_2, l_2 = l_4 \\ 2f_{ik} f_{il2} f_{il4}), k \neq l_2, l_2 \neq l_4, k \neq l_4 \end{cases},$$

$i=1...m,\ k,l_2=1...p+1,\ l_1=1...M.$ (2.7.13)

$$\frac{\partial f_{ik}}{\partial w^{(1)}_{l1,l2}} = x_{il1} g^{(1)}_{l1}(\mathbf{x}_i, W)(1 - g^{(1)}_{l1}(\mathbf{x}_i, W)) f_{ik}(w^{(2)}_{l2,k} - \sum_{j=1}^{p+1} w^{(2)}_{l2,j} f_{ij}),$$

$i=1...m,\ k=1...p+1,\ l_1=1...q+1,\ l_2=1...M.$ (2.7.14)

$$\frac{\partial^2 f_{ik}}{\partial w^{(1)}_{l1,l2}\partial w^{(1)}_{l3,l4}} = \begin{cases} x_{il1}x_{il3}g^{(1)}_{l2}(1-g^{(1)}_{l2})f_{ik}[(1-2g^{(1)}_{l2}+g^{(1)}_{l2}(1-g^{(1)}_{l2})(w^{(2)}_{l2,k}-\sum_{j=1}^{p+1} w^{(2)}_{l2,j}f_{ij}))(w^{(2)}_{l2,k} \\ -\sum_{j=1}^{p+1} w^{(2)}_{l2,j}f_{ij})-g^{(1)}_{l2}(1-g^{(1)}_{l2})\sum_{j=1}^{p+1}(w^{(2)2}_{l2,j}-\sum_{jj=1}^{p+1} w^{(2)}_{l2,j}w^{(2)}_{l2,jj}f_{ijj})],l_2=l_4 \\ x_{il1}x_{il3}g^{(1)}_{l2}g^{(1)}_{l4}f_{ik}[(w^{(2)}_{l2,k}-\sum_{j=1}^{p+1} w^{(2)}_{l2,j}f_{ij})(w^{(2)}_{l4,k}-\sum_{j=1}^{p+1} w^{(2)}_{l4,j}f_{ij})- \\ \sum_{j=1}^{p+1} w^{(2)}_{l2,j}f_{ij}(w^{(2)}_{l4,j}-\sum_{jj=1}^{p+1} w^{(2)}_{l4,jj}f_{ijj})],l_2\neq l_4 \end{cases} \tag{2.7.15}$$

$$\frac{\partial^2 f_{ik}}{\partial w^{(1)}_{l_1,l_2}\partial w^{(2)}_{l_3,l_4}} = \begin{cases} x_{il3}g^{(1)}_{l1}(1-g^{(1)}_{l1})f_{ik}[1-f_{ik}+(1-2f_{ik})(w^{(2)}_{l1,k}-\sum_{j=1}^{p+1} w^{(2)}_{l1,j}f_{ij})],k=l_2,l_1=l_4 \\ x_{il3}g^{(1)}_{l1}g^{'(1)}_{l4}f_{ik}(1-2f_{ik})(w^{(2)}_{l4,k}-\sum_{j=1}^{p+1} w^{(2)}_{l4,j}f_{ij})],k=l_2,l_1\neq l_4 \\ -x_{il3}g^{'(1)}_{l4}f_{ik}f_{il2}[1+g^{(1)}_{l1}(w^{(2)}_{l4,k}+w^{(2)}_{l4,l2}-2\sum_{j=1}^{p+1} w^{(2)}_{l4,j}f_{ij})],k=l_2,l_1=l_4 \\ -x_{il3}g^{(1)}_{l1}g^{'(1)}_{l4}f_{ik}f_{il2}(w^{(2)}_{l4,k}+w^{(2)}_{l4,l2}-2\sum_{j=1}^{p+1} w^{(2)}_{l4,j}f_{ij}),k=l_2,l_1\neq l_4 \end{cases} \tag{2.7.16}$$

As shown in the above formulas, the Wald error function is more sophisticated than sum-square-error and entropy error. It may indicate that more computing power is need to obtain $\hat{W}$.

2.8 Linear Approximation to Neural Network Imputation

One of the disadvantages of neural networks imputation is time-consuming training process. To deal with this problem, a linear approximation is developed. The EM algorithm can be used to estimate weights (Schafer, 1997).

Here only the simplest case p=1 is listed. The multivariate expansion has the similar expression.

The following series are used in the expansion:

$$\frac{1}{1-x} = 1 + x + x^2 + x^3 + \cdots, \ |x|<1. \tag{2.8.1}$$

$$\frac{1}{1-x} = -\frac{1}{x} - \frac{1}{x^2} - \frac{1}{x^3} - \cdots, \ |x|>1. \tag{2.8.2}$$

$$exp(x) = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots, \ x \in R. \tag{2.8.3}$$

Then the logistic sigmoid function defined in (2.3.1) may be expanded as follows

$$g_1(x) = \frac{1}{1+\exp(-x)}$$

$$= \begin{cases} 1 - \exp(-x) + \exp(-2x) - \cdots, & x > 0 \\ \exp(x) - \exp(2x) + \exp(3x) - \cdots, & x < 0 \end{cases}$$

$$= a_0 + a_1 x + a_2 x^2 + \cdots, \tag{2.8.4}$$

where $a_0, a_1, \ldots, \in R$, $a_n \xrightarrow{n \to \infty} 0$. Plugging (2.8.4) into (2.3.3), the MLP function $f(x_i, W)$ can be written as,

$$f(x_i, W) = I_{(W^{(1)}xi>0, W^{(2)}g(W^{(1)}xi)>0)} f(x_i, W)$$

$$+ I_{(W^{(1)}xi>0, W^{(2)}g(W^{(1)}xi)<0)} f(x_i, W)$$

$$+ I_{(W^{(1)}xi<0, W^{(2)}g(W^{(1)}xi)>0)} f(x_i, W)$$

$$+ I_{(W^{(1)}xi<0, W^{(2)}g(W^{(1)}xi)<0)} f(x_i, W)$$

$$= b_0 + b_1 x_i + b_3 x_i^2 + \cdots, \tag{2.8.5}$$

where $b_n \to 0$. For convenience in the above expression the covariate x is assumed to be scalar. If we neglect the infinitesimal terms, it turns out to be a polynomial expression. We can increase the order of polynomial equation to get the closest model to MLP in terms of likelihood ratio. Since the MLP expression in (2.8.5) is linear in coefficients, the

training process will be simplified by using the linear regression routine. The potential difficulty in this approach is to decide where to cut the expansion series. If two few terms are used, the approximation may be too distant from the original MLP function. On the other hand, too many terms may lead to near saturated model when the training data is small. The decision is really a trade-off between precision and the computing efficiency.

For the multivariate **x**, MLP can be expressed as:

$$f(\mathbf{x}_i, W) = b_0 + \sum_{j=1}^{p} b_{1j} x_{ij} + \sum_{j,k=1}^{p} b_{2jk} x_{ij} x_{ik} + \cdots. \tag{2.8.6}$$

One of the disadvantages of neural networks is the black-box feature of model expression. It makes the explanation of model properties difficult. For example, in linear models, the main effects and interactions can be explained by the corresponding coefficients. Unfortunately in neural networks there are no parameters directly relating to main effects and interactions. One way to extract the main effects and interactions from trained neural networks is to feed the neural network with corresponding 0-1 input. For example, if the j^{th} covariate variable is concerned, the vector for the covariates is (0...1...0) with the j^{th} element equals 1. The network output reflects the main effect of j^{th} covariate variable. We denote it by α_j.

$$\alpha_j = f(0\cdots \overset{j^{th}}{1} \cdots 0), j=1,\ldots,p$$

$$\alpha_{ij} = f(0\cdots \overset{i^{th}}{1} \cdots \overset{j^{th}}{1} \cdots 0) - f(0\cdots \overset{i^{th}}{1} \cdots 0) - f(0\cdots \overset{j^{th}}{1} \cdots 0), i,j=1,\ldots,p$$

The systematic part of the MLP model is a continuous smooth function which has any order derivatives. To investigate the variance of MLP prediction, the first order approximation is developed.

$$f(X,W) \approx f(X,W^0) + VW, \tag{2.8.7}$$

where W^0 is the estimator of W from neural network training, $V_{n\times q}$ is the first order derivative of $f(X,W)$, q is the number of weights. $W_q=(W^{(1)T}, W^{(2)T})^T$,

$$V_{i,j} = \frac{\partial f(\mathrm{x}_i, W)}{\partial w_j},\ i=1,\ldots,n, j=1,\ldots, q. \tag{2.8.8}$$

$$V_{i,w_{ji}^{(1)}} = \frac{\partial f(\mathrm{x}_i, W)}{\partial w_{ji}^{(1)}} = \frac{w_j^{(2)} x_{ij} \exp(-w_0^{(2)} - \sum_{j=1}^{M} w_j^{(2)} g_1(\sum_{i=1}^{p} w_{ji}^{(1)} x_{ij})}{(1+\exp(-w_0^{(2)} - \sum_{j=1}^{M} w_j^{(2)} g_1(\sum_{i=1}^{p} w_{ji}^{(1)} x_i))^2} \cdot \frac{\exp(-\sum_{i=1}^{p} w_{ji}^{(1)} x_{ij})}{(1+\exp(-\sum_{i=1}^{p} w_{ji}^{(1)} x_{ij}))^2} \cdot$$

$$V_{i,w_j^{(2)}} = \frac{\partial f(\mathrm{x}_i, W)}{\partial w_j^{(2)}} = \frac{\exp(-w_0^{(2)} - \sum_{j=1}^{M} w_j^{(2)} g_1(\sum_{i=1}^{p} w_{ji}^{(1)} x_{ij})}{(1+\exp(-w_0^{(2)} - \sum_{j=1}^{M} w_j^{(2)} g_1(\sum_{i=1}^{p} w_{ji}^{(1)} x_i))^2} \cdot g_1(\sum_{i=1}^{p} w_{ji}^{(1)} x_{ij}) \cdot$$

Then

$$Y - f(X, W^0) = VW + \varepsilon. \tag{2.8.9}$$

The LSE of W is

$$\hat{W} = (V^T V)^{-1} V^T Z, \tag{2.8.10}$$

where $Z = Y - f(X, W^0)$.

Then the EM-like method (Schafer, 1997) can be used to estimate W.

Step 1: Generate initial estimates of W,

Step 2: Impute missing values in Y by $f(X,W) \approx f(X,W^0) + VW$,

Step 3: Estimate W by $\hat{W} = (V^T V)^{-1} V^T Z$,

Step 4: Go to step 2 and repeat until converge.

The above procedure is an exact EM method if the normality assumption about Y is true. Otherwise $\hat{W}$ in step 3 may not be the estimator based on the likelihood of Y. The EM method may complicate the benefit of the linear approximation given in (2.8.9), since it may be as enduring as neural networks. After all it provides an alternative to neural networks method. It is possible that the convergence of EM method based on the linear approximation is faster than that of neural networks in some circumstances.

3 Nearest Neighbour Imputation with Weighted Distance

We discussed nearest neighbour imputation in section 1.4. The performance of this kind of imputation relies on the distance measurement. In this chapter we introduce an alternative distance measurement, weighted distance, to deal with data sets with more than one covariate variable in a record. When there are multiple covariate variables, the individual covariate may have unequal contribution in defining the overall distance between two observations. Therefore assigning a weight to each covariate may provide a way of improving distance measurement. Standard nearest neighbour imputation either assumes that the covariates are equally important in calculating distance, or defines a distance measure in an arbitrary non-data-dependent way.

3.1 Multiple Variables and Weighted Distance

Suppose the matrix X containing the n records of q covariates. Let $\mathbf{x}_i$ and $\mathbf{x}_j$ be the vector of values of i^{th} record and j^{th} record. The distance between these two records is defined as d_{ij}

$$d_{ij} = d_W(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{q} w_k d_k(x_{ik}, x_{jk}) = D(\mathbf{x}_i, \mathbf{x}_j)^T W, \qquad (3.1.1)$$

where $D(\mathbf{x}_i, \mathbf{x}_j)^T = (d_1(x_{i1}, x_{j1}), ..., d_q(x_{iq}, x_{jq}))$. The vector W is the weight vector that remains to be determined. $d_k(x_{ik}, x_{jk})$ is the distance metric for k^{th} covariate and is assumed given. Usually the weight w_k k=1,...,q are assumed to be given. In our approach we consider choosing the w_k to improve prediction. The distances between all possible pairs of records in each variable are put into the following matrix $D(X)$,

$$D(X)=\begin{bmatrix} d_1(x_{2,1},x_{1,1}) & \cdots & d_q(x_{2,q},x_{1,q}) \\ \vdots & \vdots & \vdots \\ d_1(x_{m,1},x_{m-1,1}) & \cdots & d_q(x_{m,q},x_{m-1,q}) \end{bmatrix}_{\frac{m(m-1)}{2}\times q}. \tag{3.1.2}$$

Similarly, a distance matrix of y is considered as,

$$D(Y)=\begin{bmatrix} d(y_2,y_1) \\ \vdots \\ d(y_m,y_{m-1}) \end{bmatrix}_{\frac{m(m-1)}{2}\times 1}, \tag{3.1.3}$$

where $d(y_i,y_j)$ is a given measure of the distance between y_i and y_j. We consider the following model to determine W.

$$D(Y)=f(D(X),W)+\varepsilon, \tag{3.1.4}$$

where ε is a residual term. In the special case of $f(D(X),W)=D(X)W$, the OLS estimator of W can be obtained as follows

$$\hat{W}=(D(X)^T D(X))^{-1} D(X)^T D(Y). \tag{3.1.5}$$

If relationship between $D(X)$ and W is not linear, more flexible models such as neural networks may give better estimate of W.

To help understand $\hat{W}$ in (3.1.5), we consider its form under some simplifying assumptions. Here we consider the special case of multivariate normal distribution, and seek the asymptotic value of $\hat{W}$. Specifically we assume that $\mathbf{x}$ and y are realisations of a joint normal distribution $N(\mu,\Sigma)$. For simplicity we let q=2. Then, we write

$$(\mathbf{x},y)\sim N(\mu,\Sigma), \tag{3.1.6}$$

$$\boldsymbol{\mu} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{bmatrix} = E\begin{bmatrix} \mathbf{x} \\ y \end{bmatrix}, \tag{3.1.7}$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_2^2 & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_3^2 \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho_{12} & \sigma_1\sigma_3\rho_{13} \\ \sigma_1\sigma_2\rho_{12} & \sigma_2^2 & \sigma_2\sigma_3\rho_{23} \\ \sigma_1\sigma_3\rho_{13} & \sigma_2\sigma_3\rho_{23} & \sigma_3^2 \end{bmatrix}, \tag{3.1.8}$$

$$\mathrm{var}(x_1) = \sigma_1^2, \mathrm{var}(x_2) = \sigma_2^2, \mathrm{var}(y) = \sigma_3^2, \tag{3.1.9}$$

$$\rho_{12} = \frac{\mathrm{cov}(x_1, x_2)}{\sqrt{\mathrm{var}(x_1)\,\mathrm{var}(x_2)}}, \rho_{13} = \frac{\mathrm{cov}(x_1, y)}{\sqrt{\mathrm{var}(x_1)\,\mathrm{var}(y)}}, \rho_{23} = \frac{\mathrm{cov}(x_2, y)}{\sqrt{\mathrm{var}(x_2)\,\mathrm{var}(y)}}. \tag{3.1.10}$$

To obtain an expression for $\hat{W}$ we need the detailed versions of $(D(X)^T D(X))^{-1}$ and $D(X)^T D(Y)$. For simplicity, we choose $d(y_i,y_j)=(y_i-y_j)^2$ and $d_k(x_{ik},x_{jk})=(x_{ik}-x_{jk})^2$.

$$D(X)^T D(X) = \begin{bmatrix} \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^4 & \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(x_{i2}-x_{j2})^2 \\ \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(x_{i2}-x_{j2})^2 & \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i2}-x_{j2})^4 \end{bmatrix}, \tag{3.1.11}$$

$$(D(X)^T D(X))^{-1} = \Big[\Big(\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^4\Big)\Big(\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i2}-x_{j2})^4\Big) - \Big(\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(x_{i2}-x_{j2})^2\Big)^2 \Big]^{-1}$$

$$\begin{bmatrix} \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i2}-x_{j2})^4 & -\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(x_{i2}-x_{j2})^2 \\ -\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(x_{i2}-x_{j2})^2 & \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^4 \end{bmatrix}, \tag{3.1.12}$$

$$D(X)^T D(Y) = \begin{bmatrix} \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(y_i-y_j)^2 \\ \sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i2}-x_{j2})^2(y_i-y_j)^2 \end{bmatrix}. \tag{3.1.13}$$

With the normality assumption, the following asymptotic results are obtained.

$$\frac{2}{m(m-1)}\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^4 \xrightarrow{m\to\infty} 3\sigma_1^4, \tag{3.1.14}$$

$$\frac{2}{m(m-1)}\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i2}-x_{j2})^4 \xrightarrow{m\to\infty} 3\sigma_2^4, \tag{3.1.15}$$

$$\frac{2}{m(m-1)}\sum_{i=2}^{m}\sum_{j=1}^{i-1}(y_i-y_j)^4 \xrightarrow{m\to\infty} 3\sigma_3^4, \tag{3.1.16}$$

$$\frac{2}{m(m-1)}\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(x_{i2}-x_{j2})^2 \xrightarrow{m\to\infty} \sigma_1^2\sigma_2^2+2\sigma_{12}^2, \tag{3.1.17}$$

$$\frac{2}{m(m-1)}\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i1}-x_{j1})^2(y_i-y_j)^2 \xrightarrow{m\to\infty} \sigma_1^2\sigma_3^2+2\sigma_{13}^2, \tag{3.1.18}$$

$$\frac{2}{m(m-1)}\sum_{i=2}^{m}\sum_{j=1}^{i-1}(x_{i2}-x_{j2})^2(y_i-y_j)^2 \xrightarrow{m\to\infty} \sigma_2^2\sigma_3^2+2\sigma_{23}^2. \tag{3.1.19}$$

The last three formulas are obtained using the theorem of the variable decomposition of multinormal distribution, by which $\mathbf{x}_1$ and $\mathbf{x}_2-\Sigma_{21}\Sigma_{11}^{-1}\mathbf{x}_1$ are independent (Anderson,1984). Plugging (3.1.14) to (3.1.19) in (3.1.11) and (3.1.13), the following asymptotic values are obtained.

$$D(X)^T D(X) \approx \frac{m(m-1)}{n}\begin{bmatrix} 3\sigma_1^4 & \sigma_1^2\sigma_2^2+2\sigma_{12}^2 \\ \sigma_1^2\sigma_2^2+2\sigma_{12}^2 & 3\sigma_2^4 \end{bmatrix}, \tag{3.1.20}$$

$$D(X)^T D(Y) \approx \frac{m(m-1)}{n}\begin{bmatrix} \sigma_1^2\sigma_2^2+2\sigma_{13}^2 \\ \sigma_2^2\sigma_3^2+2\sigma_{23}^2 \end{bmatrix}. \tag{3.1.21}$$

Plugging (3.1.20) and (3.1.21) into (3.1.5), the asymptotic value of $\hat{W}$ is obtained.

$$\hat{W} \xrightarrow{m\to\infty} \begin{bmatrix} \dfrac{\frac{\sigma_3^2}{\sigma_1^2}(1+3\rho_{13}^2-\rho_{23}^2-\rho_{12}^2-2\rho_{12}^2\rho_{23}^2)}{4-2\rho_{12}^2-2\rho_{12}^4} \\ \dfrac{\frac{\sigma_3^2}{\sigma_2^2}(1+3\rho_{23}^2-\rho_{13}^2-\rho_{12}^2-2\rho_{12}^2\rho_{13}^2)}{4-2\rho_{12}^2-2\rho_{12}^4} \end{bmatrix}. \tag{3.1.22}$$

Imagine the situation that principle component analysis is employed initially to produce new variables x_1 and x_2. The resulting variables are mutually independent. Assuming such an approach is used, we set ρ_{12}=0. The above result can then be further simplified as

$$\hat{W} \xrightarrow{m\to\infty} \begin{bmatrix} \frac{\frac{\sigma_3^2}{\sigma_1^2}(1+3\rho_{13}^2-\rho_{23}^2)}{4} \\ \frac{\frac{\sigma_3^2}{\sigma_2^2}(1+3\rho_{23}^2-\rho_{13}^2)}{4} \end{bmatrix} = W. \qquad (3.1.23)$$

The new distance based on the weight given above takes the variances of the component variables in to account. It eliminates the effect of scale, and gives more weight to the one with higher correlation with the dependent variable.

This approach can be extended to more sophisticated models if the linear assumption is not sufficient. For example a neural network model can be used instead. Then, the aim to is to fit the following model

$$D(Y) = f(D(X), W) + \mathbf{e}. \qquad (3.1.24)$$

$f(D(X), W)$ could be approximated by either RBF or MLP. The potential difficulty is that this will train a big neural network, and the searching time could be much longer than conventional Euclidean distances. The practical way needs to be developed. One of the possible solutions is the iterative algorithm, by which the weight from the linear regression model is used as the initial values, and is iteratively adjusted by cross validation. The distribution of $D(Y)$ is described in next section.

3.2 A Comparison of Weighted Distance Nearest Neighbour Imputation and Other Distance-based Imputations

In this section we compare the nearest neighbour imputation based on the weighted distance with other distance based imputations such as the conventional nearest neighbour imputation with Euclidean distance, the nearest neighbour imputation based on Mahalanobis distance, and predictive mean match imputation. The main difference lies in how they measure the distance between two data points. This comparison is based on the normality assumption given in previous section.

3.2.1 Nearest Neighbour Imputation with Euclidean Distance

The Euclidean distance of the i^{th} unit and j^{th} unit can be written as $(x_{i1}-x_{j1})^2+(x_{i2}-x_{j2})^2$. If we set $W=(1,1)^T$, the weighted distance becomes the standard Euclidean distance. Therefore Euclidean distance is a special case of the weighted distance we proposed. In practice the raw data is standardised before it is used for imputation. In this situation the Euclidean distance between the i^{th} unit and j^{th} unit can be denoted as

$$\frac{(x_{i1}-x_{j1})^2}{\sigma_1^2}+\frac{(x_{i2}-x_{j2})^2}{\sigma_2^2}. \qquad (3.2.1)$$

It is the summation of the squared differences between the standardised values of the auxiliary variables of the two units. No reliance on the variable to be imputed is involved, but it is widely used and in many situations it is good enough. If we set $W=(\frac{1}{\sigma_1^2},\frac{1}{\sigma_2^2})$, the weighted distance becomes the Euclidean distance given in (3.2.1). Therefore it is also a special case of weighted distance.

3.2.2 Mahalanobis Distance

The Euclidean distance can be extended to the Mahalanobis Distance, which is defined as

$$(\mathbf{x}_i - \mathbf{x}_j)^T V^{-1}(\mathbf{x}_i - \mathbf{x}_j), \tag{3.2.2}$$

where $\mathbf{x}_i^T = (x_{i1}, x_{i2})$, $\mathbf{x}_j^T = (x_{j1}, x_{j2})$, V=cov($\mathbf{x}_i$, $\mathbf{x}_j$)=$\begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}$. Plugging in the inverse of V, the following result is obtained,

$$\begin{aligned}
&(\mathbf{x}_i - \mathbf{x}_j)^T V^{-1}(\mathbf{x}_i - \mathbf{x}_j) \\
&= (\sigma_1^2\sigma_2^2 - \sigma_{12}^2)^{-1}(\sigma_2^2(x_{i1} - x_{j1})^2 - 2\sigma_{12}(x_{i1} - x_{j1})(x_{i2} - x_{j2}) + \sigma_1^2(x_{i2} - x_{j2})^2) \\
&= (1 - \rho_{12}^2)^{-1}[(\frac{(x_{i1} - x_{j1})^2}{\sigma_1^2} - 2\rho_{12}(x_{i1} - x_{j1})(x_{i2} - x_{j2}) + \frac{(x_{i2} - x_{j2})^2}{\sigma_2^2})] \\
&\propto (\frac{(x_{i1} - x_{j1})^2}{\sigma_1^2} - 2\rho_{12}(x_{i1} - x_{j1})(x_{i2} - x_{j2}) + \frac{(x_{i2} - x_{j2})^2}{\sigma_2^2}).
\end{aligned} \tag{3.2.3}$$

The constant $(1-\rho_{12}^2)^{-1}$ has no effect in searching for the nearest neighbour. Similar to Euclidean distance, the Mahalanobis distance does not depend on the variable to be imputed, but it takes the correlation between the two component variables in to account. If the imputation is carried after data preparation by principal component analysis, the x_{i1} x_{i2} x_{j1} x_{j2} are the transformed variables, and are independent (ρ_{12}=0). The Mahalanobis distance is then equivalent to the Euclidean distance. Therefore in practice the performance of an imputation method can be affected by the way the raw data is pre-processed.

3.2.3 Predictive Mean Matching

There is a similar approach to nearest neighbour imputation based on regression, which selects the imputation by nearest predictive mean (Little, 1987). If we take it as another version of distance, it becomes a variation of nearest neighbour imputation. Following the

previous notation the distance between the i^{th} unit and j^{th} unit based on the predictive mean can be denoted as

$$[(x_{i1}-x_{j1})\beta_1+(x_{i2}-x_{j2})\beta_2]^2$$
$$=(x_{i1}-x_{j1})^2\beta_1^2+2\beta_1\beta_2(x_{i1}-x_{j1})(x_{i2}-x_{j2})+(x_{i2}-x_{j2})^2\beta_2^2, \qquad (3.2.4)$$

where β_1 and β_2 are the regression coefficients of y on x_1 and x_2. Under the normality assumption, they are the function of the population parameters.

$$\beta_1^2=\frac{\sigma_3^2}{\sigma_1^2}(\rho_{13}-\rho_{12}\frac{\rho_{23}(1-\rho_{12}\rho_{13})}{1-\rho_{12}^2})^2, \qquad (3.2.5)$$

$$\beta_2^2=\frac{\sigma_3^2}{\sigma_2^2}\frac{\rho_{23}^2(1-\rho_{12}\rho_{13})^2}{(1-\rho_{12}^2)^2}. \qquad (3.2.6)$$

The constant term β_0 is omitted, because it is cancelled by the subtraction in the distance equation. This distance involves both the variance and the correlation of the covariates and the variable to be imputed. If ρ_{12}=0, the distance becomes,

$$(x_{i1}-x_{j1})^2\frac{\sigma_3^2}{\sigma_1^2}\rho_{13}^2+(x_{i2}-x_{j2})^2\frac{\sigma_3^2}{\sigma_2^2}\rho_{23}^2+2\frac{\sigma_3^2}{\sigma_1\sigma_2}\rho_{13}\rho_{23}(x_{i1}-x_{j1})(x_{i2}-x_{j2}). \qquad (3.2.7)$$

When it is employed to search the nearest neighbour, it is equivalent to

$$\frac{(x_{i1}-x_{j1})^2}{\sigma_1^2}\rho_{13}^2+\frac{(x_{i2}-x_{j2})^2}{\sigma_2^2}\rho_{23}^2+2\rho_{13}\rho_{23}\frac{(x_{i1}-x_{j1})}{\sigma_1}\cdot\frac{(x_{i2}-x_{j2})}{\sigma_2}. \qquad (3.2.8)$$

The predictive mean matching is a modified Mahalanobis distance that gives more weight to the one that is more correlated to the variable to be imputed.

3.2.4 Weighted Distance

Nearest neighbour imputation with weighted distance imputes the missing value by the corresponding value of its nearest neighbour in terms of weighted distance. In the case of q=2, the distance between the i^{th} unit and j^{th} unit is as follows,

$$(x_{i1}-x_{j1})^2 w_1+(x_{i2}-x_{j2})^2 w_2. \tag{3.2.9}$$

When the above expression is used for imputation, it is equivalent to $c(x_{i1}-x_{j1})^2 w_1+c(x_{i2}-x_{j2})^2 w_2$, where c is a non-zero constant, since c only changes the scale of distance, it has no influence on the relative distance between two units. For simplicity we multiply it by four (c=4), it then becomes,

$$(x_{i1}-x_{j1})^2 \frac{\sigma_3^2}{\sigma_1^2}(1+3\rho_{13}^2-\rho_{12}^2-\rho_{23}^2+2\rho_{12}^2\rho_{23}^2)+$$

$$(x_{i2}-x_{j2})^2 \frac{\sigma_3^2}{\sigma_2^2}(1+3\rho_{23}^2-\rho_{12}^2-\rho_{13}^2+2\rho_{12}^2\rho_{13}^2). \tag{3.2.10}$$

If x_{i1} x_{i2} x_{j1} x_{j2} are the transformed variables by principal component analysis, they are independent, therefore ρ_{12}=0. The above distance then becomes,

$$(x_{i1}-x_{j1})^2 \frac{\sigma_3^2}{\sigma_1^2}(1+3\rho_{13}^2-\rho_{23}^2+(x_{i2}-x_{j2})^2 \frac{\sigma_3^2}{\sigma_2^2}(1+3\rho_{23}^2-\rho_{13}^2)$$

$$=(x_{i1}-x_{j1})^2 \frac{\sigma_3^2}{\sigma_1^2}+(x_{i2}-x_{j2})^2 \frac{\sigma_3^2}{\sigma_2^2}+2[(x_{i1}-x_{j1})^2 \frac{\sigma_3^2}{\sigma_1^2}\rho_{13}^2+(x_{i2}-x_{j2})^2 \frac{\sigma_3^2}{\sigma_2^2}\rho_{23}^2]+$$

$$\sigma_3^2[\frac{(x_{i1}-x_{j1})^2}{\sigma_1^2}-\frac{(x_{i2}-x_{j2})^2}{\sigma_2^2}](\rho_{13}^2-\rho_{23}^2)$$

$$\propto \frac{(x_{i1}-x_{j1})^2}{\sigma_1^2}+\frac{(x_{i2}-x_{j2})^2}{\sigma_2^2}+2\{\frac{(x_{i1}-x_{j1})^2}{\sigma_1^2}\rho_{13}^2+\frac{(x_{i2}-x_{j2})^2}{\sigma_2^2}\rho_{23}^2\}+$$

$$[\frac{(x_{i1}-x_{j1})^2}{\sigma_1^2}-\frac{(x_{i2}-x_{j2})^2}{\sigma_2^2}](\rho_{13}^2-\rho_{23}^2). \tag{3.2.11}$$

In some circumstances the third term could be small enough to neglect, for example if $x_{i2}=\alpha+\beta x_{i1}+\varepsilon_i^*$, $\varepsilon_i^* \sim N(0,\sigma^{*2})$ and $\sigma^{*2}=o(\sigma_1^2)$. This expression turns out to be a combination of the Euclidean distance and the predictive mean matching. For example, when the correlation between the covariate and the dependent variable is small it is more like an Euclidean distance, otherwise it is more like a predictive mean matching.

From the perspective of predictive mean square error (PMSE), the objective of the selection of a donor value y_j for the missing value y_i is to minimise the PMSE. Suppose the normality assumption holds. The PMSE of y_j as an imputed value for y_j is then

$$\begin{aligned} E(y_j - y_i)^2 &= E[(\mathbf{x}_j - \mathbf{x}_i)\beta + \varepsilon_j - \varepsilon_i]^2 \\ &= [(\mathbf{x}_j - \mathbf{x}_i)\beta]^2 + 2\sigma^2 \\ &= (x_{j1} - x_{i1})^2 \beta_1^2 + (x_{j2} - x_{i2})^2 \beta_2^2 + 2\beta_1\beta_2(x_{j1} - x_{i1})(x_{j2} - x_{i2}) + 2\sigma^2 . \end{aligned} \quad (3.2.12)$$

The weighted distance imputation misses out the cross product term, therefore generally it can not outperform the predictive mean matching imputation under these assumptions, although they are approximately equivalent if the cross product term is small.

We now consider an alternative set of assumptions where the weighted distance method may be optimal. We assume that y_i and y_j are not independent and do not depend on the covariate $\mathbf{x}$, in which their correlation is a function of the weighted distance. We describe this situation as follows. We assume

$$\begin{aligned} &E(y_i) = \mu \text{ (does not depend on } \mathbf{x}_i), \\ &corr(y_i, y_j) = \rho(\mathbf{x}_i, \mathbf{x}_j), \\ &\text{var}(y_i) = \sigma^2, i = 1 \cdots n, \end{aligned} \quad (3.2.13)$$

where $\rho(\mathbf{x}_i, \mathbf{x}_j)$ is a continuous monotone function of $\mathbf{x}_i$ and $\mathbf{x}_j$. For example,

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = \lambda_1 (x_{i1} - x_{j1})^2 + \lambda_2 (x_{i2} - x_{j2})^2, \tag{3.2.14}$$

or,

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = \phi \exp(-\{\lambda_1 (x_{i1} - x_{j1})^2 + \lambda_2 (x_{i2} - x_{j2})^2\}), \tag{3.2.15}$$

where λ_1, λ_1 and ϕ are some constants. Under these assumptions, the PMSE becomes,

$$\begin{aligned} E(y_j - y_i)^2 &= E[(y_j - \mu) - (y_i - \mu)]^2 \\ &= \mathrm{var}(y_i) + \mathrm{var}(y_j) - 2\,\mathrm{cov}(y_i, y_j) \\ &= 2\sigma^2 - 2\sigma^2 \rho(y_i, y_j), \end{aligned} \tag{3.2.16}$$

Therefore minimising (3.2.16) is equivalent to maximising the correlation function which is also equivalent to minimising the weighted distance.

In spatial data analysis, such as ore reserve assessment study, one of the main interests is to estimate the covariogram, the covariance of two spatial points, which is a function of the distance between the two points. This gives the validity of the weighted distance in practical applications. Matern (1960) derives several valid covariance models in R^q, where R^q is the real space with q dimensions. Here we assume q=2. One of them can be used to construct the valid weighted distance.

$$\begin{aligned} \mathrm{cov}^{(1)}(y_i, y_j) &= \sigma^2 \exp(-a^2 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2) \\ &= \sigma^2 \exp(-a^2 \{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2\}), \; a \in R^1. \end{aligned}$$

For more details see Yaglom (1957). The linear combination of any valid covariogram is also a valid covariogram; the product of any valid covarigram is also a valid covariogram (Cressie, 1993, p85). Suppose we have two valid type I covariance function as follows,

$$\mathrm{cov}^{(11)}(y_i, y_j) = \sigma^2 \exp(-a_1^2 (x_{i1} - x_{j1})^2), \; a_1 \in R^1,$$

$$\mathrm{cov}^{(12)}(y_i, y_j) = \sigma^2 \exp(-a_2^2 (x_{i2} - x_{j2})^2), \; a_2 \in R^1.$$

Then the product of these two is also a valid covariogram.

$$\mathrm{cov}^{(1p)}(y_i, y_j) = \sigma^4 \exp(-\{a_1^2(x_{i1} - x_{j1})^2 + a_2^2(x_{i2} - x_{j2})^2\}),\ a_1, a_2 \in R^1.$$

The exponential covariance function of type (3.2.15) is obtained. With this covariance structure, minimising the mean square error object is equivalent to minimising the weighted distance.

Although the weighted distance based nearest neighbour imputation neglects the cross product term in the PMSE, it still has some improvement compared with the Euclidean distance and Mahalanobis distance imputation. One can add the cross product term to the weighted distance to form a modified version of weighted distance imputation. The main advantage of the weighted distance is the ability of dealing with different types of variables and embedding different type of distance accordingly. Especially the more robust distances like absolute value distance can be employed to improve the performance in the situations where outliers exist.

3.3 The Distributions of Distances

This section explores the distribution of the distance between y_is, with the aim of considering suitable models relating this distance to $D(x)$ and hence to estimating W.

Nominal y

Suppose a nominal variable y, has the following distribution

$$\pi_i = \Pr(y=i),\ i=1,\ldots,p+1. \qquad (3.3.1)$$

Let the distribution function for y be binary with $d(y_i,y_j)=0$ if $y_i=y_j$ and 1 if $y_i \neq y_j$. Then the distribution of d for a pair of randomly chosen y is

$$\Pr(d(y)=1)=\sum_{i\neq j}\pi_i\pi_j \qquad (3.3.2)$$

$$\Pr(d(y)=0)=1-\sum_{i\neq j}\pi_i\pi_j=\sum_i\pi_i^2 . \qquad (3.3.3)$$

The logistic model or a neural network could be used to predict this distance.

Ordinal y

If y is an ordinal variable taking values $1,\ldots,p+1$, then the distance function may be defined as $d(y_i,y_j)=|y_i-y_j|$. $D(y)$ is still a ordinal variable, but likely has more distance levels. Suppose the distribution of y is

$$\pi_i=\Pr(y=a_i),\ i=1,\ldots,p+1. \qquad (3.3.4)$$

Then the distribution of $d(y)$ for randomly chosen y_i and y_j is

$$\Pr(d(y)=k)=2\sum_{i=1}^{p+1-k}\pi_i\pi_{i+k},\ k=1\ldots m(m-1)/2. \qquad (3.3.5)$$

The Multinomial logit model or neural network model can be used to describe the relationship between this distance measure and $D(\mathrm{x})$.

Continuous y

For continuous y, the simplest case, normal distribution case, is considered. Suppose $y\sim N(\mu(\mathrm{x}),\sigma^2)$, and Euclidean distance is used. Denote the distance between y_i and y_j by $d(y_i,y_j)$ as before.

$$d(y_i,y_j)=(y_i-y_j)^2. \qquad (3.3.6)$$

Notice $y_i-y_j\sim N(\mu(x_i)-\mu(x_j),2\sigma^2)$, then the distribution of $d(y_i,y_j)$ is

$$\Pr(d(y_i,y_j)<a)=\Pr(|y_i-y_j|<\sqrt{a})=\Pr(y_i-y_j>-\sqrt{a})+\Pr(y_i-y_j<\sqrt{a})$$

$$=1-\frac{1}{2\sigma\sqrt{\pi}}\int_{-\infty}^{-\sqrt{a}} e^{-\frac{(y-(\mu(x_i)-\mu(x_j)))^2}{4\sigma^2}}dy+\frac{1}{2\sigma\sqrt{\pi}}\int_{-\infty}^{\sqrt{a}} e^{-\frac{(y-(\mu(x_i)-\mu(x_j)))^2}{4\sigma^2}}dy. \quad (3.3.7)$$

The density function is

$$f_D(a)=\frac{1}{4\sigma\sqrt{\pi a}}e^{-\frac{(-\sqrt{a}-(\mu(x_i)-\mu(x_j)))^2}{4\sigma^2}}+\frac{1}{4\sigma\sqrt{\pi a}}e^{-\frac{(\sqrt{a}-(\mu(x_i)-\mu(x_j)))^2}{4\sigma^2}}. \quad (3.3.8)$$

It is not a member of the exponential family. μ(x) can either be approximated by a linear function or by non-linear functions like neural networks. The likelihood estimate of W can only be obtained numerically. When the computing burden is concerned, we would rather model y with respect to x directly. Another way is to check the distribution of $D(y)$ non-parametrically, and model $D(y)$ with respect to D(x) instead.

4 Properties of Imputation Methods

4.1 Introduction

In previous chapters we have discussed various non-parametric imputation methods, such as distance based nearest neighbour imputation (donor imputation) and neural networks imputation, and parametric imputation methods such as linear regression based imputation. A natural question is to ask how these methods perform. There is no easy answer. It depends on various conditions including the mechanism of missingness and the characteristics of the population from which the data are generated. More importantly, it depends on criteria for what is a good imputation method. Taking a continuous variable as an example, one can judge the performance of an imputation method by evaluating how it preserves the properties of the population such as mean and variance in terms of the expectation and variance of estimators of these parameters under imputation. On the other hand, one can consider estimators of quantiles.

If the missing mechanism depends on the variable which contains missing values, or if the observed data and the missing data are from two different populations, the evaluation will be complicated. For example, in the annual consumer survey of China, income is more likely to be missing for low income owners. In this situation if a good imputation is defined as one that leads to accurate estimates of population parameters based on the complete cases, many imputation methods will perform poorly and will fail to find the genuinely good imputations that give much lower average of income than the observed average. In this chapter we assume the missing mechanism is missing at random (MAR) defined in Chapter I, where the missingness doesn't depend on the variable with missing values, but may depend on observed covariates. That makes the evaluation of imputation methods sensible, because the covariates may provide a reliable basis for imputation.

The main results in this chapter are about the bias and variance of estimators containing imputed values. Our attention is focused on continuous variables and superpopulation parameters. Finite population parameters are also discussed. In section 4.8, we give some initial results on imputation for categorical variable. But these results are descriptive.

Further work is needed to reveal the difference between methods such as RBF imputation and logistic imputation.

Let consider a continuous variable y. Assume μ is the mean of y, τ is the variance of y in either a superpopulation or finite population. $y_1 ... y_n$ are a realisation of y, where n is the size of sample. For convenience, we assume $y_1 ... y_n$ are independent and identically distributed (*i.i.d.*)

$$E(y)=\mu, \tag{4.1.1}$$

$$\text{var}(y)=\tau, \tag{4.1.2}$$

$y_1 ... y_n$ are *i.i.d.* (4.1.3)

The evaluation will be based on how an imputation can preserve μ and τ in terms of the biases and variances of its estimators. A good imputation is expected to preserve the population mean as well as not to inflate variance much. We first introduce the estimators of population mean and variance with full sample. The comparison is based on the properties of estimators with missing values (assumed missing) substituted by imputed values. Let's consider two scenarios, superpopulation and finite population respectively.

4.2 Estimators with Full Observations

Superpopulation

As described in section 4.1, suppose we have a realisation $Y=(y_1 ... y_n)^T$ from a superpopulation model, where $y_1 ... y_n$ are independent realisations of a random variable y with mean μ and variance τ, where n is the number of observations. Here τ is scalar. The reason of using τ instead of σ^2 is that later on σ^2 is used to denote the variance of regression residual. The parameter μ can be estimated by

$$\hat{\mu} = n^{-1} \sum_{i=1}^{n} y_i = n^{-1} \mathbf{1}_n^T Y, \tag{4.2.1}$$

where $\mathbf{1}_n^T = (1,...,1)$. Under the assumptions given in (4.1.3), the above estimator is unbiased with variance τ/n. The variance of y can be estimated by

$$\hat{\tau} = n^{-1}\sum_{i=1}^{n}(y_i - \hat{\mu})^2 = n^{-1}Y^T(I_n - n^{-1}\mathbf{1}_n\mathbf{1}_n^T)Y\,, \qquad (4.2.2)$$

where I_n is the $n\times n$ identity matrix. The estimator $\hat{\tau}$ is asymptotically unbiased (Larson, 1982).

For imputation purposes we usually make use of a covariate **x** to construct imputation methods. We assume there are q covariates. All values of covariates are assumed observed, and these values are written as

$$X = \begin{pmatrix} x_{11} & \cdots & x_{1q} \\ & \vdots & \\ x_{n1} & \cdots & x_{nq} \end{pmatrix}.$$

We further assume the conditional expectation of y given these covariates may be expressed as $\mu(\mathrm{x})$, and we may write

$$y_i = \mu(\mathrm{x}_i) + \varepsilon_i, i = 1...n\,, \qquad (4.2.3)$$

where $\mathrm{x}_i=(x_{i1}\ldots x_{iq})^{\mathrm{T}}$, $E(\varepsilon_i)=0$, $var(\varepsilon_i)=\sigma^2$, $\varepsilon_1 \ldots \varepsilon_n$ are independent. For simplicity we treat x as fixed. The function $\mu(\mathrm{x}_i)$ becomes $\mathrm{x}_i\beta$ in regression model (section 1.3) or $\Phi(\mathrm{x}_i)W$ in RBF model (section 2.1). It depends on model assumptions. For superpopulation, the definition of μ may be extended in this case to

$$\mu = \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\mu(\mathrm{x}_i) = \lim_{n\to\infty}E(\bar{y})\,. \qquad (4.2.4)$$

Similarly the definition of τ may be extended as

$$\tau = \lim_{n\to\infty} E(n^{-1}\sum_{i=1}^{n}(y_i - \bar{y})^2)$$

$$= \lim_{n\to\infty} E(n^{-1}\sum_{i=1}^{n}(y_i - \mu(\mathbf{x}_i) + \mu(\mathbf{x}_i) - \bar{y})^2)$$

$$= \lim_{n\to\infty} E(n^{-1}\sum_{i=1}^{n}((y_i - \mu(\mathbf{x}_i))^2 + (\mu(\mathbf{x}_i) - \bar{y})^2)$$

$$= \sigma^2 + \sigma_\mu^2, \qquad (4.2.5)$$

where $\sigma_\mu^2 = \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}(\mu(\mathrm{x}_i) - \mu)^2$. The expectation of the mean estimator in (4.2.1) is as follows

$$E(\hat{\mu}) = \frac{1}{n}\sum_{i=1}^{n}\mu(\mathrm{x}_i) \overset{n\to\infty}{=} \mu. \qquad (4.2.6)$$

(4.2.6) indicates $\hat{\mu}$ in (4.2.1) is an asymptotically unbiased estimator of μ.

The expectation of the estimator of τ in (4.2.2) is as follows

$$E(\hat{\tau}) = n^{-1}\sum_{i=1}^{n}E(y_i - \hat{\mu})^2$$

$$= n^{-1}\sum_{i=1}^{n}E(y_i - \mu(\mathbf{x}_i))^2 + 2n^{-1}\sum_{i=1}^{n}E(y_i - \mu(\mathbf{x}_i))(\mu(\mathbf{x}_i) - \mu) +$$

$$n^{-1}\sum_{i=1}^{n}E(\mu(\mathbf{x}_i) - \mu)^2$$

$$= \sigma^2 + n^{-1}\sum_{i=1}^{n}(\mu(\mathbf{x}_i) - \mu)^2, \qquad (4.2.7)$$

and $\hat{\tau}$ is also asymptotically unbiased for τ.

Finite Population

In the census scenario, suppose we have a finite population of size n, the same as the sample size. The population mean and variance can thus be written as follows,

$$\mu = n^{-1}\sum_{i=1}^{n} y_i , \tag{4.2.8}$$

$$\tau = n^{-1}\sum_{i=1}^{n} (y_i - \mu)^2 , \tag{4.2.9}$$

and $\hat{\mu} = \mu$, $\hat{\tau} = \tau$. So there is no estimation error if there are no missing values.

4.3 Estimators in the Presence of Imputation

In section 4.2, we gave the estimators based on fully observed Y. In this section we consider the corresponding estimators when some of the observations in Y are missing and replaced by imputed values. Suppose the first m units are observed, and the remaining n-m units are missing. For convenience, we denote the data matrix X and Y as

$$X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} x_{11} & \cdots & x_{1q} \\ & \vdots & \\ x_{m1} & \cdots & x_{mq} \\ x_{m+1,1} & \cdots & x_{m+1,q} \\ & \vdots & \\ x_{n1} & \cdots & x_{nq} \end{pmatrix}$$

$$Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_m \\ y_{m+1} \\ \vdots \\ y_n \end{pmatrix}.$$

The estimator of μ (4.2.1) can be expressed as the sum of two parts as follows

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} y_i = \frac{1}{n}\sum_{i=1}^{m} y_i + \frac{1}{n}\sum_{j=m+1}^{n} y_j$$

$$= \frac{m}{n}\bar{Y}_1 + \frac{n-m}{n}\bar{Y}_2 = (1-w)\bar{Y}_1 + w\bar{Y}_2, \qquad (4.3.1)$$

where $\bar{Y}_1$ and $\bar{Y}_2$ are the means of Y_1 and Y_2 respectively, $w=(n-m)/n$.

If Y_2 is missing, it can be imputed and plugged into (4.3.1). We denote the imputed value of Y_2 by $\hat{Y}_2$, where

$$\hat{Y}_2 = \begin{pmatrix} \hat{y}_{m+1} \\ \vdots \\ \hat{y}_n \end{pmatrix}.$$

Then

$$\bar{\hat{Y}}_2 = \frac{1}{n-m}\sum_{j=m+1}^{n} \hat{y}_j, \qquad (4.3.2)$$

and

$$\hat{\mu}_I = (1-w)\bar{Y}_1 + w\bar{\hat{Y}}_2, \qquad (4.3.3)$$

The estimator of τ in (4.2.2) can be expressed as follows

$$\hat{\tau} = \frac{1}{n}\sum_{i=1}^{n} (y_i - \bar{Y})^2$$

$$= \frac{1}{n}\sum_{i=1}^{m} (y_i - \bar{Y})^2 + \frac{1}{n}\sum_{j=m+1}^{n} (y_j - \bar{Y})^2 \qquad (4.3.4)$$

$$= A + B,$$

where A and B can be written as follows

$$A=\frac{1}{n}\sum_{i=1}^{m}(y_i-\bar{Y})^2$$

$$=\frac{1}{n}\sum_{i=1}^{m}(y_i-(1-w)\bar{Y}_1-w\bar{Y}_2)^2$$

$$=\frac{1}{n}\sum_{i=1}^{m}(y_i-\bar{Y}_1+w(\bar{Y}_1-\bar{Y}_2))^2$$

$$=\frac{1}{n}\sum_{i=1}^{m}(y_i-\bar{Y}_1)^2+\frac{2w}{n}\sum_{i=1}^{m}(y_i-\bar{Y}_1)(\bar{Y}_1-\bar{Y}_2)+\frac{w^2}{n}\sum_{i=1}^{m}(\bar{Y}_1-\bar{Y}_2)^2$$

$$=(1\text{-}w)\ \hat{\tau}_1+w^2(1-w)(\bar{Y}_1-\bar{Y}_2)^2,$$

where $\hat{\tau}_1$ is the estimator of τ based on Y_1. Similarly,

$$B=\frac{1}{n}\sum_{j=m+1}^{n}(y_j-\bar{Y})^2$$

$$=\frac{1}{n}\sum_{j=m+1}^{n}(y_j-(1-w)\bar{Y}_1-w\bar{Y}_2)^2$$

$$=\frac{1}{n}\sum_{j=m+1}^{n}(y_j-\bar{Y}_2+(1-w)(\bar{Y}_2-\bar{Y}_1))^2$$

$$=\frac{1}{n}\sum_{j=m+1}^{n}(y_j-\bar{Y}_2)^2+2\frac{1-w}{n}\sum_{j=m+1}^{n}(y_j-\bar{Y}_2)(\bar{Y}_2-\bar{Y}_1)+w(1-w)^2(\bar{Y}_2-\bar{Y}_1)^2$$

$$=w\ \hat{\tau}_2+w(1-w)^2(\bar{Y}_2-\bar{Y}_1)^2,$$

where $\hat{\tau}_2$ is the estimator of τ based on Y_2. Plugging A and B into (4.3.4), $\hat{\tau}$ can be written as

$$\hat{\tau}=(1\text{-}w)\ \hat{\tau}_1+w\ \hat{\tau}_2+w(1-w)(\bar{Y}_1-\bar{Y}_2)^2. \qquad (4.3.5)$$

If Y_2 is missing and replaced by $\hat{Y}_2$, the estimator of τ based on Y_1 and $\hat{Y}_2$ becomes

$$\hat{\tau}_1=(1\text{-}w)\ \hat{\tau}_1+w\ \hat{\tau}_2^{\,I}+w(1-w)(\bar{Y}_1-\bar{\hat{Y}}_2)^2. \qquad (4.3.6)$$

With the decomposition given above, the comparison can be concentrated on the term containing the imputed values. A simple way to consider the effect of imputation is to compare the differences between the estimators based on true values and those containing imputations. In the case of a census and the finite population parameters, this difference is simply the estimation error. Here we assume the true values are known. The difference between $\hat{\mu}$ and $\hat{\mu}_I$ is

$$\begin{aligned}\hat{\mu}_I - \hat{\mu} &= \bar{\hat{Y}}_2 - \bar{Y}_2 \\ &= \frac{1}{n-m}\sum_{j=m+1}^{n}(\hat{y}_j - y_j) \\ &= \frac{1}{n-m}\mathbf{1}_{n-m}^T(\hat{Y}_2 - Y_2).\end{aligned} \tag{4.3.7}$$

In the census situation, $\hat{\mu}=\mu$, therefore (4.3.7) is the error introduced by imputation. Since τ is a scalar, the ratio of estimators of τ can be used to compare the effect of imputation,

$$\frac{\hat{\tau}_I}{\hat{\tau}} = 1 + \frac{w(\hat{\tau}_2^I - \hat{\tau}_2) + w(1-w)(\bar{\hat{Y}}_2 - \bar{Y}_2)(\bar{\hat{Y}}_2 + \bar{Y}_2 - 2\bar{Y}_1)}{\hat{\tau}}. \tag{4.3.8}$$

In (4.3.8), from the imputation point view, the first term is constant; the third term is likely to be a small quantity under MCAR assumption. The key effect of imputation is determined by the second term, which is the estimator of population variance based on $\hat{Y}_2$.

To compare the estimators based on full sample (in the sense of both Y_1 and Y_2 are available) and the estimators under imputation (Y_2 is missing and replaced by $\hat{Y}_2$), we need a criterion to measure the differences between them. Here we define the unbiasedness of the estimators in the presence of imputation with respect to the estimator of full sample as follows

$$E(T_I - T) = 0, \tag{4.3.9}$$

where T is the estimator of the parameter of interest based on Y_1 and Y_2. T_I is the estimator based on Y_1 and $\hat{Y}_2$. In census situation T becomes a quantities of finite population, such as population total or mean, which is actually the parameter of the finite population. Therefore this definition becomes the conventional unbiasedness definition. For example if the parameter of interest is μ, $\hat{\mu}$ is the estimator of full sample, and $\hat{\mu}_I$ is the estimator containing imputed values, the unbiasedness of $\hat{\mu}_I$ with respect to $\hat{\mu}$ can be expressed as

$$E(\hat{\mu}_I - \hat{\mu}) = 0. \qquad (4.3.10)$$

In census situation, (4.3.10) becomes $E(\hat{\mu}_I - \mu) = 0$. Similarly the unbiasedness of $\hat{\tau}_I$ can be defined as

$$E(\hat{\tau}_I - \hat{\tau}) = 0. \qquad (4.3.11)$$

If $E(T_I - T) \neq 0$, we call T_I is a biased estimator of T. We call $E(T_I\text{-}T)$ predictive bias, and denote it by *pbias*,

$$pbias = E(T_I - T). \qquad (4.3.12)$$

Potentially there are many ways to obtain $\hat{Y}_2$ (see section 1.4). Since our main objective is to compare regression-based imputations (1.4.1, 1.4.2) and neural networks based imputations (2.5.1, 2.5.2), we shall focus our discussion on these two kinds of imputations. Also for mathematical simplicity, we choose RBF neural networks imputations instead of MLP imputations.

4.4 Bias Properties of Estimators of μ

In this section we shall be investigating the biases of estimators of μ when imputed values are used. The imputations we consider are regression-based imputation and neural network based imputation such as linear regression imputation and random regression imputation and the counterparts of neural network based imputations. The neural network

model we chose is RBF neural network. We discuss the two kinds imputations respectively.

Regression Based Imputation

Two imputation methods based on linear regression are considered, regression imputation and random regression imputation. The regression imputation replaces missing values by the conditional mean given by regression model (see 1.4.1). Suppose Y_2 is missing, the regression imputation is

$$\hat{Y}_{2l} = X_2\hat{\beta}, \tag{4.4.1}$$

where $\hat{\beta} = (X_1^T X_1)^{-1} X_1^T Y_1$.

Random regression imputation is regression imputation with an additional residual term, which can be obtained by a random draw from the residuals of complete cases (see 1.4.2). The random regression imputation can be written as

$$\hat{Y}_{2rl} = X_2\hat{\beta} + \hat{\mathbf{e}}_2, \tag{4.4.2}$$

where $\hat{\mathbf{e}}_2 = (\hat{\varepsilon}_{m+1} \cdots \hat{\varepsilon}_n)^T$ is the estimated residual vector of length *n-m* (see 1.4.2). $\hat{\varepsilon}_i$ can be obtained from the residuals of observed cases. The simplest way is a random draw from the available residuals $\{\hat{\varepsilon}_j, j = 1...m\}$, where $\hat{\varepsilon}_j = y_j - x_j\hat{\beta}, j = 1...m$. Other methods include predictive mean matching and a random draw from the distribution of residual term estimated from observed residuals. For predictive mean matching method, $\hat{\varepsilon}_i$ can be written as $\hat{\varepsilon}_i = \hat{\varepsilon}_{j:\left|x_i\hat{\beta} - x_j\hat{\beta}\right| = \min(\left|x_i\hat{\beta} - x_l\hat{\beta}\right|, l=1...m)}$.

The assumptions about variable *y* are $E(y) = \mu(\mathbf{x})$ and $var(y) = \tau$ (see 4.2.3), where $\mathbf{x}$ is the covariate of *y*. The regression imputation is a linear combination of Y_1 and can be written as

$$\hat{Y}_2 = X_2(X_1^T X_1)^{-1} X_1^T Y_1 = H_{21} Y_1, \tag{4.4.3}$$

where

$$H_{21} = X_2(X_1^T X_1)^{-1} X_1^T. \tag{4.4.4}$$

If we plug the imputation in (4.4.3) into (4.3.3), the estimator $\hat{\mu}_l$ under regression imputation is obtained.

$$\begin{aligned}
\hat{\mu}_l &= (1-w)\overline{Y}_1 + w\overline{\hat{Y}}_{2l} \\
&= \frac{1}{n}(\mathbf{1}_m{}^{\mathrm{T}} Y_1 + \mathbf{1}_{n-m}{}^{\mathrm{T}} H_{21} Y_1) \\
&= \frac{1}{n}(\mathbf{1}_m^T + \mathbf{1}_{n-m}^T H_{21}) Y_1 .
\end{aligned} \tag{4.4.5}$$

Plugging (4.4.2) into (4.3.3), the estimator of μ based on random regression imputation is obtained. We denote it by $\hat{\mu}_{rl}$. It can be written as

$$\begin{aligned}
\hat{\mu}_{rl} &= \frac{1}{n}(\mathbf{1}_{\mathbf{m}}^{\mathrm{T}} + \mathbf{1}_{\mathbf{n-m}}^{\mathrm{T}} H_{21}) Y_1 + \frac{1}{n}\mathbf{1}_{\mathbf{n-m}}^{\mathrm{T}} \hat{\mathbf{e}}_{\mathbf{2}} \\
&= \hat{\mu}_l + \frac{1}{n}\mathbf{1}_{\mathbf{n-m}}^{\mathrm{T}} \hat{\mathbf{e}}_{\mathbf{2}} .
\end{aligned} \tag{4.4.6}$$

We discuss the regression imputation in (4.4.1) first. If the linear assumption is true, which leads to $\mu(\mathbf{x})=\mathbf{x}\beta$, the expectation of $\hat{\mu}_l$ becomes

$$\begin{aligned}
E(\hat{\mu}_l) &= \frac{1}{n}(\mathbf{1}_m^T + \mathbf{1}_{n-m}^T H_{21}) X_1 \beta \\
&= \frac{1}{n}(\mathbf{1}_m^T X_1 \beta + \mathbf{1}_{n-m}^T X_2 (X_1^T X_1)^{-1} X_1^T X_1 \beta) \\
&= \frac{1}{n}(\mathbf{1}_m^T X_1 \beta + \mathbf{1}_{n-m}^T X_2 \beta) \\
&= \frac{1}{n}\mathbf{1}_n^T X\beta .
\end{aligned} \tag{4.4.7}$$

Comparing (4.4.7) with (4.2.4), we find $\hat{\mu}_l$ is an asymptotically unbiased estimator of μ. Furthermore, under the definition of (4.3.11) $\hat{\mu}_l$ is also unbiased with respect to $\hat{\mu}$.

The unbiasedness of $\hat{\mu}_l$ is a direct result of linear assumption. What if the assumption is invalid? We would like to know the robustness of $\hat{\mu}_l$ under misspecification. Specifically, what is the bias if the true model is RBF (see 2.1.4)? One reason we consider RBF instead of other models is that RBF can be adjusted to be any linear and non-linear functions at least in theory. Therefore it has the potential to represent a wide range of models. The other reason is that the main objective of this chapter is to compare regression imputations and RBF imputations. We would like to know what would happen if one model is true and the other one is used.

Under RBF model, the expectation of **y** is $E(y)=\mu(\mathrm{x})=\Phi(\mathrm{x})W$ (see 2.1.4). According to (4.2.4), μ becomes

$$\mu = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \Phi(\mathrm{x}_i) W . \tag{4.4.8}$$

The expectation of $\hat{\mu}_l$ becomes

$$E(\hat{\mu}_l) = \frac{1}{n} (\mathbf{1}_m^T + \mathbf{1}_{n-m}^T H_{21}) \Phi(X_1) W . \tag{4.4.9}$$

From (4.4.9) and (4.4.8) we obtain the following expression for the bias of $\hat{\mu}_l$ under RBF model assumption.

$$bias(\hat{\mu}_l) \approx \lim_{n\to\infty} (\frac{1}{n} \mathbf{1}_{n-m}^T H_{21} \Phi(X_1) W - \frac{1}{n} \mathbf{1}_{n-m}^T \Phi(X_2) W) . \tag{4.4.10}$$

Suppose the term inside the limit sign is a good approximation of $bias(\hat{\mu}_l)$, and if the underlying $\Phi(X)$ is far from linear, the bias is likely large. In the ideal situation where $\Phi(X_1)W \approx X_1\beta$ and then $\Phi(X_2)W \approx X_2\beta$, the bias will be very small.

Based on the unbiasedness definition in (4.3.11), the predictive bias of $\hat{\mu}_l$ with respect to $\hat{\mu}$ is exactly the term inside the limit sign in (4.4.10).

$$pbias(\hat{\mu}_l) = \frac{1}{n}\mathbf{1}_{n-m}^T H_{21}\Phi(X_1)W - \frac{1}{n}\mathbf{1}_{n-m}^T \Phi(X_2)W, \qquad (4.4.11)$$

Again the predictive bias depends on the difference between $X\beta$ and $\Phi(X)$. If $X\beta \approx \Phi(X)$, it is small, otherwise it might be large. Therefore the performance of regression imputation depends on the validity of model assumption. If the underlying model is a linear regression model, the regression imputation gives unbiased mean estimator with respect to the estimator based on true values. Otherwise if the underlying model is RBF, regression imputation may give biased estimator. Because RBF model is very flexible, it can be tuned to be any non-linear function. Therefore the assumption of RBF model can cover a wide range of models. This eases the damage of model misspecification.

From now on we shall focus on predictive bias other than the ordinary bias, which is difference between the expectation of estimator and the parameter of interest. The expansion of (4.4.11) is discussed for several special cases in section 4.7.

The above results of $\hat{\mu}_l$ are also applicable to random regression imputation $\hat{\mu}_{rl}$. Since the residual term is assumed to be independent to Y_1 and has expectation of zero (see 1.3.2). Therefore we have $E(\hat{\mu}_l) = E(\hat{\mu}_{rl})$.

RBF Based Imputation

If Y_2 is missing, it can be imputed by RBF imputations. Similar to regression-based imputation, the two methods based on the RBF model are RBF imputation, which imputes

missing values by RBF prediction (2.5.1) and random RBF imputation which adds a residual term to RBF imputation (see 2.5.2). The RBF imputation can be denoted as follows

$$\hat{Y}_2^R = \Phi(X_2)\hat{W}, \tag{4.4.12}$$

where

$$\hat{W} = ((\Phi(X_1)^T \Phi(X_1))^{-1} \Phi(X_1)^T Y_1 \tag{4.4.13}$$

The random RBF imputation is $\hat{Y}_2^R$ with additional residual term randomly drawn from the residuals of the complete cases. We denote it by $\hat{Y}_2^{rR}$,

$$\hat{Y}_2^{rR} = \hat{Y}_2^R + \hat{\mathbf{e}}_2^R, \tag{4.4.14}$$

where $\hat{\mathbf{e}}_2^R$ is the residual vector drawn from the residuals of the complete cases and assumed to be independent to Y_1 and centred at zero. If we plug (4.4.12) and (4.4.14) into (4.3.3), the estimators of mean based on RBF imputation and random RBF imputation are obtained. We denote them by $\hat{\mu}_R$ and $\hat{\mu}_{rR}$ respectively.

$$\hat{\mu}_R = \frac{1}{n}(\mathbf{1}_m^T Y_1 + \mathbf{1}_{n-m}^T \Phi(X_2)\hat{W}), \tag{4.4.15}$$

$$\hat{\mu}_{rR} = \hat{\mu}_R + \frac{1}{n}\mathbf{1}_{\mathbf{n-m}}^{\mathbf{T}} \hat{\mathbf{e}}_2^R. \tag{4.4.16}$$

Compare the RBF estimator in (4.4.15) to the estimator in (4.3.1), the predictive bias of $\hat{\mu}_R$ is obtained,

$$pbias(\hat{\mu}_R) = E(\frac{1}{n}\mathbf{1}_{n-m}^T(\Phi(X_2)\hat{W} - Y_2))$$

$$= \frac{1}{n}\mathbf{1}_{n-m}^T(\Phi(X_2)((\Phi(X_1)^T\Phi(X_1))^{-1}\Phi(X_1)^T\mu(X_1) - \mu(X_2)) \qquad (4.4.17)$$

where $\mu(X_1)=(\mu(x_1)\ldots\ \mu(x_m))^T$, $\mu(X_2)=(\mu(x_{m+1})\ldots\ \mu(x_n))^T$. If the underlying model is RBF, $\mu(X_1)=\Phi(X_1)$, $\mu(X_2)=\Phi(X_2)$, $pbias(\hat{\mu}_R) = 0$, then $\hat{\mu}_R$ is unbiased with respect to $\hat{\mu}$. As aforementioned, RBF model can represent a wide range of models, although it may not be easy to obtain, in the sense of finding an exact representation of the underlying model in practice. If the underlying model is a linear regression model, $\mu(X_1)= X_1\beta$, $\mu(X_2)= X_2\beta$, then $pbias(\hat{\mu}_R) \neq 0$, therefore the mean estimator based on regression imputation is biased. In the special case when $\Phi(X_1)W \approx X_1\beta$, regression imputation may also give promising results. This may explain in some situations when computing time is a big concern regression imputation can be a good replacement to RBF.

The random RBF estimator $\hat{\mu}_{rR}$ shares the properties of $\hat{\mu}_R$ discussed above, since the residual terms are assumed to have zero expectations.

4.5 Bias Properties of Estimators of τ

In this section we discuss the properties of estimators of the variance τ in the presence of imputation. Specifically, we investigate the predictive unbiasedness of the estimators of τ based on regression imputation and RBF imputation. The imputation methods considered are regression imputation (4.4.1), random regression imputation (4.4.2), RBF imputation (4.4.12) and random RBF imputation (4.4.14). They will be discussed in turn.

Regression Based Imputation

The variance estimator based on regression imputation can be obtained by plugging (4.4.1) into (4.3.7) and is denoted by $\hat{\tau}_l$,

$$\hat{\tau}_I = (1-w)\,\hat{\tau}_1 + w\,\hat{\tau}_2^{\,I} + w(1-w)(\bar{Y}_1 - \bar{\hat{Y}}_2)^2$$

$$= \frac{1}{n} Y_1^T H_{11} Y_1 + \frac{1}{n} Y_1^T H_{21}^T H_{22} H_{21} Y_1$$

$$+ \frac{m(n-m)}{n^2} Y_1^T \left(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T \mathbf{1}_{n-m}}{n-m}\right)\left(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T \mathbf{1}_{n-m}}{n-m}\right)^T Y_1$$

$$= Y_1^T C_H Y_1, \qquad (4.5.1)$$

where C_H is the matrix containing constants and hat matrices,

$$C_H = \frac{H_{11}}{n} + \frac{H_{21}^T H_{22} H_{21}}{n} + \frac{m(n-m)}{n^2}\left(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T \mathbf{1}_{n-m}}{n-m}\right)\left(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T \mathbf{1}_{n-m}}{n-m}\right)^T \qquad (4.5.2)$$

$$H_{11} = \mathbf{I}_m - \frac{1}{m}\mathbf{1}_m \mathbf{1}_m^T,$$

$$H_{22} = \mathbf{I}_{n-m} - \frac{1}{n-m}\mathbf{1}_{n-m}\mathbf{1}_{n-m}^T.$$

Under the assumptions given in (4.1.1)-(4.1.3), the expectation of $\hat{\tau}_I$ becomes,

$$E(\hat{\tau}_I) = \sigma^2 tr(C_H) + \mu_1^T C_H \mu_1, \qquad (4.5.3)$$

where $\mu_1^T = (\mu(\mathbf{x}_1), \cdots, \mu(\mathbf{x}_m))$. The term $tr(C_H)$ can be written as

$$\frac{m-1}{n} + \frac{1}{n} tr(X_2 (X_1^T X_1)^{-1} X_2^T) +$$

$$\frac{n-m}{n^2} - \frac{2}{n^2}\mathbf{1}_{n-m}^T H_{21} \mathbf{1}_m - \frac{1}{n^2} tr(\mathbf{1}_{n-m}\mathbf{1}_{n-m}^T X_2 (X_1^T X_1)^{-1} X_2^T). \qquad (4.5.4)$$

C_H is a function of the matrix X_1, the matrix X_2, the size of the observed sample m and the sample size n. It is useful to investigate the asymptotic behaviour of C_H in order to understand the influence of the different model assumptions. To achieve that we need more assumptions about the relationship between X_1 and X_2 as well as between m and n. One of the assumptions is that the proportion of missing cases tends to a constant as n

increases. The other one is that X_2 falls in to the linear space spanned by X_1. Let us write down the two conditions

$$w = \frac{n-m}{n} \xrightarrow{n\to\infty} r_0, \tag{4.5.5}$$

$$X_2 = UX_1, \tag{4.5.6}$$

where U is a $(n\text{-}m)\times m$ matrix. The validity of (4.5.6) is based on the fact of $m>>q$. Therefore it is likely that there exist q rows in X_1, which form a R^q space. For simplicity we assume the first q rows $\mathbf{x}_1,\ldots,\mathbf{x}_q$ span the R^q space. Then the jth row in X_2 can be represented by

$$\mathbf{x}_{2j} = a_1\mathbf{x}_{11} + \cdots + a_q\mathbf{x}_{1q} + 0\cdot\mathbf{x}_{1q+1} + \cdots + 0\cdot\mathbf{x}_{1m}, j = 1,\ldots,n-m, a_1,\ldots,a_q \in R^1,$$

where $\mathbf{x}_{2j}$ is the jth row of X_2, $\mathbf{x}_{1i}$ is the ith row of X_1, i=1,…,m. The above expression can be denoted in the form of (4.5.6). If the number of observed cases (m) is much larger than the number of unobserved cases (n-m), for each row vector in X_2, it is highly possible to find a same row in X_1. The assumption can be described by the following expression,

$$\mathbf{x}_{2j} = \mathbf{x}_{1k}, j = 1,\cdots,n-m, k \in \{1,\ldots,m\}. \tag{4.5.7}$$

That means all the elements of U are just 0 or 1.

The term $tr(C_H)$ may be expressed as

$$tr(C_H) = \frac{m-1}{n} + \frac{1}{n}tr(P_1U^TU) +$$
$$\frac{n-m}{n^2} - \frac{2}{n^2}\mathbf{1}_{n-m}^T UP_1\mathbf{1}_m - \frac{1}{n^2}tr(P_1U^T\mathbf{1}_{n-m}\mathbf{1}_{n-m}^T U), \tag{4.5.8}$$

where $P_1 = X_1(X_1^T X_1)^{-1} X_1^T$ is the hat matrix based on X_1. P_1 has the following properties.

(1) Idempotent

(2) Symmetric

(3) Non negative.

Therefore there exists a matrix V such that

$$P_1 = V\Lambda_m V^T, \tag{4.5.9}$$

where $V^T V = \mathbf{I}_m$, Λ_m is a diagonal matrix with only 0 or 1 on the diagonal (Lütkepohl, 1996). Based on (4.5.9), we can obtain the following result.

$$tr(P_1) = tr(V\Lambda_m V^T) = tr(\Lambda_m V^T V) = tr(\Lambda_m) = rank(X_1) \le q, \tag{4.5.10}$$

where q is the number of covariates.

It will be helpful if the asymptotic value or limit of $tr(\boldsymbol{C_H})$ is obtained. We employ the properties of a symmetric non negative matrix to determine the limit of $tr(C_H)$.

For symmetric non-negative matrixes A and B (Lütkepohl, 1996),

$$tr(A) = \sum_{i=1}^{rank(A)} \lambda_i \ge 0, \tag{4.5.11}$$

$$tr(AB) = tr(BA) \le tr(A) + tr(B), \tag{4.5.12}$$

where λ_i is the ith eigenvalue of A, (4.5.12) assumes A B are exchangeable. Hence

$$0 \le tr(P_1 U^T U) = tr(U^T U P_1)$$

$$= tr(U^T U V \Lambda_m V^T)$$

$$= tr(\Lambda_m V^T U^T U V)$$

$$= tr(\Lambda_m (UV)^T UV) \tag{4.5.13}$$

$$0 \le tr(\mathbf{1}_{n-m}^T U P_1 \mathbf{1}_m) = tr(P_1 \mathbf{1}_m \mathbf{1}_{n-m}^T U) \le q + tr(\mathbf{1}_m \mathbf{1}_{n-m}^T U), \tag{4.5.14}$$

$$0 \le tr(P_1 U^T \mathbf{1}_{n-m} \mathbf{1}_{n-m}^T U) \le q + tr(U^T \mathbf{1}_{n-m} \mathbf{1}_{n-m}^T U). \tag{4.5.15}$$

All the elements of U are assumed to be 0 or 1 as explained in (4.5.7). In the situation of $m>>n-m$, U is a sparse matrix. In another situation when each row of X_2 can be found in X_1, we have $U^T U = \Lambda_m'$, where Λ_m' is a diagonal matrix with elements 0 or 1 on the diagonal. In both situations, the trace of matrices involving U can be further expressed as

$$tr(\Lambda_m (UV)^T UV)$$

$$\le tr(\Lambda_m V^T V) = tr(\Lambda_m) = tr(P_1) \tag{4.5.16}$$

$$tr(\mathbf{1}_m \mathbf{1}_{n-m}^T U) \le tr(\mathbf{1}_m \mathbf{1}_m^T) = m, \tag{4.5.17}$$

$$tr(U^T \mathbf{1}_{n-m} \mathbf{1}_{n-m}^T U) = tr(\mathbf{1}_{n-m} \mathbf{1}_{n-m}^T U U^T) \le tr(\mathbf{1}_{n-m} \mathbf{1}_{n-m}^T) = n - m. \tag{4.5.18}$$

If we plug (4.5.16) into (4.5.13), (4.5.17) into (4.5.14), and (4.5.18) into (4.5.15), then plug (4.5.13) to (4.5.16) to (4.5.8), the asymptotic value of $tr(C_H)$ is obtained as follows

$$tr(C_H) = \frac{m-1}{n} + \frac{1}{n} tr(P_1 U^T U) + o(\frac{1}{n})$$

$$= \frac{m-1}{n} + O(\frac{q}{n}) + o(\frac{1}{n})$$

$$= \frac{m-1}{n} + O(\frac{1}{n}) \xrightarrow{n \to \infty} 1 - r_0. \tag{4.5.19}$$

For general U, the similar result to (4.5.9) is expected. Due to mathematical complexity, the result for general U is not included here. But one can imagine the general U is a linear combination of a series of orthogonal matrix U. For orthogonal U, we have $tr(P_1 U^T U) = tr(P_1)$. Therefore the result of (4.5.16) can be extended to the general U.

The second term in $E(\hat{\tau}_l)$ (see 4.5.3) is $\mu_1^T C_H \mu_1$. This is the variance component generated by the regression mean (see 4.2.7). If the linear model is true, it becomes $\beta^T X_1^T C_H X_1 \beta$. It can be written as

$$\begin{aligned}
&\mu_1^T C_H \mu_1 = \beta^T X_1^T C_H X_1 \beta \\
&= \beta^T [\frac{X_1^T X_1 + X_2^T X_2}{n} - \frac{X_1^T \mathbf{1}_m \mathbf{1}_m^T X_1 + X_2^T \mathbf{1}_{n-m} \mathbf{1}_{n-m}^T X_2 + 2X_1^T \mathbf{1}_m \mathbf{1}_{n-m}^T X_2}{n^2}] \beta \\
&= n^{-1} \beta^T X^T (\mathbf{I}_n - n^{-1} \mathbf{1}\mathbf{1}^T) X\beta . \qquad (4.5.20)
\end{aligned}$$

Plugging (4.5.19) and (4.5.20) into (4.5.3), $E(\hat{\tau}_l)$ is approximately equal to

$$E(\hat{\tau}_l) \approx \frac{m-1}{n}\sigma^2 + n^{-1}\beta^T X^T (\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T) X\beta . \qquad (4.5.21)$$

Comparing (4.5.21) to (4.2.9), the predictive bias of $\hat{\tau}_l$ with respect to $\hat{\tau}$ is as follows

$$pbias(\hat{\tau}_l) \approx -\frac{n-m+1}{n}\sigma^2 . \qquad (4.5.22)$$

From (4.5.22), we find $\hat{\tau}_l$ underestimates variance by nearly $\sigma^2(n\text{-}m\text{+}1)/n$. In the special case when the covariate **x** is constant c (**x**=c), then X^{T}=(c...c) and $X^{\mathrm{T}}(\mathbf{I}^n - n^{-1}\mathbf{1}\mathbf{1}^{\mathrm{T}})X$= $c^2\mathbf{1}^{\mathrm{T}}(\mathbf{I}^n - n^{-1}\mathbf{1}\mathbf{1}^{\mathrm{T}})\mathbf{1}$=$c^2(\mathbf{1}^{\mathrm{T}} - n^{-1}\mathbf{1}^{\mathrm{T}}\mathbf{1}\mathbf{1}^{\mathrm{T}})\mathbf{1}$=0. The above term becomes zero. $E(\hat{\tau}_l)$ is just the first term in (4.5.3).

$$E(\hat{\tau}_l) = \sigma^2 tr(C_H) \approx \frac{m-1}{n}\sigma^2 . \qquad (4.5.23)$$

Random Regression Imputation

To preserve τ, one can use random regression imputation instead of regression imputation. The estimator of τ based on random imputation is obtained by replacing $\hat{Y}_2$ in (4.3.7) with (4.4.2). We denote it by $\hat{\tau}_{rl}$.

$$\hat{\tau}_{rl} = \hat{\tau}_{l} + \frac{1}{n}\hat{\mathbf{e}}_2^T H_{22}\hat{\mathbf{e}}_2 + \frac{m}{n^2(n-m)}\hat{\mathbf{e}}_2^T \mathbf{1}_{n-m}\mathbf{1}_{n-m}^T\hat{\mathbf{e}}_2$$

$$+\frac{2}{n}Y_1^T H_{21}^T H_{22}\hat{\mathbf{e}}_2 - \frac{2m(n-m)}{n^2}Y_1^T\left(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T\mathbf{1}_{n-m}}{n-m}\right)\mathbf{1}_{n-m}^T\hat{\mathbf{e}}_2, \qquad (4.5.24)$$

where the expression for $\hat{\tau}_{l}$ is given in (4.5.1). Plugging (4.5.21) into (4.5.24), the expectation of $\hat{\tau}_{rl}$ can be written as

$$E(\hat{\tau}_{rl}) = E(\hat{\tau}_{l}) + \frac{n-m-1}{n}\sigma^2 + \frac{m}{n^2}\sigma^2$$

$$= \sigma^2(tr(C_H)) + \mu_1^T C_H \mu_1 + \frac{n-m-1}{n}\sigma^2 + \frac{m}{n^2}\sigma^2$$

$$\approx \sigma^2 + n^{-1}\beta^T X^T(\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)X\beta\,. \qquad (4.5.25)$$

Comparing (4.5.25) to (4.2.9), we find $\hat{\tau}_{rl}$ is unbiased with respect to $\hat{\tau}$. This result suggests random regression should be used instead of regression imputation if the underlying model is regression model.

If the true model is RBF with $E(y)=\mu(\mathrm{x})=\Phi(\mathrm{x})W$, the simplified expression of $\mu_1^T C_H \mu_1$ seems very difficult to obtain. When $\mu(\mathrm{x})$ is far from $\mathrm{x}\beta$, $\mu_1^T C_H \mu_1$ is likely to have larger value, which leads to bigger $E(\hat{\tau}_{l})$ and $E(\hat{\tau}_{rl})$.

From above results, one can find the expectation of variance estimator is quite complex, because it contains both the variance term and mean term. The variance term is a product of σ^2 and the trace of C_H. The mean term is determined by the form of mean function. For regression model we have, $E(y)=\mu(\mathrm{x})=\mathrm{x}\beta$, $\mu_1^T = (\mathbf{x}_1\beta,\cdots,\mathbf{x}_m\beta)$. It can be simplified. For RBF neural networks model, $E(y)=\mu(\mathrm{x})=\Phi(\mathrm{x})W$, $\mu_1^T = (\Phi(\mathbf{x}_1)W,\cdots,\Phi(\mathbf{x}_m)W)$. It is

not an easy task to obtain the simple expression in general situation. But we can investigate the result in some special cases. This will be detailed in the next section.

RBF Based Imputation

As in section 4.4, the two imputation methods based on RBF model are RBF imputation described in (4.4.12) and random RBF imputation described in (4.4.14). If we plug (4.4.12) and (4.4.14) into (4.3.5), the estimators of τ based on RBF imputation and random RBF imputation are obtained. We denote them by $\hat{\tau}_R$ and $\hat{\tau}_{rR}$ respectively.

$$\hat{\tau}_R = Y_1^T C_H^R Y_1, \tag{4.5.26}$$

$$\hat{\tau}_{rR} = \hat{\tau}_R + \frac{1}{n}\hat{\mathbf{e}}_2^T H_{22}\hat{\mathbf{e}}_2 + \frac{m}{n^2(n-m)}\hat{\mathbf{e}}_2^T \mathbf{1}_{n-m}\mathbf{1}_{n-m}^T \hat{\mathbf{e}}_2$$

$$+\frac{2}{n}Y_1^T H_{21}^{(R)T} H_{22}\hat{\mathbf{e}}_2 - \frac{2m(n-m)}{n^2}Y_1^T\left(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^{(R)T}\mathbf{1}_{n-m}}{n-m}\right)\mathbf{1}_{n-m}^T\hat{\mathbf{e}}_2, \tag{4.5.27}$$

where

$$C_H^R = \frac{H_{11}}{n} + \frac{(H_{21}^{(R)})^T H_{22} H_{21}^{(R)}}{n} +$$

$$\frac{m(n-m)}{n^2}\left(\frac{\mathbf{1}_m}{m} - \frac{(H_{21}^{(R)})^T\mathbf{1}_{n-m}}{n-m}\right)\left(\frac{\mathbf{1}_m}{m} - \frac{(H_{21}^{(R)})^T\mathbf{1}_{n-m}}{n-m}\right)^T, \tag{4.5.28}$$

$$H_{21}^{(R)} = \Phi(X_2)(\Phi(X_1)^T\Phi(X_1))^{-1}\Phi(X_1)^T. \tag{4.5.29}$$

H_{22} and H_{11} are defined in (4.5.2). Since the form of RBF mean function is linear in W, if we give the same assumptions for X_1 and X_2 to $\Phi(X_1)$ and $\Phi(X_2)$, and assume the underlying model is RBF with $E(\mathbf{y})=\mu(\mathbf{x})=\Phi(\mathbf{x})W$, $E(\hat{\tau}_R)$ and $E(\hat{\tau}_{rR})$ have similar expressions of (4.5.21) and (4.5.25) respectively,

$$E(\hat{\tau}_R) = \sigma^2 tr(C_H^R) + \mu_1^T C_H^R \mu_1$$

$$= \frac{m-1}{n}\sigma^2 + n^{-1}W^T\Phi(X)^T(\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)\Phi(X)W, \quad (4.5.30)$$

$$E(\hat{\tau}_{rR}) = \sigma^2 + n^{-1}W^T\Phi(X)^T(\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)\Phi(X)W. \quad (4.5.31)$$

Comparing (4.5.30) and (4.5.31) with (4.2.11), the predictive biases of $\hat{\tau}_R$ and $\hat{\tau}_{rR}$ are obtained as,

$$pbias(\hat{\tau}_R) = -\frac{n-m+1}{n}\sigma^2, \quad (4.5.32)$$

$$pbias(\hat{\tau}_{rR}) = 0. \quad (4.5.33)$$

If the unbiased estimator of τ is used in (4.5.32), $pbias(\hat{\tau}_R)$ becomes $\frac{n-m}{n}\sigma^2$. We can conclude from (4.5.32) and (4.5.33) that RBF imputation deflates variance by $\sigma^2(n-m+1)/n$. Random RBF imputation gives unbiased estimator of τ under the assumption of RBF being the underlying model.

If the underlying model is a linear regression model, both RBF imputation and random RBF imputation are likely biased. The predictive biases can be written as,

$$pbias(\hat{\tau}_R) = -\frac{n-m+1}{n}\sigma^2 + \sigma^2 tr(C_H^R) +$$

$$\beta^T X_1^T C_H^R X_1 \beta - n^{-1}\beta^T X^T(\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)X\beta, \quad (4.5.34)$$

$$pbias(\hat{\tau}_{rR}) = \sigma^2 tr(C_H^R) + \beta^T X_1^T C_H^R X_1 \beta - n^{-1}\beta^T X^T(\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)X\beta. \quad (4.5.35)$$

It is difficult to simplify (4.5.34) and (4.5.35) in the general situation. That makes it hard to judge the magnitude of the predictive biases given by RBF imputations when the underlying model is actually a linear regression model. However we can investigate the scale in some special cases that may help us to understand the advantages and disadvantages of RBF imputation over regression imputations. This is detailed in section 4.7. In theory RBF model has the potential to be a linear regression model, it may suggest RBF imputations are robust to misspecifications of model assumptions, although

specifying RBF to be the underlying model might be an easy task in practice. In the ideal situation when the underlying model is a linear regression model, and RBF model is specified to be equivalent to the underlying model described by $E(y)=\mu(\mathbf{x})=\mathbf{x}\beta=\Phi(\mathbf{x})W$, the random RBF imputation still gives relatively unbiased estimator of Σ. Furthermore, if the underlying model is a non-linear model, the advantage of RBF imputations is evident by the possibility of being trained to represent the non-linear model.

One disadvantage with RBF model is the difficulty of finding the right specification for the underlying model, linear or non-linear. It is not guaranteed to obtain the right specification in terms of number of nodes and definition of centres. An efficient training algorithm is needed to get a reasonable specification.

4.6 Variance Properties of Estimators of μ

In this section we discuss the variance of estimators of μ. Under the assumptions described in (4.1.1) and (4.1.3), the variance of $\hat{\mu}_I$ in the form of $H_x Y_1$ can be expressed as

$$\begin{aligned} \mathrm{var}(\hat{\mu}_I) &= H_x^T \,\mathrm{var}(Y_1) H_x \\ &= \tau H_x^T H_x, \end{aligned} \qquad (4.6.1)$$

where H_x is a vector containing covariate $\mathbf{x}$, Y_1 is defined in the beginning of section 4.3, τ is the variance defined in (4.1.2). (4.6.1) tells that the variance of $\hat{\mu}_I$ is determined by the scale of $H_x^T H_x$. For convenience, we call it *variance coefficient of imputation* (*VCI*).

$$VCI = H_x^T H_x. \qquad (4.6.2)$$

Although *VCI* contains the covariate $\mathbf{x}$, it is independent of the assumption about the form of the mean function $\mu(\mathrm{x})$. Meanwhile, different imputations may give different *VCI*s. We expect *VCI* to be close to $1/n$ which is the *VCI* based on the true values of Y. One can judge the performance of an imputation by its *VCI* along with its predictive bias property

defined in (4.3.13). If *VCI* is bigger than $1/n$, the imputation gives less reliable estimator of μ. We discuss (4.6.1) and *VCI* with the regression imputations in (4.4.5) (4.4.6) and the RBF imputations in (4.4.15) (4.4.16) respectively.

Regression Imputation

For $\hat{\mu}_I$ based on regression imputation, H_x can be written as

$$H_I = \frac{1}{n}(\mathbf{1}_m^T + \mathbf{1}_{n-m}^T H_{21})^T, \tag{4.6.3}$$

where H_{21} is defined in (4.4.4). According to (4.6.1) the variance of $\hat{\mu}_I$ can be written as

$$\text{var}(\hat{\mu}_I) = \tau VCI_I, \tag{4.6.4}$$

where VCI_I is the variance coefficient of regression imputation,

$$\begin{aligned} VCI_I &= H_I^T H_I \\ &= \frac{1}{n^2}(m + 2\mathbf{1}_{n-m}^T H_{21}\mathbf{1}_m + \mathbf{1}_{n-m}^T H_{21}H_{21}^T\mathbf{1}_{n-m}) \\ &= \frac{1}{n^2}(m + \mathbf{1}_{n-m}^T X_2(X_1^T X_1)^{-1}(2X_1^T\mathbf{1}_m + X_2^T\mathbf{1}_{n-m})) \\ &= \frac{1}{n^2}(m + 2\mathbf{1}_{n-m}^T H_{21}\mathbf{1}_m + \mathbf{1}_{n-m}^T H_{212}\mathbf{1}_{n-m}), \end{aligned} \tag{4.6.5}$$

$$H_{212} = X_2(X_1^T X_1)^{-1}X_2^T. \tag{4.6.6}$$

The magnitude of VCI_I is determined by the matrix H_{21} and H_{212}. It is difficult to simplify (4.6.5). However the expression of VCI_I can be used as the basis of comparison in some special scenarios. For example if **x**=1, $H_{21} = \frac{1}{m}\mathbf{1}_{n-m}\mathbf{1}_m^T$, $H_{212} = \frac{1}{m}\mathbf{1}_{n-m}\mathbf{1}_{n-m}^T$, therefore VCI_I

$=1/m$. In this case, $\hat{\mu}_l$ has bigger variance than the estimator of μ based on true values (4.2.1).

The variance of $\hat{\mu}_{rl}$ (see 4.4.6) is the sum of var($\hat{\mu}_l$) and $\frac{n-m}{n^2}\tau$, since we assume $\hat{\mathbf{e}}_2$ is independent of Y_1 (see 4.4.2). The resulting variance coefficient of random regression imputation (VCI_{rl}) is obtained as follows

$$VCI_{rl} = VCI_l + \frac{n-m}{n^2}$$

$$= H_l^T H_l + \frac{n-m}{n^2}. \qquad (4.6.7)$$

(4.6.7) reveals that the estimator of μ based on random regression imputation has bigger variance than that based on regression imputation by a term of $\frac{n-m}{n^2}\tau$, but the gain is the unbiasedness for variance estimation described in (4.5.25).

RBF Imputation

For RBF imputation, the variance of $\hat{\mu}_R$ (4.4.15) can also be put in to the form of (4.6.1) with H_x given below

$$H_R = \frac{1}{n}(\mathbf{1}_m^T + \mathbf{1}_{n-m}^T H_{21}^{(R)})^T. \qquad (4.6.8)$$

H_R is obtained by applying (4.6.1) to (4.4.15), where $H_{21}^{(R)}$ is defined in (4.5.29). Plugging (4.6.8) into (4.6.2), the resulting VCI_R is obtained as

$$VCI_R = H_R^T H_R$$

$$= \frac{1}{n^2}(m + 2\mathbf{1}_{n-m}^T H_{21}^{(R)} \mathbf{1}_m + \mathbf{1}_{n-m}^T H_{21}^{(R)} H_{21}^{(R)T} \mathbf{1}_{n-m})$$

$$= \frac{1}{n^2}(m + 2\mathbf{1}_{n-m}^T H_{21}^{(R)} \mathbf{1}_m + \mathbf{1}_{n-m}^T H_{212}^{(R)} \mathbf{1}_{n-m}), \qquad (4.6.9)$$

where

$$H_{212}^{(R)} = \Phi(X_2)(\Phi(X_1)^T \Phi(X_1))^{-1} \Phi(X_2)^T. \qquad (4.6.10)$$

Again, the magnitude of VCI_R is not obvious, which makes the comparison of VCI_R and VCI_l more difficult. However we can look at some special situations to find out whether RBF based imputation can give some improvement over regression based imputation.

Similar to var($\hat{\mu}_{rl}$), under the independence assumption about $\hat{\mathbf{e}}_2$ and Y_1, var($\hat{\mu}_{rR}$) can be written as the sum of var($\hat{\mu}_R$) and $\frac{n-m}{n^2}\tau$.

$$\text{var}(\hat{\mu}_{rR}) = \text{var}(\hat{\mu}_R) + \frac{n-m}{n^2}\tau. \qquad (4.6.11)$$

From (4.6.11), the variance coefficient of random RBF imputation (VCI_{rR}) can be obtained as

$$VCI_{rR} = VCI_R + \frac{n-m}{n^2}. \qquad (4.6.12)$$

Comparing VCI_{rR} in (4.6.12) to VCI_R in (4.6.9), the estimator of μ based on random RBF imputation has bigger variance than that of RBF imputation. But the comparison of VCI_{rR} to VCI_l and VCI_{rl} remains difficult, and will be discussed in some special situations in section 4.7. For example if $\mathbf{x}$=1, there exists $\Phi(\cdot)$ such that $\Phi(\mathbf{x})$=1. Then $H_{21}^{(R)} = \frac{1}{m}\mathbf{1}_{n-m}\mathbf{1}_m^T = H_{21}$, $H_{212}^{(R)} = \frac{1}{m}\mathbf{1}_{n-m}\mathbf{1}_{n-m}^T = H_{212}$, $VCI_l = VCI_R = 1/m$, $VCI_{rR} = VCI_{rl} = 1/m+(n-m)/n^2$. Therefore, in this special case, regression imputation is equivalent to the counterparts of RBF imputation in terms of the variance of estimators of μ.

4.7 Comparison of Linear Regression Imputation and RBF Imputation in two Special Situations

In the last three sections we obtained some general expressions for the expectation and the variance of the mean estimator and the expectation of the estimator of population variance based on linear regression imputation and RBF neural network imputation. These general results did not enable the performance of the two methods to be compared very easily. In this section we compare them in two special scenarios, where the difference is easier to understand.

Before we compare the difference between the two models, we should mention the fact that the RBF model can be trained (or adjusted) to be either linear or non-linear functions (Bishop, 1996). Here we give a special RBF model that is equivalent to linear regression model in terms of model prediction.

Suppose the centres (or nodes) of the RBF model are defined as the original data points $x_1,\ldots,x_n$. We also assume $x_1,\ldots,x_n$ are scalar and distinct. The basis function is $\phi_c(x)=exp(-(x-c)^2/\lambda)$, where c is the node or centre. For an arbitrary small number α, there exists a value λ with $\lambda<md/\alpha$, where $md=min(|x_i-x_j|^2, i\neq j,\ i,j=1\ldots n)$ such that $\phi_c(x_i)\approx 1$, $c=x_1,\ldots,x_n$, $i=1\ldots n$. For simplicity we use a similar basis function that can be approximated by the above basis function,

$$\phi_c(x) = \begin{cases} 1 & |x-c| < \lambda\alpha \\ 0 & otherwise \end{cases}. \qquad (4.7.1)$$

With this basis function, the $\Phi(X)=(\phi_{xj}(x_i))_{\substack{i=1,\ldots,n \\ j=1,\ldots,n}}$ becomes an elementary matrix $\mathbf{I}_n$. If the weight vector W is set to be $X\beta$, $\Phi(X)W$ becomes $X\beta$, which is the mean function of regression model. In practice the above basis function can be obtained approximately in RBF training process if the proper initial value of λ is provided.

The following comparison is based on assumptions given in (4.1.1) to (4.1.3). One additional assumption is about the mean of **y** given **x**. In regression model we assume $E(y|\mathbf{x})=\mathbf{x}\beta$, in RBF model we assume $E(y|\mathbf{x})=\Phi(\mathbf{x})W$, which is where the difference comes from.

4.7.1 Scenario I: x=1

In this special case we can write $E(y|\mathbf{x})=\mu$. The variance of mean estimator and the expectation of the estimator of population variance do not depend on the mean function assumption.

Regression Based Imputation

When x equals one, the matrix H_{21} in (4.4.4) becomes

$$H_{21} = \frac{1}{m}\mathbf{1}_{n-m}\mathbf{1}_m^T. \qquad (4.7.2)$$

If we plug (4.7.2) into (4.4.5), $\hat{\mu}_l$ becomes,

$$\hat{\mu}_l = \frac{1}{n}(\mathbf{1}_m^T + \mathbf{1}_{n-m}^T H_{21})Y_1 = \frac{1}{m}\mathbf{1}_m^T Y_1, \qquad (4.7.3)$$

which is the mean of the observed y_i values so that regression imputation reduces to mean imputation. Under assumptions (4.1.1) to (4.1.3), $E(\hat{\mu}_I)$ and $\text{var}(\hat{\mu}_I)$ are obtained as,

$$E(\hat{\mu}_I) = \mu, \tag{4.7.4}$$

$$\text{var}(\hat{\mu}_I) = \frac{\sigma^2}{m}. \tag{4.7.5}$$

From (4.7.4) and (4.7.5), we find $\hat{\mu}_I$ is an unbiased estimator, but the variance of $\hat{\mu}_I$ is increased by imputation compared to the estimator based on true values. In this special case $\tau=\sigma^2$.

If we plug (4.7.2) into (4.5.2), the resulting C_H is obtained as follows,

$$C_H = \frac{H_{11}}{n}. \tag{4.7.6}$$

Applying (4.7.6) to (4.5.3), $E(\hat{\tau}_I)$ can be written as

$$E(\hat{\tau}_I) = \frac{m-1}{n}\sigma^2. \tag{4.7.7}$$

The above result (4.7.7) shows that the mean imputation deflates. The drawback of mean imputation can be overcome by using random regression imputation. If we plug (4.7.7) into (4.5.25), we find $\hat{\tau}_{rI}$ is an unbiased estimator of τ. Meanwhile the variance of mean estimator increases further by $\frac{n-m}{n^2}\tau$ (see 4.6.7).

In this special case, the term containing the conditional mean given **x** is equal to zero, therefore the result doesn't depend on the mean expression assumption, it only depends on the assumptions given in (4.1.1) to (4.1.3).

RBF Imputation

In this special case, the RBF model becomes a linear regression model, if we express the model using the following notation,

$$\phi(x)= \frac{1}{1+(x-1)^2}, \tag{4.7.8}$$

where $\phi(x)$ is a radial basis function defined at c=1. Since x_i=1, i=1...m then $\phi(x_i)$=1, i=1...m. The RBF becomes a linear regression model. Since

$$\mu(\mathbf{x})=E(y|\mathbf{x})=\phi(\mathbf{x})w=w=\mu. \tag{4.7.9}$$

It follows that $C_H^R = C_H$. Even if the centre is not at **x**=1, a minor modification in basis function gives the same result. The other thing is the scale of the basis function (like the kernel function in non-parametric regression, Hardle, 1989). It does not change the prediction given **x**. When the basis function is multiplied by a factor k, $\hat{W}$ is multiplied by $1/k$. Later on we will change the scale when we need simple expressions.

Now let the centre be $c\neq1$ and let $\phi(x)= \frac{1+(1-c)^2}{1+(x-c)^2}$. This leads to the same conclusion of (4.7.9). By now we assume only one node (or centre) is defined. What will happen if two or more centres are used in this particular situation? Let us consider the situation with two centres defined. Whatever centres are defined, the new covariates transformed by the basis functions are collinear. A perturbation is needed to tackle the singularity problem when estimating W. The side effect of perturbation is that it may distort the true result.

Suppose two centres c_1 c_2 are defined. A perturbation is applied to $\Phi(X)$. The perturbation is generated from the standard normal distribution $\Delta\sim N(0,1)$, then the perturbed data of the transformed X is:

$$\Phi(X)+\Delta=\begin{bmatrix} 1+\Delta_{11} & \cdots & \dfrac{1+(1-c_1)^2}{1+(1-c_2)^2}+\Delta_{21} \\ \vdots & \cdots & \vdots \\ 1+\Delta_{m1} & \cdots & \dfrac{1+(1-c_1)^2}{1+(1-c_2)^2}+\Delta_{mm} \end{bmatrix} \tag{4.7.10}$$

The asymptotic estimator of W_Δ is

$$W_\Delta \approx \begin{bmatrix} \bar{\mathbf{y}}_1 \\ 0 \end{bmatrix} + \begin{bmatrix} 2\bar{\Delta}\bar{\mathbf{y}}_1 + \mathrm{cov}(\mathbf{y},\Delta) + \bar{\Delta}\,\mathrm{cov}(\mathbf{y},\Delta) \\ \mathrm{cov}(\mathbf{y},\Delta) \end{bmatrix}. \tag{4.7.11}$$

Where $\bar{\mathrm{y}}_1$ is the sample mean of **y** based on the observed data, $\bar{\Delta}$ is the sample mean of Δ, $\mathrm{cov}(\mathrm{y},\Delta)$ is the sample covariance of y and Δ. Apparently the perturbation could distort the estimation by the second term in the above expression. This result reveals when the data centres result in collinearity, the estimation could be distorted. Therefore the selection of the centres should be carefully carried out.

4.7.2 Scenario II: x has two different values (a, b)

Suppose **x** is still a scalar covariate taking two possible values a and b $(a\neq b)$. For convenience we give the following notation.

$S_{ax1}=\{i,\ x_i=a,\ i=1,m\}$, $S_{bx1}=\{i,\ x_i=b,\ i=1,m\}$, $S_{ax2}=\{i,\ x_i=a,\ i=m+1,n\}$, $S_{bx2}=\{i,\ x_i=b,\ i=m+1,n\}$, $n_{ax1}=^{\#}\{i,\ i\in S_{ax1}\}$, $n_{bx1}=^{\#}\{i,\ i\in S_{bx1}\}$, $n_{ax2}=^{\#}\{i,\ i\in S_{ax2}\}$, $n_{bx2}=^{\#}\{i,\ i\in S_{bx2}\}$.

Regression Based Imputation

The unbiasedness property of estimators of μ and τ under the linear regression model has been shown in (4.4.6) and (4.5.22). Here we focus on the property of predictive bias when the underlying model is a RBF model (see 4.4.11 and 4.5.3).

The regression imputation of Y_2 is given by (4.4.1). In this special case it becomes

$$\hat{Y}_2 = X_2\hat{\beta} = (a^2 n_{ax_1} + b^2 n_{bx_1})^{-1} X_2 X_1^T Y_1 = H_{21} Y_1, \tag{4.7.12}$$

where

$$\begin{aligned}
H_{21} &= (a^2 n_{ax_1} + b^2 n_{bx_1})^{-1} X_2 X_1^T \\
&= (a^2 n_{ax_1} + b^2 n_{bx_1})^{-1} (a\mathbf{1}_{ax_2} + b\mathbf{1}_{bx_2})(a\mathbf{1}_{ax_1} + b\mathbf{1}_{bx_1})^T \\
&= (a^2 n_{ax_1} + b^2 n_{bx_1})^{-1} (a^2 \mathbf{1}_{ax_2}\mathbf{1}_{ax_1}^T + ab\mathbf{1}_{ax_2}\mathbf{1}_{bx_1}^T + ab\mathbf{1}_{bx_2}\mathbf{1}_{ax_1}^T + b^2\mathbf{1}_{bx_2}\mathbf{1}_{bx_1}^T), \qquad (4.7.13)
\end{aligned}$$

$$X_2 = a\mathbf{1}_{ax2} + b\mathbf{1}_{bx2},$$

$$X_1 = a\mathbf{1}_{ax1} + b\mathbf{1}_{bx1},$$

$$\mathbf{1}_{ax2} = (\delta_{a2,i})_{(n-m)\times 1},$$

$$\mathbf{1}_{bx2} = (\delta_{b2,i})_{(n-m)\times 1},$$

$$\mathbf{1}_{ax1} = (\delta_{a1,i})_{m\times 1},$$

$$\mathbf{1}_{bx1} = (\delta_{b1,i})_{m\times 1},$$

$$\delta_{a_2,i} = \begin{cases} 1, x_i = a, i = m+1,\ldots,n \\ 0, \quad otherwise \end{cases},$$

$$\delta_{b_2,i} = \begin{cases} 1, x_i = b, i = m+1,\ldots,n \\ 0, \quad otherwise \end{cases},$$

$$\delta_{a_1,i} = \begin{cases} 1, x_i = a, i = 1,\ldots,m \\ 0, \quad otherwise \end{cases},$$

$$\delta_{b_1,i} = \begin{cases} 1, x_i = b, i = 1,\ldots,m \\ 0, \quad otherwise \end{cases}.$$

This notation can be easily extended to the multiple value case of x. Therefore the conclusion about unbiasedness of estimators in the presence of imputation obtained from the two-value case is applicable to more general situations.

For convenience, we assume the underlying model is

$$E(y|\mathbf{x})=\phi(\mathbf{x})w, \tag{4.7.14}$$

where w is assumed to be 1 for mathematical simplicity, there are two nodes at a and b respectively and $\phi(\mathbf{x})$ is as follows

$$\phi(x)=\frac{1+(b-a)^2}{(b-a)^2[1+(x-a)^2]}-\frac{1}{(b-a)^2}=\begin{cases}1 & if \quad x=a\\ 0 & if \quad x=b\end{cases}. \tag{4.7.15}$$

The basis function $\phi(x)$ will be re-used in the following paragraph. This will make the comparison easier. If we plug (4.7.13) to (4.7.15) into (4.4.11), the predictive bias of $\hat{\mu}_l$ is obtained,

$$\begin{aligned}
pbias(\hat{\mu}_l) &= \frac{1}{n}\mathbf{1}_{n-m}^T H_{21}\Phi(X_1)W-\frac{1}{n}\mathbf{1}_{n-m}^T\Phi(X_2)W\\
&=\frac{1}{n}\mathbf{1}_{n-m}^T H_{21}(\Phi_a(X_1)+\Phi_a(X_1))-\frac{1}{n}\mathbf{1}_{n-m}^T(\Phi_a(X_2)+\Phi_a(X_2))\\
&=\frac{1}{n}\mathbf{1}_{n-m}^T H_{21}\mathbf{1}_m-\frac{1}{n}\mathbf{1}_{n-m}^T\mathbf{1}_{n-m}\\
&=\frac{m(n-m)}{n}\bar{X}_1\bar{X}_2-\frac{n-m}{n}.
\end{aligned} \tag{4.7.16}$$

(4.7.16) shows the predictive bias of $\hat{\mu}_l$ may not be neglectable if the underlying model is a RBF model.

To understand the magnitude of the predictive bias of the estimator of τ based on regression imputation, we need a simple expression of C_H defined in (4.5.2). We start with $H_{21}^T H_{22} H_{21}$ and $(\frac{\mathbf{1}_m}{m}-\frac{H_{21}^T}{n-m})(\frac{\mathbf{1}_m}{m}-\frac{H_{21}^T}{n-m})^T$. To simplify the notation, we give a further assumption that is likely to hold in reality, $\frac{n_{ax_1}}{n_{bx_1}}=\frac{n_{ax_2}}{n_{bx_2}}=r$. In this case

$$H_{21}^T H_{22} H_{21} = \lambda_1 \mathbf{1}_{ax_1} \mathbf{1}_{ax_1}^T + \lambda_2 \mathbf{1}_{bx_1} \mathbf{1}_{bx_1}^T,$$

$$(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T}{n-m})(\frac{\mathbf{1}_m}{m} - \frac{H_{21}^T}{n-m})^T = \lambda_3 \mathbf{1}_{ax_1} \mathbf{1}_{ax_1}^T + \lambda_4 \mathbf{1}_{bx_1} \mathbf{1}_{bx_1}^T,$$

where

$$\lambda_1 = \frac{n_{ax2} n_{bx2} a^2 (a-b)^2}{(n-m)(a^2 n_{ax2} + b^2 n_{bx2})^2},$$

$$\lambda_2 = \frac{n_{ax2} n_{bx2} b^2 (a-b)^2}{(n-m)(a^2 n_{ax2} + b^2 n_{bx2})^2},$$

$$\lambda_3 = \frac{b^2 (a-b)^2 (n-m)}{n^2 m (a^2 r + b^2)^2},$$

$$\lambda_4 = \frac{a^2 r^2 (a-b)^2 (n-m)}{n^2 m (a^2 r + b^2)^2}.$$

Then (4.5.2) becomes

$$C_H = \frac{H_{11}}{n} + (\frac{\lambda_1}{n} + \frac{m(n-m)\lambda_3}{n^2}) \mathbf{1}_{ax_1} \mathbf{1}_{ax_1}^T + (\frac{\lambda_2}{n} + \frac{m(n-m)\lambda_4}{n^2}) \mathbf{1}_{bx_1} \mathbf{1}_{bx_1}^T. \qquad (4.7.17)$$

If we plug (4.7.17) into (4.5.3), the expectation of the estimator of τ under regression imputation becomes

$$E(\hat{\tau}_l) = \sigma^2 (\frac{m-1}{n} + \frac{(n^2 - m^2)(a-b)^2 r}{n^2 m (a^2 r + b^2)(1+r)}) +$$

$$\frac{(n-m)(a-b)^2 r^2 (a^2 rn + b^2 m)}{n^2 (a^2 r + b^2)^2 (1+r)^2}. \qquad (4.7.18)$$

The expectation of the estimator of τ based on true values is obtained by applying (4.2.11) to this special situation,

$$E(\hat{\tau}) \approx \sigma^2 + n^{-1} W^T \Phi(X)^T (\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)\Phi(X) W$$

$$= \sigma^2 + n^{-1}\mathbf{1}^T(\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T)\mathbf{1}$$

$$= \sigma^2 . \qquad (4.7.19)$$

The predictive bias of $\hat{\tau}_l$ can be obtained by (4.7.18)-(4.7.19)

$$pbias(\hat{\tau}_l) = \sigma^2(-\frac{n-m+1}{n} + \frac{(n^2-m^2)(a-b)^2 r}{n^2 m(a^2 r + b^2)(1+r)}) +$$

$$\frac{(n-m)(a-b)^2 r^2 (a^2 rn + b^2 m)}{n^2 (a^2 r + b^2)^2 (1+r)^2} . \qquad (4.7.20)$$

Based on (4.5.25), the predictive bias of $\hat{\tau}_{rl}$ can be obtained from (4.7.20) as

$$pbias(\hat{\tau}_{rl}) = \frac{(n^2-m^2)(a-b)^2 r}{n^2 m(a^2 r + b^2)(1+r)}\sigma^2 +$$

$$\frac{(n-m)(a-b)^2 r^2 (a^2 rn + b^2 m)}{n^2 (a^2 r + b^2)^2 (1+r)^2} . \qquad (4.7.21)$$

To simplify the above results, we use the assumption given by (4.5.5). The asymptotic value of $pbias(\hat{\tau}_l)$ is $\frac{r_0(a-b)^2 r^2 (a^2 r + b^2(1-r_0))}{(a^2 r + b^2)^2 (1+r)^2} - r_0\sigma^2 < \frac{r_0(a-b)^2 r^2}{(1+r)^2} - r_0\sigma^2$. In general, if the proportion of missing values (r_0) is small, the predictive bias of regression imputation could be negligible. Meanwhile if $r = \frac{\sigma}{\sigma + |a-b|}$ holds, $pbias(\hat{\tau}_l)$ turns to be 0, therefore regression imputation is unbiased. In this special case, regression imputation does better than random regression imputation. When $r < \frac{\sigma}{\sigma + \sqrt{2}|a-b|}$, random regression imputation produces less predictive bias than regression imputation. Otherwise regression imputation does better than random regression imputation.

RBF Imputation

For RBF neural networks there are several ways to specify it, mainly the ways to determine the centres. We classify it into three possible situations:

(1) Only one centre is selected,

(2) Two centres,

(3) Three or more centres.

As before, our interest is the relative basis of estimators based on RBF imputation when the underlying model is a regression model. We denote the underlying model as $E(y|\mathbf{x})=\mathbf{x}\beta$. Let us explore each of the three specifications.

RBF with one centre

If the centre is coincidentally defined to be $c=\frac{a+b}{2}$, let the radial basis function be

$$\phi(x)=\frac{1+(\frac{a-b}{2})^2}{1+(x-\frac{a+b}{2})^2}. \qquad (4.7.22)$$

Again the term in the numerator of (4.7.22) is just for simplicity. Then the transformed data becomes $\Phi(X)=1$. If we plug (4.7.22) into (4.4.17), the predictive bias of $\hat{\mu}_R$ is obtained as

$$pbias(\hat{\tau}_R)=\frac{1}{n}\mathbf{1}_{n-m}^T(\Phi(X_2)((\Phi(X_1)^T\Phi(X_1))^{-1}\Phi(X_1)^T\mu(X_1)-\mu(X_2))$$

$$=\frac{1}{n}\mathbf{1}_{n-m}^T(\mathbf{1}_{n-m}(m)^{-1}\mathbf{1}_m^T\mathbf{1}_m^T\beta-\mathbf{1}_{n-m}^T\beta)$$

$$=0.$$

Therefore RBF imputation gives unbiased mean estimator. The variance of the mean estimator is increased to $\frac{\sigma^2}{m}$. (see 4.7.5, 4.2.1). The expectation of the variance estimator

based on this RBF imputation is $\frac{(m-1)\sigma^2}{n}$. It underestimates the variance by an amount of $\frac{(n-m+1)\sigma^2}{n}$. The predictive bias can be obtained by subtracting (4.2.9) from (4.7.7) after plugging (4.7.13) into (4.2.9).

$$pbias(\hat{\tau}_R) = -\frac{n-m+1}{n}\sigma^2 - \frac{\beta^2}{n}\sum_{i=1}^{n}(x_i-\bar{x})^2 . \qquad (4.7.23)$$

When a and b are far apart, this one node RBF imputation gives very biased variance estimation. In other words the estimator of variance based on RBF imputation underestimates variance; in the meantime it likely inflates the variance of the mean estimator compared to that based on true values. The likeliness depends on the magnitude of the ratio r. If r shrinks to zero or grows to infinity the covariate data becomes the constant $\mathbf{x}=a(or\ b)$, the difference will be very small. The advantage of the regression imputation is most significant when r equals 1(r=1).

If the centre is defined not equal to $c=\frac{a+b}{2}$, the RBF model is equivalent to the underlying linear regression model. The estimator of W is different from $\hat{\beta}$ just in scale. Therefore regression imputation is equivalent to RBF imputation, which results random regression equivalent to random RBF imputation. This is reflected in the following basis function.

$$\phi(x)=\frac{1}{1+(x-c)^2}=\begin{cases}\frac{1}{1+(a-c)^2} & x=a\\ \frac{1}{1+(b-c)^2} & x=b\end{cases}. \qquad (4.7.24)$$

RBF with two centres

For convenience we define the two centres as $c_1=a$, $c_2=b$. The different definition of c_1 and c_2 and the scale of the kernel function will not change the RBF imputation. It will

change the scale of $\hat{W}$ only. We slightly modify the previous basis function to make the mathematical expressions simpler. The new basis functions are as follows,

$$\phi(x,c_1)=\frac{1+(b-a)^2}{(b-a)^2[1+(x-c_1)^2]}-\frac{1}{(b-a)^2}=\begin{cases}1 & if \quad x=a\\ 0 & if \quad x=b\end{cases}, \tag{4.7.25}$$

$$\phi(x,c_2)=\frac{1+(b-a)^2}{(b-a)^2[1+(x-c_2)^2]}-\frac{1}{(b-a)^2}=\begin{cases}0 & if \quad x=a\\ 1 & if \quad x=b\end{cases}. \tag{4.7.26}$$

Under this specification the RBF neural networks transforms the original covariate data into the following data matrix,

$$\Phi(X)=\lfloor\delta_{ij}\rfloor_{m\times 2}, \tag{4.7.27}$$

where

$$\delta_{ij}=\begin{cases}1 & x_i=c_j\\ 0 & others\end{cases}, i=1\ldots m, j=1,2.$$

If we plug (4.7.27) into (4.4.17), the predictive bias of $\hat{\mu}_R$ is obtained as

$$pbias(\hat{\tau}_R)=\frac{1}{n}\mathbf{1}_{n-m}^T(\Phi(X_2)((\Phi(X_1)^T\Phi(X_1))^{-1}\Phi(X_1)^T X_1-X_2)\beta$$

$$=\frac{1}{n}\mathbf{1}_{n-m}^T(\Phi(X_2)\begin{bmatrix}n_{ax1} & 0\\ 0 & n_{bx1}\end{bmatrix}^{-1}\Phi(X_1)^T X_1-X_2)\beta$$

$$=\frac{1}{n}(\mathbf{1}_{n-m}^T\Phi(X_2)\begin{bmatrix}n_{ax1} & 0\\ 0 & n_{bx1}\end{bmatrix}^{-1}\begin{bmatrix}an_{ax1}\\ bn_{bx1}\end{bmatrix}-\mathbf{1}_{n-m}^T X_2)\beta$$

$$=\frac{1}{n}((n_{ax2},n_{bx2})\begin{bmatrix}a\\ b\end{bmatrix}-\mathbf{1}_{n-m}^T X_2)\beta$$

$$=\frac{n-m}{n}(\bar{X}_2-\bar{X}_2)\beta=0. \tag{4.7.28}$$

Hence RBF imputation gives an unbiased mean estimator when the centres are properly defined.

Under the two centre RBF model, the estimator of W becomes

$$\hat{W} \approx \begin{bmatrix} \bar{y}_{S_a} \\ \bar{y}_{S_b} \end{bmatrix}, \tag{4.7.29}$$

where

$$\bar{y}_{S_a} = n_a^{-1} \sum_{i \in S_a} y_i ,$$

$$\bar{y}_{S_b} = n_b^{-1} \sum_{i \in S_b} y_i .$$

The estimators and the imputations are actually the local averages of y at x=a and x=b. Following the previous notation, the C_H becomes

$$C_H = \frac{H_{11}}{n} + \lambda_5 \mathbf{1}_{ax_1} \mathbf{1}_{ax_1}^T + \lambda_6 \mathbf{1}_{bx_1} \mathbf{1}_{bx_1}^T , \tag{4.7.30}$$

where

$$\lambda_5 = \frac{nmn_{ax_2} n_{bx_2} + (n_{ax_1} n_{bx_2} - n_{ax_2} n_{bx_1})^2}{n^2 m(n-m) n_{ax_1}^2},$$

$$\lambda_6 = \frac{nmn_{ax_2} n_{bx_2} + (n_{ax_1} n_{bx_2} - n_{ax_2} n_{bx_1})^2}{n^2 m(n-m) n_{bx_1}^2}.$$

Plugging (4.7.30) into (4.5.34), the predictive bias of $\hat{\tau}_R$ is obtained,

$$\begin{aligned} pbias(\hat{\tau}_R) &= -\frac{n-m+1}{n}\sigma^2 + \sigma^2 tr(C_H^R) \\ &\beta^T X_1^T C_H^R X_1 \beta - n^{-1} \beta^T X^T (\mathbf{I}_n - n^{-1}\mathbf{1}\mathbf{1}^T) X \beta \\ &= \frac{r(n-m)}{n(1+r)^2} \bullet (-2ab\beta^2). \end{aligned} \tag{4.7.31}$$

From (4.7.31), we can conclude that when n-m is much smaller than n, RBF imputation gives a variance estimator with small bias. However the promising performance of RBF imputation depends on how it is specified in terms of the way the centres are defined. The results in this section provide evidence that the RBF model can be trained to give promising imputation even when the underlying model is not RBF.

RBF with three or more centres

If three or more different centres are defined, the transformed data matrix is collinear. Although the perturbation can be used to obtain estimation practically, as explained in the beginning of this section, the parameter estimation is likely to be distorted. Therefore we should avoid setting too many centres since this may result in bad conditions in the transformed data. The difficulty is we never know where the limit is in practice. That is one of the disadvantages of RBF neural networks imputation.

Based on above result, there is no easy answer for the properties of the regression imputation and the RBF imputation. Basically under the linear assumption, if there are too many nodes (or data centres) in RBF models, collinearity could occur, which leads to distorted results. On the other hand, if too few nodes are selected, the variation in covariates will not be fully represented, which also gives bad result. When the nodes are properly defined to represent the functional relationship between y and its covariates, the RBF neural networks imputation is likely to outperform linear regression imputation.

4.8 Multinomial Logit Imputation

The last seven sections are concerned with imputation for continuous variables. In this section we outline some considerations for categorical data imputation. This is more like a rough idea than precise results. Further work needs to be done to make the idea more precise.

Suppose $\mathbf{y}$ is a multinomial variable with expectation $\mu(\mathbf{x})$.

$$\boldsymbol{\mu}(\mathbf{x}) = \boldsymbol{\pi}(\mathbf{x}) = \begin{bmatrix} \pi_1(\mathbf{x}) \\ \vdots \\ \pi_{p+1}(\mathbf{x}) \end{bmatrix}. \tag{4.8.1}$$

For convenience we denote $E(Y_1)$ as

$$E(Y_1) = \begin{bmatrix} \pi_1(\mathbf{x}_1) \cdots \pi_{p+1}(\mathbf{x}_1) \\ \vdots \\ \pi_1(\mathbf{x}_m) \cdots \pi_{p+1}(\mathbf{x}_m) \end{bmatrix}, \tag{4.8.2}$$

and

$$\text{var}(\mathbf{y} \mid \mathbf{x}) = \begin{bmatrix} \pi_1(\mathbf{x})(1-\pi_1(\mathbf{x})) \cdots \pi_1(\mathbf{x})\pi_{p+1}(\mathbf{x}) \\ \vdots \\ \pi_{p+1}(\mathbf{x})\pi_1(\mathbf{x}) \cdots \pi_{p+1}(\mathbf{x})(1-\pi_{p+1}(\mathbf{x})) \end{bmatrix}. \tag{4.8.3}$$

A linear logit model can be used to predict class membership $\boldsymbol{\pi}(\mathbf{x})$. The imputation can be made based on the predicted distribution $\hat{\boldsymbol{\pi}}(\mathbf{x}_i)$.

$$\hat{\boldsymbol{\pi}}(\mathbf{x}_i) = \begin{bmatrix} \hat{\pi}_1(\mathbf{x}_i) \\ \vdots \\ \hat{\pi}_{p+1}(\mathbf{x}_i) \end{bmatrix}. \tag{4.8.4}$$

One way of imputation is to choose the category with highest probability. This mechanism can be described as

$$\Pr(\hat{y}_{ik} = 1 \mid \hat{\pi}(\mathbf{x}_i)) = \begin{cases} 1, \hat{\pi}_k(\mathbf{x}_i) = \max_j(\hat{\pi}_j(\mathbf{x}_i)) \\ 0 \qquad \textit{otherwise} \end{cases} \quad i=\text{m+1}\ldots\text{n},\ k=1\ldots\text{p+1}. \tag{4.8.5}$$

The other way of imputation is a random draw from the predicted distribution, which is carried out by calculating the cumulative distribution based on the predicted distribution, generating a random number in (0, 1) and finding the corresponding category based on the

location of the random number in the cumulative distribution. If this method is used, the following result can be obtained

$$E(\hat{Y}_2 \mid Y_1) = \begin{bmatrix} \hat{\pi}_1(\mathbf{x}_{m+1}) \cdots \hat{\pi}_{p+1}(\mathbf{x}_{m+1}) \\ \vdots \\ \hat{\pi}_1(\mathbf{x}_n) \cdots \hat{\pi}_{p+1}(\mathbf{x}_n) \end{bmatrix}, \tag{4.8.6}$$

$$\text{var}(\hat{\mathbf{y}}_i \mid \mathbf{x}_i, Y_1) = \begin{bmatrix} \hat{\pi}_1(\mathbf{x}_i)(1-\hat{\pi}_1(\mathbf{x}_i)) \cdots \hat{\pi}_1(\mathbf{x}_i)\pi_{p+1}(\mathbf{x}_i) \\ \vdots \\ \hat{\pi}_{p+1}(\mathbf{x}_i)\hat{\pi}_1(\mathbf{x}_i) \cdots \hat{\pi}_{p+1}(\mathbf{x}_i)(1-\hat{\pi}_{p+1}(\mathbf{x}_i)) \end{bmatrix}. \tag{4.8.7}$$

$i=m+1 \ldots n.$

Based on (4.3.3) and (4.8.2), we can get the following results

$$E(\hat{\mu}_I) = \frac{1}{n}\left(\sum_{i=1}^{m} E(y_i \mid \mathbf{x}_i) + \sum_{j=m+1}^{n} E(\hat{y}_j \mid \mathbf{x}_j)\right)$$

$$= \frac{1}{n}(\mathbf{1}_m^T E(Y_1) + \mathbf{1}_{n-m}^T E(\hat{Y}_2)), \tag{4.8.8}$$

$$\text{var}(\hat{\mu}_I) = \frac{1}{n^2}\left(\sum_{i=1}^{m} \text{var}(\mathbf{y}_i \mid \mathbf{x}_i) + \sum_{j=m+1}^{n} \text{var}(\hat{\mathbf{y}}_j \mid \mathbf{x}_j)\right). \tag{4.8.9}$$

Consider the census situation as a special case, and suppose that for large N the parameter of interest is

$$\mu = \frac{1}{N}\mathbf{1}_N^T \pi(X) = \frac{1}{N}(\mathbf{1}_m^T \pi(X_1) + \mathbf{1}_{N-m}^T \pi(X_2)). \tag{4.8.10}$$

The bias of $\hat{\mu}_I$ can be written as follows

$$bias(\hat{\mu}_I \mid Y_1) = \frac{\mathbf{1}_{N-m}^T}{N}(E(\hat{Y}_2 \mid Y_1) - \pi(X_2)),$$

$$= \frac{\mathbf{1}_{N-m}^T}{N}(\hat{\pi}(X_2) - \pi(X_2)). \tag{4.8.11}$$

If the imputation is to choose the category with highest probability, then

$$E(\hat{Y}_2 \mid Y_1) = \hat{Y}_2, \tag{4.8.12}$$

$$\text{var}(\hat{\mathbf{y}}_i \mid \hat{\pi}(\mathbf{x}_i)) = 0. \tag{4.8.13}$$

Therefore

$$E(\hat{\mu}_I) = \frac{1}{n}(\sum_{i=1}^{m} E(\mathbf{y}_i \mid \mathbf{x}_i) + \sum_{j=m+1}^{n} E(\hat{\mathbf{y}}_j \mid \mathbf{x}_j))$$

$$= \frac{1}{n}(\mathbf{1}_m^T E(Y_1) + \mathbf{1}_{n-m}^T E\hat{Y}_2), \tag{4.8.14}$$

$$\text{var}(\hat{\mu}_I) = \frac{1}{n^2}\sum_{i=1}^{m} \text{var}(\mathbf{y}_i \mid \mathbf{x}_i). \tag{4.8.15}$$

Again, in census situation, the bias of $\hat{\mu}_I$ becomes

$$bias(\hat{\mu}_I) = \frac{\mathbf{1}_{N-m}^T}{N}(E\hat{Y}_2 - \pi(X_2)). \tag{4.8.16}$$

As mentioned in the beginning of this section, the results in (4.8.10) and (4.8.16) have not provided much help to distinguish different imputation methods, especially between RBF imputation and logistic regression imputation. However we can use the above results in simulation and numerical studies to compare different imputations.

Neural Network Imputation for Categorical Data

Neural network imputation for a categorical missing value involves selecting a category based on the predicted membership probability. The strategy is the same as in the multinomial logit model. The difference is that a neural network model is used instead of a linear logit model for the probability. Therefore the expectation, bias and variance formulas in previous section are still valid. In this section the attention is focused on the performances of the two approaches. Let's consider the log-value of the probability ratio

of an imputed value $\hat{y}_i$ (i=m+1...n) as a distance measurement. Suppose in both multinomial logit and neural networks models, the k^{th} category is imputed with probabilities $\hat{\pi}_k^{(1)}(\mathbf{x}_i)$ (multinomial logit) and $\hat{\pi}_k^{(2)}(\mathbf{x}_i)$ (neural networks) respectively. Then

$$
\begin{aligned}
LR_{ik} &= \ln \hat{\pi}_k^{(2)}(\mathbf{x}_i) - \ln \hat{\pi}_k^{(1)}(\mathbf{x}_i) \\
&= \ln \frac{\exp(\Phi(\mathbf{x}_i)W_k)}{\sum_{j=1}^{p+1} \exp(\Phi(\mathbf{x}_i)W_j)} - \ln \frac{\exp(\mathbf{x}_i\beta_k)}{1+\sum_{j=1}^{p} \exp(\mathbf{x}_i\beta_j)} \\
&= \Phi(\mathbf{x}_i)W_k - \mathbf{x}_i\beta_k - (\ln \sum_{j=1}^{p+1} \exp(\Phi(\mathbf{x}_i)W_j) + \ln(1+\sum_{j=1}^{p} \exp(\mathbf{x}_i\beta_j))) \\
&\approx \Phi(\mathbf{x}_i)W_k - \mathbf{x}_i\beta_k + const. \qquad (4.8.17)
\end{aligned}
$$

In the above formula, the last term consists of two probability normalisation terms, which can be treated as a constant. The performances of the two models depend on whether they can approximate the data generating mechanism. If the underlying mechanism is closer to linear, the neural networks model could overfit data, and affect the prediction efficiency. Otherwise if the mechanism is more non-linear, the linear model will not be sufficient.

4.9 Variance Estimation of Estimators

In section 4.6, we discussed the variance properties of estimators of the population mean under the model assumptions given in (4.1.1) to (4.1.3) in the presence of imputation and saw that the variance coefficient of imputation (VCI) could lead to different variances than with no imputation. Due to mathematical difficulty, the variances of $\hat{\tau}_I$, $\hat{\tau}_{rI}$, $\hat{\tau}_R$ and $\hat{\tau}_{rR}$ have not been discussed. One approach to estimating this variance is multiple imputation proposed by Rubin (1978). The complexity of analysis resulted from multiple imputation put an obstacle to wide adoption. Therefore single imputation is still widely used. For this reason, various approaches have been proposed to address the underestimation of variance problem of single imputation. Ford (1983) suggested reimputation for the replication variance estimators under hot-deck imputation. Särndal

(1992) investigated the precision of the generalised regression estimator (GREG) under imputation. Fay (1991) suggested a probability description for the sampling and response process, which provided the foundation for variance estimation. Rao and Sitter (1995) provided a two-phase approach for the Horvitz-Thompson estimator under SRS and the response mechanism given by Fay. A literature review on this area was given by Lee, Rancourt and Särndal (2002). The basic idea is to decompose the total variance of an estimator under single imputation into two components, the sampling variance and the variance due to imputation under the unbiasedness assumption of the estimator. Then approaches such as adjustment and reimputation can be used to improve variance estimation.

The Jackknife estimators can be used to estimate the variance of estimators based on nonparametric imputations such as neural network imputation and weighted distance nearest neighbour imputation in our case. Chen and Shao (2001) suggested using the Jackknife estimators to estimate the variance of population mean estimator based on nearest neighbour imputation. Chen and Shao (2000) showed that in the design-based context the variance of population mean estimator based on nearest neighbour imputation can be expressed as a function of $E(y|x)$ and $var(y|x)$. Since nearest neighbour imputation is nonparametric, which results in that the analytical expressions of $E(y|x)$ and $var(y|x)$ are not available, therefore the Jackknife variance estimator is suggested. Each time when the Jackknife pseudoreplicate is generated, the sampling weight is adjusted according to the size of imputation class (poststratum size). The Jackknife estimator that treats the imputed value as true value is likely to underestimate the variance of the estimator of interest (Rao and Shao, 1992). Rao and Shao (1992) proposed a method of adjusting the imputed values in calculating each Jackknife pseudoreplicate. Since adjusting the imputed values leads to overestimation of the variance of population mean estimator, Chen and Shao (2001) proposed a modified adjustment method, which performs partial reimputation that produces the right amount of variation among the Jackknife pseudoreplicates. Chen and Shao (2001) showed the partial reimputation method gives better result than the previous adjustment methods.

4.10 Conclusion

We have showed the bias and variance of the estimators of population mean under regression and RBF imputation. We also gave two special cases to simplify the theoretical results. These results indicate that there is no simple answer about which method is better than the other. The performance of individual imputation method depends on the validity of the assumptions about the underlying model. For regression imputation, the performance depends on whether the linear assumption is valid. Under the linear assumption we showed regression imputation produces unbiased mean estimation but underestimates population variance. In this situation random regression imputation gives better results in terms of preserving population variance. Meanwhile in this situation the estimators of population mean and variance based on RBF imputation are biased. We also showed that if the underlying model can be expressed by a RBF model, RBF imputation will give the unbiased estimators of population mean and variance. In this later situation the estimators based on regression imputation are biased. The bias is measured in terms the difference between the estimator based on the true data and that based on the imputed data. We term it predictive bias (*pbias*). Since RBF model can be specified to be a given non-linear model as well as linear model, it is more sensible to ask how it can be specified to be the unknown underlying model rather than whether it can outperform regression model. For example, how the centres are defined is crucial to the performance of RBF imputation. Since RBF imputation can outperform regression imputation in some circumstances, it therefore deserves further study.

5 Simulation Study

In Chapter 4, the properties of linear regression imputation and RBF neural networks imputation were discussed. The asymptotic results showed that the performance of different imputation methods depends on the underlying model of the data. Simply put, if the true model is linear the random linear imputation method gives better imputations than that of RBF imputation, in terms of preserving the variance of population. Otherwise if the underlying model can be well approximated by a RBF model, the random linear imputation can't compete with RBF. In this chapter we test these conclusions with the simulated data and a real data set, derived from the 1991 household census.

5.1 Simulation Study Based on Predetermined Models

5.1.1 Design of Simulation Study

The design of the simulation is based on the intuitive idea that each imputation method is assessed both for data generated from its own model and for data from the underlying models of the competing methods. An additional data set generated from an independent model, which is none of the true models of all candidate methods, is also tested. This is to test the performance when the real model does not happen to be any of them.

Eight models are considered for the variable y containing missing values. This variable is called the response variable. Four of the models are for continuous variables; the remaining four models are for categorical response variables. For simplicity we denote the eight models as "Simulation I"... "Simulation VIII". All models depend on the same covariate variables x_1 and x_2, generated from normal distributions, $x_1 \sim N(10,2^2)$, $x_2 \sim N(5,1.5^2)$, x_1 and x_2 are independent. The residual term when needed is also assumed to follow a normal distribution, $\varepsilon \sim N(0,3^2)$. The models are summarised in Table 5.1.1.

Table 5.1.1 Simulation Design

No.	Name	Model expression	Type of response y
I	Linear regression	$y=.5x_1+x_2+\varepsilon$	continuous
II	Correlated response	$Y\sim N(10\times\mathbf{1}_{100},\Sigma_{100\times100})$ $\mathrm{cov}(y_i, y_j) = 9\exp(-\{5\frac{(x_{i1}-x_{j1})^2}{2^2}+20\frac{(x_{i2}-x_{j2})^2}{1.5^2}\})$	continuous
III	RBF	$y=\Phi(x_1, x_2,\mu)W+\varepsilon$	continuous
IV	Non-linear	$y=.5x_1+x_2+5\sin(x_1)+5\sin(x_2)+1.1^{X_1}+1.2^{X_2}+\varepsilon$	continuous
V	Logistic regression	$\mathrm{Logit}(\Pr(y=0))=-10+.5x_1+x_2$	y=0,1
VI	Associated response	$y_i=\begin{cases}1, & z_i > E(z)\\ 0, & others\end{cases}$, where z_i the value of y_i from simulation II	y=0,1
VII	Binary RBF	$\mathrm{Logit}(\Pr(y=0))=\Phi(x_1, x_2,\mu)$	y=0,1
VIII	Logistic non-linear regression	$\mathrm{Logit}(\Pr(y=0))=-12+.5x_1+x_2+5\sin(x_1)$ $+5\sin(x_2)+1.1^{X_1}+1.2^{X_2}$	y=0,1

Note: $\Phi(x_1, x_2,\mu)=(\phi(x_1,x_2,\mu_1)\ldots\phi(x_1,x_2,\mu_j)\ldots\phi(x_1,x_2,\mu_k))$, $\phi(x_1, x_2,\mu_j)=\exp(-((x_1-\mu_{j1})^2+(x_2-\mu_{j2})^2)/\lambda)$, k is the number of nodes, λ is the shape parameter, $\boldsymbol{\mu}=(\mu_1\ldots\mu_k)^T$ are the centres, $W=(w_1\ldots w_k)^T$.

In each simulation, three hundred independent samples are generated. In each sample, one hundred vectors of values of x_1, x_2, ε and y are generated independently from the given model. The resulting 100×1 vector of values of y is denoted by Y. The first thirty cases of the response variable are assumed missing, the remaining seventy are assumed observed. The covariate values of all cases are assumed observed.

Imputation is implemented by twelve methods (see Table 5.1.2). Six of them are nearest neighbour methods based on different distance definitions (see 3.1.1). The first distance is the square value Euclidean distance (NNIEU). The second is Mahalanobis distance (NNIMH). The third to the sixth are as follows: weighted distance without cross term where the distance is the square value of the difference between two points (WD21), weighted distance with cross term where the distance is the square value of the difference

between two points (WD22), weighted distance without cross term where the distance is the absolute value of the difference between two points (WD11), and weighted distance with cross term where the distance is the absolute value of the difference between two points (WD12). The distance based nearest neighbour imputation methods have the same imputation strategy for both the continuous response and the categorical response variable, which is to impute the missing value with the corresponding value of the observed case that has the smallest distance between its covariates. The other six imputation methods are based on the linear model and the RBF model (see section 2.1). The imputation approaches for continuous response are different to these for categorical response. In both linear regression model and RBF model three imputation methods are employed, the prediction based imputation (lm, RBF), the random imputation (the predicted value plus an error term: Rlm, RRBF) and the predictive mean match imputation (PMMlm, PMMRBF) which imputes a missing value by the observed value of response that has the nearest predicted value. For categorical response, the imputations are the category with the highest probability and a random draw from the predicted distribution. Therefore ten imputation methods are included for categorical responses (six nearest neighbour imputations, two logistic regression imputations and two RBF imputations).

The MLP neural network imputation and the tree model imputation are not included in the simulations, but are included in the real data study in the next section. Two concerns motivated this decision. One is that there are no theoretical results about these two imputation methods in the previous chapters. The aim of the simulation is to test the theories developed in previous chapters. The other concern is computing time. The MLP imputation is very time consuming. Running a single MLP model could take several days. Some numerical results of these two methods are provided in the next section.

Among the imputation methods considered in this simulation, regression based imputations and the distance-based imputations are uniquely defined. A unique set of results can be reproduced given the same data set. Unlike these methods, the RBF based imputations are dependent upon their specification, and the results may be sensitive to this specification. With different initial values of weights, centres and shape parameters, the imputations are likely to be different. Only in the ideal situation the global minimum of the error function can be achieved with any initial tuning specification. Therefore the strategy of setting up RBF model needs consideration. In fact it is the main area where

computer scientists put their efforts. They have been developing algorithms to find the best solutions efficiently. These algorithms are also called machine-learning algorithms.

Table 5.1.2 Imputation methods

Method	**Abbreviation**	**Type of response the method can be applied for**
Regression	lm	Continuous
Predictive mean match by regression	PMMlm	Continuous
Random regression	Rlm	Continuous
RBF	RBF	Continuous
Predictive mean match by RBF	PMMRBF	Continuous
Random RBF	RRBF	Continuous
Nearest neighbour imputation with Euclidean distance	NNIEU	Continuous and categorical
Nearest neighbour imputation with Mahalanobis distance	NNIMH	Continuous and categorical
Nearest neighbour imputation based on the weighted distance of absolute value differences without cross term	WD11	Continuous and categorical
Nearest neighbour imputation based on the weighted distance of absolute value differences with cross term	WD12	Continuous and categorical
Nearest neighbour imputation based on the weighted distance of square value differences without cross term	WD21	Continuous and categorical
Nearest neighbour imputation based on the weighted distance of square value differences with cross term	WD22	Continuous and categorical
Logistic imputation based on the highest probability	LogisticHP	categorical
Logistic imputation based on random draw	LogisticRD	categorical
RBF imputation based on the highest probability	RBFHP	categorical
RBF imputation based on random draw	RBFRD	categorical

These provide an elegant theory for approximating non-linear functions. But in reality, they may not be achievable.

The base software used for the RBF simulation and the numerical study with MLP in the next setion is the neural computing tool box obtained from the website of the NEURAL COMPUTING RESEARCH GROUP of Aston University (http://www.ncrg.aston.ac.uk/netlab/index.html). The source code in MATLIB (Pratap, 2001) format is also available from this website. The reason of using this tool box is that the algorithms of RBF neural network and MLP neural work in this package are the same algorithms described in Bishop's (1996) book. The simulation code used in this section and the code for WALD error neural networks are both built upon this tool box.

The two factors that affect the performance of RBF model are the centres and the shape (smooth) parameter λ. In this simulation, the centres are defined by the following method. Suppose the number of centres (k) is known, and the number of the centres in each individual covariate is $\sqrt[q]{k}$, where q is the number of covariates. The centres for the ith covariate (i=1...q) are then defined as the quantiles of the covariate distribution for probabilities equal to $\frac{1}{\sqrt[q]{k}+1},\cdots,\frac{\sqrt[q]{k}}{\sqrt[q]{k}+1}$. The final centres are the full combination of the individual centres. For example, if $(6,12)^T$ are centres of x_1, $(4,6)^T$ are centres of x_2, the final centres of the RBF model are obtained as follows.

x_1	x_2
6	4
6	6
12	4
12	6

The number of centres k can be optimised by a cross-validation method. The cross-validation procedure is carried out by splitting the original data in to 10 approximately equal subsets. Then each time pick one as the test data, and the remaining 9 subsets as training data. The optimal parameter is the one that gives best prediction to the test data. Due to computing hardware limitation, k is fixed to be 9. For q=2, the number of centres of individual covariate becomes $\sqrt[q]{k}=3$. With large k, RBF model can be trained to be a better local regression model. On the other hand, if k is too large, such as $k>n$ (the number of observations), RBF becomes saturated which leads to inefficient prediction. One can

see the performance of the RBF imputation could be improved if k is included in the optimisation process. The shape parameter λ is given an initial value $\lambda_0 \in \Theta=[-20, -19...20]$, then the optimal value is determined by the cross-validation method. The two cross-validation processes are independently implemented because of algorithm constraints. Also the initial value involves subjective judgement. The true value may not lie in this interval. This may limit the capability of RBF model. As we will see in the following simulations, when the true values of the RBF parameters are known, not only the training process is quick, the imputation result is also very promising. The difficulty is if a wide Θ such as $\Theta=[-100, -19...100]$ is used, the training time could be increased exponentially.

5.1.2 Criteria Used to Evaluate Properties of Imputation Methods

For both the continuous response and the categorical response, the evaluation is based on the thirty imputed values. The measures used to evaluate the consistency of imputations for continuous variables are the predictive mean square error (PMSE), the mean, the variance and quantiles at .25, .50, .75. Suppose y_{ij} is the true value of y for the jth unit in the ith repetition, and $\hat{y}_{ij}$ is the imputed value of y for the jth unit in the ith repetition, $i=1...300, j=1...30$. Then the measures can be denoted as

$$PMSE = \frac{1}{300}\sum_{i=1}^{300}\frac{1}{30}\sum_{j=1}^{30}(\hat{y}_{ij} - y_{ij})^2, \tag{5.1.1}$$

$$mean_{300} = \frac{1}{300}\sum_{i=1}^{300}\frac{1}{30}\sum_{j=1}^{30}\hat{y}_{ij}, \tag{5.1.2}$$

$$\text{var}iance_{300} = \frac{1}{300}\sum_{i=1}^{300}\frac{1}{30}\sum_{j=1}^{30}(\hat{y}_{ij} - \bar{\hat{y}}_{i.})^2, \tag{5.1.3}$$

$$Q(p=.25) = \frac{1}{300}\sum_{i=1}^{300}\inf\{t : \hat{F}_i(t) > .25), \tag{5.1.4}$$

$$Q(p=.50) = \frac{1}{300}\sum_{i=1}^{300}\inf\{t : \hat{F}_i(t) > .50), \tag{5.1.5}$$

$$Q(p=.75) = \frac{1}{300}\sum_{i=1}^{300}\inf\{t : \hat{F}_i(t) > .75), \tag{5.1.6}$$

where $\bar{\hat{y}}_{i.} = \frac{1}{30}\sum_{j=1}^{30}\hat{y}_{ij}$, i=1...300, $\hat{F}_i(t) = \frac{1}{30}\sum_{j=1}^{30} I(\hat{y}_{ij} \le t)$, $I(\cdot)$ is an indicator function.

The *PMSE* is used to measure the average prediction error due to imputation. Smaller *PMSE*s may be expected to lead to smaller estimation errors. Measures (5.1.2) to (5.1.6) are designed to assess how the distribution is preserved. Since mean and variance determine the properties of normal distribution, they are used to measure the distribution of imputed values. Considering the fact that the actual distribution of 100 realisations may not duplicate the distribution of the underlying model due to sampling error, quantiles are used to evaluate how imputation can preserve the actual distribution. For simplicity, when y follows a normal distribution, it is considered to evaluate the quantiles at only three points, 25%, 50% and 75%.

The measures for categorical data imputation consist of the expected marginal distribution and the proportion of correct imputations (the percentage of the imputed values equal to the true values: p_c).

$$\hat{p}_{300} = \frac{1}{300}\sum_{i=1}^{300}\hat{p}_i \,,\ \hat{p}_i = \frac{1}{30}\sum_{j=1}^{30} I(\hat{y}_{ij} = 0), \tag{5.1.7}$$

$$p_{c300} = \frac{1}{300}\sum_{i=1}^{300}\frac{1}{30}\sum_{j=1}^{30} I(\hat{y}_{ij} = y_{ij}). \tag{5.1.8}$$

The measure p_{c300} corresponds to the measure PMSE for the continuous case, since both assess how well the imputed values $\hat{y}_{ij}$ predict the true values y_{ij}. For computing consideration the categorical responses are taken to be binary. Therefore the marginal distribution is just the mean of the imputed values if the probability of the response equal to one is the interest, or 1-mean ($\hat{y}_{mis}$) if the opposite is the parameter of interest.

Since the three hundred repetitions are independent, a t-test can be used to test the significance of the differences between the measures based on the imputed values and the measurements based on the true values. Take $mean_{300}$ as an example, the t statistic is

$$t = \frac{mean_{300} - mean_{true}}{\sqrt{\text{var}(\frac{1}{30}\sum_{j}(\hat{y}_{ij} - y_{ij}))/300}}, \tag{5.1.9}$$

where

$$mean_{true} = \frac{1}{300}\sum_{i=1}^{300}\ \frac{1}{30}\sum_{j=1}^{30} y_{ij}\ . \tag{5.1.10}$$

If an imputation well preserves the mean, the t value will be small, and the p-value should be fairly large. For simplicity only the p-values are presented in following simulations.

As defined in (5.1.1) to (5.1.8), the overall measures for both continuous and categorical variables are the averages of the measures over the three hundred repetitions. In each simulation, two tables are provided; the first one is the values of the overall measures. The second table is the p-values of the t-test based on the individual values of the three hundred repetitions and the measures based on the true values.

5.1.3 Results of Simulations for Continuous Variables

Simulation I: Linear Regression Model

Model:

y=.5 x_1+ x_2+ε.

In the first simulation, the simplest situation is considered. The response variable is a linear combination of its covariates and a residual. The results are given in Table 5.1.3 with the first line containing the results when the imputed values $\hat{y}_{ij}$ are given by the true values y_{ij}. The mean and variance of these true values correspond closely to the values of $E(y)$=10 and var(y)=12.25 under this model.

In this simulation we would expect the methods Rlm and PMMlm to perform well, since the data is generated from a linear regression model. The lm imputation is also expected to preserve the population mean very well, but to deflate the variance.

The simulation shows that the random regression imputation gives the best imputation in terms of preserving the population mean, variance and quantiles (see Table 5.1.3, Table 5.1.4). The predictive mean match imputation of linear regression also gives good imputation with variance deflated a little bit. In the meantime, the random RBF imputation also performs remarkably well in preserving the population mean and variance.

Table 5.1.3 The predictive mean square errors, means, variances and quartiles of imputed values (based on 300 repetitions of the linear model)

Imputation Method	**PMSE**	**Mean$_{300}$**	**variance$_{300}$**	**Q(p=.25)**	**Q(p=.50)**	**Q(p=.75)**
True	0	9.9965	12.23	7.65	9.9821	12.32
lm	9.2361	10.0072	3.3532	8.8169	10.0291	11.1796
PMMlm	18.0841	10.0085	11.9917	7.8719	10.0151	12.2085
Rlm	17.9566	10.0042	12.2377	7.7977	10.0007	12.3155
RBF	10.2184	10.0115	3.6158	8.7816	10.0362	11.2626
PMMRBF	18.7063	9.9960	11.9242	7.7708	10.0793	12.2336
RRBF	18.6299	10.0087	12.1945	7.7707	10.0486	12.2329
NNIEU	18.0809	9.9942	11.4284	7.8366	10.0438	12.1955
NNIMH	18.2231	10.0098	11.4356	7.8702	10.0680	12.2126
WD11	18.0758	10.0193	11.4679	7.8314	10.0702	12.1513
WD21	18.0316	9.9982	11.4533	7.8579	10.0699	12.1454
WD12	18.1626	10.0013	11.4690	7.8586	10.0555	12.0431
WD22	18.1522	9.9782	11.4929	7.8613	10.0572	12.0342

Note: True—the original values that are assumed missing.

For both linear regression imputation and RBF imputation the variance is severely deflated. It is strong evidence to avoid mean imputations. It is consistent with the theories in Chapter 4, which show that both random imputations based on linear regression model and RBF model are much better than the corresponding mean imputation methods in preserving the population variance and the distribution (quantiles). Random imputation here is implemented by adding a term drawn from the residuals of the complete cases to the mean given by the regression or RBF prediction.

The weighted distance based nearest neighbour imputations (WD11, WD12, WD21, WD22) can also preserve the mean very well but slightly deflate the variance. Another interesting phenomenon in this simulation is that the nearest neighbour imputations with Euclidean distance and Mahalanobis distance are almost as good as the weighted distance imputations, while all of them deflate the population variance. It may be a good idea to use simple methods if the data is following a simple model such as a linear model. It also reveals that Euclidean distance is as good as Mahalanobis distance. They are almost identical in terms of imputation performance.

A measure of overall performance is the percentage of p-values above .05. Only the regression mean imputation and the RBF mean imputation display significant lack of fit. All other imputation methods display not significant evidence of lack of fit. This is consistent with our expectation. Meanwhile high p-values may also indicate large variances of the estimator used in the t-test. If this is true, the imputation method that produces high p-value imputes the missing values with much variant values, which may give unstable imputation in single imputation. It will be helpful to include the variances in the future simulation to clarify the exact reason of high p-values.

Table 5.1.4 *P* values of t-test: the means, variances and quantiles of imputed values vs. the true values or the expected value

Imputation method	$Mean_{300}$	$variance_{300}$	Q(p=.25)	Q(p=.50)	Q(p=.75)	Overall
lm	0.9005	0.0000	0.0029	0.7906	0.0033	2/5
PMMlm	0.8947	0.4647	0.3297	0.8479	0.5726	5/5
Rlm	0.9141	0.9283	0.4778	0.9112	0.9778	5/5
RBF	0.8813	0.0000	0.0035	0.7630	0.0051	2/5
PMMRBF	0.9476	0.3796	0.5466	0.6151	0.6492	5/5
RRBF	0.8938	0.8541	0.5469	0.7171	0.6469	5/5
NNIEU	0.9392	0.0858	0.3934	0.7345	0.5366	5/5
NNIMH	0.8889	0.0877	0.3325	0.6508	0.5845	5/5
WD11	0.8477	0.0965	0.4037	0.6437	0.4302	5/5
WD21	0.9419	0.0924	0.3536	0.6447	0.4177	5/5
WD12	0.9275	0.0969	0.3524	0.6928	0.2504	5/5
WD22	0.8670	0.1041	0.3477	0.6869	0.2395	5/5

Note: The notation is in Table 5.1.1.

Simulation II: Correlated Normal Distribution Model

Model:

$$Y \sim N(10\times\mathbf{1}_{100}, \Sigma_{100\times100}),$$

where

$$\Sigma_{100\times100} = [\mathrm{cov}(y_i, y_j)]_{100\times100}, \mathrm{cov}(y_i, y_j) = 9\exp(-\{\frac{(x_{i1}-x_{j1})^2}{2^2}\times 5 + \frac{(x_{i2}-x_{j2})^2}{1.5^2}\times 20\}).$$

This simulation describes a situation where the response variable y doesn't depend on its covariates directly but only through the covariance between different units of y. The more distant the covariate values of $\mathbf{x}_1$ and $\mathbf{x}_2$, the less correlated are the response y_i and y_j. Figure 5.1 is a coarse grid graph based on one simulation. The grid is defined by x_1 and x_2 with equal intervals in each axis. The y value at each joint of the grid is the average of the y values in the neighbouring squares. The shape could be different when different data generated from the same correlated model is used. Also the correlation nature is not clearly displayed. It should be a flat surface, because less distant covariate values imply less distant response values.

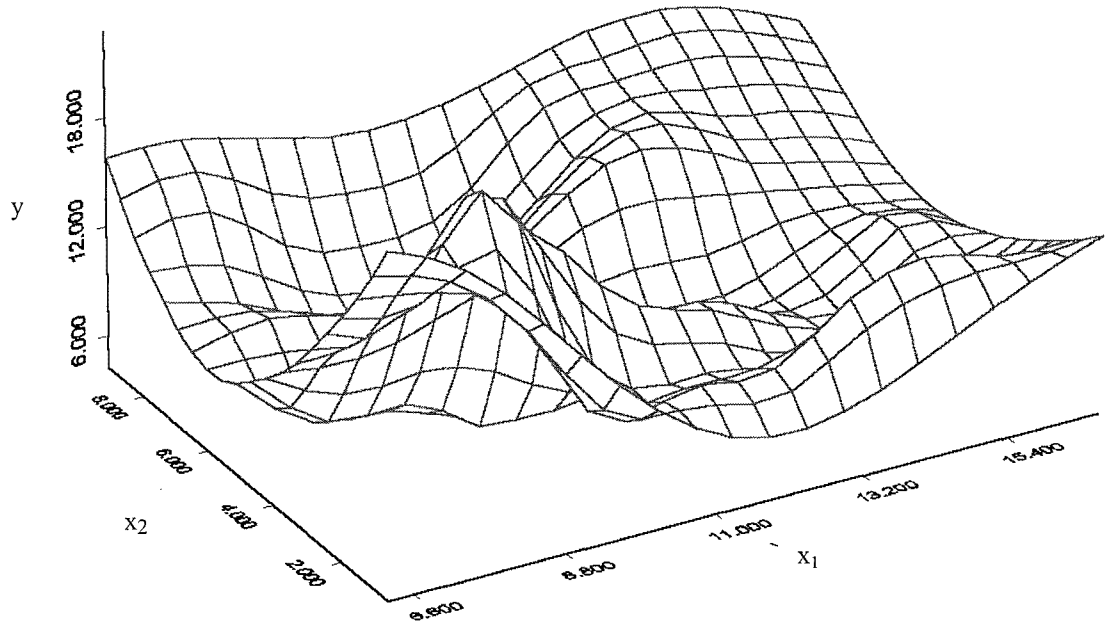

Figure 5.1 The smoothed response surface for the correlated model.

In theory the nearest neighbour imputation based on weighted distance should perform very good and it does. The average performances of the twelve imputations based on 300 repetitions are given in Table 5.1.5.

All of the six distance-based imputations give better results than other imputations in terms of preserving the distribution and true values (see table 5.1.6). The imputations with absolute value distance are even better than that of square value distance in this simulation. This could indicate robust feature of absolute value distance. The cross term in the weighted distance doesn't give significant contribution to the performance. This may be because there is no cross term in the model. It is hard to judge whether it should be included in real situation. There seems no loss in including it.

Table 5.1.5 The predictive mean square errors, means, variances and quartiles of imputed values (based on 300 repetitions) of the correlated response model

Imputation Method	PMSE	$Mean_{300}$	$variance_{300}$	Q(p=.25)	Q(p=.50)	Q(p=.75)
True	0	10	9.01	7.7	10	11.4
lm	14.8654	9.9802	2.8593	8.5708	9.8376	10.7146
PMMlm	20.1434	10.1335	9.4090	7.9906	10.1931	11.9163
Rlm	26.6562	10.0803	9.4874	7.4422	10.1329	12.3282
RBF	12.5534	9.9921	1.2409	9.4175	10.1334	10.4688
PMMRBF	20.1218	10.0685	9.2921	7.8827	10.1491	11.9138
RRBF	20.9850	10.0858	9.2370	7.9134	10.1086	11.8571
NNIEU	15.0530	10.0209	9.3467	7.8031	10.0615	11.4738
NNIMH	15.0765	10.0159	9.3349	7.8112	10.0628	11.4671
WD11	14.4736	10.0189	9.1465	7.7249	10.0417	11.4215
WD21	14.6711	10.0214	9.1457	7.7329	10.0505	11.4345
WD12	15.0531	10.0449	9.2088	7.8366	10.1200	11.3682
WD22	15.0210	10.0760	9.2076	7.8253	10.0722	11.4013

Note: The notation is in Table 5.1.1.

The nearest neighbour imputations based on Euclidean distance (NNIEU) and Mahalanobis distance (NNIMH) as well as the random imputation based on RBF model (RRBF) are also good methods in this simulation. For NNIEU and NNIMH, their good performance can be explained by the similar distance measure based on the covariates. These two imputations are almost identical. This phenomena happened in simulation I, therefore it is probably safe to say NNIEU is a good replacement of NNIMH. The good performance of random RBF imputation can be explained by the local regression nature of RBF, which means the RBF model can be trained to be a combination of local regressions in the neighbours of its centres. If the centres are properly defined, the combination will be a good representation of the underlying model.

The imputations based on the linear regression model seem unable to deal with the covariance structure of the response, and gives poor imputations. Meanwhile Rlm and PMMlm give acceptable imputation in terms of preservation of the variance. The quantiles are not preserved by all three regression based imputation, which may indicate the inadequacy of the regression model.

From the p-values in Table 5.1.6, the overall performances of the twelve imputation methods are summarised by the percentage of p-values above .05. In the overall evaluation, the weighted distance based nearest neighbour imputation methods outperform the other methods since they display no significant lack of fit. Again the conclusion based on p-values should is not fully convincing without providing the variances. This statement applies to the remaining simulations for the continuous variable.

In summary, when the covariance of response is a function of the distance of its covariates, the weighted distance imputations are the right choice. If the actual data set is very large, such that the distance matrix can not be manipulated, RBF and NNIEU are good replacements.

Table 5.1.6 *P* values of t-test: the means, variances and quantiles of imputed values vs. the true values or the expected values of the correlated response model

Imputation Method	$Mean_{300}$	$variance_{300}$	Q(p=.25)	Q(p=.50)	Q(p=.75)	Overall
lm	0.7794	0.0000	0.0009	0.1971	0.0325	2/5
PMMlm	0.2500	0.1292	0.0978	0.1450	0.0457	4/5
Rlm	0.4256	0.0873	0.1271	0.2647	0.0096	3/5
RBF	0.8778	0.0000	0.0000	0.2634	0.0095	2/5
PMMRBF	0.4789	0.2318	0.2319	0.2251	0.0466	4/5
RRBF	0.4028	0.3053	0.1814	0.3376	0.0487	4/5
NNIEU	0.7708	0.1764	0.4383	0.5406	0.6914	5/5
NNIMH	0.8104	0.1872	0.4108	0.5335	0.7150	5/5
WD11	0.7864	0.4801	0.8194	0.6590	0.8981	5/5
WD21	0.7669	0.4820	0.7686	0.6035	0.8416	5/5
WD12	0.7014	0.3516	0.4353	0.3012	0.8530	5/5
WD22	0.7293	0.3537	0.4670	0.4858	0.9935	5/5

Note: The notation is same as that of simulation I.

Simulation III: RBF Model

Model:

$$y=\Phi(x_1, x_2,\mu)+\varepsilon=\Sigma w_j\phi(x_1, x_2,\mu_j)+\varepsilon,$$

where $\phi(x_1, x_2, \mu_j)=\exp(-((x_1-\mu_{j1})^2+(x_2-\mu_{j2})^2)/\lambda)$, $\boldsymbol{W}=(5,2.5,5)$, $\mu=\{(8,4),(12,6)\}$, $\lambda=1$.

The systematic part of the RBF model given above has two peaks (centres in RBF model). One is at $(x_1,x_2)=(8,4)$, the other one is at $(x_1,x_2)=(12,6)$ (see Figure 5.2). The following picture describes the RBF model without noise. The vertical axis is the value of the RBF function with out the residual, which is the expected value of response y. Axis x_1 and axis x_2 are the two covariates. The response surface has the shape of the density function of a mixture of two normal distributions.

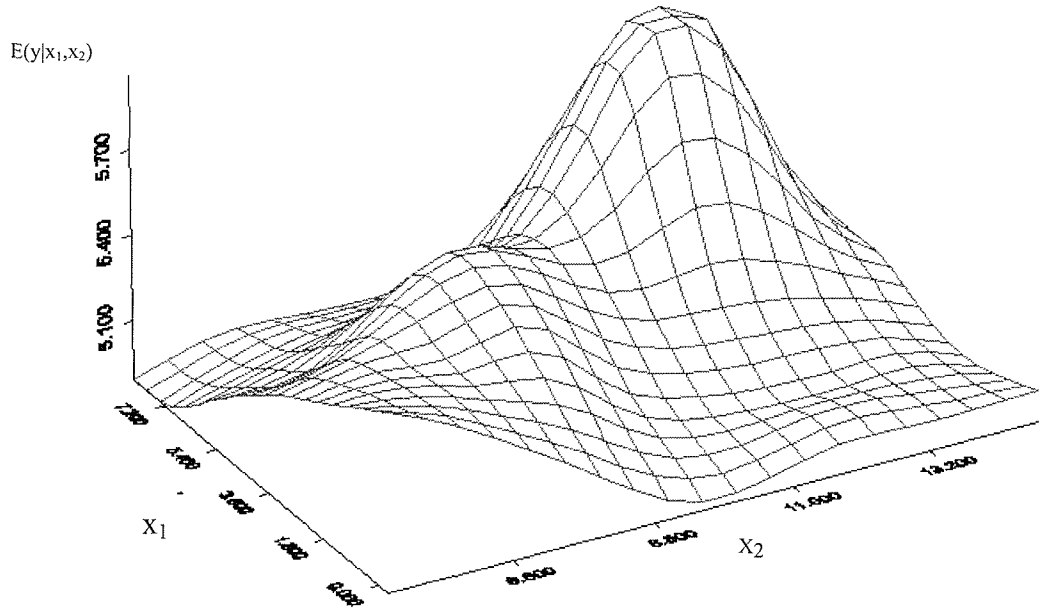

Figure 5.2 The RBF model without noise.

The results of the simulation are given in Table 5.1.7 and Table 5.1.8. Under the RBF model, the imputations from both the predictive mean match and the random imputation based on the RBF mean are better than other methods in preserving the variance and the distribution. The population mean is well preserved by all imputation methods. It is not a dimension which discriminates them. The random RBF and the predictive mean match imputation based on RBF are nearly equally good. But the random RBF is easier to implement in terms of the complexity of the computing procedure.

Table 5.1.7 The predictive mean square errors, means, variances and quartiles based on the imputed values (based on 300 repetitions) of the RBF model

Imputation Method	PMSE	$Mean_{300}$	$variance_{300}$	Q(p=.25)	Q(p=.50)	Q(p=.75)
True	0	6.8911	10.3757	4.7433	6.8864	9.0240
lm	11.0641	6.7839	1.4927	6.0087	6.7712	7.5605
PMMlm	19.4238	6.9137	9.8027	4.9092	6.8529	8.9355
Rlm	20.8859	6.8670	11.7873	4.6472	6.9264	9.0763
RBF	10.5250	6.8876	1.6284	6.0220	6.7565	7.5495
PMMRBF	18.3239	6.8669	10.4503	4.7734	6.8517	8.9579
RRBF	17.5411	6.8689	10.1602	4.7968	6.8494	8.9614
NNIEU	18.0352	6.9765	10.1070	4.9255	6.9637	9.0235
NNIMH	17.9833	6.9935	10.0909	4.9477	6.9897	9.0312
WD11	17.4217	6.3684	9.7153	4.7358	6.3454	8.4272
WD21	17.8465	6.4681	9.8248	4.7344	6.4976	8.6213
WD12	16.8401	6.2237	9.4793	4.7599	6.2586	8.1282
WD22	17.1440	6.1998	9.8875	4.6788	6.1997	8.4590

Note: The notation is same as that of simulation I.

The imputations based on Rlm and PMMlm are also acceptable. To understand this phenomenon, let's examine the variance of y. The overall variance based on the three hundred repetitions is 10.3757. The contribution of the residual is 9, nearly 87%. Therefore the contribution from the RBF mean function is small (only 1.3757), which makes the residual term dominant. This might be the cause of the promising performance of imputations from the linear regression model.

In the meantime, the nearest neighbour imputations based on the Euclidean and Mahalanobis distances are also surprisingly good. This can be explained by the similar strategy used by these two methods. The RBF model is a weighted average over all the data points that gives more weight to the less distant ones, while the nearest neighbour imputation simply picks the nearest one as donor. The other fact is the promising performance of imputations based RBF model is not guaranteed if the initial locations of centres are not correctly given. This happened when we ignored the true centres and let

RBF choose it randomly. This is the very nature of complexity in RBF. In the real situation the true centres are rarely known. It doesn't mean the true centres can not be derived. Actually much of the time in training RBF is spent on finding the true (or nearly true) centres. That could be a difficult task for practitioners who do not posses the knowledge of neural networks and programming skills. Fortunately this restriction could be removed when the future computing power is big enough, so that the whole process can be automated.

Table 5.1.8 *P* values of t-test: the means, variances and quantiles of imputed values vs. the true values or the expected values of the RBF model

Imputation Method	$Mean_{300}$	$variance_{300}$	Q(*p*=.25)	Q(*p*=.50)	Q(*p*=.75)	Overall
lm	0.5557	0.0000	0.0018	0.5622	0.0007	2/5
PMMlm	0.8487	0.0753	0.4364	0.8460	0.6423	5/5
Rlm	0.8419	0.0564	0.6183	0.8187	0.7699	5/5
RBF	0.9333	0.0000	0.0017	0.5225	0.0006	2/5
PMMRBF	0.8414	0.6542	0.8601	0.8408	0.7184	5/5
RRBF	0.8500	0.3234	0.7653	0.8315	0.7313	5/5
NNIEU	0.6199	0.2479	0.4021	0.6794	0.9974	5/5
NNIMH	0.5693	0.2288	0.3600	0.5965	0.9646	5/5
WD11	0.1980	0.0350	0.9633	0.0669	0.0506	4/5
WD21	0.2670	0.0604	0.9564	0.1431	0.1335	5/5
WD12	0.1283	0.0107	0.9206	0.0433	0.0113	2/5
WD22	0.1194	0.0827	0.7243	0.0323	0.0593	4/5

In Table 5.1.8, although seven out of the twelve imputations have the best overall performance, the imputations based on RBF model, namely PMMRBF and RRBF, have much higher p-values than the other five methods, especially in the variance measure. NNIEU and NNIMH also give high p-values. Simply put, when the underlying model is RBF, PMMRBF and RRBF are the best choice.

Simulation IV: Non-linear Model

Model:

$$y=.5*x_1+x_2+5*\sin(x_1)+5*\sin(x_2)+1.1^{X_1}+1.2^{X_2}+\varepsilon$$

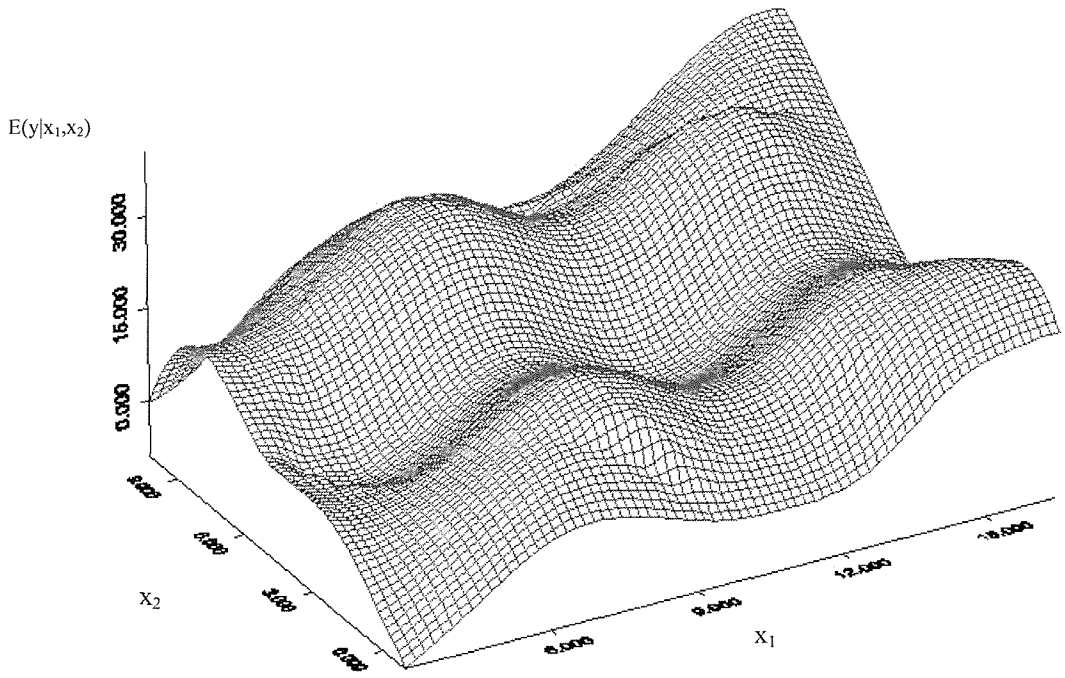

Figure 5.3 The response surface of the non-linear data generating mechanism without noise.

This model is created to test the performances of the twelve imputation methods when the underlying model is not any of them. To understand the data generating mechanism, Figure 5.3 plots the regression function of y given $\mathbf{x}_1$ and $\mathbf{x}_1$ with the noise term not included. It has two linear components, two periodical function components and two exponential function components corresponding to the two covariates. Overall it looks like a multi-peak function. We can see in the above picture, there are four peak points and two or three bottom points. One can imagine if these points are defined to be the centres of the RBF model, the random RBF imputation should perform well.

Table 5.1.9 The predictive mean square errors (or mean distance), means, variances and quartiles based on the imputed values (based on 300 repetitions) of a non-linear model

Imputation Method	PMSE	$Mean_{300}$	$variance_{300}$	Q(p=.25)	Q(p=.50)	Q(p=.75)
True	0	13.3032	38.9381	8.8596	12.7623	17.2173
lm	39.6228	13.3302	10.1457	11.1462	13.1929	15.2648
PMMlm	43.7937	13.3310	40.3691	9.0755	12.9489	17.1543
Rlm	60.4180	13.2483	40.4634	8.9788	13.1144	17.3221
RBF	38.9827	13.2673	19.6400	9.9400	12.9340	16.3285
PMMRBF	42.5489	13.2840	38.4357	8.9071	12.7698	17.2734
RRBF	43.3386	13.2785	39.5749	8.9081	12.7919	17.2700
NNIEU	22.6986	13.0899	37.0403	8.8814	12.5841	16.8099
NNIMH	22.6588	13.0988	36.8990	8.8838	12.5973	16.8129
WD11	22.3540	13.3379	37.4542	8.9007	12.7764	16.8167
WD21	22.5193	13.3353	37.3552	8.8920	12.7742	17.0407
WD12	28.8887	13.3630	37.6112	9.0902	12.6340	16.9987
WD22	29.6415	13.3540	37.2200	9.1019	13.6709	17.6000

Note: The notation is same as that of simulation I.

The other feature one might have found is that the surface is not jumping up and down dramatically especially when the noise term is added. It is quite like a linear surface. This data-generating model might be close to many real situations where the underlying model is not far from a linear model, although they are not exactly linear. Therefore you can expect the imputations based on the linear regression model might also be acceptable. This is exactly what happens in Table 5.1.9. Interestingly the weighted distance based nearest neighbour imputations are also promising. All of the three types of imputations are equally good in terms of preserving the distributions (see Table 5.1.10). But when we look at PMSE the distance-based imputations are much smaller than the other imputations, especially WD11 and WD21. That might explain why the distance-based imputations such as the widely used donor imputation work well in real situations. Meanwhile we expect the weighted distance is a way to improve the performance of the methods in this category.

Table 5.1.10 *P* values of t-test: the means, variances and quantiles of imputed values vs. the true values or the expected values of the non-linear model

Imputation Method	$Mean_{300}$	$variance_{300}$	Q(*p*=.25)	Q(*p*=.50)	Q(*p*=.75)	Overall
lm	0.7254	0.0000	0.0000	0.0057	0.0000	2/5
PMMlm	0.7194	0.4646	0.0750	0.1066	0.4694	5/5
Rlm	0.5484	0.4431	0.2392	0.0146	0.2843	4/5
RBF	0.6632	0.0001	0.0000	0.1274	0.0000	3/5
PMMRBF	0.7838	0.7390	0.5657	0.9143	0.5103	5/5
RRBF	0.7421	0.6909	0.5589	0.7010	0.5315	5/5
NNIEU	0.1126	0.3678	0.7694	0.1178	0.0075	4/5
NNIMH	0.1230	0.3428	0.7476	0.1380	0.0078	4/5
WD11	0.6714	0.4524	0.6110	0.8448	0.0082	4/5
WD21	0.6892	0.4305	0.6777	0.8666	0.1202	5/5
WD12	0.5224	0.4893	0.0629	0.2144	0.0726	5/5
WD22	0.5717	0.4024	0.0546	0.0000	0.0101	3/5

From the four simulations, we obtain the findings that are consistent with the theories in Chapter 4.

- The performance of the imputation methods depends on the validity of the assumptions of these methods.
- The mean imputation methods deflate the population variance, and distort the distribution.
- Under the true model, random imputation can preserve the mean, variance and distribution, including random imputation based on the linear regression model and RBF model.
- With proper set up, the RBF model can be very close to the underlying model therefore possesses the possibility of being a regression model. Meanwhile finding the right RBF is also difficult.
- In many situations, imputations based on linear regression are acceptable. It can be regarded as the first order approximation in terms of covariates.

5.1.4 Results for Categorical Variables

Simulation V: Logistic Regression Model

Model: $\Pr(y=0)=\exp(-10+.5\ x_1+ x_2)/(1+\exp(-10+.5\ x_1+ x_2))=p$.

This simulation repeatedly (300 repetitions) generates 100 records from the linear logistic model. The first 30 cases of y are assumed missing. Ten imputation methods, logistic regression imputation based on the highest category probability and random draw, RBF imputation based on the highest probability and random draw, nearest neighbour imputation based on Euclidean distance, Mahalanobis distance and four weighted distance, are applied to impute the missing values. The evaluation is based on the marginal distribution and the ratio of true values. In the binary case the marginal distribution can be represented by the percentage in one category. Here we use the percentage of the responses equal to zero. The simulation results are given in Table 5.1.11 and Table 5.1.12.

Table 5.1.11 The marginal distribution (probability of y=0) and the proportion of correct imputations (based on 300 repetitions) for the logistic linear model with marginal distribution at .5.

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
True	0.4987	1.0000
LogisticHP	0.4966	0.9864
LogisticRD	0.4983	0.9860
RBFHP	0.4897	0.9050
RBFRD	0.4856	0.8959
NNIEU	0.4942	0.9399
NNIMH	0.4936	0.9394
WD11	0.4970	0.9411
WD21	0.4940	0.9490
WD12	0.4938	0.9481
WD22	0.4940	0.9501

Note: The notation is the same as that of simulation I.

Under this model, the linear logistic imputations are the best ones in terms of the two evaluation criteria. In fact all of the imputation approaches perform well. The weighted

distance based methods are the second best methods. The distance embedded in the weighted distance methods is the regression distance. If more flexible distance such as neural networks distances is used, the performance could be even better. But the disadvantage is the computing effort needed to obtain the weights could be huge.

One may expect striking difference between the random imputation and the mode imputation. This does not happen in this study. Here the random imputation is implemented by generating *n-m* random values from $U(0,1)$, the uniform distribution between 0 and 1, and find the corresponding category of the interval in the cumulative distribution. Since the *n-m* missing values are imputed simultaneously, this may reduce the difference between the two imputation methods.

According to the representation theory of RBF (Bishop, 1996), the RBF model can be adjusted to be any function including the linear logistic function. Why does it not happen in this simulation? The answer is in the implementation. In practice, it is very difficult to obtain the true model, only an approximation can be obtained because of the incapability of the algorithm or the time needed to reach the true representation is off the acceptable limit. Therefore the linear logistic imputation may be a reliable and practical approach. Similar situation happens to the weighted distance based nearest neighbour imputations, if the linear regression weight is used, the implementation is stable and quick, but the performance is not always good because of the inflexibility. The neural networks weight can be used to increase the flexibility but the drawback is that the computing effort increases dramatically.

Table 5.1.12 P values of t-test: The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the linear logistic model with marginal distribution at .5

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
LogisticHP	0.9386	0.8732
LogisticRD	0.9900	0.8694
RBFHP	0.7634	0.3867
RBFRD	0.6748	0.3531
NNIEU	0.8752	0.5482
NNIMH	0.8578	0.5458
WD11	0.9512	0.5549
WD21	0.8694	0.6005
WD12	0.8636	0.5952
WD22	0.8694	0.6072

Note: The notation is same as that of simulation I. The above t-test is based on statistic defined in (5.1.9), since the response is binary taking values 0 and 1.

The value of Pr(y=0) for the simulation respond above 0.5. To vary this assumption, the situation when the marginal distribution is far from the middle is also considered. The following simulation replace the intercept of –10 by –8,

$$\Pr(y=0)=\exp(-8+.5\,x_1+x_2)/(1+\exp(-8+.5\,x_1+x_2))=p,$$

Which leads to about 13.5% of y values being 1. From Table 5.1.13 and Table 5.1.14 the pattern of the results is similar to the previous simulation where the imputations based on logistic regression are better than other methods. Overall all ten methods give promising results.

Table 5.1.13 The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the linear logistic model with marginal distribution at .865

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
True	0.8653	1.0000
LogisticHP	0.8682	0.9822
LogisticRD	0.8671	0.9809
RBFHP	0.8652	0.9167
RBFRD	0.8738	0.8917
NNIEU	0.8763	0.9583
NNIMH	0.8756	0.9584
WD11	0.8767	0.9662
WD21	0.8764	0.9653
WD12	0.8739	0.9643
WD22	0.8732	0.9649

Table 5.1.14 P values of t-test: The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the linear logistic model with marginal distribution at .865.

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
LogisticHP	0.9855	0.9149
LogisticRD	0.9910	0.9089
RBFHP	0.9996	0.6592
RBFRD	0.9585	0.5818
NNIEU	0.9463	0.8119
NNIMH	0.9500	0.8124
WD11	0.9448	0.8446
WD21	0.9458	0.8409
WD12	0.9580	0.8365
WD22	0.9612	0.8389

Note: The notation is same as that of simulation I.

Simulation VI: Associated Response Model

Model:

$$y_i = \begin{cases} 1, & z_i > E(z) \\ 0, & otherwise \end{cases},$$

where z is the variable described in simulation II. This is the counterpart of simulation II for the categorical data. Under this model the true marginal distribution $p=E(y=1)=.5$. The marginal distribution and the percentage of the true values imputed by the ten methods are as follows (see Table 5.1.15, Table 5.1.16).

Table 5.1.15 The marginal distribution (probability of $y=1$) and the proportion of correct imputation (based on 300 repetitions) of the correlated response model

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
True	0.4931	1.0000
LogisticHP	0.4995	0.6760
LogisticRD	0.4925	0.5948
RBFHP	0.5046	0.6823
RBFRD	0.4940	0.6125
NNIEU	0.4928	0.7846
NNIMH	0.4923	0.7855
WD11	0.4989	0.8899
WD21	0.4998	0.8676
WD12	0.5011	0.8035
WD22	0.4983	0.8657

Note: The notation is same as that of simulation I.

The weighted distance based nearest neighbour imputations outperform logistic regression based imputations in terms of preservation of the true values. Overall the weighted distance based imputation with the absolute value distance and cross term gives more consistent imputation in terms of the marginal and the true values. The RBF imputations also show it flexibility of being a local regression. They also outperform the linear logistic based imputations. But due to the computing limitation, the RBF model has not been

trained to be the underlying model, therefore it can not outperform the weighted distance based nearest neighbour imputations.

Table 5.1.16 P values of t-test: The marginal distribution (probability of y=1) and the proportion of correct imputation (based on 300 repetitions) of the correlated response model

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
LogisticHP	0.9688	0.0541
LogisticRD	0.9971	0.0261
RBFHP	0.9443	0.0573
RBFRD	0.9954	0.0306
NNIEU	0.9985	0.1439
NNIMH	0.9956	0.1451
WD11	0.9715	0.3713
WD21	0.9674	0.3037
WD12	0.9608	0.1707
WD22	0.9745	0.2986

Note: The notation is same as that of simulation I.

This simulation gives the evidence that the weighted distance based nearest neighbour imputation can be the best choice in situations that the correlation between the response relates to the distance of its covariates.

Simulation VII: Binary RBF Model

Model:

$$\Pr(y=0)=exp(\Phi(x_1, x_2,\mu))/(1+exp(\Phi(x_1, x_2,\mu)))= logit(\Sigma w_j\phi(x_1, x_2,\mu_j)),$$

where $\phi(x_1, x_2,\mu_j)=\exp(-((x_1-\mu_{j1})^2+(x_2-\mu_{j2})^2)/\lambda)$, W=(-.3,.5,1), μ={(8,4),(12,6)}, λ=2.

The expression inside the *logit* function is the same as the systematic part of the continuous RBF model in simulation III. With the true values of the RBF model known, the RBF imputations demonstrate remarkable performance in terms of predicting the true values (see Table 5.1.17 Table 5.1.18). The marginal distribution is well preserved by all imputations. Meanwhile the distance-based imputations also show its strength. In the meantime, if the true values of model parameters are not set as the initial values in the

training process of the RBF model, the promising results may have not been obtained especially when the true data centres are not achieved in the training process. After all in the ideal situation when the true model is represented by a RBF model and the true values of its parameters are achieved in the training process, RBF imputations could be the best approach.

Table 5.1.17 The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the RBF model.

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
True	0.4932	1.0000
LogisticHP	0.4750	0.5997
LogisticRD	0.4807	0.5242
RBFHP	0.4892	0.9877
RBFRD	0.4897	0.9846
NNIEU	0.4660	0.8924
NNIMH	0.4667	0.8927
WD11	0.5159	0.9017
WD21	0.5110	0.9102
WD12	0.5192	0.8962
WD22	0.5132	0.9006

Note: The notation is same as that of simulation I.

Table 5.1.18 P values of t-test: The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the RBF model

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
LogisticHP	0.9132	0.1351
LogisticRD	0.9394	0.0927
RBFHP	0.9805	0.9402
RBFRD	0.9827	0.9257
NNIEU	0.8730	0.5840
NNIMH	0.8759	0.5847
WD11	0.8926	0.6116
WD21	0.9148	0.6383
WD12	0.8779	0.5952
WD22	0.9046	0.6083

Note: The notation is same as that of simulation I.

Simulation VIII: Logistic Non-linear Regression Model

Model:

$$\Pr(y=0)=logit(-12+.5*x_1+x_2+5*\sin(x_1)+5*\sin(x_2)+1.1^{X_1}+1.2^{X_2})$$

This model is the categorical counterpart of model IV. The expression inside the *logit* function is a non-linear function. It is created to compare the performances of the ten imputation methods when the true model does not fit into any of them.

Table 5.1.19 The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the non-linear model

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
True	0.5007	1.0000
LogisticHP	0.4933	0.6610
LogisticRD	0.5060	0.5686
RBFHP	0.5059	0.7033
RBFRD	0.5038	0.7450
NNIEU	0.5127	0.8870
NNIMH	0.5130	0.8873
WD11	0.5000	0.9028
WD21	0.5057	0.9067
WD12	0.5028	0.8893
WD22	0.5053	0.8897

Note: The notation is same as that of simulation I.

The results in Table 5.1.19 and Table 5.1.20 show that the imputations of the distance-based nearest neighbour methods are much better than that of the linear logistic imputations and the RBF imputations. It is easy to understand that the linear logistic imputations do not work well under this model assumption, as the true model is not linear. The inferior performance of the RBF imputations in preserving the true values can be explained as that the true values of the parameters of the equivalent RBF model to the non-linear model are not reached in the training process. Although the theory shows that the RBF model has the potential, in practice it could be an unachievable task. It is this

reason that the scientists in the neural networks area pay much attention to the algorithms which are frequently referred as machine learning methods.

Table 5.1.20 P values of t-test: The marginal distribution (probability of y=0) and the proportion of correct imputation (based on 300 repetitions) of the non-linear model

Imputation Method	$\hat{p}_{300}$	Proportion of Correct Imputation
LogisticHP	0.9425	0.0337
LogisticRD	0.9587	0.0134
RBFHP	0.9596	0.0515
RBFRD	0.9759	0.0781
NNIEU	0.9089	0.3230
NNIMH	0.9065	0.3241
WD11	0.9942	0.3782
WD21	0.9613	0.3932
WD12	0.9838	0.3304
WD22	0.9639	0.3318

Note: The notation is same as that of simulation I.

Among the six nearest neighbour imputation methods based on four different distance measures, it seems difficult to improve upon the Euclidean distance measure. The imputation based on the Mahalanobis distance is almost identical to that based on the Euclidean distance, and the improvements from the weighted distance methods are minor. One can improve the performance of weighted distance imputation by embedding neural networks in it, but the computing task will be increased exponentially, and becomes a big difficulty in implementation.

From the simulations for the categorical response data, the following findings can be summarised.

- The performance of an imputation method depends on the validity of the model assumptions.
- RBF imputations are very flexible, and could approximate many underlying models in practice.

- The imputation strategy of taking the value with the highest probability gives similar results to the random draw strategy, although the latter method tends to preserve the distribution somewhat better.
- The nearest neighbour imputations based on the Euclidean distance and the Mahalanobis distance always give similar results.
- The improvement of the weighted distance based on regression over the Euclidean distance may not be important if the true model is not the correlated response model.

In practice if the nearest neighbour imputation is acceptable, it might not be necessary to use the weighted distance. Otherwise if the Euclidean is not good enough, one could choose the weighted distance and use neural networks distance to improve the performance of the nearest neighbour imputation.

5.2 Numerical Study Based on 1991 Household Census Data

5.2.1 Data Source

Data from the 1991 population census of Great Britain are used. The original census data contains both personal and household variables (Dale, 1993). The data set we use here is a derived one that only contains variables at the household level. For computing concern, we only use the data from one area which has 9980 records. 7598 of them are complete records.

There are twenty-four variables in the derived data. Ten of them are indicators from the data checking and editing process carried out by the Office for National Statistics (ONS). Of the remaining fourteen variables, five variables are nearly constant (more than 99%). Therefore they are excluded in this study. Also a variable indicating geography is not included. In the end, there are eight variables left, as listed in Table 5.2.1. The two variables with most missingness are "number of rooms" (11.68%) and "cars and vans" (7.52%). The variable "number of rooms" has fourteen different values (from 1 to 14), and the variable of "cars and vans" has four different values (from 1 to 4). To speed up the computing, the categories of these variables are collapsed in the following way, Number of rooms: 1→2, 8-14→7, Number of cars: 2+→2. The collapsing of categories of "number of rooms" and "cars and vans" is shown in Table 5.2.2 and Table 5.2.3.

Table 5.2.1 Census Variables Used

Variables	**Values**	**Pct. of missing(%)**
Number of Rooms	1-14	11.68
Cars and vans	0-4	7.52
Type of building	2-8	5.81
Type of tenure	1-8	1.90
Type of heating	1-3	2.52
Number of household	01,02,11,12	4.55
Type of bathshower	1-3	2.53
Number of person in the house	01-09	2.52

Note: The percentages are based on the 9980 cases in one area.

The new distribution is as follows.

Table 5.2.2 Distribution of Number of Rooms

Number of Rooms	Original classification (%)	Collapsed categories (%)
1	2.00	8.92
2	6.92	
3	21.88	21.88
4	28.36	28.36
5	22.37	22.37
6	9.11	9.11
7	3.64	9.36
8	2.61	
9	1.37	
10+	1.74	
All	100.00	100.00

Note: The percentages are based on the 9980 cases in one area.

Table 5.2.3 Distribution of Number of Cars

Number of Cars	Original classification (%)	Collapsed categories (%)
0	54.23	54.23
1	34.64	34.64
2	9.58	11.13
3	1.55	
All	100.00	100.00

Note: The percentages are based on the 9980 cases in one area.

5.2.2 Criteria Used to Evaluate Properties of Imputation Methods in This Numerical Study

In this numerical study, the response variables are categorical variable with more than two categories. We use Wald statistic to evaluate the consistency of marginal distributions (Stuart 1955, Chambers 1996). If we denote the marginal counts of the contingency table of the imputed values and the true values of Y_{mis} by R and S, the upper-left matrix (the contingency table without the last column and the last row) by T, the Wald statistic can be written as

$$\chi_W^2 = [R-S]\left[R+S-T-T'\right]^{-1}[R-S]. \qquad (5.2.1)$$

Under some regular conditions χ_W^2 follows a χ^2 distribution. From (5.2.1), we can conclude when the marginal distribution is well preserved, R and S are likely to be close which leads to small χ_W^2.

The statistic used for testing the consistency of true values is the proportion of off-diagonal values.

$$D = 1 - n^{-1}\sum_{j=1}^{n} I(\hat{y}_j = y_j). \qquad (5.2.2)$$

A small D indicates a good imputation in terms of predicting true values. The relationship between p_{c300} in (5.1.8) and D can be described by the following expression.

$$p_{c300} = \frac{1}{300}\sum_{i=1}^{300}(1-D_i),$$

where

$$D_i = 1-30^{-1}\sum_{j=1}^{30} I(\hat{y}_{ij} = y_{ij}).$$

Therefore these two measures are consistent in nature.

5.2.3 Numerical Study

As mentioned in section 5.1, neural network methods are very time-consuming to train. We can not afford to build a neural networks model using all the available cases. Instead, two simple random samples are drawn from the 7598 complete cases as the training data and test data respectively. The training data are used to estimate the parameters of parametric models and build nonparametric models such as tree model and neural network models. The test data, whose responses are assumed missing, are used to compare the performance of competing imputation methods. The sample for training data has 600 cases, and the sample for test data has 200 cases. There is no overlap between the two data sets.

We choose the two variables with most missing values, "number of rooms" and "cars and vans", as response variables in this study. The first situation assumes only the variable "number of rooms" is missing in the test data, and all other seven variables are present. The second situation assumes the variable "cars and vans" in the test data is missing, all other variables are complete. The third situation deals with simultaneous missingness of "number of rooms" and "cars and vans", where both response variables in the test data are assumed missing.

For computing hardware restriction, this numerical study is just a one-time sample with no repetition. Once the training data and test data are created, they will keep unchanged. The missing values are created from the test data by assuming one or two variables missing in corresponding situations. That gives another dimension of variation in the results. In that some of the promising imputations could be resulted from the sample selection instead of good performance of the imputation method.

5.2.4 Imputation Methods

The imputation methods included in this numerical study are multinomial regression imputation based on the highest probability (Multinomial), nearest neighbour imputation (NNI), tree model imputation (Tree) (see section 1.6.2), RBF neural networks imputations (with sum-square-error, RBF:SSE, and Wald error, RBF:WALD, respectively), MLP neural networks (with sum-square-error, MLP:SSE, and Wald error, MLP:WALD, respectively) and weighted distance imputation based on nearest neighbour method (WD12). The weighted distance is the square value distance without cross term. It is built on regression model.

As aforementioned, the comparison of different imputation methods is mainly based on *Wald* statistic (see 5.2.1), because these variables are categorical. The values of the statistic D for evaluating correct imputation are also provided. The imputations are carried out in the three circumstances described in previous section.

5.2.5 Results when the Values of "number of rooms" are Missing

Table 5.2.4. Wald Statistics and *D*: "Number of Rooms"

Imputation Method	Wald Statistic (df=5)	P Value of Wald	*D*	Time	Model specification
Multinomial	6.86	.2316	0.31	10 mins	
NNI	9.24	.0998	0.27	3 mins	
Tree model	4.01	.5479	0.28	2 mins	Node:54,1e-3
RBF: SSE	3.41	.6373	0.17	2 days	Node:25,1e-4
RBF: Wald	3.45	.6303	0.29	7 days	Node:25,1e-4
MLP: SSE	1.26	.9388	0.15	3 days	Node:30,1e-4
MLP:Wald	2.00	.8487	0.25	4 days	Node:30,1e-4
WD12	4.21	.5186	0.28	3 mins	

Note: The above results are based on the imputed values of the test data which has 200 cases.

The values of the Wald Statistic in Table 5.2.4 reveal that the MLP imputation with sum-square-error gives the lowest value, and thus preserves the distribution of this variable best. In Table 5.2.5, the distributions based on the imputed values and the true values are displayed along with the means and variances. It also shows that the MLP imputation based on SSE error function preserves the mean and variance very well. Meanwhile the MLP with Wald error function best preserves the variance. Considering the p-values, the results do not show significant differences except for nearest neighbour imputation (NNI). In Table 5.2.5, NNI does preserves the mean very well but gives very biased variance estimation. The poor performance of NNI might be caused by the small sample size of training data, in the sense the best donor is not included in the training data. Also, in the previous simulations, the nearest neighbour imputation gives promising results.

The statistic *D* is used to assess how well the true values are predicted. Smaller *D* indicates better prediction for true values. In Table 5.2.5, RBF imputation and MLP imputation with sum-square-error predict the true value better than the other imputation methods.

The RBF imputations seem not to outperform MLP imputations. It could be the result of improper specification for the centres. In theory both of them can be configured to be any functions, while the disadvantage of both neural networks models is the time needed to

approximate the underlying model. This is caused by the complexity of the neural networks model. Generally speaking, the best model is rarely known before one begins to build it. One of the practical approaches is to use cross validation method to try many models and find the best one. It is the cross validation process that takes substantial time. The other problem is the multi peak phenomenon. Because the optimisation target (error function) is a complex non-linear function of the neural networks weights, therefore there is no guarantee the global minimum can be obtained numerically. For RBF model the peak points are actually the locations of RBF centres. The computing time is also affected by the initialisation of the weights and centres, which is time-consuming.

If the training time is a big concern, the multinomial logistic imputation and tree model imputation may be a good choice.

Table 5.2.5.Marginal distribution of imputation: "Number of Rooms"

Imputation Method	1-2	3	4	5	6	7+	mean	Vari-ance	Departure from the true mean	Departure from the true variance
Original	12.5	19.0	25.0	22.0	12.0	9.5	4.29	35.77	0.00	0.00
Multinomial	7.5	21.5	28.0	26.5	8.0	8.5	4.32	20.53	0.03	-15.24
NNI	10.0	21.0	23.5	29.0	4.5	12.0	4.34	16.20	0.05	-19.57
Tree model	9.5	26.5	24.0	21.0	11.0	8.0	4.21	31.88	-0.08	-3.89
RBF: SSE	10.0	20.5	27.5	23.5	7.5	11.0	4.32	25.05	0.03	-10.72
RBF: Wald	11.5	20.5	32.0	18.5	8.5	9.0	4.18	29.29	-0.11	-6.48
MLP: SSE	11.0	17.5	28.5	23.5	11.5	8.0	4.30	28.77	0.00	-7.00
MLP: Wald	10.0	22.0	24.0	20.5	11.0	12.5	4.39	35.69	0.10	-0.08
WD12	9.0	23.5	27.0	23.5	8.5	8.5	4.24	24.87	-0.05	-10.90

Note: The above results are based on the imputed values of the test data which has 200 cases. The numbers under the room numbers are percentages.

5.2.6 Results when the Values of "cars and vans" are Missing

Different imputation methods may perform well with some variables but not for other variables. To investigate this issue we consider the situation of another variable that has a high percentage of missing values, the "cars and vans". The same study as in the previous

section is carried out, now with this variable missing. The results are presented in Table 5.2.6 and Table 5.2.7.

Table 5.2.6. Wald Statistics and *D*: "Cars and Vans"

Imputation Method	Wald Statistic Df=2	P Value of Wald	*D*	Time
Multinomial	2.62	.2703	0.51	8 mins
NNI	8.98	.0112	0.56	3 mins
Tree model	4.13	.1266	0.45	2 mins
RBF: SSE	2.40	.3013	0.57	36 hours
RBF: Wald	3.11	.2114	0.48	5 days
MLP: SSE	.70	.7063	0.61	2 days
MLP:Wald	1.68	.4311	0.46	3 days
WD12	.15	.9258	0.52	2 mins

Note: The above results are based on the imputed values of the test data which has 200 cases.

The results in Table 5.2.6 show that the weighted distance based nearest neighbour imputation (WD12) provides a very good distributional fit in terms of Wald statistic. Again the p-values suggest all methods perform well except NNI. From the evaluations in Table 5.2.7, the eight imputation methods give quite similar results in terms of the biases of the mean estimators and the variance estimators based on the imputed values. As indicated by the p-value the imputed values for NNI appear to lead to systematic under-prediction. Also there is possibility of WD12 outperforming others in this study.

From the *D* values, one can find that all imputations do not preserve the true values very well with the best result given by MLP.

In terms of the time needed for computing, the Wald error based neural networks take much longer time than other methods including the SSE based neural networks. This may suggest not use Wald error. Because it makes training more difficult. If both efficiency and accuracy are concerned, we may prefer to the weighted distance imputation and multinomial imputation based on the results this study.

Table 5.2.7.Marginal distribution of Imputation: "Cars and Vans"

Imputation Method	0	1	2+	Mean	Variance	Departure from the true mean	Departure from the true variance
Original	55.5	35.0	9.5	0.59	2.26	0.00	0.00
Multinomial	52.0	41.5	6.5	0.58	1.95	-0.01	-0.31
NNI	67.0	28.0	5.0	0.41	2.29	-0.19	0.03
Tree model	51.0	33.5	15.5	0.72	1.53	0.13	-0.73
RBF: SSE	55.0	31.5	13.5	0.65	1.99	0.06	-0.27
RBF: Wald	59.5	35.5	5.0	0.48	2.10	-0.11	-0.16
MLP: SSE	53.0	38.5	8.5	0.60	2.11	0.01	-0.15
MLP:Wald	54.5	39.0	6.5	0.55	2.09	-0.04	-0.17
WD12	56.0	35.5	8.5	0.57	2.29	-0.02	0.03

Note: The above results are based on the imputed values of the test data which has 200 cases. The numbers under the car numbers are percentages.

5.2.7 Results when Both Values of "number of rooms" and "cars and vans" are Missing

Table 5.2.8. Wald Statistics and *D*: both are missing

Imputation Method	ROOMS(df=5)			CARS(df=2)			Time
	Wald Statistic	P Value of Wald	*D*	Wald Statistic	P Value	*D*	
Multinomial	5.94	.3119	0.27	1.86	.3940	0.57	30 mins
NNI	7.86	.1641	0.28	.49	.7837	0.49	5 mins
Tree model(57,1e-3)	4.45	.4870	0.34	.32	.8539	0.53	4 mins
RBF: SSE(25,1e-4)	6.54	.2572	0.22	3.13	.2089	0.53	3 days
RBF: Wald(25,1e-4)	8.07	.1522	0.19	4.96	.0834	0.54	10 days
MLP: SSE(30,1e-4)	5.23	.3879	0.29	1.80	.4075	0.50	5 days
MLP:Wald(16,1e-4)	10.02	.155	0.19	5.90	.03	0.55	6 days
WD12	3.21	.6225	0.24	.29	.8668	0.53	10 mins

Note: The above results are based on the imputed values of the test data which has 200 cases.

In previous sections we investigated the situation of one variable containing missing values. In the census data, the simultaneous missingness of multiple variables is also very common. Here we deal with the situation with two variables missing. In imputing the multivariable missing values, not only the marginal distributions of single variables should be preserved, the joint distribution should be preserved as well. For the categorical

variables, the joint distribution is simply the distribution of the combined categories. Therefore it can be treated as a new single categorical variable. We impute the values for the derived variable. The imputed category for individual response variable can be obtained accordingly. There is one practical problem that makes the imputation more unreliable. It is the number of complete cases. When two single variables are joined together the complete cases are much less than any of the single variable. Some imputation methods that perform very well with large datasets may not give good result to the joined variable. Fortunately, this does not happen in these data.

Table 5.2.9.Marginal distribution of Rooms based on Imputations

Imputation Method	ROOMS									
	1-2	3	4	5	6	7+	mean	variance	Departure from the true mean	Departure from the true variance
Original	12.5	19.0	25.0	22.0	12.0	9.5	4.29	35.77	0.00	0.00
Multinomial	8.0	18.5	32.5	23.5	9.5	8.0	4.32	22.79	0.03	-12.98
NNI	9.5	23.5	23.0	25.0	6.0	13.0	4.35	22.31	0.06	-13.46
Tree model	11.5	20.0	30.5	22.0	8.5	7.5	4.17	25.12	-0.13	-10.65
RBF: SSE	12.5	21.0	29.5	22.5	5.5	9.0	4.13	21.05	-0.16	-14.72
RBF: Wald	12.5	20.5	17.0	19.0	15.0	16.0	4.53	47.24	0.24	11.47
MLP: SSE	11.5	23.5	26.5	19.0	7.0	12.5	4.25	29.91	-0.04	-5.86
MLP:Wald	14.5	14.0	20.0	15.0	17.5	18.5	4.65	50.03	0.36	14.26
WD12	9.0	24.0	24.0	24.5	9.5	9.0	4.29	27.23	0.00	-8.54

Note: The above results are based on the imputed values of the test data which has 200 cases. The numbers under the room numbers are percentages.

Table 5.2.8 contains the evaluation statistics based on imputations when both the "number of rooms" and the "cars and vans" are assumed missing. The weighted distance imputation gives the most consistent imputation in terms of Wald statistic. MLP with sum-square-error and tree imputation are also promising. The MLP imputation with Wald error gives the highest Wald value for both variables and displays significant lack of fit ($p<0.05$) in the case of both variables missing. It may tell that the Wald error is not a suitable error function. Again the D values indicate all imputation methods do not predict the true values very well.

In Table 5.2.9 and Table 5.2.10, all imputations preserve the mean very well except MLP with Wald error for the variable of Number of Rooms. The performances in preserving the variance are much more variable with the smallest deviation given by WD12.

Table 5.2.10. Marginal distribution of "Cars and vans" based on Imputations

Imputation Method	CARS						
	0	1	2+	mean	variance	Departure from the true mean	Departure from the true variance
Original	55.5	35.0	9.5	0.59	2.26	0.00	0.00
Multinomial	61.0	29.5	9.5	0.53	2.54	-0.06	0.28
NNI	52.5	37.0	10.5	0.63	2.04	0.04	-0.22
Tree model	54.0	35.0	11.0	0.63	2.11	0.04	-0.15
RBF: SSE	52.0	33.0	15.0	0.71	1.65	0.12	-0.61
RBF: Wald	51.0	32.5	16.5	0.74	1.43	0.15	-0.83
MLP: SSE	56.5	30.5	13.0	0.63	2.12	0.04	-0.14
MLP:Wald	38.0	40.0	22.0	0.95	0.40	0.36	-1.86
WD12	53.5	37.5	9.0	0.60	2.14	0.01	-0.12

Note: The above results are based on the imputed values of the test data which has 200 cases. The numbers under the car numbers are percentages.

If the computing time is considered, the right choices might be the tree imputation and weighted distance imputation, especially tree imputation.

5.2.8 Conclusions from the Study with Census Data

This numerical study based on census data shows that the neural networks imputation methods such as RBF imputation and MLP imputation are capable of dealing with the unknown model. The tree imputation method is also attractive especially when the computing time is concerned. In the first situation, the linear logistic imputation seems not very successful compared to other imputations except NNI in terms of the values of Wald statistic. It may imply that the linear assumption doesn't hold in this data. This is the situation where neural networks and tree model can outperform other methods by their flexibility, in which they are very flexible to be adjusted (trained) to be the underlying model. The latter method can be difficult and time-consuming, which can be seen from the time listed in the last columns in above tables. Based on the performance of NNI in

previous simulations, the poor performance of NNI may result from the distance measure, Euclidean distance, since the weighted distance based nearest neighbour imputation does perform better in this situation.

Overall, the results of Wald evaluation show that all imputations can adequately preserve the marginal distributions very well in some circumstances. In the meantime, the D values don't vary dramatically among these imputations. The sharp contrast is in computing time, where neural networks take much more time than the other imputations.

Both the simulation study in the previous section and this numerical study using census data show that neural network imputation can outperform regression-based imputation in some circumstances. The results of the simulation study give more dimensions to compare the competing imputation methods. Meanwhile in the simulation study the true models are predetermined, it is possible to see how an imputation method preserves the population mean and variance, although the lack of variance in the T-test makes the conclusion less convincing. The numerical study shows how neural work imputation method and other imputation methods perform in the real situation. The practical concern about excessive computing time needed to train neural networks is revealed by the numerical study. Among the competing imputation methods, NNI demonstrates quite different performances in the two studies. In the simulation study, NNI performs very well. On the other hand, in the numerical study, NNI gives the highest Wald value, which indicates poor performance. The poor performance of NNI in the numerical study may be resulted from the type of covariates. In the simulation study, the two covariates are continuous. In the numerical study the majority covariates are categorical which may lead to the poor performance of Euclidean-based NNI. Meanwhile the weighted distance-based NNI gives better imputation result. This may suggest the weighted distance-based imputation method is capable of coping with different types of covariates. Meanwhile the performance of an imputation method not only depends on which model is chosen but also depends on how it is implemented.

6 Conclusions and Future Research

6.1 Conclusions

The theoretical results in chapter 4 and the simulation results in chapter 5 show how the performance of an imputation method depends on the validity of the model assumptions. If the underlying model which generates the data coincides with the model that the imputation method is based on, the performance is good in terms of preserving population properties like the mean variance and distribution. If the two models do not coincide, the performance is not much different to other imputations. However, the neural network model has the flexibility to approximate many unknown data-generating models, which may give the advantage of RBF imputation.

Among the model-based imputation methods, such as imputation based on a linear regression model or the RBF model, random imputation methods, such as random regression imputation and random RBF imputation, outperform other variations based on the same model (such as mean imputation) in terms of preserving the variance and distribution (specifically the quantiles in the simulations). Mean imputation deflates the variance and distort the distribution in both simulations.

The neural network imputation methods such as RBF and MLP are very flexible, and can be trained to the underlying model of the given data. However, the flexibility leaves many factors to be optimised in the training process, which makes the implementation very difficult and reaching the true model not guaranteed. With the rapid growing of computing power, the difficulty could be eased in the future. The simulation results are favourable to the RBF imputation. That may encourage us to find the way to specify the RBF model properly. One of the strategies is to link the parameters in RBF to the properties of the underlying model that generates the data. This will be discussed in the next section.

The weighted distance based nearest neighbour imputation method outperforms the Euclidean distance based imputation. In one extreme case, one can set all weights to be

one, then the weighted distance becomes Euclidean distance. There seems to be no loss in using the weighted distance. The striking disadvantage is the computation task needed in obtaining the weights, especially when neural network distance is used. With n records in the original data set, the resulting distance matrix has $n(n-1)/2$ units. It makes the computing very hungry of computer memory. On the other hand, one can improve the performance of the weighted distance imputation method by some iterative algorithms. One way to improve the performance of the weighted distance imputation method is to add a cross term, although the gain is not always significant, especially when the underlying model does not contain a cross term.

In the situation where the underlying model is not far from the linear regression model, the random regression imputation is a good choice in term of performance and computing time required. In this situation, the difference between the performances of the random linear regression and the random RBF imputation is minor. Meanwhile the distance-based nearest neighbour imputations are also promising. This phenomenon makes the decision in real situation more difficult, because it seems there is no obvious winner. It might be necessary to test the validity of model assumptions before you adopt it.

The simulations for categorical variable lead to similar conclusions that the performance of an imputation method depends on the validity of model assumptions. The RBF model demonstrates the flexibility of potentially being the true model, linear and non-linear. Meanwhile the distance based non-parametric imputations are also promising in predicting the categorical missing values in terms of preserving the true value and marginal distribution.

6.2 Ideas for Future Research

Given the difficulties experienced in the simulation studies of chapter 5 in using the RBF method, we found it would be a great help if some guidelines about how to specify the initial values of centres could be given. Some practitioners regard neural networks as black box for its indefinite structure and various training algorithms. This raises a concern about the application of neural networks. One way to deal with this problem is to link the

parameters in neural networks to the properties of the underlying model. In the simulation studies, we discovered that the peaks of the RBF model are located at the RBF centres in the covariates. If the weight is negative, the peaks will become the bottom points. Suppose we use RBF model to approximate an unknown model. The best centres of RBF should be the points in the covariates corresponding to the peaks and lowest values of the unknown model. The question is how to find them. These points could be the local maximum/minimum or global maximum/minimum. Therefore the question becomes how to find the local minimum and maximum. If the local maximum and minimum can be obtained efficiently, the performance of the RBF imputation could be improved, and the training process sped up. That would make the RBF model more attractive. Although this does not appear to be an easy task, it seems worth exploring.

In the census situation, the data contain millions of records. It is unrealistic to model the whole data by one big neural networks model. One practical approach is to split the original data into smaller subsets and then apply the neural networks to the subsets. There are several concerns about this strategy. One question is whether it is a good approach. How does it perform compared with the hot-deck imputation, which is widely used in census organisations? There could be no yes-no answer. The performance quite likely depends on how you build the combination of splitting and modeling. There comes the second question how to optimise the combination of splitting and modeling.

Imputation is about filling the holes in a data set. Outlier identification is about finding points that distort data analysis and providing the basis of robust estimation techniques. Editing is about modifying potentially wrong values or outliers in the data set. If we put the three methods together we may address a general situation where the data set can be regarded as a combination of usable units and unusable units which could be missing values, outliers or wrong values. That may suggest the methods in each area may be combined or used as alternative methods. Specifically we probably can use imputation approaches to solve the problems of in outlier identification and editing. For example, in the outlier identification problem, an initial idea is assuming the outliers are missing, then impute them. That may lead to a sensible result. This could be an interesting area to work on in future research.

References

Agresti, A. (1990). *Categorical Data Analysis*. New York, Wiley.

Anderson, R. L. (1946). Missing-Plot Techniques. *Biometrics Bulletin*, **2**, 41-47.

Anderson, T.W. (1984). *An introduction to Multivariate Statistical Analysis*. New York: Wiley.

Bard, Y. (1974). *Nonlinear Parameter Estimation*. New York: Academic Press.

Bartlett, M. S. (1937). Some Examples of statistical methods of Research in Agriculture and Applied Biology. *SUPPLEMENT TO THE ROYAL STATISTICAL SOCIETY*, **4**, 20.

Bates, Douglas M. (1988). *Nonlinear Regression Analysis and Its Applications*. New York: John Wiley & Sons, Inc.

Bishop, Christopher M. (1996). *Neural Networks for Pattern Recognition*. Oxford: Clarendon Press.

Breiman, Leo (1998). Bias, Variance, and Arcing Classifiers. *Annals of Statistics*. **26**, 801-823.

Chambers, R. (1996). Evaluation of Imputation. Manuscript.

Chan, L. S. and Dunn, O. J. (1972). The Treatment of Missing Values in Discriminant Analysis. The Sampling Experiment (in Theory and Methods). *Journal of the American Statistical Association*, **67**, 473-477.

Chen, Jiahua and Jun Shao (2000). Nearest Neighbor Imputation for Survey Data. *Journal of Official Statistics*, **16**, 113-131.

Chen, Jiahua and Jun Shao (2001). Jackknife Variance Estimation for Nearest-neighbor Imputation. *Journal of the American Statistical Association*, **96**, 260-269.

Cheng, Bing, and D. M.Titterington (1994). Neural Networks: A Review from a Statistical Perspective. *Statistical Science*, **9**, 2-54.

Clausen, J.A. and Ford, R. N. (1947). Controlling Bias in Mail Questionnaires. *Journal of the American Statistical Association*, **42**, 497-511.

Clifton, R. Wharton, Jr. (1960). Processing Underdeveloped Data from an Underdeveloped Area. *Journal of the American Statistical Association*, **55**, 23-37.

Creezy, R. H., Massand, B.M., Smith, S.J. and Waltz, D.L. (1992). Trading MIPS and Memory for Knowledge Engineering. *Communications of the ACM*, **35**, 48-63.

Cressie, Noel A. C (1993). *Statistics for Spatial Data*. New York : John Wiley & Sons, Inc..

David, A. P. and Skene, A.M. (1979). Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Applied Statistics*, **28**, 20-28.

David, M., Little, R. J. A., Samuhel, M. E., Triest, R. K. (1986). Alternative Methods for CPS Income Imputation (in Applications). *Journal of the American Statistical Association*, **81**, 29-41.

Dillman, D. A., Eltinge, J. L., Groves, R. M., Little, R. J. A. (2002). Survey Nonresponse in Design, Data Collection, and Analysis. *Survey Nonresponse*. John Wiley & Sons, Inc.

Fay, R. E. (1991). A Design-based Perspective on Missing Data Variance. *Proceedings of the 1991 Annual Research Conference*, U.S. Bureau of the Census, 381-440.

Fellegi, I. P. and Holt, D. (1976). A Systematic Approach to Automatic Edit and Imputation. *Journal of the American Statistical Association*, **71**, 17-35.

Ford, B. M. (1983). An Overview of Hot-deck Procedures. *Incomplete Data in Sample Surveys*. New York: Academic Press.

Freund, R. J., Hartley, H. O. (1967). A Procedure for Automatic Data Editing. *Journal of the American Statistical Association*, **62**, 341-352.

Gregor, P.J. Schmitz, Chris Aldrich, and Francois S. Gouws (1999). ANN-PT: An Extraction of Decision Trees from Artificial Neural Networks. *IEEE transaction on Neural Networks*, **10**, 201-208.

Groves, Robert M. and Robert Louis Kahn (1979). *Surveys by Telephone: A National Comparison with Personal Interviews*. New York: Academic Press.

Hansen, Morris H, Hurwitz, William N. (1946). The Problem of Non-Response in Sample Surveys. *Journal of the American Statistical Association*, **41**, 517-529.

Hardle, Wolfgang (1989). *Applied Nonparametric Regression*. Cambridge: Cambridge University Press.

Haitovsky, Y. (1968). Missing Data in Regression Analysis. *Journal of the Royal Statistical Society. Series B (Methodological)*, **30**, 67-82.

Jaszi,. G. (1951). National Income: Status and Prospects as Seen by an Estimator. *Journal of the American Statistical Association*, **46**, 345-357.

Johnson, Richard A (1992). *Applied Multivariate Statistical Analysis*. New York : Springer.

Kaufman, G. M. and King, B. (1973). A Bayesian Analysis of Nonresponse in Dichotomous Processes. *Journal of the American Statistical Association*, **68**, 670-678.

Kay, J. W. and D. M. Titterington (1999). *Statistics and Neural Networks*. Oxford : Oxford University Press

Kohonen, T. (1995). *Self-Organizing maps*. Berlin: Springer.

Lee, H., Rancourt, E. and Sarndal, C. E. (2002). Variance Estimation from Survey Data under Single Imputation. *Survey Nonresponse*. New York: John Wiley &Sons, Inc.

Lessler, J. and Kalsbeek, W. D. (1992). *Nonsampling Errors in surveys*. New York: John Wiley & Sons, Inc.

Lindsey James K. (1996). *Parametric Statistical Inference*. Oxford : Clarendon Press.

Little, Roderick J. A. (1982). Models for Nonresponse in Sample Surveys. *Journal of the American Statistical Association*, **77**, 237-250.

Little, Roderick J. A. (1992). Regression With Missing X's: A Review (in Theory and Methods). *Journal of the American Statistical Association*, **87**, 1227-1237.

Little, Roderick J. A. and Donald B.Rubin (1987). *Statistical Analysis with Missing Data*. New York: John Wiley & Sons, Inc.

Lury, D. B. (1946). The Analysis of Latin Squares when Some Observations are Missing. *Journal of the American Statistical Association*, **41**, 370-389.

Mason, R., Lesser, V. and Traugott, M. W. (2002). Effect of Item Nonresponse on Nonresponse Error and Inference. *Survey Nonresponse*. New York: John Wiley & Sons, Inc.

Matern, B (1960). Spatial Variation. *Lecture Notes in Statistics*, **36**, Springer, New York.

McCullock, W., S., Pitts, W. (1943). A logical Calculus of the Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, **5**, 115-117.

Mojirsheibani, Majid(1999). Combining Classifiers via Discretization. *Journal of American Statistical Association*, **94**, 600-609.

Nordbotten, Svein (1995). Editing Statistical Records by Neural Networks. *Journal of Official Statistics*, **11**, 391-411.

Nordbotten, Svein (1996). Neural Network Imputation Applied to the Norwegian 1990 Population Census Data. *Journal of Official Statistics*, **12**, 385-401.

Peng, F. C., Jacobs R. A. and Tanner, M. A. (1996). Bayesian Inference in Mixtures-of-Experts and Hierarchical Mixtures-of-Experts Models With an Application to Speech Recognition (in Applications and Case Studies). *Journal of the American Statistical Association*, **91**, 953-960.

Phillips, H. S. (1956). United Kingdom Indices of Wholesale Prices, 1949-1955. *Journal of the Royal Statistical Society. Series A (General)*, **119**, 239-283.

Pratap, Rudra (2001). *Getting Started with MATLAB 6: A Quick Introduction for Scientists and Engineers*. Oxford: Oxford University Press Inc.

Rao, J. N. K. and Shao, J. (1992). Jackknife Variance Estimation with Survey Data Under Hot Deck Imputation. *Biometrika*, **79**, 811-822.

Rao, J. N. K. and Sitter, R. R. (1995). Variance Estimation Under Two-phase Sampling with Application to Imputation for Missing Data. *Biometrika*, **82**,453-460.

Reilly, M. and Pepe, M. S. (1995). A Mean Score Method for Missing and Auxiliary Covariate Data in Regression Models. *Biometrika*, **82**, 299-314.

Ripley, B. D. (1997). *Pattern Recognition and Neural Networks*. Cambridge University Press.

Ripley, B.D., (1999). *Modern Applied Statistics with S-PLUS*. Third Edition. New York : Springer-Verlag.

Rogers, Theresa F. (1976). Interviews by Telephone and in Person: Quality of Responses and Field Performance. *Public Opinion Quarterly*, **40**, 51-65.

Rockwell, R. C. (1975). An Investigation of Imputation and Differential Quality of Data in the 1970 Census (in Applications). *Journal of the American Statistical Association*, **70**, 39-42.

Rubin, D. B. (1977). Formalising Subjective Notions About the Effect of Nonrespondents in Sample Surveys. *Journal of the American Statistical Association*, **72**, 538-543.

Rubin, D. B. (1978). Multiple Imputations in Sample Surveys—a Phenomenological Bayesian Approach to Nonresponse. *Proceedings of the survey research methods section of the American Statistical Association*, 20-34.

Rubin, D.B. (1976). Inference and Missing Data. *Biometrika*, **63**, 581-592.

Rubin, Donald B. (1987). *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley & Sons, Inc.

Rubin, D. B. and Schenker, N. (1986). Multiple Imputation for Interval Estimation From Simple Random Samples With Ignorable Nonresponse (in Survey Research Methods). *Journal of the American Statistical Association*, **81**, 366-374.

Särndal, C-E (1992). Methods for Estimating the Precision of Survey Estimates when Imputation has been Used. *Survey Methodology*, **18**,241-252.

Schafer, J. L. (1997). *Analysis Incomplete Multivariate Data*. CHAPMAN &HALL.

Scott, Christopher (1961). Research on Mail Surveys. *Journal of the Royal Statistical Society*, Series A, **124**, 143-205.

Skinner, C. J. and Coker, O. (1996). Regression Analysis for Complex Survey Data with Missing Values of a Covariate. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, **159**, 265-274.

Smith, T. W. (2002). Developing Nonresponse Standards. *Survey Nonresponse.* New York: John Wiley & Sons, Inc.

Wald, A. (1948). Asymptotic Properties of the Maximum Likelihood Estimate of an Unknown Parameter of a Discrete Stochastic Process. *Annals of Mathematical Statistics*, **19**, 40-46.

Warner, Stanley L. (1965). Randomised Response: A Survey Technique for Eliminating Evasive Answer Bias. *Journal of the American Statistical Association,* **60**, 63-69.

Watson, G. S. (1956). Missing and "Mixed-Up" Frequencies in Contingency Tables. *Biometrics*, **12**, 47-50.

Wilkinson, G. N. (1958). The Analysis of Variance and Derivation of Standard Errors for Incomplete Data. *Biometrics*, **14**, 360-384.

Yaglom, A. M. (1957). Some Classes of Random Fields in N-dimensional Space, Related to Stationary Random Processes. *Theory of Probability and Its Applications*, **2**, 273-320.

Zhu, Qiuming, Yao, C., Liu, L. (1999). A Global Learning Algorithm for a RBF Network. *Neural Networks*, **12**, 527-540.