Paper 3643-2019

Exploration of missing data imputation methods

Humphrey Brydon, Rénette Blignaut, University of the Western Cape

ABSTRACT

The study presented in this paper looked at possible methods and processes involved in the imputation of complete missing blocks of data. A secondary aim of the study was to investigate the accuracy of various predictive models constructed on the blocks of imputed data.

Hot-deck imputation resulted in less accurate predictive models, whereas a single or multiple Monte Carlo Markov Chain or the fully conditional specification imputation methods resulted in more accurate predictive models.

An iterative bagging technique applied to variants of the neural network, decision tree and multiple linear regression improved the estimates produced by the modelling procedures. A stochastic gradient boosted decision tree was also constructed as a comparison to the bagged decision tree.

The results indicated that the choice of an imputation method as well as the selection of a predictive model is dependent on the data and hence should be a data-driven process.

INTRODUCTION

Missing data are frequently observed in data, even large data sets can contain incomplete and missing data. Many analytical procedures are unable to model or cater for incomplete observations and, as a result, omit such observations from the modelling procedure and any derived analysis (i.e. a complete case analysis approach is followed).

Missing data imputation methods were developed for instances where the omission of observations with missing data would lead to a loss in information. Imputation methods, by definition are techniques that can be used to estimate missing data from the available data. The question then is: Would the model constructed on the imputed data or on the original incomplete data result in the best predictive model?

The first section of this paper discusses the data source used in this study. The second section discusses the various missingness mechanisms that should ideally be identified prior to the adoption of any imputation method. This is followed by a discussion of the various imputation methods explored in this study.

The third section of this paper describes the different predictive models constructed on the imputed blocks of data. The fourth and fifth sections of this paper discuss the results of the various modelling procedures with an emphasis on the effect of the imputed data mechanism on model accuracy.

DATA SOURCES

The specific data sets used in this study contained partially observed data as well as complete missing blocks of data (i.e. no data observed for a given observation). Since the data used in this study is a real-world data set. In order to preserve the confidentiality of the data source and the data set components, it was necessary to mask the data. This did not result in any loss of interpretability of the results in this paper.

The first data set (referred to as '*Data set 1*') contained 57517 observations and 5 variables. The second data set (referred to as '*Data set 2*') contained 76637 observations and the

same 5 variables as well. All 5 variables, of which one was the target variable, are continuous type variables.

It was found that the two original data sets each contained two separate distributions (w.r.t the target variable) and needed to be split to predict a specific target. The splitting of the data resulted in four separate data sets for implementation of the imputation and modelling phase of this study. These data sets will be referred to as '*Data set 1 Split 1*', '*Data set 1 Split 2*', '*Data set 2 Split 1*' and '*Data set 2 Split 2*' in this paper.

IMPUTATION METHODS

Missingness within data can be identified by or related to a missingness mechanism. If the identification of the missingness mechanism is done prior to data imputation, then the selection of an appropriate imputation method to impute the missing data can be done appropriately. As mentioned by Little and Rubin (2002), the three missingness mechanisms that can be identified in the presence of missing data are: missing completely at random (MCAR), missing at random (MAR) and not missing at random (NMAR). Note that the missingness mechanisms do not assume that the occurrence of missing data is random but are more of an indication of the relationship between the missing data and the observed data.

If the full data set of size n, containing p variables can be expressed as $Y = (y_{ij})$, where $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, p$ then a missing indicator matrix for this data set can be expressed as $M = (m_{ij})$, where $m_{ij} = 1$ is indicative of y_{ij} being missing and $m_{ij} = 0$ been indicative of y_{ij} being observed.

The missingness mechanism can be expressed as the conditional distribution of the missing indicator matrix given the complete data set, i.e. $f(M|Y, \emptyset)$, where $\emptyset$ denotes unknown parameters (Little & Rubin, 2002).

If missingness is neither dependent on the observed values of the data set Y nor those of the missing data, then the missing mechanism is said to be MCAR and can therefore be expressed as (Little & Rubin, 2002):

$$f(M|Y, \emptyset) = f(M|\emptyset) \; for \; all \; Y, \emptyset. \qquad \textbf{Equation 1}$$

If the presence of missing data is dependent on only the values of the observed data of Y the missing mechanism is said to be MAR and the missing mechanism can be described as (Little & Rubin, 2002):

$$f(M|Y, \emptyset) = f(M|Y_{obs}, \emptyset) \; for \; all \; Y_{mis}, \emptyset, \qquad \textbf{Equation 2}$$

where Y_{obs} are the observed components of Y and Y_{mis}, the missing components of Y (Little & Rubin, 2002). If the missing data value itself is causing the data value to be missing then the missing mechanism is said to be NMAR.

If the missingness mechanism can be identified, the ensuing step would be to identify a possible imputation method to impute the missing values. Depending on the pattern of the missing data (i.e. the order of the partially observed and missing values), whether that be monotone or random, as shown in Table 1 and Table 2, various imputation methods are

available. If the data allow for it, variables with missing data can be re-arranged in such a manner that the missing pattern becomes monotone (Little & Rubin, 2002).

Observation	Variable 1	Variable 2	Variable 3
1	X	X	X
2	X	X	.
3	X	.	.

Table 1. Monotone missing pattern

Observation	Variable 1	Variable 2	Variable 3
1	X	X	X
2	X	X	.
3	.	.	X
4	.	X	.
5	X	X	X
6	.	.	.

Table 2. Random missing pattern

For the data used in this study, a random missing data pattern was observed. Due to the nature of the variables contained in the data, the variables could also not be re-arranged to obtain a monotone missing data pattern. As a result of the random missing data pattern in the data, a hot-deck or multiple imputation method was deemed appropriate for the imputing of missing values.

TRUE IMPUTATIONS

Due to the fact that the two main data sets used in this study shared information w.r.t the variables contained in the data, observed values for variables contained in one data set could be imputed or substituted in the second data set where the value for that specific observation and variable was missing (in this study a time variable was also used to match observations in the different data sets). Figure 1 provides a depiction of this type of imputation method.

The true imputation, as it is referred to in this study, is more of a substitution method than an imputation method. This method of imputation or substitution avoided unrealistic imputations by substituting observed values contained within the original data (this is not to be confused with the hot-deck imputation method).

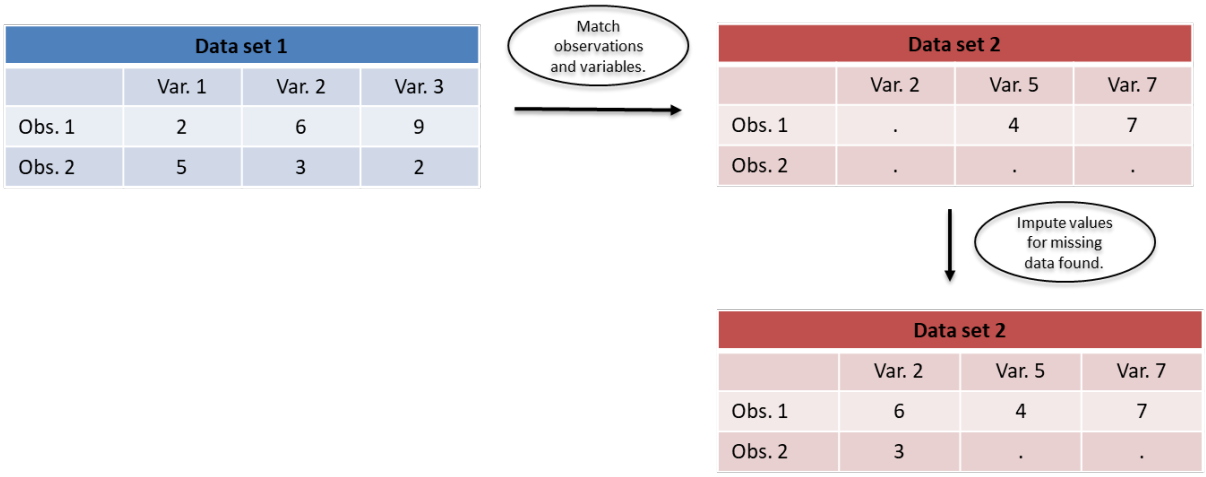

Figure 1. True imputation depiction

As was the case for the true imputation method discussed in this paper, where the use of the true imputation method was dependent on the fact that the two data sets shared information, in practice this could also be the case for other data sets and would therefore be dependent on specific domain knowledge. For those missing values for which the true imputation method did not find an observed value, either the hot-deck or multiple imputation method, discussed in the following sections, was followed. For the hot-deck and multiple imputation methods mentioned in the following sections, a total of 25 imputed data sets were created for each of the imputation methods.

HOT-DECK IMPUTATION

The hot-deck imputation method involves replacing the missing data value of an observation with that of a 'donor' observation's (i.e. substitution of values). The relationship between the two variables is therefore that of donor and recipient (Fuller, 2009).

Donors can either be selected using a 'with' or 'without' replacement sampling technique and this replacement technique can then either be random or non-random in nature (Bethlehem, 2009). The sampling of donors should ideally be done with replacement so as to maintain the randomness of donors being selected. For this study, sampling of donor observations was done with replacement.

As mentioned by Bethlehem (2009), the random donor selection method does increase the variance of the estimator due to the sampling technique and imputation method, however, this method does produce a slightly less biased estimator when compared to other single imputation techniques.

Another use of the hot-deck imputation method, not widely covered in the literature, would be to look at the effect when all variable values for a specific observation are imputed or donated from a donor observation where all the data values are missing for the recipient observation (i.e. imputing complete missing blocks of data).

MULTIPLE IMPUTATION

An improvement on the hot-deck single imputation method mentioned previously is the multiple imputation (MI) method. Although the MI method is an improvement on the single imputation method, the single imputation method is able to impute values when all values for a given observation are missing (i.e. 'donates' all variable values) whereas the MI method cannot perform such imputations.

MI is a method of imputation more preferred by many authors as the multiple imputation method incorporates the uncertainty of the imputed value by imputing multiple values for the missing value (Schafer, 1997). Both the single imputation and MI method can be used in a data set that has either a monotone or random missing patterned data as the imputation methods themselves are not dependent on this pattern.

The Monte Carlo Markov Chain (MCMC) is a common multiple imputation method used in the literature as it allows the user to set prior and posterior distributions of the missing data that are imputed. The main assumption underlying the MCMC method is that the data are of a multivariate normal distribution.

As mentioned by Schafer, when there is a departure from the multivariate normality assumption, the method is still robust enough to provide good estimates (Schafer, 1997). If the data can be represented as a vector y_i in the form (Schafer, 1997):

$$\boldsymbol{y}_i = (Y_1, Y_2, \dots, Y_p)^T, \qquad \textbf{Equation 3}$$

where i is the observation and p is the number of variables. A prior distribution needs to be chosen in order to estimate the values of μ, the mean vector and Σ, the covariance matrix of the estimated missing values of the parameters θ.

As the focus of multiple imputation is more on the uncertainty or variance of the estimate (i.e. the covariance matrix Σ), the use of Jeffrey's prior (also known as a non-informative prior) as a starting point, given in Equation 4, is suitable when little information is known about the prior distribution of the missing values (Schafer, 1997, p. 155). Jeffrey's prior can be represented as:

$$\pi(\theta) \propto |\boldsymbol{\Sigma}|^{-(\frac{p+1}{2})}, \qquad \textbf{Equation 4}$$

even though the use of Jeffrey's prior does not provide any information regarding the mean vector $\boldsymbol{\mu}$, the selection of any prior distribution should assist in stabilising the mean vector (SAS Institute Inc., 2015, p. 5905).

By using the non-informative Jeffrey's prior, the posterior distribution of Σ and μ becomes (Schafer, 1997):

$$\boldsymbol{\Sigma}^{(t+1)}|Y \sim W^{-1}[n-1, (n-1)S], \qquad \textbf{Equation 5}$$

$$\boldsymbol{\mu}^{(t+1)}|(\boldsymbol{\Sigma}^{(t+1)}, \boldsymbol{Y}) \sim N(\bar{y}, \frac{1}{n}\Sigma^{(t+1)}), \qquad \textbf{Equation 6}$$

where $\boldsymbol{Y}$ is the data matrix and W^{-1} the inverted Wishart distribution. The sample mean vector and sample covariance matrix are expressed as $\bar{\boldsymbol{y}}$ and $\boldsymbol{S}$ respectively. The expectation-maximization (EM) algorithm is used in order to compute the initial values for $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$, which are then used in the MCMC method. The number of iterations defined for the imputation method is expressed as t (where $t = 1, 2, \dots$).

MCMC can also be specified to conduct either a single or multiple chain imputation (i.e. parallel runs of the estimation process) (Schafer, 1997). This study will compare the effect of both single and multiple MCMC chains on model accuracy.

FULLY CONDITIONAL SPECIFICATION

Another multiple imputation method that can be used when the data contain a random missing pattern, is the fully conditional specification (FCS) method, specifically the predictive mean matching (PMM) approach of this imputation method (Van Buuren, 2007) (SAS Institute Inc., 2015).

The FCS PMM method imputes a missing value by selecting the closest fully observed observations and using their values to impute a value, in this instance, the mean of the closest observations' values (in this study the 5 closest observations were used). This method also carries the assumption that a joint distribution is present in the data for all observations (Van Buuren, 2007) (Heitjan & Little, 1991) (Schenker & Taylor, 1996).

The FCS method involves two steps in its process, namely; the filled-in phase and the imputation phase. In the filled-in phase, missing values are replaced or filled with observed values from the closest fully observed variables. This process is then re-run numerous times to create multiple filled-in data sets. At the imputation phase, the filled-in variables are used to impute the missing value using the specified model, in our case, the mean. These phases are run multiple times, depending on the number of iterations and imputations requested.

The FCS PMM is one of the simpler MI methods to apply, from a theoretical perspective. However, since a unique imputation model is created for each imputation and depending on the number of imputations, this method generally requires a longer computational time when compared to the MCMC MI method. Although, with the use of a high-end or supercomputer, this time is negligible.

COMBINING IMPUTATION METHODS

Each of the imputation methods mentioned in this section were carried out on either the *True* or *All* data set (as shown in Table 3). Where the *True* data set contained imputed data using the true imputation method as mentioned previously and the *All* data set contained the original data plus the inserted missing blocks of data (the complete missing blocks of data were discarded in the original data set due to the missing values). The reason that each imputation method was carried out on each of these two data sets was to ensure that combinations of all the various imputations were available for analysis, which would then assist in determining which combination or which single imputation method produces more accurate models.

Since the multiple imputation methods did not impute any values for the missing blocks of data, which was an expected outcome, the hot-deck imputation was used on these newly imputed data sets from the multiple imputation methods to generate complete data sets. A list of all the data sets created and their specific combination of imputation methods are listed in Table 3. A total of 17 data sets were generated for each of the four data sets ('*Data set 1 Split 1*', '*Data set 1 Split 2*', '*Data set 2 Split 1*' and '*Data set 2 Split 2*'), resulting in a total of 68 data sets available for the modelling process.

Data set	Imputation methods		
1	None		
2	All		
3	True		
4	All	Hot-Deck	
5	True	Hot-Deck	
6	All	MCMC Single	
7	All	MCMC Multiple	
8	All	FCS PMM	
9	True	MCMC Single	
10	True	MCMC Multiple	
11	True	FCS PMM	
12	All	MCMC Single	Hot-Deck
13	All	MCMC Multiple	Hot-Deck
14	All	FCS PMM	Hot-Deck
15	True	MCMC Single	Hot-Deck
16	True	MCMC Multiple	Hot-Deck
17	True	FCS PMM	Hot-Deck

Table 3. List of generate data sets

MODELLING TECHNIQUES

MULTILAYER PERCEPTRON

The multilayer perceptron (MLP) is arguably the most commonly used neural network due to its simplistic architecture as compared to other neural network architectures. A graphical representation of the MLP and the normalized radial basis function (NRBF) used in this study is given in Figure 2 (SAS Institute Inc., 2015).

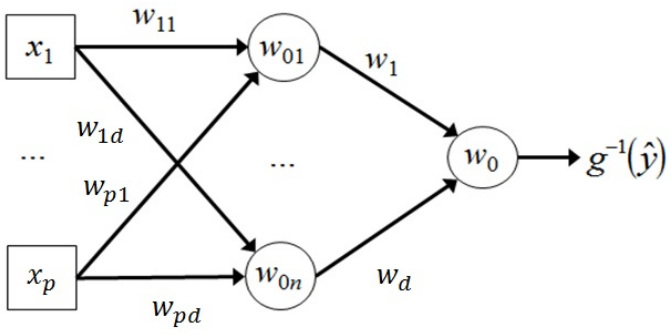

Figure 2. Neural network with single hidden layer

The output function $g^{-1}(\hat{y})$ of the MLP neural network is given in Equation 7 (SAS Institute Inc., 2015):

$$g^{-1}(\hat{y}) = w_0 + \sum_{i=1}^{d} w_i g_i(w_{0i} + \sum_{j=1}^{p} w_{ij} x_j), \qquad \textbf{Equation 7}$$

where w_{ij} is the weight of the input variable x_j. The bias of the hidden unit i in the hidden layer is defined as w_{0i} and w_i the weight of the hidden unit to the output function $g^{-1}(\hat{y})$. The function $g_i(\cdot)$ is defined as the activation function in the neural network (e.g. a sigmoidal function). The corresponding weight of the output unit is defined as w_0.

For the MLP neural network, the exponential function can be used as the target layer activation function to avoid negative numbers being outputted from the model (i.e. forced range from 0 to $+\infty$). Activation functions are also used in the mapping of input variables to the hidden layer to achieve desirable constraints of input variables and/or output from the hidden layer.

A single hidden layer was used in both the architecture of the MLP and NRBF neural network constructed in this study. The activation function of the hidden layer was determined autonomously by the *proc neural* procedure available in SAS. This procedure determines the optimal activation function based on the target and the number of hidden units in the hidden layer. The activation function of the target layer in the neural network was set to the exponential function so as to constrain the estimates of the neural network to be in the interval 0 to $+\infty$.

As mentioned by Principe et al., the number of units to include in the first hidden layer should be set at twice the number of inputs to the hidden layer (Principe, et al., 2000). Even though the number of hidden units could have been determined through some trial-and-error tests, for comparison purposes and due to time constraints the number of units for both the MLP and NRBF neural networks was set at 8.

NORMALIZED RADIAL BASIS FUNCTION

The properties of the NRBF neural network were comparable to those of the MLP neural network to make comparisons in the fit statistics of the neural networks. The main difference between the MLP and the NRBF neural networks is the treatment of the activation functions of the hidden units. In the MLP architecture, no constraint was applied to these activation functions; however, in the NRBF architecture, the sum of the hidden unit activation functions in a specific hidden layer must sum to 1.

This constraint in the NRBF architecture is handled by the $softmax$ function. As mentioned by Schwenker et al, the MLP is more of a 'rule-based' neural network and the NRBF neural network is more case-based (Schwenker, et al., 2001) and this allows for easier translation of the output generated. The formula for the output of the NRBF neural network used in this study is given in Equation 8 (SAS Institute Inc., 2015).

$$g^{-1}(\hat{y}) = w_0 + \sum_{i=1}^{d} w_i softmax\{f.\ln(a_i) - {w_{0i}}^2(\sum_{j=1}^{p}(w_{ij} - x_j)^2)\} \qquad \textbf{Equation 8}$$

The f parameter of the NRBF neural network function given in Equation 8 is determined by the number of inputs to a specific unit. For example, in the hidden layer, of which there are 8 hidden units, there are 4 inputs feeding into any given hidden unit. The parameter a_i is considered as the altitude parameter and is determined by dividing the activation function for a given unit by the sum of the activation functions in that specific layer.

For both the MLP and NRBF neural networks, the number of training iterations was set at 50. This value was selected due to the time constraints of running each neural network on the data sets mentioned previously.

The use of an appropriate optimization technique to determine the weights and bias of the neural network functions was also considered in this study. Efficient techniques help

determine the minimum error of the non-linear target surface space in the most efficient and quickest time possible (i.e. least number of iterations). The resilient back propagation (RPROP) as well as the Levenberg-Marquadt (LM) optimization techniques were investigated in this study for both architectures mentioned.

STOCHASTIC GRADIENT BOOSTED DECISION TREE

Gradient boosted decision trees (GBT) combine smaller decision trees iteratively by constructing a new decision tree at the terminal node of some previous decision tree, with the objective of minimizing some loss function. GBTs have been shown to be more accurate than some more theoretically intensive predictive models (Persson, et al., 2017).

GBT can be thought of as the decision tree alternative to the bagged neural network. GBT like bagged neural networks, iteratively update the decision tree by sequentially fitting a new decision tree to the decision tree of the previous iteration taking into consideration the residuals of the estimates of the previous iteration. This sequential updating of the decision can, however, lend itself to overfitting of the GBT. Therefore, the stochastic gradient boosted decision tree (SGBT) was developed and is often a more preferred modelling technique than the more common GBT (Friedman, 2002).

At each terminal node of each iteration in the SGBT modelling process, a random sample of the data is taken and the next split in the decision tree is determined using this random sample. This characteristic of the SGBT tends to avoid over-fitting as is the case with the normal GBT. Both forms of the GBT use a parameter called the learning rate or shrinkage parameter, which controls the amount of information that the current decision tree 'learns' from the fitted decision tree of the previous steps to obtain a new fitted decision tree.

The learning rate or shrinkage parameter as a value is constrained in the interval 0 to 1 (not including 0), where smaller values for the learning rate tend to lead to longer computational times and larger values lead to overfitting of the decision tree (Dubossarsky, et al., 2016). Dubossarsky et al. (2016) and Sayegh et al. (2016) mention that a learning rate value of 0.1 is a practical value for the learning rate to be set at to avoid over-fitting, although this will lead to longer computational times.

Persson et al. (2017) mention that the differences in the models when the learning rate is less than 0.1 is negligible after 100 iterations. As highlighted by Friedman (2002), shrinkage parameter values less than or equal to 0.1 should also lead to more robust estimates of the model.

The formula used for the updating of the final GBT as well as the final SGBT is given in Equation 9 (Friedman, 2002),

$$F_m(x) = F_{m-1}(x) + v \cdot \gamma_{lm} 1(x \in R_{lm}), \qquad \textbf{Equation 9}$$

where m is the iteration number and v the shrinkage parameter. $F_m(x)$ is the current decision tree constructed by combining the results of the previous decision tree $F_{m-1}(x)$ with the "pseudo-residuals" γ_{lm} (Friedman, 2002, p. 368), for the current disjoint region R_{lm}. The indicator function $1(\,)$ in Equation 9 will take on the value 1 if the variable x falls in the disjoint otherwise the indicator function will take on the value 0.

As with bagging, the choice of number of iterations and sampling size is an important decision to be made when constructing SGBT. The literature does not provide a clear answer to the number of iterations to be used as this value is also dependent on the learning rate value and therefore numerous combinations can be considered. The choice of sampling size,

as with bagging, needs to be large enough so that the sample can be considered as representative but small enough to avoid over-fitting.

MULTIPLE LINEAR REGRESSION

The multiple linear regression modelling procedure constructs models that effectively cater for linear relationships between input variables and a target variable. As such the MLR modelling procedure requires a high correlation between the independent variables and target variable to produce a model with high accuracy (Khademi & Behfarnia, 2016). The formula given in Equation 10 shows this relationship (Atici, 2011):

$$Y = \alpha + \sum_{i=1}^{p} \beta_i x_i + \varepsilon, \qquad \textbf{Equation 10}$$

where α is the intercept of the model and β_i the partial coefficients of the independent/input variables x_i. The index variable i represents a specific independent variable, of which there are a total of p independent variables. The error associated with the model is represented as ε.

Although, the multiple regression linear (MLR) procedure is usually outperformed in terms of model accuracy when the data is of a non-linear nature, there are circumstances where the MLR procedure outperforms (in terms of model accuracy) more advanced modelling techniques that cater for non-linear data (Khademi, et al., 2015) (Khademi, et al., 2017).

In a study conducted by Khademi et al. (2015), which looked at predicting the compressive strength of concrete, it was found that the MLR procedure was more accurate than that of an artificial neural network (ANN) for specific ranges of values of the independent variables. In the case of Khademi, the MLR procedure was found to be more accurate in prediction when the ratio between water and cement was greater than 0.45 (ANN was more accurate for ratios $\leq$ 0.45). The Khademi study highlighted that although the MLR procedure is a more basic modelling approach and is traditionally a linear type model, there are circumstances in which the MLR procedure can produce a model more accurate than more advanced modelling techniques.

MODELS CONSTRUCTED

Model	Properties
Bagged decision tree	• Branch split set to binary • Tree depth set a maximum of 6 • Bagging procedure with 200 iterations applied
Stochastic gradient boosted decision tree	• Branch split set to binary • Tree depth set a maximum of 6 • Shrinkage parameter value of 0.1 used • Sampling rate within the training algorithm set to 60% • Number of iterations set to 200
Multiple linear regression	• Only a bagging procedure with 200 iterations applied to the basic modelling procedure
Neural network	• Multilayer perceptron and Normalized radial basis function architectures investigated • Resilient back propagation and Levenberg-Marquadt optimization techniques used in each of the above mentioned architectures • Single hidden layer with 8 hidden units • Number of iterations for the training set to 50 • Bagging procedure with 200 iterations applied

Table 4. List of models constructed

The various properties of each of the predictive models constructed in this study is given in Table 4. Figure 3 provides a visual depiction of the diagram flow in SAS Enterprise Miner.

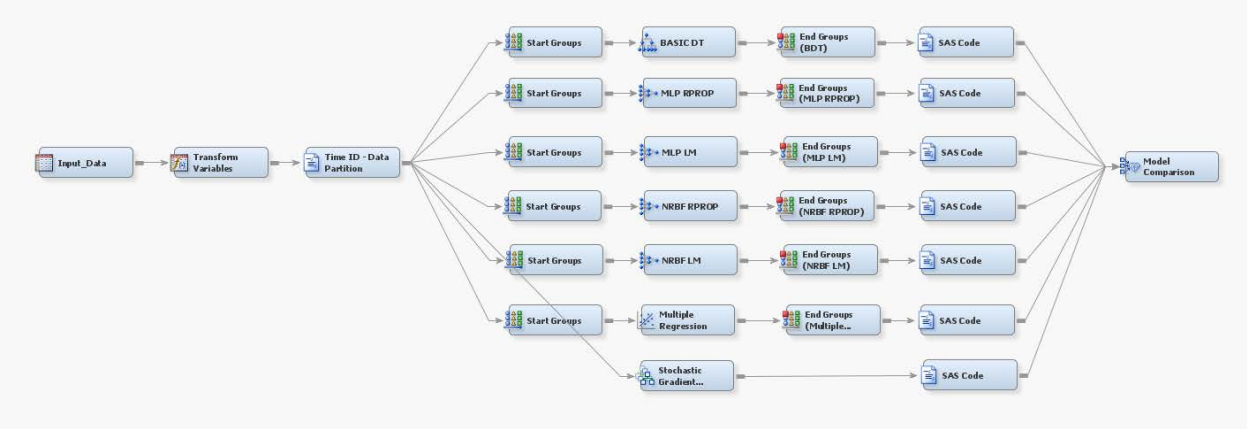

Figure 3. Diagram flow of modelling procedure

As shown in Table 4 and depicted in Figure 3, a total of 7 different predictive models were constructed on the 17 different data sets mentioned in Table 3. This resulted in a total of 119 models that needed to be evaluated for each data set and split type (a total of 476 models overall).

RESULTS

The results presented in Figures 4 to 7 are those from the predictive models producing the best result for the different fit statistics evaluated. As shown in Figure 4, the average squared error (ASE) for both data set split types were relatively consistent for the more advanced imputation method types. Although, the ASE values for *Data set 1 Split 2* did increase for its hot-deck and multiple imputation data sets.

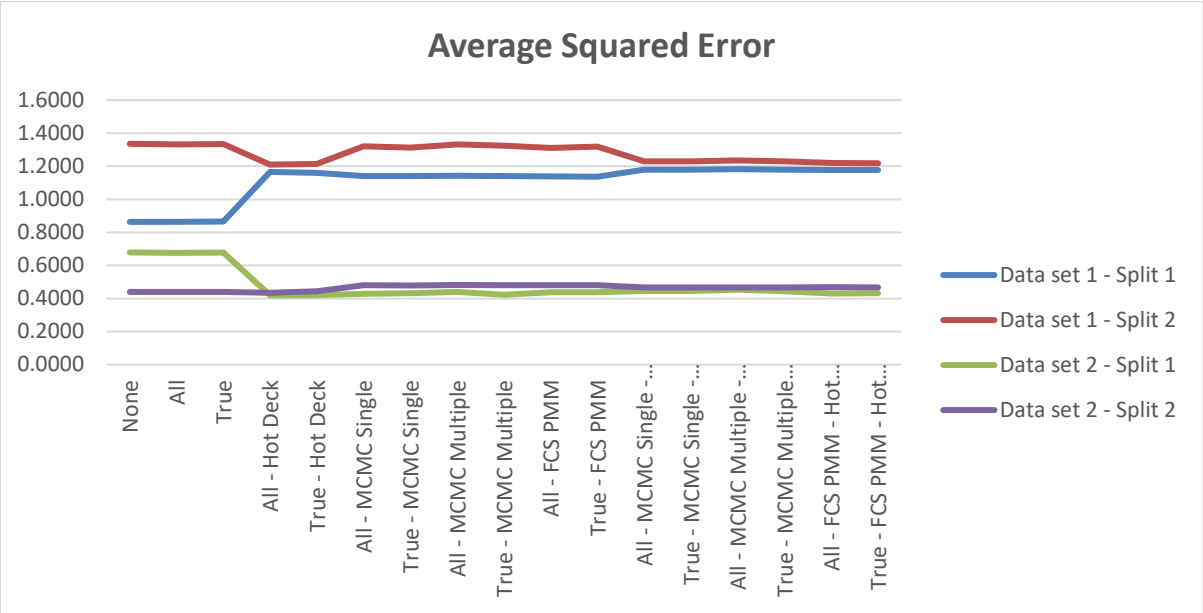

Figure 4. Average squared errors

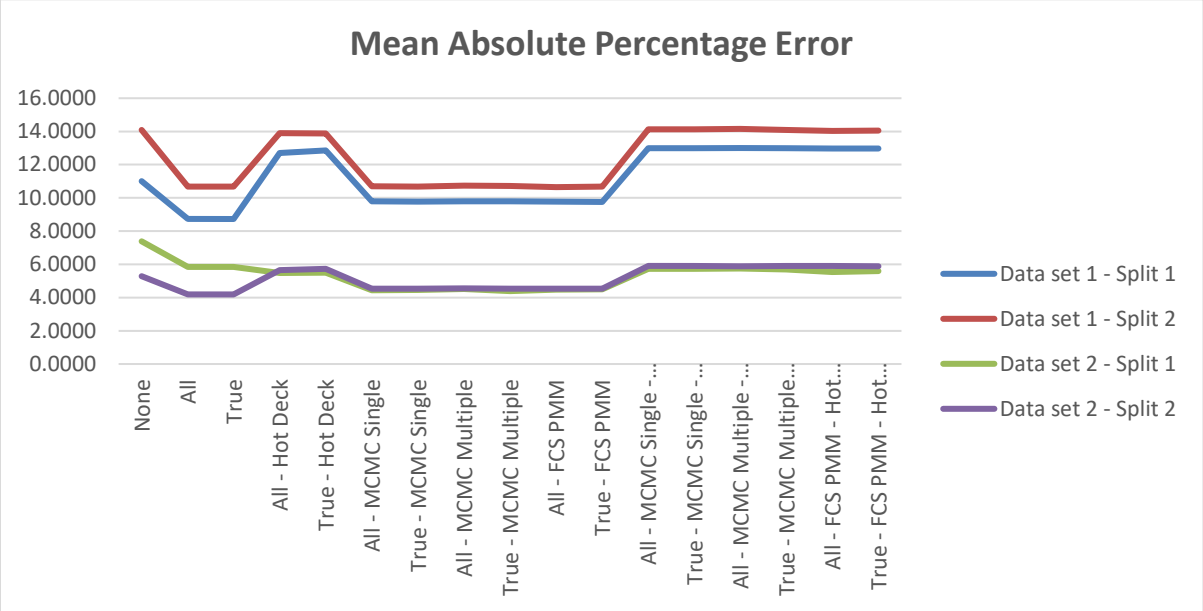

Figure 5. Mean absolute percentage errors

As shown in Figure 5, the original data sets, not employing an advanced imputation method produced slightly better mean absolute percentage error (MAPE) values. The models constructed on the hot-deck imputed data sets (including the final data sets which were a combination of either MCMC or FCS method with hot-deck imputation) tended to be less accurate than others.

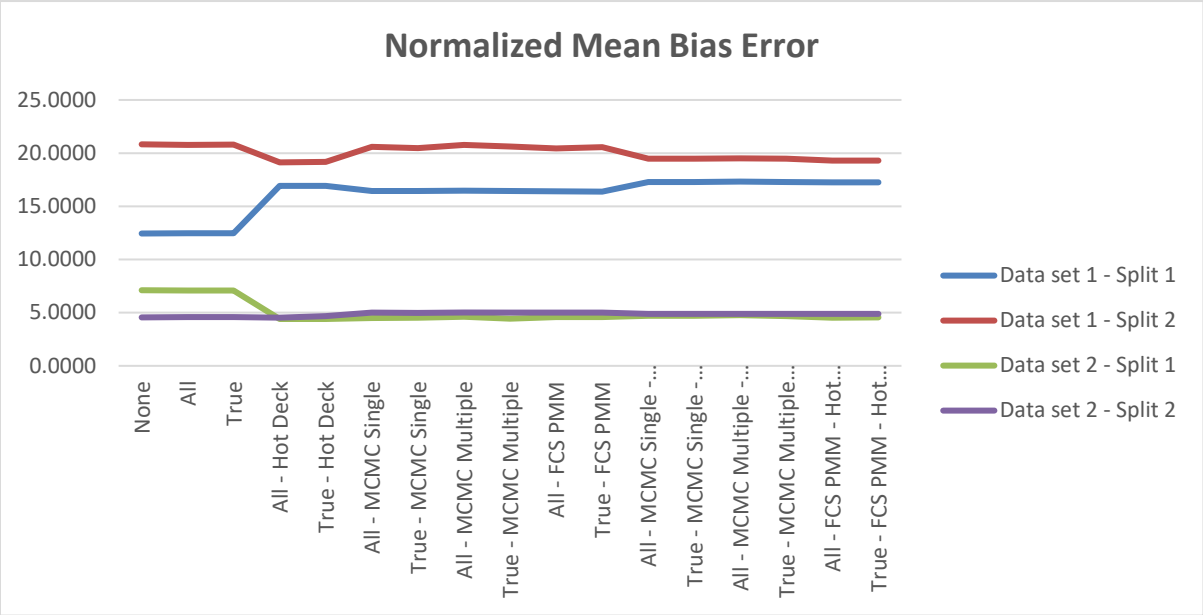

Figure 6. Normalized mean bias errors

As shown in Figure 6, all models constructed tended to produce estimates that were an over prediction. The degree of over prediction varied based on the data set and split type.

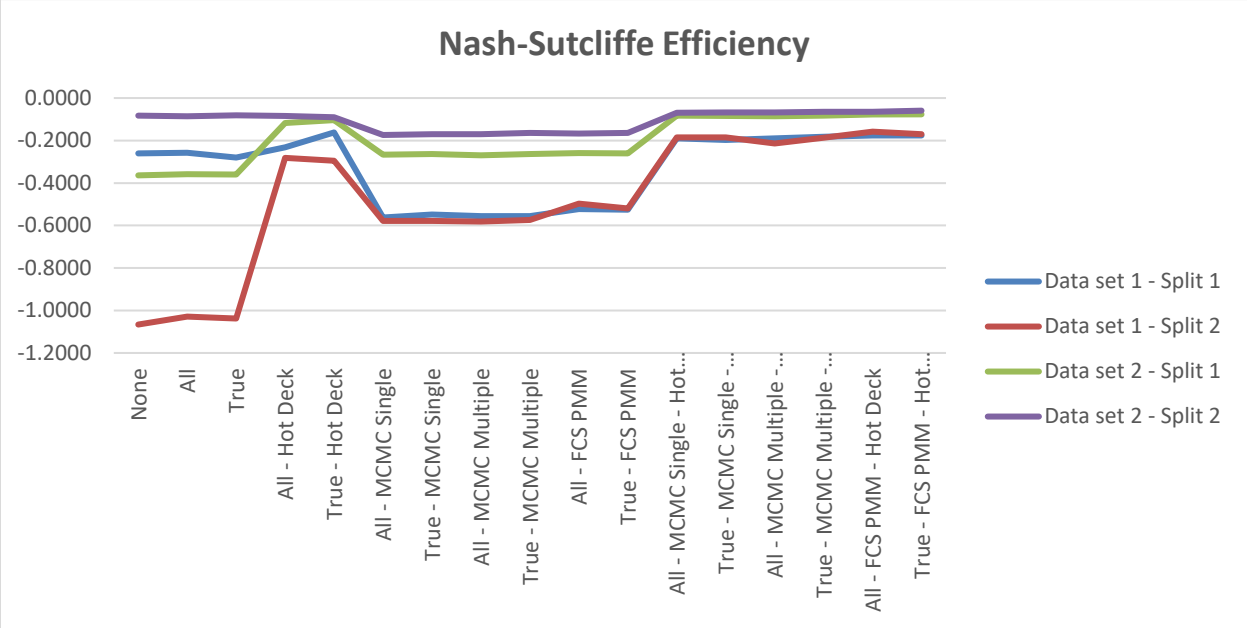

Figure 7. Nash-Sutcliffe Efficiency

As can be seen in Figure 7 all data sets, regardless of the split or missing data imputation method produced low NSE values. An interesting observation in Figure 7 is that for each of

the data set and split types, higher NSE values, although still low, were reported for those data sets employing a hot deck imputation method either before or after having employed one of the other imputation methods. The data sets employing a hot deck imputation method after having employed one of the multiple imputation methods produced larger NSE values for each of the data set and split types.

In order to determine a final best model for each vessel and its relevant catch type, the fit statistic values, specifically those of the test data sets, were examined. A final model was selected by comparing the fit statistics of the 17 different data sets' models of each data set and split type and selecting that model which produced the more accurate fit statistic. The frequency counts of the final selected models for the test data sets for both data sets and each split type are represented in Table 5 and Table 6.

Model	Data set 1 – Split 1	Data set 1 – Split 2	Data set 2 – Split 1	Data set 2 – Split 2
BDT	0	12	11	51
MLP RPROP	0	0	34	0
MLP LM	0	0	8	5
NRBF RPROP	0	56	12	0
NRBF LM	0	0	3	0
MLR	68	0	0	12
SGBT	0	0	0	0

Table 5. Model selection – Test data sets

As can be seen in Table 5, each data set and split type seemed to prefer a unique modelling technique. For example, *Data set 1 Split 1* preferred the bagged MLR modelling procedure and this modelling technique was, surprisingly, the only modelling technique identified, based on the results of the test models' fit statistics.

For *Data set 1 Split 2*, the NRBF RPROP neural network modelling procedure was selected more frequently compared to the other modelling techniques. *Data set 2 Split 1* favoured the MLP RPROP modelling technique over the others while *Data set 2 Split 2* preferred the more basic bagged decision tree. Even though these results, in terms of an overall best model, are not ideal, they show that for a specific data set, different modelling techniques were indeed needed. The final model selected for each data set and split type is summarised in Table 6.

Data set 1 – Split 1	Data set 1 – Split 2	Data set 2 – Split 1	Data set 2 – Split 2
MLR	NRBF RPROP	MLP RPROP	BDT

Table 6. Final model selection based on test data sets

FINAL BEST IMPUTED DATA SET

In order to determine the best overall imputation method, the fit statistics of the test data sets for each of the data sets were compared. As can be seen in Table 7, the selection of a single imputation for each data set was not consistent across the fit statistics of the test data (as was the case for the model selection as well).

Especially in the case of *Data set 1 Split 1*, the more accurate values for the ASE and NMBE fit statistics were unexpectedly from the original data set for these two fit statistics. As highlighted previously, *Data set 1 Split 1* also seemed to prefer the simpler MLR modelling procedure to the more advanced techniques deployed.

	ASE	MAPE	NMBE	NSE
Data set 1 – Split 1	None	True	None	None
Data set 1 – Split 2	All - Hot Deck	All – FCS PMM	All - Hot Deck	All - FCS PMM – Hot Deck
Data set 2 – Split 1	All - Hot Deck	True – MCMC Multiple	True - Hot Deck	True - FCS PMM – Hot Deck
Data set 2 – Split 2	All - Hot Deck	All	All - Hot Deck	None

Table 7. Imputation method selection

All of the other three data set types preferred more advanced imputation methods. As previously mentioned, a consistent imputation method could not be identified for any of the data set types. Because of the nature of the imputation process it is however possible for the results of Table 7 to be pooled in order to determine a final best imputation method and therefore, a final best imputed data set.

As also mentioned previously, imputation methods were applied sequentially during the imputation process in order to achieve a complete data set. This process therefore resulted in complete data sets that were based on a combination of imputation methods.

	Final imputed data set	Final model selected
Data set 1 – Split 1	True	MLR
Data set 1 – Split 2	True – FCS PMM - Hot Deck	NRBF RPROP
Data set 2 – Split 1	All – FCS PMM / MCMC Multiple – Hot Deck	MLP RPROP
Data set 2 – Split 2	All – Hot Deck	BDT

Table 8. Final test data set and model selections

As shown in Table 8, a final best imputation method could be determined for each data set's split type by combining the imputation methods identified in Table 7. As was the case for the modelling techniques, each data set's split type seemed to prefer a unique imputation method process. The selection of this final best imputed data set and combining it with the selection of the best modelling technique allowed for the extraction of imputed values in order to determine what value could be gained based on the imputed missing values.

CONCLUSION

The imputation methods used in this study to impute the missing blocks of data were carried out with the ultimate goal of producing a complete data set with no missing data. Since the original data sets contained the usual missing data patterns (i.e. at least one value observed per observation) and also more importantly, missing blocks of data with no values present, combinations of imputation methods were needed in order to achieve a final complete data set.

The evaluation of the ASE fit statistic showed that all models, for a given data set and split type, produced similar ASE values across data sets and, in some cases, resulted in larger ASE values. The MAPE fit statistic provided similar information to the ASE fit statistic, but as a percentage, making it easier to interpret and compare the accuracy of a given model.

As can be seen in Figure 5, the data sets employing the hot deck imputation method produced less accurate models based on the MAPE fit statistic, which could not be seen in the case of the ASE fit statistic (see Figure 4). Although the hot deck imputation method provided imputations of realistic values (Andridge & Little, 2010), the evaluation of the MAPE fit statistic showed that this imputation method, even when carried out in conjunction with other imputation methods (MCMC and FCS), produced models that were less accurate

than those models based on data sets employing either of the MCMC or FCS imputation methods on their own.

Although the results of the NMBE fit statistic, as with the ASE fit statistic, were similar across data sets for a given data set and split type, it did report positive values for all data sets, which provided some insight into the predictions. As mentioned by Chandwani et al. (2015), the positive values of NMBE showed that the predictions of the models selected were an over-prediction (average NMBE value of 11.50% for all data sets), even though these were the more accurate models selected. This result could also be due to the partitioning ratio followed in this study (50:25:25 data partition used).

The assessment of the NSE fit statistic showed that the data sets employing the hot deck imputation method produced more accurate models, although all NSE values produced were still considered to be quite low. It was also further found that the data sets employing both a multiple and hot deck imputation method produced the larger of the NSE values.

An initial selection of models showed that each data set and split type seemed to prefer a specific modelling technique. This was also evident in the final model selection based on the fit statistics of the test data. As shown in Table 5, the data of each of the four different data and split types preferred a specific modelling technique.

A surprising result in the final model selection process was the selection of two simpler modelling techniques for the four final models selected. The selection of the bagged MLR and BDT modelling techniques was a surprising result as it was expected that either of the neural network modelling techniques or the SGBT technique would be selected as a final model. The selection of these models could be due to the fact that these simpler modelling techniques were coupled with a bagging technique (although the neural networks were also bagged and the SGBT technique employs a boosting technique).

Based on the selection of final models, final imputation methods were determined for each data set and split type. Since the selection of an imputation method was not consistent across all data sets as shown in Table 7, results were pooled in order to achieve a final best imputed data set. As shown in Table 8, each data set and split type preferred not only a specific modelling technique but also a specific imputation method to impute its missing data.

REFERENCES

Andridge, R. R. & Little, R. J., 2010. A Review of Hot Deck Imputation for Survey Non-repsonse. *International Statistical Review,* 78(1), pp. 40-64.

Atici, U., 2011. Prediction of the strength of mineral admixture concrete using multivariable regression analysis and an artificial neural network.. *Expert Systems and Applications,* Volume 38, pp. 9609-9618.

Bethlehem, J., 2009. *Applied Survey Methods: A Statistical Perspective.* Hoboken, New Jersey: John Wiley and Sons, Inc.

Chandwani, V., Agrawal, V. & Nagar, R., 2015. Modelling slump of ready mix concrete using genetic algorithms assisted training of Artificial Neural Networks. *Expert Systems with Applications,* Volume 42, pp. 885-893.

Dubossarsky, E., Friedman, J., Ormerod, J. & Wand, M., 2016. Wavelet-based gradient boosting. *Statistical Computing,* Volume 26, pp. 93-105.

Friedman, J. H., 2002. Stochastic gradient boosting. *Computational Statistics and Data Analysis,* Volume 38, pp. 367-378.

Fuller, W. A., 2009. *Sampling Statistics.* Hoboken, New Jersey: John Wiley and Sons Inc.

Heitjan, F. & Little, R. J. A., 1991. Multiple Imputation for the Fatal Accident Reporting System. *Journal of the Royal Statistical Society, Series C,* Volume 40, pp. 13-29.

Khademi, F., Akbari, M. & Jamal, S. M., 2015. Prediction of Compressive Strength of Concrete by Data-Driven Models. *Journal on Civil Engineering,* 5(2), pp. 16-23.

Khademi, F., Akbari, M., Jamal, S. M. & Nikoo, M., 2017. Multiple linear regression, artificial neural network, and fuzzy logic prediction of 28 days compressive strength of concrete.. *Frontiers of Structural and Civil Engineering,* 11(1), pp. 90-99.

Khademi, F. & Behfarnia, K., 2016. Evaluation of concrete compressive strength using artificial neural network and multiple linear regression models.. *International Journal of Optimization in Civil Engineering,* 6(3), pp. 423-432.

Little, R. J. A. & Rubin, D. B., 2002. *Statistical Anlysis with Missing Data.* 2nd Edition ed. Hoboken, New Jersey: John Wiley and Sons Inc.

Persson, C., Bacher, P., Shiga, T. & Madsen, H., 2017. Multi-site solar power foreasting using gradient boosted regression trees. *Solar Energy,* Volume 150, pp. 423-436.

Principe, J. C., Euliano, N. R. & Lefebvre, W. C., 2000. *Neural and Adaptive Systems.* New York: Wiley.

SAS Institute Inc., 2015. *SAS/STAT® 14.1 User's Guide.* Cary, NC: SAS Institute Inc.

Sayegh, A., Tate, J. E. & Ropkins, E., 2016. Understanding of roadside concentrations of NOx are influenced by the background levels, traffic density, and meteorological conditions using Boosted Regression Trees. *Atmospheric Environment,* Volume 127, pp. 163-175.

Schafer, J. L., 1997. *Analysis of Incompete Multivariate Data.* First Edition ed. Boca Raton, FL: Chapman and Hall/CRC.

Schenker, N. & Taylor, J. M. G., 1996. Partially Parametric Techniques for Multiple Imputation. *Computational Statistics and Data Analysis,* Volume 22, pp. 425-446.

Schwenker, F., Kestler, H. A. & Palm, G., 2001. Three learning phases for radial-basis-function networks. *Neural Networks,* Volume 14, pp. 439-458.

Van Buuren, S., 2007. Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical Methods in Medical Research,* Volume 16, pp. 219-242.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Humphrey Brydon
Statistics and Population Studies Department
University of the Western Cape
+27 21 959 3023
hbrydon@uwc.ac.za

Rénette Blignaut
Statistics and Population Studies Department
University of the Western Cape
+27 21 959 3034
rblignaut@uwc.ac.za