Flexible and Robust Treatments for Missing Traffic Sensor Data

Kristian Henrickson

A thesis
submitted in partial fulfillment of the
requirements for the degree of
Master of Science in Civil Engineering

University of Washington

2014

Committee:
Yinhai Wang
Cynthia Chen

Program Authorized to Offer Degree:
Department of Civil and Environmental Engineering

University of Washington

Abstract

Flexible and Robust Treatments for Missing Traffic Sensor Data

Kristian Henrickson

Chairs of the Supervisory Committee:

Professor Yinhai Wang

Department of Civil and Environmental Engineering

The focus of the work contained in this thesis is missing data treatments in traffic loop detector datasets. This work is motivated by the need to improve data quality and coverage for performance reporting and system management decisions. Missing data, whether due to hardware malfunction or error detection and removal, is a critical concern in loop detector data quality control in Washington State and elsewhere, and can quickly become the controlling factor in overall data quality as the rate of missingness increases.

First, the various causal factors and resulting patterns of missingness in loop detector datasets are discussed with respect to the assumptions underlying common missing data treatments. Next, two multiple imputation methodologies are introduced for loop detector data, which have seen use in a number of fields but have not yet been applied to traffic data. These methods are able to take advantage of the various spatial correlation structures present in volume and speed data, and can produce reliable imputation even under high rates of missingness and missing entire days and months. The proposed imputation algorithms are demonstrated in different locations, time periods, and missing data patterns, and are shown to be capable of

reliably representing the statistical properties of the true data. Aggregation levels, model structure, and limitations of the proposed methods are discussed, and some guidelines for implementation are presented. The proposed algorithms are designed to be incorporated into a comprehensive quality control process for traffic data, to be implemented as part of the STAR Lab DRIVE Net data analysis, visualization, and dissemination platform.

Contents

List of Figures

List of Tables

ACKNOWLEDGEMENTS

A great deal of thanks to my supervisor, colleagues, friends, and family for their support and guidance throughout my time at the University of Washington. Thanks to my supervisor, Dr. Yinhai Wang, for his guidance and encouragement, as well as for the exciting opportunities he has provided at the STAR Lab. Thanks is due to Dr. Cynthia Chen, who graciously agreed to serve on my committee and has supported my efforts as a student and researcher. I would like to thank the Washington State Department of Transportation, for their financial support of the DRIVE Net project.

Thanks is also due to my fellow students and researchers in the STAR Lab, for being the foundation of the supportive and dynamic learning environment that I enjoy at the University of Washington. I would like to offer special thanks to Nathan Henrickson, for his constant support, input, and unique perspectives. Most importantly, a most heartfelt thanks to my wife, Alicia, for her unwavering support and positive attitude throughout this process.

Chapter 1: Introduction

This work is primarily focused on missing data imputation in transportation applications for the purpose of improving the coverage and accuracy of performance estimation. Loop detector volume and speed data are used in Washington State Department of Transportation (WSDOT) Gray Notebook Performance Reporting, travel time reliability estimation, and many other management, reporting, and research applications. Data quality is a constant concern in loop detector datasets, and substantial resources have been committed to the treatment of missing and erroneous data Wright (Duane & Ishimaru, 2007). Though a number of methods are used to detect and remove suspect data, it is readily apparent that, as the rate of "missingness" increases due to detector malfunction or removal during quality control processing, the way that missing data is dealt with quickly becomes the controlling factor in overall data quality.

While a great deal of research work has been completed on identifying and correcting the various sources of error and imputation of missing values, there remains a substantial gap in terms of a) the relationship between missing data mechanisms/patterns, aggregation levels, and imputation accuracy and b) statistically principled methodologies that deal with missing transportation data in a way that is efficient both in terms of computational complexity and analyst time investment.

This work addresses these needs by first investigating the relationship between aggregation levels, imputation approach, and imputation accuracy. Different missing data patterns are described in relation to the underlying causal factors and the assumptions underlying common imputation methods. Two multiple imputation methodologies are proposed that are

computationally efficient, flexible, and statistically principled. The performance of the proposed imputation methods are investigated under varied missing data patterns, time periods, and locations.

1.1. Outline

The structure of this document is as follows: Chapter 2 provides a literature review and background information on the topic of missing data imputation in traffic sensor datasets. Chapter 3 describes the theory and implementation of the multiple imputation methods that form the foundation for this work. In Chapter 4, the topic of missing data patterns is introduced and discussed with respect to the mechanisms that contribute to missingness in loop detector data. In Chapter 5:, the dataset used in algorithm development and testing is described, and data conversions, aggregation levels, and assumptions are discussed. Chapter 6: describes the imputation methodology, including prepossessing, test case development, and multiple imputation. Chapter 7 presents the results obtained for a number of test cases, as well as analysis and discussion of results. Chapter 8 summarizes the objectives and results, and offers concluding remarks and suggestions for future work.

1.2. Research Objectives

The overarching objective of this thesis is to provide a principled and practical framework for dealing with missing data in traffic sensor networks. Specifically, this work applies proven imputation methods to loop detector data, and demonstrates how the accuracy of these methods is affected by aggregation levels and missing data patterns that result from the various error types that are common in loop detector datasets. Though the detection and communications technologies

employed by loop detector networks are essentially obsolete, performance reporting and management decisions still rely them to a large extent. Further, despite the body of previous work that has been completed on this topic, many public agencies (including WSDOT) are still relying on elementary missing data treatments (Duane & Ishimaru, 2007).

This work is unique in the following respects: first, by applying non-parametric classification and regression tree multiple imputation to freeway volume/occupancy and speed data, the complex time-varying interactions are preserved without requiring time consuming explicit specification. Of particular interest is that the ability to accurately impute volume/occupancy as a proxy for speed is demonstrated, such that one of several single loop speed estimation methods may be used on the resulting complete datasets. Second, a fast and principled semi-parametric predictive mean matching multiple imputation method is applied to freeway volume data. Unlike any other methods identified in literature, the performance of the two proposed algorithms demonstrated under multiple challenging missing data scenarios including missing completely at random, missing days, and missing months. Additionally, the impact of aggregation levels (20-second vs. 5-minute) on imputation accuracy and related factors are investigated. The methodologies presented in this work will be applied to loop detector data as part of the University of Washington DRIVE Net System, an E-Science traffic data visualization, quality control, and analysis platform.

Chapter 2: State of The Art

A number of methods have been developed in recent years to deal with the ubiquitous problem of missingness in traffic sensor datasets. This provides an overview of recent research on the topic of missing data treatments in transportation sensor networks.

Pair-wise Parametric Regression

An iterative pair-wise linear regression process was developed and applied as part of the California Department of Transportation Freeway Performance Measurement System (PeMS). (Chen *et al*. 2001, Chen *et al*. 2003). For every detector of interest, a separate linear regression model is formed for each of its neighbors, both adjacent and up/down stream. Each missing value is then imputed using the median of the linear regression estimates for all neighboring loop detectors, restricted to only those reporting "good" values. By using the median of pair-wise linear regression models instead of a joint distribution model, the issue of missing predictor values is effectively dealt with. The majority of the test data showed > 0.80 correlation between neighboring detectors for both volume and occupancy, which based on Washington State data is rarely observed at the 20-second level in practice. Only a single day of data was used for testing, and the algorithm was shown to compare favorably with linear interpolation. Other researchers have investigated more complex pairwise linear regression models, for example, good results were reported for dual loop detector data using a pair-wise second order models with speed, volume, and occupancy interaction terms (Al-Deek *et al.*, 2004). It was found that the relationship between speed at the detector of interest and speed at upstream/downstream detectors changed with traffic conditions, so a selective median algorithm was developed to avoid using the upstream and downstream detectors when adjacent detector data was available. This method was shown to improve significantly on the pairwise linear model, which performed rather poorly especially in congested conditions. This method is specifically tailored for dual loop detectors, which report speed, occupancy, and volume at each time interval, and cannot be applied to single loop detectors. Additionally, while the selective median solution for time varying effects improved the imputation results for the test scenario, it cannot be assumed generalizable. This illustrates an obvious problem with using parametric

models of increasing complexity to describe the relationship between neighboring detectors, as the relationships vary by geometry, time, and traffic conditions. As a result, models needs to be carefully constructed for each location, which is both time consuming and prone to over specification. The second order pairwise imputation algorithm was only tested on a single detector cabinet, and so generality cannot be assumed. Neither pairwise regression single imputation methods were tested in scenarios with extended time periods of missing data as is often observed in practice (e.g. > 1 day), and it is likely that substantial retooling would be required.

Low Dimensional Models

A number of imputation approaches have been developed which utilize a low dimensional representation of road network detector data in order to take advantage of key spatial and temporal correlation structures. Missing detector records are then modeled as a function of a set of latent predictors in a lower dimensional space, thereby reducing the influence of random noise and risk of overfitting. In Qu *et al.* (2009), a method was developed based on Probabilistic Principle Component Analysis (PPCA), a data mining approach which seeks to represent high dimensional data as a set of principle components or linear combinations of predictors. This method was shown to be robust for volume imputation and it outperformed elementary historical and pairwise cubic regression imputation, especially in cases with high missing rates (>20%). Li *et al*. (2013) developed a Kernel Probabilistic Principle Component model, expanding on the basic PPCA approach to incorporate both spatial and temporal predictors. While these principle component methods were shown to be statistically valid in terms of the underlying data distribution, the accuracy was only reported in terms of RMSE which is informative for model comparison but less so as an objective measure of model performance in different traffic conditions. Asif *et al*. (2013)

developed two imputation methods for large, interconnected road networks using dimension reduction data mining techniques. Specifically, Fixed Point Continuation with approximate single value decomposition and Canonical Polyadic decomposition were applied to project the spatial and temporal relationships between neighboring locations into a lower dimensional space. This approach was shown to perform well on both urban networks and expressway speed data, reporting approximately 6% mean absolute percentage error for missing rates up to 60%. While these data mining approaches to missing data imputation were shown to be both accurate and computationally efficient, they were all developed and tested on 5-minute aggregation intervals are not likely to perform as well on shorter time intervals. In general, 20-second data tends to be more noisy and less amenable to time series imputation compared to 5-minute data. As will be shown, better imputation accuracy can be obtained by first imputing missing data at the smallest available time interval, and performing aggregation on the completed dataset. Were such methods to be employed on 20-second data, computation time might also become problematic.

Time/Space Tensor Models

Tan *et al*. (2014) developed an imputation method using a 4-way tensor model to represent the spatio-temporal correlation structures in 5-minute loop data. By incorporating both temporal and between-detector spatial correlation, excellent results were obtained for both volume and speed at 10% – 60% missing rates. The algorithm was demonstrated on 5-minute loop data and required substantial tuning of the model complexity and parameter set. It is clear that this approach is less applicable to shorter time interval data (i.e. 20 second), where spatial and temporal correlation is diminished due to increased influence of random variation. The importance of dealing with

imputation at the minimum available aggregation level cannot be overstated for a number of reasons, as will be discussed in Section 5.3.

Data Augmentation

Smith *et al.* (2003) developed a Data Augmentation (DA) imputation algorithm for missing Intelligent Transportation Systems (ITS) data imputation. DA is an iterative approach to filling in missing data, which alternates between estimating the model parameters and filling missing values from the resulting posterior distribution until some measure of convergence is reached. As is common in DA-based imputation, Expectation Maximization (EM) is used to estimate starting values, and DA is used to refine the estimates. The results obtained by Smith *et al*. (2003) indicate that, in some scenarios, this method can produce superior imputation estimates in terms of accuracy and bias compared to several elementary methods, including historical average, average of neighboring time periods, and weighted average of neighboring detectors. However, the algorithm was implemented on pre-aggregated 10-minute data which, as described previously, often results in better correlation between neighboring detectors and time periods as well as dramatically decreased noise. While the analysis results indicate very good performance overall, the DA algorithm did not perform remarkably better than average of neighboring time periods or average of neighboring detectors. This indicates that excellent nearly linear temporal and spatial correlation exists in the test data, which is not typically the case for shorter aggregation intervals.

Ni *et al.* (2005) developed a EM/DA multiple imputation methodology for missing ITS data, and applied it to video-based volume data collected on GA 400 near the Atlanta metropolitan area. In a multiple imputation framework, this methodology is similar to that in Smith *et al.* (2003) except that the process is repeated *m* times to create *m* multiple imputed datasets. Typically, the datasets

are then combined through one of several complete data methods, though in this case the datasets were simply averaged to give a final estimate. The methodology was described as a parametric time-space composite approach, but no description was given of the model structure. It seems likely based on the descriptions given that neighboring detector observations were used as predictors, and that no time-lag terms were included. This algorithm was developed and tested on 20-second data with a range of synthetic missing rates and, although the reported imputation accuracy was much lower than that of Smith *et al*. (2003), proved comparable when post-imputation aggregation was applied. Mean absolute percent volume error of 25% - 33% was reported for 20 second observations with missing rates of 10% - 50%, which was substantially reduced by aggregation to 5-minute intervals. The authors suggest that, given the ability to quantify the uncertainty in imputed values in a multiple imputation framework, the AASHTO guidelines barring imputation in traffic detector data should be reconsidered.

Ni & Leonard (2005) developed a DA multiple imputation scheme using a Bayesian network to describe the model structure. Imputations were generated in this case under an ARIMA time series modeling framework, with DA used to iteratively draw Bayesian network and ARIMA parameters for predictions. Like Ni *et al*. (2005), Ni & Leonard (2005) stress the importance of imputation at the 20-second aggregation level, and applied their algorithm to video-based ITS data from the Atlanta area. The algorithm was tested on a synthetically generated random 30% missing pattern, and the results were reported on 5-minute aggregation intervals (with aggregation applied post imputation). Mean absolute percentage errors of approximately 4.3%, 0.8%, and 9.7% were reported for 5-minute count, speed, and density respectively, though only a single station and day were used in testing and validation. Because a time series approach was used, this method is not

applicable when extended time periods of data are missing (i.e. more than a few consecutive 20-second intervals).

Non-parametric Regression

Haworth & Cheng (2012) developed a non-parametric scheme for online missing data imputation based on K-nearest neighbor (KNN). This approach identifies the k historical records most similar to the observation of interest, and takes a weighted average of these values as the imputed value. For each detector reporting missing a missing value, similarity or distance is measured between state vectors containing records from all directly adjacent detectors. Multiple variations of the KNN algorithm as well as a kernel regression approach were developed and tested on 5-minute link travel time data in the London metropolitan area. For the most part, the non-parametric regression methods outperformed an elementary historical average imputation method, though not overwhelmingly so. Chang *et al*. (2012) took a slightly different approach to KNN imputation, defining each state vector as a set of observations from the detector of interest over a block of time. Thus, the similarity between observations can be interpreted as the inverse Euclidian distance between time series vectors. The performance of this method was compared to a seasonal Autoregressive Integrated Moving Average (ARIMA) model, and shown to perform comparably with reduced effort and computation time. As with other strictly time series approaches, this method does not take nearby detector records into account, and so is not useful when longer time periods of data are missing. In addition, this method was demonstrated using speed and volume aggregated over 1-hour intervals, and may not applicable to less aggregate data. While these non-parametric imputation approaches may not be generalizable for raw detector data or shorter time

intervals, they do demonstrate the potential of non-parametric models for fast and flexible imputation.

Chapter 3: Multiple Imputation

Ruben (1987) introduced multiple imputation (MI) as a principled way to deal with non-response in survey and census data. MI is a Monte Carlo technique, in which each missing value is replaced by $m > 1$ replacement values. This results in m completed datasets, which are then analyzed using complete data methods and the results combined to give confidence limits incorporating the uncertainty in the imputed values. Of course, in order for these results to be valid, the models used to generate imputations should be shown to produce valid results such that the imputations reflect both the true distribution of the data and a suitable level of uncertainty (Schafer, 1999). MI has seen increasing use in a variety of fields, due in part to the ever expanding availability of powerful statistical computing tools.

Initial developments in multiple imputation focused on large joint models (e.g. joint normal) of relevant variables. In Ruben (1987), it is recommended that imputation be generated in a Bayesian framework, using an approach similar to the following: Given a data set X which consists of both observed and missing values (X_{obs} and X_{mis} respectively), specify a parametric model for imputing X_{mis} and a prior distribution for the unknown model parameter set. For each of m imputations, simulate independent draws from the conditional distribution of X_{mis} given X_{obs} by Bayes theorem. As the number of variables and size of the dataset increases, or when dealing with mixed data types (i.e. binary, categorical, continuous), this approach is more complicated and may not be appropriate (Azur *et al*., 2011).

3.1. Multiple Imputation by Chained Equations

Multiple Imputation by Chained Equations (MICE) is a somewhat newer approach introduced in Van Buuren & Oudshoorn (1999) in which a predictive model is defined separately for each variable with missing data. First, the missing values are filled with some initial estimate, typically by randomly sampling from the observed values. Then, for each variable with missing values, a regression model is estimated using the observed portion of the other variables as predictors. The missing values are then replaced with random draws from the resulting posterior predictive distribution. This process is repeated for each variable with missing data, using the observed and most recently estimated imputation estimates as predictors. After completing this process for all variables, the cycle is repeated several times updating the imputation estimates as described. This then constitutes a single imputed data set, and the entire process is repeated *m* times to give *m* multiple imputed datasets (Azur *et al.*, 2011).

Under a parametric modeling framework, the MICE procedure can described as follows: Given the following dataset with p incomplete variables $Y = Y_1, Y_2, \dots Y_p$, define the observed portion of Y as $Y^{obs} = Y_1^{obs}, Y_2^{obs}, \dots Y_p^{obs}$ and the missing portion of Y as $Y^{miss} = Y_1^{miss}, Y_2^{miss}, \dots Y_p^{miss}$. Assuming the multivariate distribution of the complete dataset is specified by the vector of unknown parameters θ, the posterior distribution for θ is obtained by sampling in an iterative fashion form the conditional distributions defined for each variable as shown below (from Van Buuren & Oudshoorn, 2011):

$$P(Y_1|Y_{-1}, \theta_1)$$

$$\vdots$$

$$P(Y_p|Y_{-p}, \theta_p)$$

Imputations are drawn iteratively for each incomplete variable in $(i = 1,2, \dots p)$ steps by first simulating a random draw for the parameter, and then simulating random draws for the missing values in the variable of interest. This can be interpreted as a Gibbs sampling approach, which is a simple Marcov Chain Monte Carlo algorithm used when sampling directly from the full multivariate distribution is difficult. The procedure can be shown as follows, as described in van Buuren & Oudshoorn (2011):

$$\theta_1^{*(i)} \sim P(\theta_1|Y_1^{obs}, Y_2^{i-1}, \dots Y_p^{i-1})$$

$$Y_1^{*(i)} \sim P(Y_1|Y_1^{obs}, Y_2^{i-1}, \dots Y_p^{i-1}, \theta_1^{*(i)})$$

$$\vdots$$

$$\theta_p^{*(i)} \sim P(\theta_1|Y_p^{obs}, Y_1^{i-1}, \dots Y_{p-1}^{i-1})$$

$$Y_p^{*(i)} \sim P(Y_p|Y_p^{obs}, Y_2^{i-1}, \dots Y_{p-1}^{i-1}, \theta_p^{*(i)})$$

Note that the parameter vectors obtained in each iteration are specific to the model that has been specified for the variable of interest. That is, the joint distribution of the variables need not be specified or even known for the procedure to produce valid imputation estimates. It is possible in fact to specify a combination of models that cannot be represented by any known joint distribution (Van Buuren & Oudshoorn, 2011). This provides a great deal of flexibility, making it possible to simulate a wide variety of complex joint distributions. It should also be noted that a number of non-parametric imputation models have been incorporated in popular MICE software tools, which add additional flexibility for dealing with complex interaction effects.

The MICE algorithm is particularly well suited for transportation data, in part because the iterative procedure allows missing data in all variables, which is invariably the case in loop detector data. Also, because a separate model is defined for each variable in the dataset, MICE is capable of handling different data types simultaneously including categorical, continuous, or binary. For loop detector data, the MICE framework allows easy model specification and predictor selection for each detector, and without the assumption of multivariate normality it is easier to incorporate non-continuous predictors that may help to explain the missing data patterns. Finally, the MICE approach often takes fewer Gibbs sampler iterations to reach convergence relative to joint distribution approaches, resulting in shorter execution time. This is critically important for transportation datasets, which often consist of millions of observations.

One thing that should be pointed out is the necessity of preserving the structure of the data and any interactions that may be present. Schafer (1999) points out that, if an imputation method does not explicitly take interactions and associations into account, any subsequent analysis that may seek to describe such associations will be less informative. Schafer (1999) suggests that making unreasonable assumptions in imputation stage will lead to loss of information. For this reason, it is important to include all terms that may be used in analysis at the imputation stage.

The mice package in R is among the most popular open source tools for conducting MICE imputation, and has been used extensively in this work. In the following subsections, several different models that have been incorporated into the mice package and other MI software tools are introduced.

3.2. MICE using Linear Regression

Possibly because of its simplicity and interpretability, linear regression has been used frequently in missing traffic data imputation. Here it is assumed that the reader understands the basics of linear regression, and instead the focus is on linear regression in a multiple imputation framework. The process of generating random draws from the posterior predictive distribution of the variable of interest (y) is completed as follows:

1. Fit regression model using only observations corresponding to observed values of y
2. Generate random draws from the joint posterior distribution of the resulting parameters (i.e. σ and β)
3. Generate random draws for the response using the result of step (2)
4. Repeat for each variable with missing values to be imputed, using the observed and most recently imputed values of predictors
5. Cycle through steps 2 through 4 multiple times to achieve some measure of convergence

These steps are discussed in order in the following paragraphs. The basic premise of a linear regression model for normally distributed variables is shown below in Equation 1.

Equation 1: Form of linear regression model

$$p(y|X,\beta) \sim N(\beta X, \sigma^2)$$

Where y is some variable containing missing data which we would like to impute and X is a (complete, containing both observed and most recently imputed values) predictor matrix. For the purpose of this illustration, assume y is constituted of two components, y_{miss} and y_{obs}, corresponding to the missing and observed data respectively. Define $\hat{\beta}$ as the estimated parameter

vector obtained by fitting the linear model using only observations corresponding to y_{obs}, V as the estimated covariance matrix for $\hat{\beta}$, and $\hat{\sigma}$ as the estimated root mean squared error. We can then draw imputation parameters σ^* and β^* in sequence as described in Ruben (1987), first drawing σ^* as shown below in Equation 2.

Equation 2: Random draw for imputation parameter sigma star

$$\sigma^* = \hat{\sigma}\sqrt{(n_{obs} - k)/g}$$

Where

σ^* = imputation parameter (sample root mean squared error)

$\hat{\sigma}$ = estimated root mean squared error from fitted model

g = random draw from χ^2 with $(n_{obs} - k)$ degrees of freedom

n_{obs} = number of observations in y_{obs}

k = number of predictors + 1

With the imputation parameter σ^*, we can then draw the imputation parameter β^* as shown in Equation 3.

Equation 3: Random draw for imputation parameter β^*

$$\beta^* = \hat{\beta} + \frac{\sigma^*}{\hat{\sigma}} u_1 V^{1/2}$$

Where

u_1 = row vector containing k independent random draws from standard normal distribution

$V^{1/2}$ = Cholesky decomposition of V

With imputation parameters defined, imputed values can be obtained as random draws from the posterior predictive distribution of y as shown below in Equation 4:

Equation 4: Random draw for imputed value

$$Y_{i*} = \beta^* X_i + u_{2i}\sigma^* \quad i \in miss$$

Where

Y_{i*} = imputed value for Y_i

u_{2i} = a random draw from the standard normal distribution

By iteratively sampling parameters and imputations, assuming the models have been correctly specified, the true variability of the estimate is represented in the final m datasets. Alternatively, a non-Bayesian linear regression method is available through the mice R package, but is only recommended for very larger samples when the parameter variance will be small (van Buuren and Oudshoorn, 2011).

3.3. MICE Using Predictive Mean Matching

The linear regression method can be adapted to perform Predictive Mean Matching (PMM) by, instead of filling in each y_i as described above, first the k observed values with the closest predicted mean to y_i are identified, and the imputation is generated as a random draw from this set of candidate replacements. This approach insures that the imputed values are within the range of

observed values, and may perform better under certain violations of normality (Horton & Lipsitz, 2001; Little, 1988). PMM is described in Ruben (1987) as follows: After drawing parameter estimates (β^* and σ^*) as described previously, calculate predicted values for all missing values as shown in Equation 5:

Equation 5: Mean prediction for PMM

$$Y_{i*} = \beta^* X_i \qquad i \in miss$$

Next, for all Y_{i*} where $i \in miss$, find the observation $Y_i$from $i \in obs$ that is closest to Y_{i*} and use this as the imputed value.

In the R package "mice", this process is slightly different (Van Buuren & Oudshoorn 2011). First, matching is based on the predicted mean for both the missing and observed values. That is, predictions are made for both missing and observed values, and the observed values are assigned based on the similarity between predicted mean. In addition, instead of selecting the observed value with the closest predicted value to the missing observation of interest, the k > 1 closest observed values are identified and the imputation is made as a random draw from these k observed values. This allows more between-imputation variation, in theory providing a better estimate of the true variance of the imputed values.

3.4. MICE using Classification and Regression Trees

Due to the presence of complex, time varying interaction effects, imputing traffic sensor data using spatial and temporal correlation is challenging. It is clear that imputation can be improved by manually specifying higher order and interaction effects in the imputation model (Al-Deek *et al*., 2004), but this is both time consuming and fraught with the risk of miss-specification. In a number

of previous studies, Classification And Regression Tree (CART) models have shown to result in more reliable imputation relative to parametric main effects models in the presence of interaction effects (Burgette & Reiter, 2010; Doove *et al.,* 2014). In this approach, imputations are drawn from groups of similar observations which are identified using a recursive partitioning algorithm (described below). By grouping observations into relatively homogenous "bins", interaction effects are dealt with automatically.

In fact, early work in recursive partitioning was focused on grouping survey respondents into homogenous bins, such that interactions (both known and unknown) could be removed from consideration (Morgan & Sonquist, 1963). CART imputation can preserve interactions in the imputed values without *a priori* knowledge of the true structure, which is critically important if the data is to be suitable for use in research investigation.

3.4.1. Introduction to CART

Before covering Classification And Regression Trees (CART) in the multiple imputation framework, a general introduction to the topic is in order. The introduction given here is based on the descriptions provided in Hastie *et al*. (2009). CART is a set of non-parametric decision tree-based classification and regression methods. In the general case, the predictor space is split into J high dimensional, non-overlapping rectangular regions, such that the total sum of squares is minimized. The response is taken to be the mode (classification) or mean (regression) of the region (R_j) in which the observation is located. For a given model complexity (defined by the number of regions J), the objective is to find the optimal partitioning such that Equation 6 is minimized (Hastie *et al.*, 2009).

Equation 6: Objective function for CART

$$RSS = \sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}R_j)^2$$

Where

RSS = Residual sum of squares

y_i = Response for observation i

$\hat{y}R_j$ = Mean or mode response for region R_j

J = Number of regions

Unfortunately, it is not computationally tractable to consider every possible partition of the predictor space. Instead, a greedy solution can be constructed using recursive binary splitting. In this approach, splits are defined sequentially for individual predictors such that the incremental decrease in RSS is maximized. For example, for the first split, every possible split point for every predictor is considered and the single split resulting in the greatest decrease in RSS is selected (described in Equation 7). With this split in place, a similar search is conducted on all predictors and split points and a second split selected. As this process is repeated, a tree-like structure emerges in which each terminal node or "leaf" contains an increasingly homogeneous subset of observations. For regression, each split involves selecting the value of j and t that minimizes Equation 7. For prediction, each new observation is assigned to the terminal node corresponding to the values of its predictors, and value of the response is taken to be the mean or mode of the responses in the terminal node for regression and classification respectively.

Equation 7: Residual sum of squares for split

$$RSS_{split} = \sum_{ix_i \in R_1(j,t)} (y_i - \hat{y}R_1)^2 + \sum_{ix_i \in R_2(j,t)} (y_i - \hat{y}R_2)^2$$

Typically, a tree is grown until no region has > *m* observations, and then pruning is used to select the optimal model complexity, defined by the number of leaves. As in any regression/classification approach, model complexity is selected to balance the bias and variance of the model. A larger J will result in lower training error, but may lead to poor transferability and bad performance in testing and prediction. For regression (as opposed to classification), cost complexity pruning is used to select an optimal subtree (T) from the previously constructed full tree (T_0) by minimizing the penalized RSS. The RSS is penalized by the total number of leaves or regions in the tree scaled by the tuning parameter α as shown below in Equation 8. Note that, while this work is oriented toward regression estimation for continuous variables, the process differs for classification only in the methods used for splitting and pruning the completed trees.

Equation 8: Penalized residual sum of squares for cost complexity pruning

$$PRSS = \sum_{k=1}^{|T|} \sum_{x_i \in R_k} (y_i - \hat{y}R_k)^2 + \alpha|T|$$

Where

α = non-negative tuning parameter

$|T|$ = Number of leaves or regions in the tree

The tuning parameter α is selected by pruning over a grid of α values, and selecting the value that minimizes cross validation error.

3.4.2. CART in a MICE Framework

The algorithm implemented in the R mice package is similar to the process described above, though a somewhat simpler method is used to control complexity (Doove *et al.,* 2014). Instead of pruning, the tree is grown by splitting until one of the following occurs: 1) All terminal leaves contain less than *p* members or 2) no further splits will improve RSS by more than a given value (e.g. 0.0001). In this case, no pruning is used. Note that this has the tendency to result in suboptimal or unstable trees, because a poor split (i.e. one that results in a small reduction in RSS) can often be followed by a split that results in comparatively large improvement in RSS. However, complexity parameters are typically set such that the resulting tree is quite complex, largely eliminating this problem. Though a smaller tree may be preferable to avoid over-fitting in a modeling scenario, this is not critical in an imputation framework where over-fitting is less of a concern and a more complex model is desirable to reduce bias (Ruben, 1987).

In a missing data imputation scenario, a tree is constructed using only observations with observed response values. Each terminal node or leaf on the tree will contain a set of response values in the set of Y_j^{obs}. Each observation corresponding to a value in Y_j^{miss} is put into the tree, and assigned to a leaf based on the values of the predictors. The missing value is then filled with a random draw from the response values contained in the assigned leaf. Alternatively, single imputation can be performed by filling the missing value with a similarity-weighted average of the values contained in the assigned leaf.

This concept is illustrated below, where Table 1 contains variables that will be used for prediction as well as the target variable for which inference is to be made. By splitting the dataset predictor-wise, the outcome variable is divided into increasingly homogenous subsets resulting in a tree-like structure as shown in Figure 1. Imputations are made as random draws from the subset of observed outcomes corresponding to the leave assigned to the observation of interest, described analytically in Equation 9.

Table 1: Example dataset for Figure 1

Target	**Var2**	**Var3**	**Var4**	**Var5**
4	5	5	3	6
5	5	4	6	5
6	5	5	4	4
…	…	…	…	…

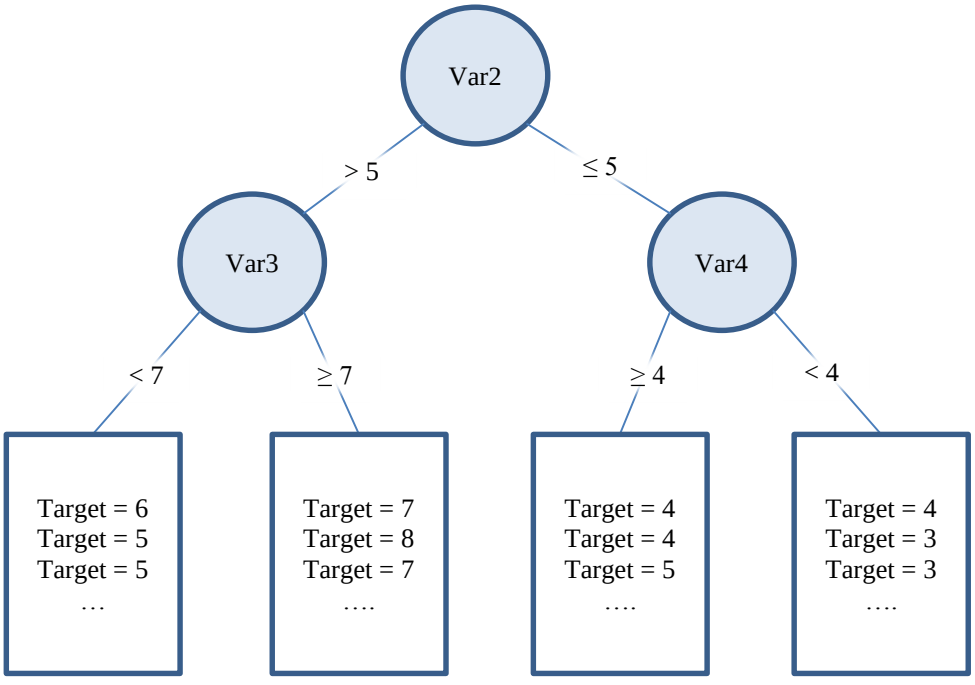

Figure 1: Simple example of cart modeling procedure for MICE

Equation 9: Random draw for CART MI

$$Y_j^{miss} \sim Y_{k \in leaf(j)}^{obs}$$

Where

$leaf(j)$ = set of index values corresponding to the Y^{obs} values in the leaf assigned to Y_j^{miss}

$Y_{k \in leaf(j)}^{obs}$ = Set of observed values contain in the leaf assigned to Y_j^{miss}

Note that this methodology is similar in concept to the KNN regression described in Haworth & Cheng (2012), which identifies similar observations is a historical database and aggregates over similar observations to form the imputation estimate. Aside from the methodology used to identify similar observations, there are two key differences between the algorithm developed in Haworth & Cheng (2012) and the CART approach taken here. First, the CART algorithm is applied iteratively to all columns in the dataset, such that the tree is constructed using most recent replacement values from the previous iteration. In theory this can result in better selection of similar observations when missingness is present in all predictors (as is most often the case in loop detector data). Second, this algorithm is applied in a multiple imputation framework, which produces less bias and makes it possible to estimate the level of confidence in the imputed values.

Finally, it is worth noting that the CART imputation method is more computationally complex than the parametric approaches described previously. Because of this, it should only be applied when it can be shown to significantly improve imputation accuracy.

Chapter 4: Missing Data Patterns

There are a number or reasons why missing data is present in traffic sensor datasets, including various types of hardware malfunction, communications failure, and events related to traffic conditions. Though in some cases the occurrence of "missingness" is predictable, others are completely random or are related to unobserved predictors. Several previous studies have focused on identifying missing data patterns for which the causal mechanisms can be ignored in the imputation process. Most current work considers the occurrence of missing data under a probabilistic framework, with the pattern described by a statistical distribution (Ruben, 1976). The mechanism driving the missing data pattern is assumed to be ignorable if data is Missing At Random (MAR), which is only true when the distribution of missingness is not dependent on the unobserved or missing values. That is, if the dataset defined as X is constituted of both observed and unobserved components (X_{obs} and X_{mis} respectively) the probability that a value is missing depends only on X_{obs} as shown in Equation 10 (Ruben, 1976; Schafer & Graham, 2002):

Equation 10: Probability of Missingness Under MAR

$$\Pr(missing|X) = \Pr(missing|X_{obs})$$

Data is described as Missing Completely At Random (MCAR) when the occurrence of missingness is independent of both observed and unobserved values, as shown in Equation 11. This is considered a special case of MAR.

Equation 11: Probability of Missingness under MCAR

$$\Pr(missing|X) = \Pr(missing)$$

If both Equation 10 and Equation 11 are violated, the data is considered to be Missing Not At Random (MNAR), which means that the missing data mechanism is not ignorable. When the mechanism or distribution of missingness is not known, it is typically assumed to be MAR to simplify the imputation process. While this is often violated, the assumption is made more plausible by including additional predictors that may help to describe the distribution of missingness. Note that these definitions seem to allow substantial blocks of time to be missing from a time series dataset without violating the MAR assumption, because there is no causal relationship between the missing data pattern and the unobserved values. However, it should be stressed that Equation 10 and Equation 11 do not describe a causal relationship (Ruben, 1976). Thus, if blocks of time are missing from a time series, this violation of the MAR assumption can only be arguably ignorable if the missing data follow a distribution identical to that of the observed data. Because the missing data is not observed, this cannot be assumed to be true, and so some bias will likely be introduced.

The majority of loop detectors in Washington State do not always produce useable data. Likewise, few detectors consistently fail to produce useable data. Instead, missing and erroneous values are reported intermittently with useable data, making it necessary to employ routines to identify and remove erroneous data and impute missing values. In part because a significant percentage of the values defined as missing have in fact been removed by error detection procedures, the MAR assumption seems hard to justify. Ruben (1987) describes the bias introduced by non-ignorable non-response as a systematic difference between observed and missing values, even given identical (observed) predictor values. In traffic data, this can be interpreted as a missingness pattern that is a function of the relationship between traffic levels on neighboring lanes. That is, the relationship between the predictors (nearby detector observations)

and the response (observations at the detector of interest) changes with the probability of response. This would likely be the case when an entire month of data is missing, due to seasonal variations in traffic patterns, including those related to weather. However, though some of the causes of missingness are related to unobserved predictors, there is often no reason to expect the missing values will be systematically different from the observed ones. What follows is a brief discussion of the various mechanisms that contribute to missing and erroneous data, and the patterns of missingness that result. Specifically, the focus is on identifying the error patterns that would result in a non-ignorable non-response.

4.1. Segmentation Error

Segmentation is caused when a vehicle detection occurs at the divide between two subsequent time intervals. When this occurs, the vehicle is counted during one interval, but the presence or occupancy is divided between the time periods. This will result in an unrealistically small occupancy, and a very high speed being computed from the observation. In reality, the true occupancy has little to no relation to the measured occupancy. For example, an observation with a volume/occupancy ratio over 120 will almost certainly be removed as a hardware error. However, the MAR assumption only requires that the "missingness" not be attributable to unobserved predictors or the missing values themselves. In the volume/occupancy example, the value is only removed because it is likely the result of segmentation error. The estimated speed, then, is an artifact from a random event (i.e. a vehicle crossing the detector at a particular time) and can be safely removed without violating the MAR assumption. While it is true that the ability to detect such errors is somewhat dependent on traffic conditions, this mechanism can be described in part by neighboring detector observations. Thus, the sensitivity of imputation accuracy to this type or

missingness pattern can be estimated by creating a synthetic error pattern in which the probability of removal is proportional to the true measured value. For all cases, it is important to set threshold values such that only those which are truly erroneous are removed (i.e. retain all values that can be considered physically possible).

4.2. Crosstalk

Cross talk occurs when two neighboring detectors interact, usually as a result of interference or short circuiting between the cables. Cross talk will usually result in very short and intermittent occupancy values, which can occur even when no vehicles cross over the detector of interest. Again, removing very low occupancy values based on value thresholds has nothing to do with the actual occupancy at the location of the detector. Instead, as in the case of segmentation, the true values are unobserved. However, the MAR assumption is in this case questionable, because there is some recurring hardware issue that is causing the values to be removed. The MAR assumption can be made more feasible by including weather (i.e. precipitation) in the predictor set, as it could be argued that increased moisture on the roadway could increase the likely hood of this type of error. Similar to segmentation error, crosstalk is often easier to detect in low traffic conditions, which increases the likelihood of removal. Again, the MAR assumption can be made more plausible by including a sufficient number of nearby detectors in the predictor set. Depending on the rate of occurrence, detectors prone to crosstalk may be removed entirely from the dataset. For those retained, cross talk should most often have exceedingly rare occurrence and can be assumed MAR if an adequate predictor set is used.

4.3. Stuck On or Off

If a detector is stuck on or off, the result will be a time interval (>> 20 seconds) during which a volume and occupancy do not change. Typically, this error type is detected by setting a daily entropy threshold (Chen *et al.,* 2001), under which a full day of data is removed. In this case, the data is not MAR, as a block of sequential observations is removed from the dataset. For this reason, the sensitivity of the employed imputation algorithm to such missing patterns should be investigated. Intuitively, it can be assumed that the accuracy of the imputation algorithm under this scenario will depend on the extent to which the missing data are representative of a typical day of operation. In any case, data imputed under such a scenario should be flagged as such, to enable this data to be excluded in subsequent analysis if needed.

4.4. Communications Failure

Communications failures will result in no data being recorded for a time interval. This may occur at the individual detector level or at the cabinet level, and may be caused by a variety of factors (see Rajagopal & Varaiya, 2007). If this occurs for a single time interval, and is not due to some consistent underlying hardware problem, the MAR assumption seems plausible. However, some detector cabinets have a greater tendency for communications failure, which is indicative of underlying hardware issues. For relatively sparse and isolated failures, the MAR assumption appears to be defensible. For extended time periods (i.e. >> 1 consecutive reporting interval or 20-seconds) or in cases with frequent communications loss, data cannot be assumed MAR. Thus, detectors which report a consistently elevated missing data rate should be flagged as suspect (which is typically done regardless of imputation method, see Chen *et al.*, 2001). The sensitivity of resulting imputations to violation of the MAR assumption can really only be investigated,

similar to stuck on/off detectors, in terms of accuracy over extended periods of missing data. All imputed values produced in such a scenario should be flagged as such for possible exclusion in subsequent analysis, depending on the requirements of the analysis.

4.5. Sensitivity and Detector Health Issues

The sensitivity of a detector systematically impacts the measured occupancy. For example, if the sensitivity is too high, vehicles will be detected before they reach the detector and the detection will remain active for a brief time after the vehicle has passed. As a result, an unrealistically high occupancy will be recorded, leading to a lower speed estimate. In this case, an unrealistically high occupancy value may just represent particularly high value instead of a random error. If such a value is removed on the basis of an occupancy threshold, it will be the case that higher occupancy values are removed with greater frequency than lower occupancy values. Thus, it is of critical importance to distinguish this scenario from the random error types in order to avoid violating the MAR assumption. One possible solution is to apply a sensitivity adjustment before error detection is performed, to insure that only truly erroneous occupancy values are removed.

For sensitivity or other detector health reasons, an entire day, week, or month of data is often discarded from the dataset. This results in a non-probability sampling mechanism, as for a block of sampling intervals the probability of inclusion is zero (regardless of the actual un-measured values). In this case, as previously mentioned, the reliability of the imputations depends to a large extent on the extent to which the missing data follows a similar distribution to that of the neighboring time periods. Of course, because the data is not observed, this cannot be assumed to be true and some bias will likely be introduced. Similar to the stuck on/off error type, the imputation accuracy in this scenario can be investigated using longer missing intervals, for

example, 1 or more months. Again, data imputed under this scenario should be flagged as such for possible exclusion in subsequent analysis.

4.6. Other Missing Data Mechanisms

Additional causes of missingness that could violate the MAR assumption include construction activity, weather, and ongoing detector hardware issues. Thus, the algorithm used to detect and eliminate erroneous values must have some mechanism for identifying the error type, in order to make the distinction between random and not random missing patterns. In any case, as noted by Schafer (2010), standard ignorable missing data procedures are superior to ad hoc solutions, as the bias that can be explained by the observed values is removed, which is not true in general for ad hoc procedures. The sensitivity of various imputation methods to violations of the MAR assumption is investigated in the Results section (Chapter 7).

Chapter 5: Data Description

The Washington State Department of Transportation (WSDOT) manages the loop detectors on state highways and interstate freeways within Washington State. The UW STAR Lab downloads and archives a great deal of data for research work using an online FTP site provided by WSDOT. Algorithm development and testing was performed using data from the Northwest region of Washington State, which contains approximately 4200 single or dual loop detectors primarily on Interstates 5, 90 and 405, as well as State Highways 520 and 167. The northwest region is comprised of a variety of urban and rural land use types, and contains the State's largest metropolitan area (Seattle). Specifically, data the Interstate 5 corridor between mileposts 150 and

165 in the year 2012 is used in testing. This corridor includes the Seattle-Tacoma International Airport and the Port of Seattle, and is just south of the Seattle Metropolitan area.

Site A:

Site A is located near milepost 152 on Interstate 5, near the 188th Street off-ramp. At this location, there are four general purpose and one HOV lane in each direction (North and South). The detector used in testing is positioned in the 3rd lane from the right traveling in the northbound direction. This detector was selected based on the quantity of useable data that it produces. Other detectors in this and nearby cabinets have varying rates of missing and erroneous data, only those reporting at least 60% useable data were used for prediction.

Site B:

Site B is located near milepost 162 on Interstate 5 near the South Seattle Industrial district, North of the King County International Airport. At this location, there are four general purpose lanes and one HOV lane in each direction. The detector used in testing is located in the 2nd lane from the right, traveling in the northbound direction. Dynamic speed control was enacted on this section of I-5 in 2010 for safety and efficiency reasons, controlled using active signage mounted on overhead gantries. This part of I5 is typically congested during a large portion of the day, due to its proximity to major interchanges, major airports, and the Seattle Metropolitan area. As in Site A (and most locations in Washington State) other detectors in this and nearby cabinets have varying rates of missing and erroneous data.

The loop data acquired from WSDOT includes a number of data fields depending on detector type. Single loop detectors simply report the volume and occupancy for each time interval,

and so observations include date/time stamp, volume, scan count, and error flags produced by hardware diagnostics. Scan count is the number of scans per time interval during which a vehicle was present over the detector, which can be converted to percent occupancy by dividing the scan count by the number of scans per time interval. For example, at 60 Hz scanning frequency (used in Washington State loop detectors), the percent occupancy can be computed as follows:

$$Percent\ occupancy = \frac{scan\ count}{1200} \times 100\%$$

Dual loop detectors are able to estimate vehicle length and speed by incorporating observations from two closely spaced single loop detectors. Thus, dual loop detector observations include average length, and speed as well as length-binned volume data fields. The WSDOT data also includes detector tables, which contain descriptive information including cabinet name, route, direction, milepost, and unit type (i.e. mainline, HOV, etc.) for each detector.

5.1. Single Loop Speed Estimation

Though single loop detectors are unable to report speed information directly, they are far more common than dual loop detectors in many locations including Washington State. Because they are more broadly deployed and present in far greater numbers, there is a great deal of interest in obtaining reliable speed estimates using single loop detector data (Wang & Nihan, 2003; Coifman & Kim, 2009). The basic principle for single loop speed estimation involves obtaining some estimate for average vehicle length, and using the measured volume and occupancy to compute speed as the average (travel distance)/(travel time) for each time interval. The standard WSDOT single loop speed estimation equation is shown below according to Wang & Nihan (2000) as Equation 12:

Equation 12: WSDOT Single Loop Speed Equation

$$\bar{s}(i) = \frac{N(i)}{T \cdot O(i) \cdot g}$$

Where

i = time interval

$\bar{s}(i)$ = space mean speed for interval i

$N(i)$ = Volume for interval i

T = time length per interval (in hours)

$O(i)$ = Lane occupancy for interval i

g = speed estimation parameter (a function of average effective vehicle length)

The g speed factor used in Equation 12 warrants some clarification, as it is the primary source of uncertainty in this speed estimation equation. This factor is a function of mean effective vehicle length which includes both a) the actual average vehicle length and b) the detection range of the loop detector of interest. Though it is known that the g speed factor varies somewhat between time intervals (Wang & Nihan, 2000), a constant value is typically assumed for simplicity and consistency (Ishimaru & Hallenbeck, 1999). Wang & Nihan (2000) showed that better speed estimation can be obtained by allowing the factor to adapt over time, but this methodology has not been incorporated into WSDOT performance measurement.

In this research, speed imputation is handled as follows: For dual loop detector data, speed imputation is performed separately from volume imputation. For single loop data, volume and volume/occupancy ratio (as a proxy for speed) are computed separately. Alternatively, it would be possible to impute both volume and occupancy for single loop detector data and then compute speed from the resulting imputed values. This was not done for several reasons. First, volume is more variable at the 20-second level, and as a result is less amenable to accurate imputation. Second, by imputing volume and occupancy separately, the relationship between these two values may not be accurately represented, leading to compounding of the model error. More accurate imputation can be obtained for volume/occupancy, which is more predictable at the 20-second level compared to volume and can be imputed with good accuracy. Finally, for most applications of loop data including facility performance measurement and travel time analysis, speed and volume are the primary quantities of interest.

5.2. Data Structure

This work considers discrete regions in the highway network as subsystems, each with useful temporal and spatial correlation structures. Thus, missing data reported by each detector can be imputed using observations from a set of nearby detectors as predictors. The data is queried from the loop detector database as a table in which the rows are time intervals and detectors are represented as columns. For each cabinet of interest (corresponding to $p > 1$ loop detectors), detector columns are added to the table from both the cabinet of interest as well as upstream/downstream cabinets. To illustrate, Figure 2 shows an example detector layout for a divided freeway section. To impute the missing values for each of the detectors corresponding to Cabinet 1, the observations for Cabinets 1, 2, and 3 are included in the query. Note that, for any of

the cabinets, a variety of lane configurations may be present including ramps and HOV lanes. The detector observations for all such configurations would be included in the query.

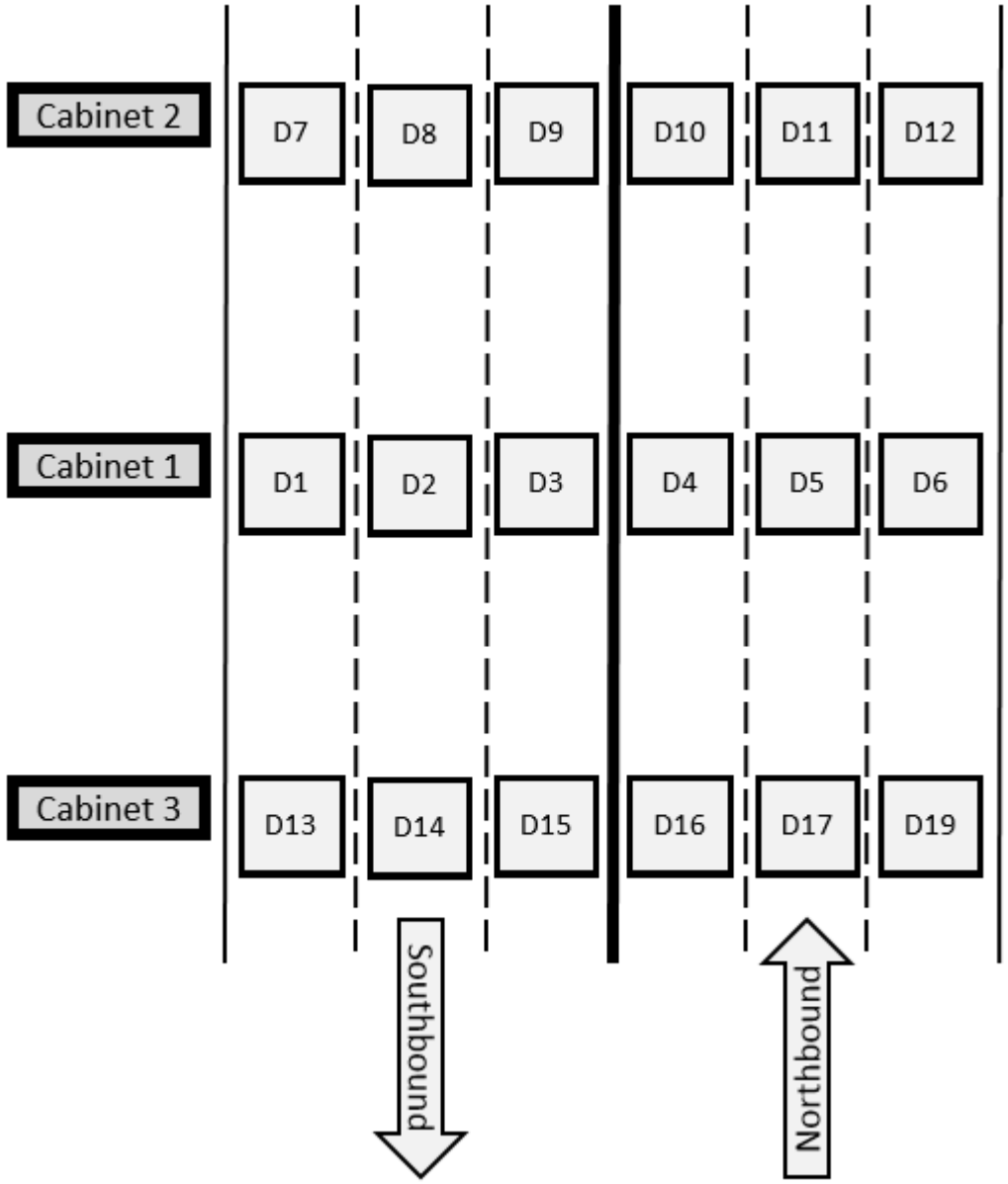

Figure 2: Example Loop Detector Layout for Freeway Section

Table 2 shows the table that would result from the configuration shown in Figure 2. In the multiple imputation framework, all missing values in all columns are imputed. However, only

those for the center cabinet (i.e. Cabinet 1) are stored and used for subsequent analysis. That is, although data from nearby detector cabinets is used for prediction in the imputation process, the imputed values for these nearby cabinets are not stored and the imputation is performed for each cabinet individually.

Table 2: Example Volume Data Structure

Date Time Stamp	**Loop Detectors**				
	D1	**D2**	**D3**	**…**	**D19**
2010-01-01 00:06:40.000	1	2	2	…	3
2010-01-01 00:07:00.000	1	3	1	…	4
2010-01-01 00:07:20.000	2	NULL	2	…	4
…	…	…	…	…	…
2010-01-01 00:11:00.000	4	2	NULL	…	3
…	…	…	…	…	…

5.3. Aggregation Levels

In Washington State, the majority of analysis and performance reporting is conducted on aggregated 5-minute data. For traffic speed and level of service maps, such as those shown in the UW DRIVE Net system, 20-second data are often used. The approach advocated here is to impute missing data at the 20-second level, and then make both the complete 20-second and aggregate 5-minute data available for analysis. To illustrate why the pre-aggregation imputation is preferable, a brief investigation of the relationship between aggregation levels and imputation accuracy is given here.

In conducting this research, it was noted that the majority of methods employing more sophisticated statistical and machine learning methods for imputation were developed and tested on pre-aggregated data at 5-minute, 10-minute, or even 1-hour intervals. In some cases, this was because this time period represented the minimum reporting interval for the available hardware. However, the majority of loop detectors in Washington State report on 20-second intervals. While it is clear that shorter time intervals result in greater random noise and lower spatial and temporal correlation, substantial information is lost in pre-aggregated data. To illustrate, consider a scenario in which a given percentage of all 20-second observations over a month of data are missing. In order to aggregate this into 5-minute averages, it will be necessary to average over many incomplete 5-minute intervals. In most cases, a minimum number of non-missing 20-second observations is established, below which the entire 5-minute period is considered missing and slated for imputation. For example, Li *et al*. (2013) considered a 5-minute interval "complete" for imputation model testing if at least half of the contributing 30-second observations were available. This brings up several problems. First, that many 5-minute intervals will be computed based on incomplete data, which is essentially equivalent to a simplistic nearest-neighbor mean imputation scheme. Second, by removing those 5-minute intervals which do not contain the minimum number of observations required for completeness, we discard information that could be used to inform the imputation procedure. Finally, even though an imputation procedure based on 20-second data might have lower accuracy on a per-observation basis, better accuracy can be achieved by applying a statistically valid imputation procedure to non-aggregate data and then aggregating the complete data.

The key here is applying a methodology capable of representing the underlying distribution of the data, such that the variability on the imputed values is smoothed in aggregation. Otherwise,

the imputation errors can be compounded in the aggregation process, leading to inferior performance. To illustrate, we compare an elementary pairwise linear regression approach to a predictive mean matching multiple imputation algorithm at both the 20-second and 5-minute aggregation levels. Missing values are randomly distributed over two weeks of loop detector volume data, with missing rates between 10 and 60 percent in 10 percent increments. Detector data was obtained from the Washington State Department of Transportation on northbound Interstate 5 in Washington State near the SeaTac Airport, over two weeks in May of 2012 (total of 60,480 20-second observations). For the detector of interest, less than 5.0% of all observations were found missing or erroneous during this time period, though varying missing rates were present on the neighboring detectors used as predictors.

Predictive Mean Matching (PMM) uses a regression model to predict the missing values, and then replaces the regression estimates with the nearest observed value. This avoids any issues with unrealistic predictions, by insuring that all replacement values are from the same value range as the observed data. For a more in depth description of the PMM algorithm, see Section 3.3. PMM is applied in a multiple imputation framework (i.e. Fully Conditional Specification or MICE) in order to better represent the statistical properties of the volume data. For comparison, the pair-wise regression model approach is applied as described in Chen *et al.* (2003), by forming separate regression models describing the detector of interest from each of the adjacent, upstream, and downstream detectors. Imputations are estimated as the median of the predicted values from all nearby detectors which are reporting useable data.

For the imputation at the 5-minute aggregation level, aggregation is performed before any imputation is completed. The simple mean of observed 20-second values is used as the 5-minute estimate, and those with missing rates of at least 40% are marked as missing. For imputation at the

20-second level, imputation is performed first, and aggregation to 5-minute intervals is performed on the completed dataset. Figure 3 shows a comparison of the four imputation approaches, with mean absolute percent error (MAPE) and Root Mean Squared Error (RMSE) estimated at the 5-minute level relative to the complete dataset at varying missing data rates. For both pair-wise regression and PMM, the same set of nearby detectors was used for prediction.

Note in Figure 3 that the pairwise linear imputation method works somewhat better at the 5-minute level, while the PMM imputation approach performs better at the 20-second level. This is as expected because, by applying a multiple regression model and restricting predictions to the range of observed values, the PMM algorithm does a better job of representing the true statistical properties of the 20-second data. That is, the imputed values introduce very little bias in the aggregate 5-minute intervals because any inaccuracies tend to cancel each other out. In the pair-wise regression approach, much of the information in the 20-second data is lost in computing the median of contributing regression estimates. As a result, the inaccuracies present at the 20-second level tend to compound in the aggregation process, resulting in inferior performance even compared to pairwise imputation at the 5-minute level. Aside from any arguments about the superiority of the PMM multiple imputation algorithm, it is clear that imputing at the lowest available aggregation level can only be assumed superior when a statistically principled imputation scheme is applied.

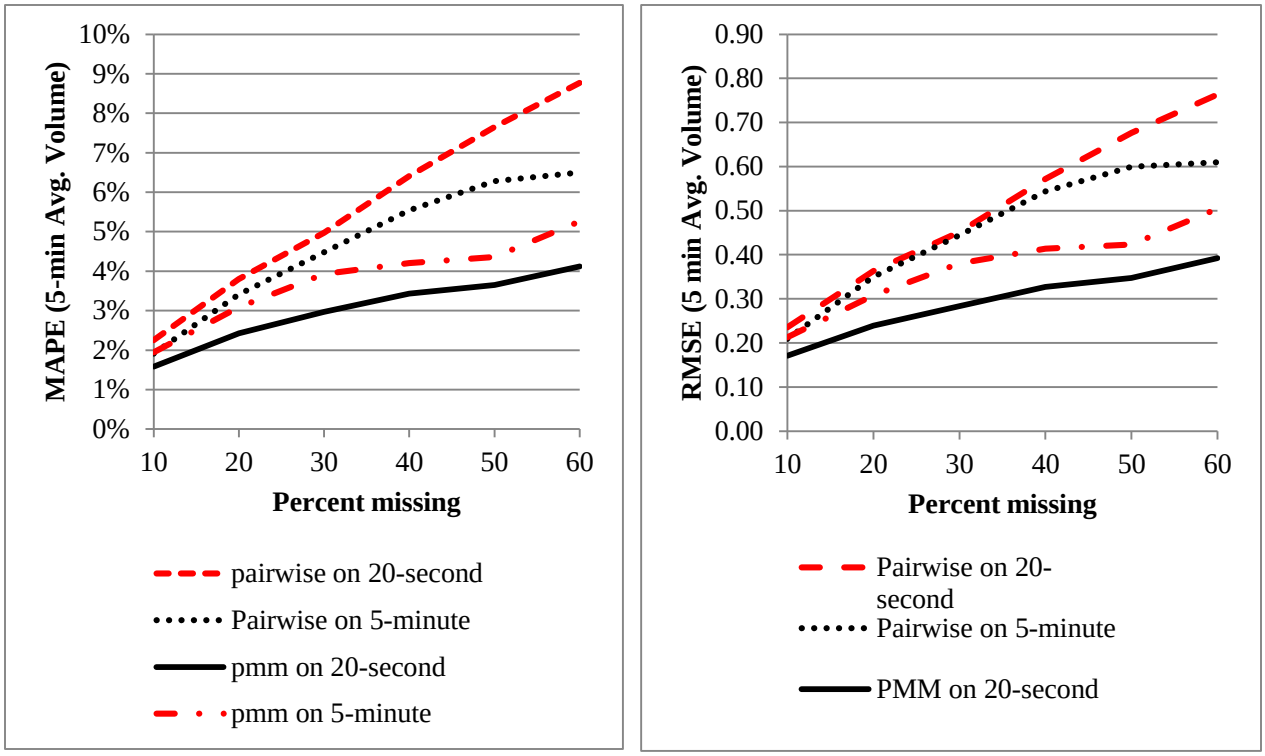

Figure 3: 5-minute MAPE (left) and RMSE (right) for 20-second and 5-minute imputation

Chapter 6: Approach

6.1. Approach Overview

In this research, a classification and regression tree multiple imputation algorithms is developed and implemented using the R statistical computing software. Specifically, several built-in functions in the Multiple Imputations by Chained Equations (mice) package are used in conjunction with several other common R packages to perform the following steps:

- Query 20-second single loop detector data from a Microsoft SQL Server database using RODBC package in R (Ripley & Lapsley, 2013)
- Preprocess data for formatting and consistency, as well as to remove erroneous observations
- Create several different error patterns for testing including missing completely at random, missing days, and missing months
- Predictor selection
- Conduct multiple imputation, generating *m* multiple imputed datasets
- Conduct aggregation and generate performance measures
- Report and compare results

Each of these steps is discussed in the following subsections, starting with a brief discussion of data acquisition and preprocessing.

6.2. Query and Preprocessing

Loop detector data is queried from the STAR Lab databases according to the procedure described in the previous section, using the RODBC R package for data source connectivity (Ripley & Lapsley, 2013). Only data from 6:00AM – 10:00PM are imputed for several reasons. Most importantly, the sparsity of late night and early morning data make it difficult to identify erroneous observations. Also, the late night and early morning time periods are of less interest for analysis because traffic is nearly always in free flow conditions.

The query structure varies based on the missing data pattern present in the cabinet of interest. For random 20-second, 5-minute, and day time interval missing data patterns, data is queried in month-long time blocks. For missing month patterns, data is queried for an entire year in 1-hour time blocks. That is, first the 6:00Am – 7:00AM time period for every day in an entire year is queried and imputed, followed by the 7:00AM – 8:00AM time period, and so on until the 6:00AM – 10:00PM time period has been imputed for an entire year.

The following preprocessing steps are completed on the resulting dataset:

1. Remove zero volume intervals
2. Data type conversions, to insure consistency
3. Rudimentary error detection procedures
    a. Visual inspection and removal of erroneous observations
    b. Volume/occupancy thresholding
4. Remove detectors for which < 40% of observations are non-null

Note that, although an automated error detection procedure is currently available in the STAR Lab DRIVE Net system, more time consuming manual error detection was undertaken to insure that the test data is in good condition. At the implementation stage, the algorithms developed here will be incorporated into a semi-automated comprehensive error detection and imputation framework within DRIVE Net.

6.3. Missing Data Generation

Several missing data patterns are analyzed in this work including completely random, missing days, and missing months. To generate random patterns, a set of observations are set to null using random uniform sampling. For missing month, one or more months are selected and set missing based on the date time stamp. In all cases, the original dataset is retained for imputation accuracy reporting. All datasets contained a certain number of truly missing or erroneous observations, but this uncertainty was minimized by selecting detectors that consistently produce complete and accurate data for algorithm development and testing.

6.4. Predictor Selection

In the MICE framework, model is defined for each detector individually. Thus, it is necessary to select those that have a reasonable level of correlation with the detector of interest from among the available predictors (D1 – D19 in Figure 2). Previous work has suggested that incorporating all available information in the imputation model will result in imputations with "…minimal bias and maximal certainty" (Van Buuren & Oudshoorn, 1999). This is based on the notion of a mindless imputation method, which results in a multiply imputed dataset that can be used in nearly any type of complete data analysis "mindless" of the imputation procedure. That is, because all existing

relationships are preserved in the imputation model, analysis can be performed on the imputed dataset under the assumption that nothing was lost in the imputation procedure. In practice, however, this is not feasible due to a number of issues including correlation between predictors and computational constraints. Instead, a number of previous works (Van Buuren & Oudshoorn, 1999; White *et al.*, 2011) have provided the following guidelines for predictor selection MICE:

- The imputation model should include no more than 15 – 25 variables
- It is important to include all predictors that will be used in the complete data analysis
- Include variables that are known to influence the missing data pattern
- Include variables that explain a significant amount of the variance in the variable of interest
- Do not include variables that themselves have too many missing values

Though the predictor selection process is not as restrictive as what would be completed for data analysis where over fitting is of greater concern, there is a need to avoid including uninformative predictors to control computation time and avoid introducing bias. For each detector, the neighboring detectors with a Pearson correlation coefficient of at least 0.08 are selected as predictors. Detectors with > 60% missing rate or that are not correlated with more than 2 of the other predictors are removed from the predictor list. With the resulting subset of detectors, it was found that better imputation could be achieved by reducing the minimum correlation to 0.03. Thus, after removing the detectors which do uncorrelated with the cabinet of interest, all available detectors Pearson correlation coefficient > 0.03 are used as predictors. This typically results in 10 – 25 predictors for each detector, depending on the lane configuration and detector layout.

Variables describing weather conditions (i.e. freezing, precipitation) and time of day (i.e. peak/off peak) were initially investigated as possible predictors of both the missing values and the

missingness patterns, but were found to consistently bias the results. This is easiest to understand in the non-parametric modeling framework, as the weather and traffic variables tend to control the tree building procedure resulting in a tree structure dominated by time of day and weather. As a result, only neighboring detector observations are used as predictors in the imputation procedure.

6.5. MICE Procedure

MICE imputation was performed using the mice package in R (Van Buuren & Oudshoorn, 2011). Multiple imputation is performed using the mice function, for which the primary inputs are described below:

- A dataframe containing the variables with missing data to be imputed
- The number of imputations to be performed
- A predictor matrix specifying the predictors to be included in the model for each variable
- A sequence of variables indicating he order in which the imputation is to be performed
- A list of imputation methods describing the model form to be used to impute each variable
- The number of iterations to be performed in each imputation

The number of imputations and iterations are both set by default to 5. A range of values were tested, the default values appear to provide a good balance of accuracy, consistency, and computation time. The visitation sequence is set to monotone, which results in imputation being performed in order of increasing missing data rate as suggested in Van Buuren & Oudshoorn (1999). The predictor matrix is developed as described in the previous subsection. Both the PMM and CART methods were used in imputation as described previously.

The results from the mice function include *m* complete datasets, each corresponding to a single imputation. These can then be analyzed individually and the results averaged to give a final result. For example, if the result of the analysis is to be aggregated 5-minute volume data, each complete dataset is aggregated into 5-minute intervals and the results averaged over all imputed datasets.

The default implementation of the R statistical computing package is single threaded, which often results in underutilization of available computing resources depending on the hardware configuration and memory requirements of the computing procedure. To reduce computation time, the doParallel and foreach packages (Revolution Analytics & Weston, 2014a; Revolution Analytics & Weston, 2014b) were used to parallelize the imputation procedure. Instead of attempting to parallelize components internal to the mice function, separate multiple imputations were performed in parallel. For example, if imputation is performed for each month of data, multiple months were imputed in entirety in parallel. With 6 parallel threads, computation time was reduced by approximately a factor of at least 3 from the default implementation (with substantial variation).

6.6. Performance Measures

Several different imputation methods were considered for comparison purposes. While it may be instructive to demonstrate the performance of the proposed methodology in comparison with the more sophisticated models described in recent literature, no such methods are widely used in practice and there is no clear "gold standard" in terms of performance. Instead, the performance of the proposed method is demonstrated in multiple challenging scenarios, allowing the reader to form their own views on the utility of the methods.

Accuracy is reported in terms of the mean error (ME), Mean Absolute Error (MAE) and mean absolute percent error (MAPE) of the imputed values at the 20-second and 5-mintute aggregation levels. ME (Equation 13) gives an indication of bias, while MAE (Equation 14) and MAPE (Equation 15) indicate the per-observation deviation from the true values. Note that, in the multiple imputation case, the imputed values represent the mean of m multiply imputed datasets. For reporting and comparison at the 5-minute aggregation interval, the true and imputed values represent the sum volume or mean volume/occupancy of each 5-minute interval.

Equation 13: Mean Error

$$ME = \frac{1}{n}\sum_{i=1}^{n}(\widehat{y_i} - y_i)$$

Equation 14: Mean absolute error

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\widehat{y_i} - y_i|$$

Equation 15: Mean absolute percent error

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{|\widehat{y_i} - y_i|}{y_i}$$

Where

ME = Mean error

MAE = Mean absolute error

$MAPE$ = Mean absolute percent error

n = number of missing values

$\widehat{y_i}$ = imputation estimate for observation i

y_i = True value for observation i

The variance of the imputed values is also compared with the true variance, as a measure of a model's ability to represent the true statistical properties of the data. The variance is also reported, to give an estimate of how well the imputed values follow the true distribution.

Chapter 7: Results and Discussion

In this section, results are presented for a number of testing scenarios involving different imputed quantities (i.e. volume and volume/occupancy) locations, missing rates, and missing patterns. Only a subset of the results obtained are presented for brevity, with emphasis on those that illustrate a key feature or limitation of the methodology. First, an investigation of recursive partitioning models for traffic data is presented. This investigation is included to illustrate the utility of CART for modeling traffic data, and aid in the selection of model structure for imputing missing data.

7.1. A Brief Investigation of Recursive Partitioning for Modeling Traffic Data

To illustrate the utility of non-parametric modeling of highway speed, a series of models were developed using a month of dual loop speed data on Interstate 5 in Washington. The purpose of this investigation is to illustrate the importance of interaction effects in predicting speed using spatial correlation, and show that non-parametric CART models are capable of representing these

interactions without explicit specification. To do this, each model is used to predict the speed at a single detector positioned in the center northbound lane on I5 near milepost 162. The available predictors include all available dual loop detectors from the cabinet of interest, as well as on the nearest upstream and downstream cabinets. Only detectors situated on northbound lanes were used for prediction.

Data for the month of May, 2013 is first split into training and testing sets, using approximately half of the data for each selected by random sampling using 5 different random seed values. Each model is then fit using the training data, and the performance is reported in terms of mean squared error (MSE) of prediction on the test set. The performance of each model, then, is the average MSE over 5 random seed values.

First, two different main effects multiple linear regression models are applied. The first is a complete main effects model, which includes all available predictors regardless of significance. This results in a total of 12 predictors plus intercept for speed. The second model includes only the predictors found to be significant at $p < 0.1$ based on a t test, which results in a variable number of predictors depending on the seed value.

The second two models are both lasso regression, fit using all of the following: all main effects, all second order terms, and all interaction terms. This results in a total of 91 possible predictors. Lasso regression, introduced in Tibshirani (1996), is similar to linear regression but has a complexity penalty added to the objective function. By tuning the complexity parameter in the objective function, the magnitude of the model coefficients can be controlled to produce an optimal complexity level, balancing bias and variance. From Tibshirani (2011), the regression coefficients ($\beta_j \; for \; j = (1,2,..p)$) are found by minimizing the following expression (Equation 16):

Equation 16: Objective Function for Lasso Regression

$$\sum_{i=1}^{N}(y_i - \sum_{j} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p}|\beta_j|$$

Where

$i = (1,2, \ldots N)$ = data row index

$j = (1,2, \ldots p)$ = predictor index

y_i = response variable value at i

x_{ij} = predictor matrix at row i and column j

λ = tuning parameter for lasso regression

In this expression, increasing λ will shrink the model coefficient estimates. Unlike ridge regression, lasso regression can result in some coefficients being reduced to zero, which can be useful for predictor selection. Typically, cross validation is used to select the optimal model complexity via λ.

The difference between the two models used in this research is in the selection of the λ parameter. For the first model, the lambda parameter is selected to produce the lowest error based on 5-fold cross validation. In the second model, the lambda parameter is selected, also using 5-fold cross validation, as the largest lambda for which the error is within 1 standard error of the minimum. This second model results in a larger lasso penalty, and so produces a somewhat less complex model (i.e. fewer non-zero predictors). These two lasso regression models are included to show the predictive accuracy that can be achieved by careful predictor selection, considering all

interactions and second order terms. Lasso regression is performed using the glmnet package in R (Friedman *et al.,* 2010).

The final model under consideration is based on classification and regression trees as described previously, and is included to demonstrate the utility of this family of models in the presence of interaction effects. In the CART model, the minimum number of observations in any terminal node was set to 5 and the complexity parameter was set to 0.0001. This complexity parameter determines the minimum increase in R^2 (indicating model "goodness of fit") that must result if a split is to be attempted. Only main effects were used as predictors in this case. The CART model was developed using the rpart package in R (Therneau *et al.,* 2014).

Table 3 below shows the testing mean squared error (MSE) for the overall average MSE over 5 random seed values. Testing results are based on slightly over 40,000 observations, or approximately half of the data for the month of May. The number of predictors includes the total number of predictors plus the intercept term if applicable. Note that, although a comparatively larger number of predictors are used in both lasso models, this cannot be directly compared to the complexity that would result in the same number of predictors in a non-penalized linear regression. This is because of the coefficient shrinkage produced by the penalty term.

While it is clear that the lasso regression with interaction effects produced better results that either of the main effects models, the CART model performed comparably in all cases. This indicates that the interaction effects are able to improve the predictive power of the model, and that the CART model is able to capture these interactions to a large extent without the need to specify them in the model. While this example is limited in scope and transferability, it illustrates the difficulty of specifying an optimal parametric model for speed prediction, as well as the utility

of CART models in overcoming this difficulty. In an imputation framework, it is critical to preserve interactions that are present, so that the data is useful for analysis related to interactions.

Table 3: Speed Modeling Results

Model	**MSE**
CV Lasso (1 SE Lambda)	6230.6
CV Lasso (min Lambda)	6161.4
Main Effects (all predictors)	6459.6
Main Effects ($p < 0.1$)	6462.2
CART	6187.6

A similar test was conducted for volume data, using detectors from the same cabinets. In all, 11 detectors produced consistent volume data for the time period in question. Again, the lasso models consider all second order and interaction effects. The optimal complexity parameter for CART was found in this case to be 0.0003.

The volume results are shown in Table 2. In this case, it is clear that the interactions are less significant than for volume data, and as a result the lasso models provide little benefit over the main effects models. The CART model performs the worst overall in this case. This is in keeping with Doove *et al.* (2014), which suggested that recursive partitioning models are sometimes less appropriate in the presence of strong linear main effects. This is clearly the case here, as nearly all main effects were found significant at $p < 0.001$. These results suggest that a main effects model can perform adequately for volume imputation and, at the very least, that there is no benefit to be gained from a more computationally intensive CART model.

Table 4: Volume Modeling Results

Model	MSE
CV Lasso (1 SE Lambda)	3.605
CV Lasso (min Lambda)	3.574
Main Effects (all predictors)	3.593
Main Effects ($p < 0.1$)	3.593
CART	3.744

In summary, the CART model provides substantial performance benefits over the linear main effects model for speed estimation, and can nearly approximate a much more complex parametric model. For volume estimation, the main effects model performs better that the CART model and only slightly worse than the more complex parametric model. Based on these results and an investigation of imputation accuracy, the CART model will be applied for speed imputation and the PMM main effects model will be applied for volume imputation. It may be useful in the future to investigate other scenarios for possible interactions in volume data, and possibly look into other imputation methods that can account for this.

7.2. CART Volume/Occupancy Imputation Results

This subsection describes a set of results obtained for volume/occupancy ratio (VOLOC) using CART multiple imputation. Results are presented at both the 20-second and 5-minute levels, to give both an indication of the raw imputation accuracy as well as the accuracy that can be expected for more aggregate measures. Note that, when the data is missing at random, the 5-minute data includes the non-missing observed values in the aggregation step. For missing day and month

patterns, the 5-minute data includes only imputed values, as all observed values for the time period of interest are set to missing.

Table 5 and Table 6 below show the imputation results for Sites A and B respectively at 40% missing during the months of January, September, and May. Note that the mean error is in all cases near zero at Site A, and slightly higher for Site B. This indicates that little bias is introduced in imputation. The mean absolute error corresponds to less than 1.6 mph and 8.3 mph in all cases at the 5-minute and 20-second level respectively, based on the WSDOT single loop speed calculation method.

Table 5: Volume/Occupancy Imputation Results for Site A at 40% missing

	Aggregation Level	Mean Error	MAPE	MAE	Variance True	Variance Impute
January	20-sec	0.062	16.5%	11.027	378.8	241.1
	5-min	0.010	2.6%	0.334	165.3	150.9
May	20-sec	-0.180	16.8%	10.840	332.0	200.9
	5-minute	-0.011	2.4%	0.322	141.9	130.8
September	20-sec	-0.021	15.5%	10.325	334.3	212.0
	5-minute	-0.021	2.2%	1.690	143.6	134.9

Table 6: Volume/Occupancy Imputation Results for Site B at 40% Missing

	Aggregation Level	Mean Error	MAPE	MAE	Variance True	Variance Impute
January	20-second	-0.213	20.4%	10.68	1129.1	963.9
	5-minute	-0.012	2.9%	2.03	821.0	793.8
May	20-second	-0.433	22.0%	9.35	1337.5	1191.4
	5-minute	-0.278	2.9%	1.80	1084.7	1054.1
September	20-second	-0.288	20.3%	8.35	1427.1	1303.0
	5-minue	-0.193	3.1%	1.79	1168.3	1142.6

Table 7 shows the imputation results for Site B, during the month of January, 2012 with entire days set missing. Note that, while the accuracy of the imputed results are similar to that of

the random missing pattern, the mean error is somewhat higher on average. This added bias can be attributed to the fact that the relationship between neighboring detectors is somewhat different for the missing days as compared to the month as a whole.

Table 7: Volume/Occupancy results for Site A, missing days in January, 2012

	Aggregation Level	Mean Error	MAPE	MAE	Variance True	Variance Impute
Saturday	20-second	-1.251	13.0%	10.66	162.2	62.5
	5-minute	-1.252	3.1%	2.74	12.0	4.7
Wednesday	20-second	1.018	19.5%	12.20	301.0	160.5
	5-minute	0.952	4.1%	3.13	76.0	49.0
Monday	20-second	0.798	16.0%	10.05	459.1	328.1
	5-minute	0.747	4.3%	2.95	257.6	221.5

Table 8 and Table 9 below shows the imputation results for Site A and Site B respectively for entire months missing. As described previously, imputation is performed for each hour in the day using an entire year of data. It should be noted that the variance for the true values at the 20-second level between missing 40% at random (Table 5 and Table 6) and missing months (Table 8 and Table 9). This is because the 20-second variance in Table 5 and Table 6 includes only values set missing, or approximately 40% of the data. Compared to missing 40% at random, the accuracy of imputation at both the 5-minute and 20-second levels is not dramatically increased. However, the bias as defined by the average error is somewhat increased in terms of absolute value. This is to be expected, as the relationship between neighboring detectors likely shifts somewhat between months. On the other hand, the maximum bias is still reasonable, and the imputation accuracy is still quite good.

Table 8: Volume/Occupancy Results for Site A, Entire Months Missing

	Aggregation Level	Average Error	MAPE	MAE	Variance True	Variance Impute
January	20-seconds	1.294	17.4%	11.36	382.0	265.1
	5-minute	1.381	4.4%	3.17	165.3	151.3
May	20-seconds	0.073	17.2%	10.97	347.4	237.4
	5-minute	0.055	3.4%	2.68	141.9	132.9
September	20-seconds	0.911	15.8%	10.56	336.9	243.2
	5-minute	0.910	3.4%	2.66	143.6	140.6

Table 9: Volume/Occupancy Results for Site B, Entire Months Missing

	Aggregation Level	Average Error	MAPE	MAE	Variance True	Variance Impute
January	20-seconds	-0.180	20.0%	10.7127	1132.0	975.8
	5-minute	-0.459	4.6%	3.192905	821.0	773.4
May	20-seconds	-0.285	22.0%	9.367795	1334.0	1195.2
	5-minute	-0.440	4.5%	2.691042	1084.7	1034.5
September	20-seconds	-0.213	22.9%	9.365586	1405.2	1269.0
	5-minute	-0.337	4.5%	2.569411	1168.3	1111.8

Figure 4 below shows the imputation results for the morning period of May, 2012 at 40% missing. The plot contains only values set missing in testing, and so does not represent a continuous time series. It is clear from this plot that the imputed values follow the true values well, even at very low values were traffic is likely in a congested state. The 5-minute aggregate plot shown in Figure 5 demonstrates that nearly perfect agreement is achieved between true and imputed values when aggregation is applied post imputation. Note also that, because this methodology is focused on imputing volume/occupancy instead of speed, there is some added random variation due to fluctuations in vehicle length. That is, there is an added random component in this data that is not amenable to prediction compared to measured speed. Despite this, it is clear that reasonably accurate prediction can be achieved at the 20-second level, and that much of the randomness is smoothed out in aggregation to the 5-minute level.

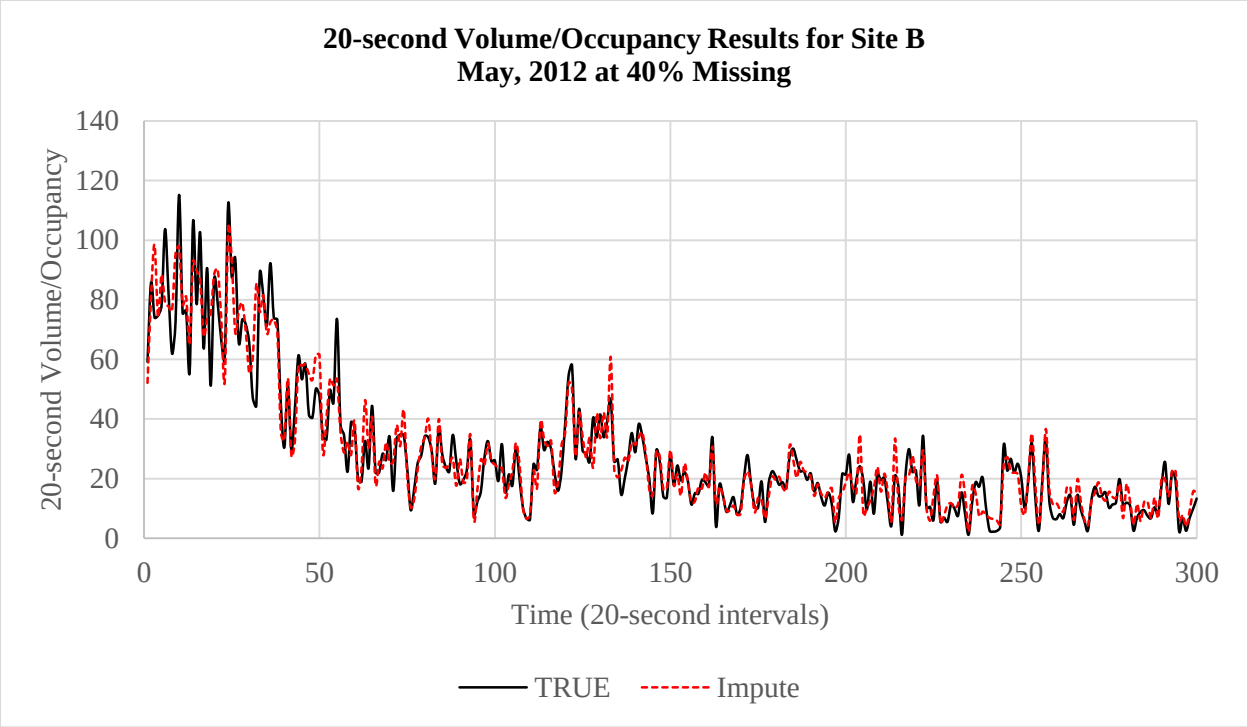

Figure 4: Results at Site B from May 1st, 2012, 20-second level at 40% missing

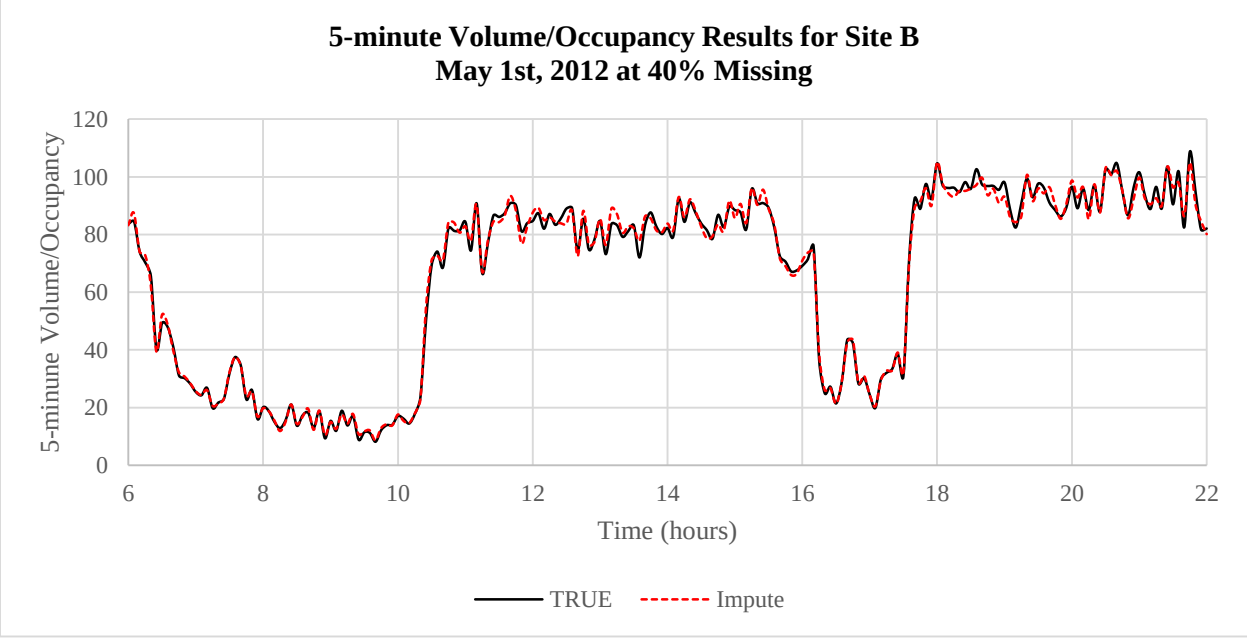

Figure 5: Results at Site B from May 1st, 2012, 5-minute level at 40% missing

Figure 7 and Figure 6 below shows the imputation results for missing Monday, the 28th of January in 2012. The imputed values follow the true values quite well at both aggregation levels, capturing both the general trends and much of the smaller variation. These results, as well as those shown in Table 7, demonstrate that imputing entire days of missing data is quite feasible. In the missing days scenario, 3 out of 31 day were set missing, which equates to a missing rate of approximately 10%. The random missing results suggest that good imputation accuracy can be expected even a substantially larger percentage of days were missing.

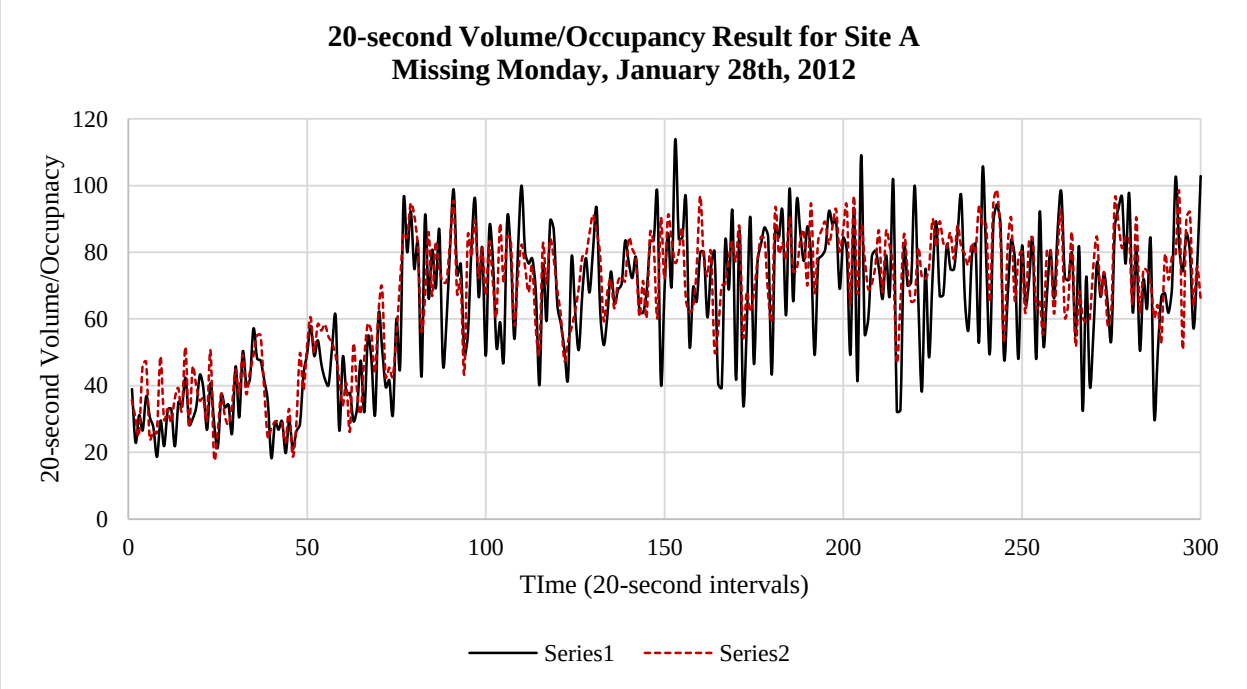

Figure 6: Results at Site A, missing Monday, May 28th, 2012, 20-second level

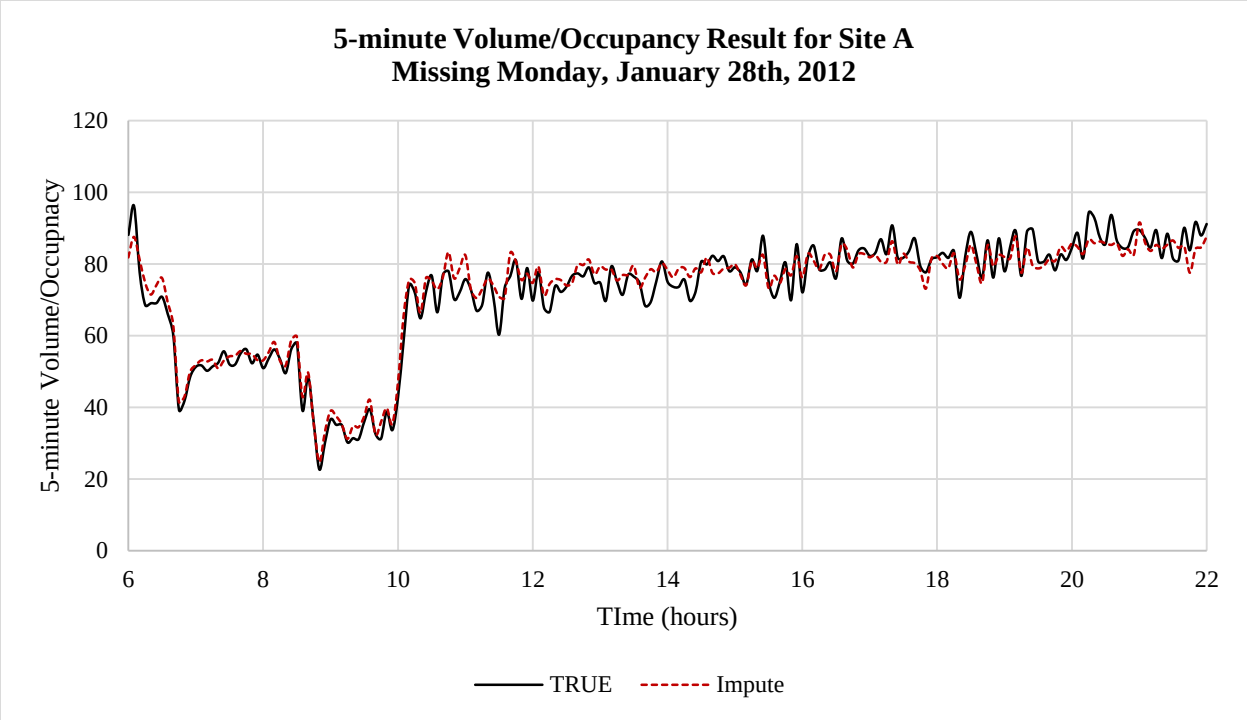

Figure 7: Results at Site A, missing Monday in January, 2012, 5-minute level

Figure 8 shows the histogram of imputed and true values for September, 2012 with 40% missing at random at the 20-second level. From this plot it is clear that, while the two distributions match relatively well at lower speed values, higher speed values tend to be underrepresented while values close to the overall mean are overrepresented. This indicates that more extreme values on the high end tend to be underestimated. This is really a best case scenario for the error distribution, because low to mid-range values are of more interest for analysis, when traffic volumes are higher and congestion is more likely.

Figure 9 below shows the histogram of imputed and true values for September, 2012 at the 5-minute level and 40% missing. It is clear in this case that the two distributions match very well, and that no speed range is significantly under or over represented. This is true even at very low values, which are in general more variable and harder to predict.

Figure 10 and Figure 11 below show histograms for the 20-second and 5-minute volume/occupancy during the month of September, 2012 with the entire month set missing. These results are quite similar to those for 40% missing, all though a greater total number of 20-second observations are present because all values (not just 40%) were set missing. In this case there tends to be less underestimation of higher values, which is likely an artifact of the difference in the imputation approach. Consider that higher values tend to occur during a certain portion of the day (e.g. early morning, night). When the month is set missing, only the historical values for each hour of interest are used to impute that hour. As a result, time periods which tend to have higher speed are represented by the data from same time period during other months.

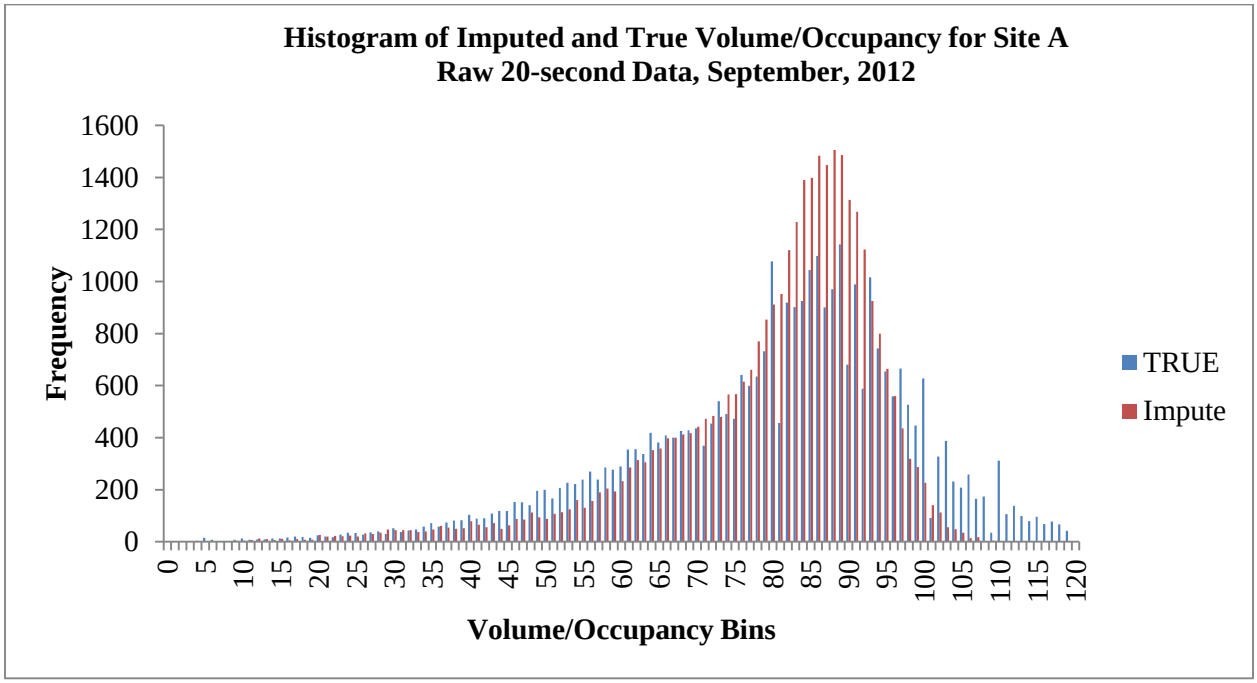

Figure 8: Histogram for Site A from September, 2012, 40% missing, 20-second level

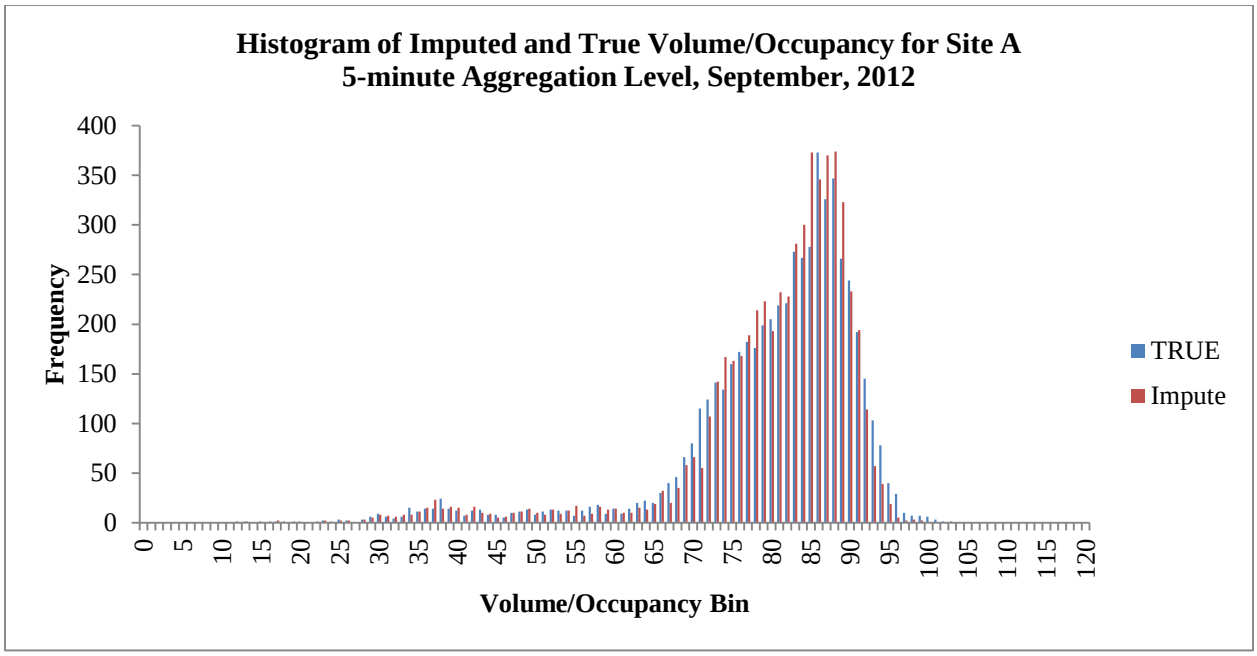

Figure 9: Histogram for Site A from September, 2012, 40% missing, 5-minute level

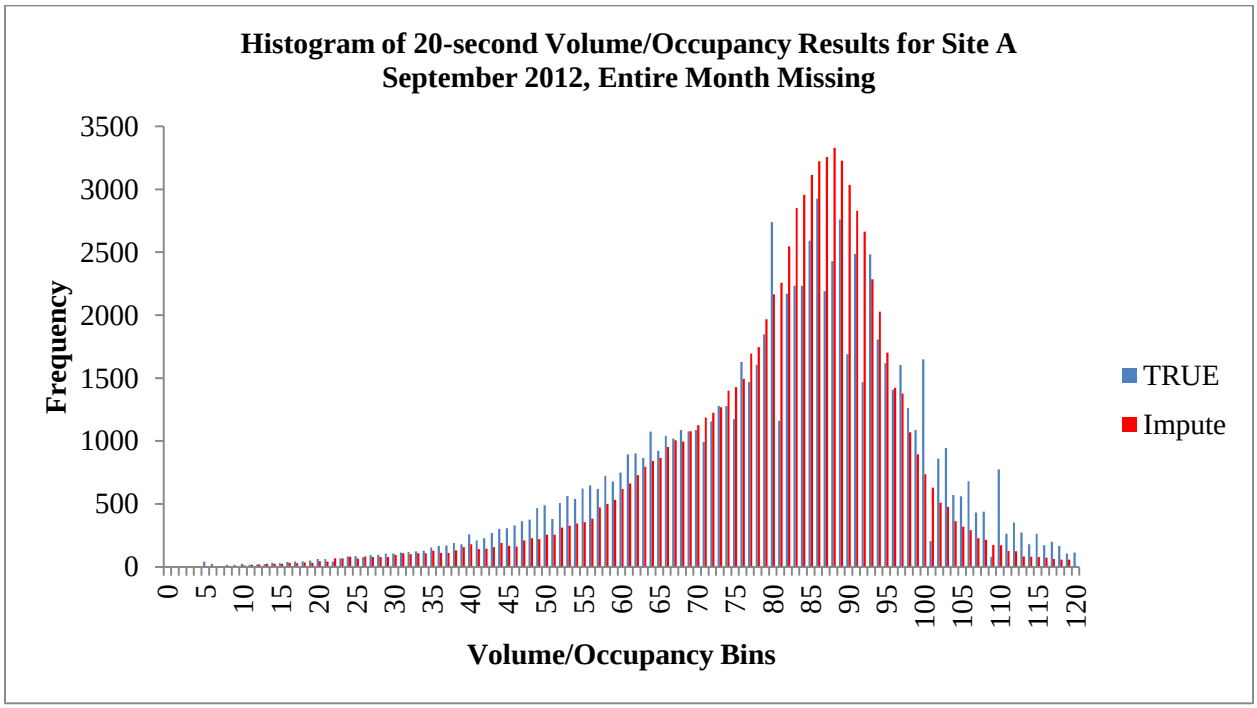

Figure 10: Histogram from Site A, month of September, 2012 missing, 20-second level

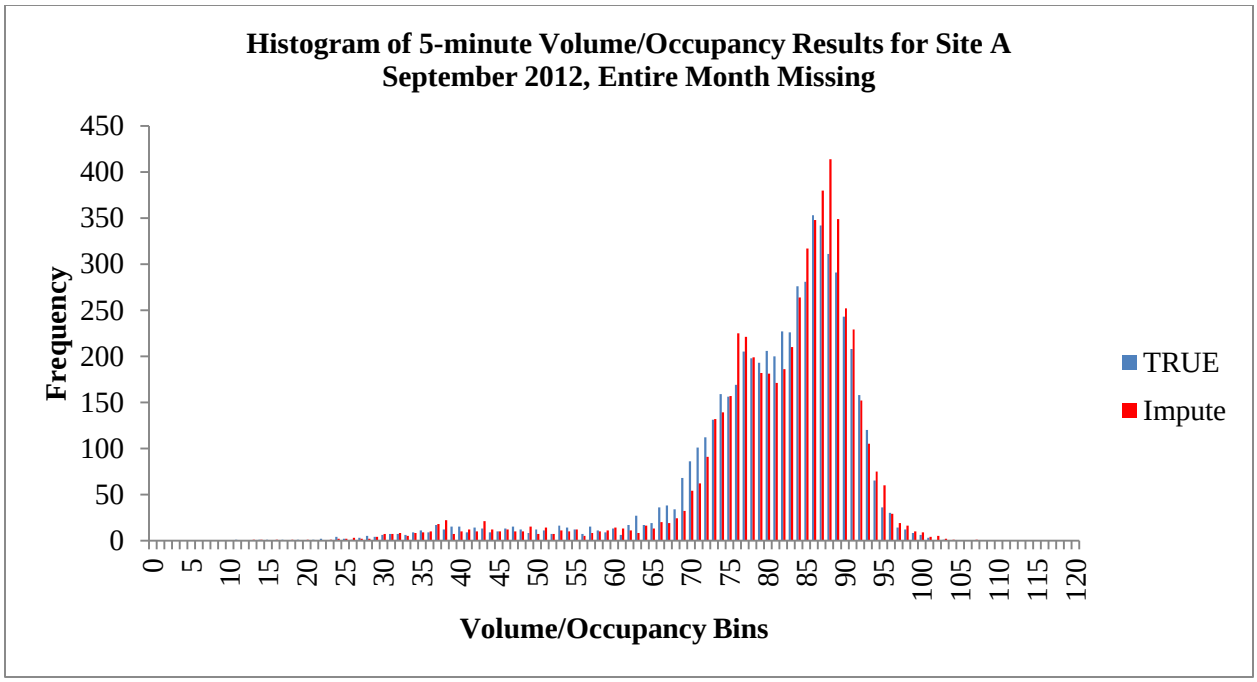

Figure 11: Histogram from Site A, month of September, 2012 missing, 5-minute level

One of the primary benefits of using a multiple imputation approach for missing data lies in the ability to quantify the uncertainty in the resulting analysis caused by the missing data. Ruben (1987) provides guidance for estimating uncertainty in parameters generated using multiply imputed data, but statistical modeling is not really the focus here. Instead, the key concern here is the ability to estimate the extent to which uncertainty in basic traffic measures (i.e. speed, volume) is increased by the presence of missing data. To do this, 95% confidence intervals for 5-minute aggregate speed are estimated using a student-t distribution with $\{m = number\ of\ imputations\} - 1$ degrees of freedom and between imputation variance estimated as described in Ruben 1987. The within-imputation variance is not considered, as each non-missing observation is considered the true value for a discrete time interval, and it is assumed that the true population values (i.e. 5-minute mean) are known for each 5-minute interval when no data is missing. Using this methodology, approximately 94.0% of values fall within the 95% confidence

bounds with 40% missing data in the month of September. This indicates that it is feasible to estimate uncertainty in the imputed values with reasonable accuracy. Figure 12: 95% Confidence Bounds from Site A for September, 2012 below shows the true observed volume/occupancy for a section of September, with 95% confidence bounds for 40% missing.

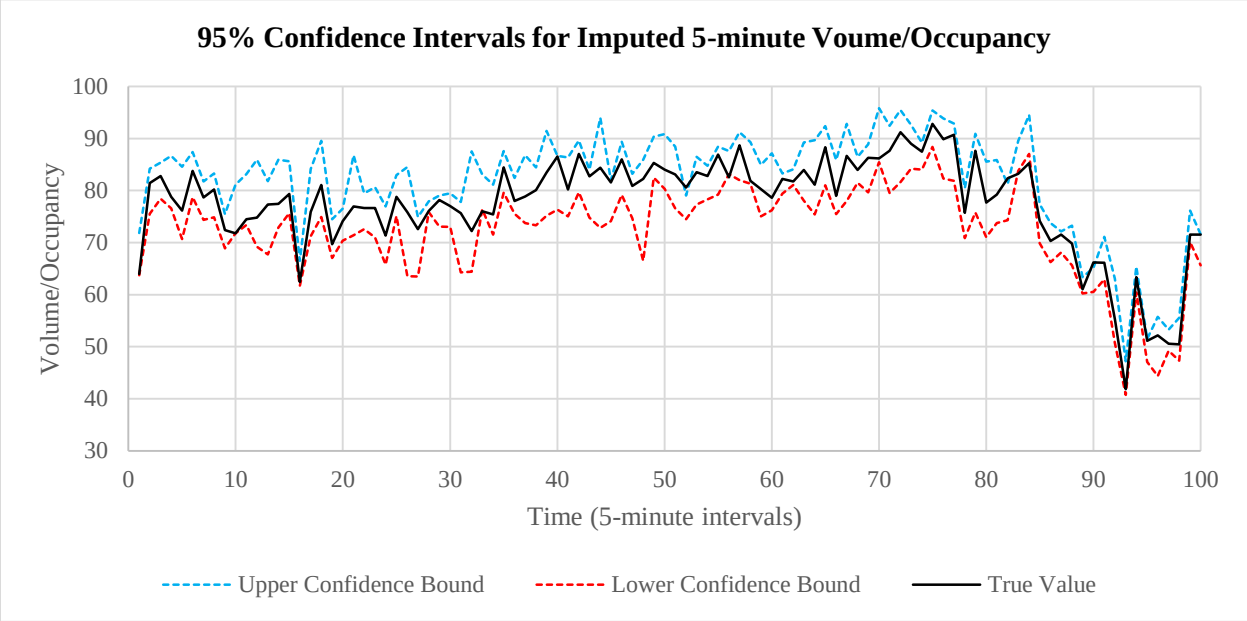

Figure 12: 95% Confidence Bounds from Site A for September, 2012, missing 40%

7.3. PMM Volume Imputation

PMM matching was used to impute volume, as it is faster computationally than CART and performs slightly better for volume. As in volume/occupancy imputation, when the data is missing at random, the 5-minute data includes the non-missing observed values in the aggregation step. For missing day and month patterns, the 5-minute data includes only imputed values, as all observed values for the time period of interest are set to missing.

Table 10 below shows the imputation results for a number of missing rates (MCAR pattern) during the month of May, 2012. Results in Table 10 are aggregated to the 5-minute level. In all cases, the MAPE is below 5% and the imputed variance is quite close to the true variance. The mean error is consistently low at all missing rates, which indicates little bias is introduced when data is missing completely at random.

Table 10: Volume Results for May, 2012 at the 5-minute level, Various Missing Rates

	missing	Mean Error	MAE	MAPE	Variance True	Variance Impute
Site A	10%	0.584	1.52	1.54%	625.1	642.2
	20%	0.587	2.13	2.12%	625.1	629.5
	30%	0.599	2.61	2.58%	625.1	623.6
	40%	0.701	2.92	2.90%	625.1	613.7
	50%	0.538	3.12	3.10%	625.1	607.8
	60%	0.657	3.37	3.35%	625.1	589.3
Site B	10%	0.478	2.15	1.91%	544.1	533.8
	20%	0.481	3.12	2.78%	544.1	529.7
	30%	0.635	3.63	3.26%	544.1	520.1
	40%	0.620	4.03	3.62%	544.1	510.7
	50%	0.626	4.43	3.99%	544.1	507.3
	60%	0.499	4.81	4.34%	544.1	494.1

Table 11 below shows the imputation results (MAPE and MAE only) for January and May of 2012. Note that he results for January are somewhat worse than for May, which is partly due to the increased rate of missingness and lower quality of the true data available during January.

Table 12 below shows the imputation results (MAE and MAPE only) for missing entire days. Three days in January and May were set missing, and imputed using only data from the month of interest. The need to impute missing days often arises from hanging detector issues, when an entire day of data is discarded due to a detector being stuck on or off. The majority of MAPE

values in this case are below 5% at the 5-minute level, with the worst being 5.2%. This indicates that good imputation accuracy is achievable for missing days, in some cases better than that for high rates of random missingness.

Table 11: Volume Results for January and May of 2012, Various Missing Rates

	missing	May 5 minute		May 20 seconds		January 5 minute		January 20 seconds	
		MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE
Site A	10%	1.52	1.5%	1.31	22.7%	1.69	2.7%	1.25	26.5%
	20%	2.13	2.1%	1.28	22.1%	2.31	3.4%	1.25	26.3%
	30%	2.61	2.6%	1.30	22.8%	2.77	4.0%	1.26	26.9%
	40%	2.92	2.9%	1.29	22.7%	3.02	4.3%	1.25	26.7%
	50%	3.12	3.1%	1.30	22.8%	3.25	4.6%	1.26	26.8%
	60%	3.37	3.4%	1.30	22.8%	3.45	4.8%	1.25	26.3%
Site B	10%	2.15	1.9%	1.63	27.7%	2.14	3.8%	1.50	32.3%
	20%	3.12	2.8%	1.64	27.8%	2.89	4.6%	1.49	32.2%
	30%	3.63	3.3%	1.63	27.7%	3.49	5.4%	1.48	32.2%
	40%	4.03	3.6%	1.63	27.5%	3.81	5.8%	1.48	31.9%
	50%	4.43	4.0%	1.62	27.4%	4.19	6.4%	1.48	32.2%
	60%	4.81	4.3%	1.63	27.8%	4.45	6.6%	1.48	32.0%

Table 12: Volume Results for Entire Days Missing

	Missing	May 2012 5 minute		May 2012 20 seconds		January 2012 5 minute		January 2012 20 seconds	
		MAE	MAPE	MAE	MAPE	MAE	MAPE	MAE	MAPE
Site A	Saturday	4.04	3.7%	1.27	21.3%	3.27	3.1%	1.27	21.8%
	Wednesday	3.81	4.2%	1.12	20.6%	3.37	3.0%	1.19	19.6%
	Monday	3.48	3.9%	1.28	26.4%	3.29	3.3%	1.18	21.3%
Site B	Saturday	5.16	4.7%	1.52	25.6%	4.21	4.5%	1.78	35.5%
	Wednesday	5.09	4.4%	1.58	24.8%	4.88	4.6%	1.69	29.7%
	Monday	4.75	5.2%	1.60	32.1%	5.02	5.0%	1.34	24.6%

Missing entire months is the most challenging missing pattern used in this analysis. For volume imputation, the months of January and May were set missing in entirety, and imputation was performed for each hour of the day using an entire year of data as described previously. Table

13 below shows the results for both sites when missing entire months. Note that, even in the worst case (i.e. Site A), relatively little bias is introduced in the imputed values. The worst MAPE at the 5-minute level is 11.36%, but the quality of the underlying "true" likely contributes somewhat to the inaccuracy. In some cases, such as during a month of particular low quality data, it will be preferable to impute the entire month rather than attempt to impute only missing values.

Table 13: Volume Results for Entire Months Missing

	Aggregation Level	Mean Error	MAPE	MAE	Variance True	Variance Impute
January Site A	20-sec	0.246	34.50%	1.49	8.11	5.61
	5-min	4.492	11.36%	6.53	924.34	728.23
May Site A	20-sec	-0.141	25.99%	1.57	8.02	5.47
	5-minute	-1.526	4.80%	5.41	625.70	610.97
January Site B	20-sec	0.094	27.87%	1.27	7.50	5.47
	5-min	1.915	6.72%	4.44	866.44	709.58
May Site B	20-sec	0.004	23.00%	1.31	6.49	4.50
	5-minute	0.276	4.69%	3.82	550.99	475.53

Based on the results shown in Figure 13, it is clear that the imputed volume follows the trend of the true values closely. The accuracy is substantially improved by aggregation to the 5-mintue level as shown in Figure 14.

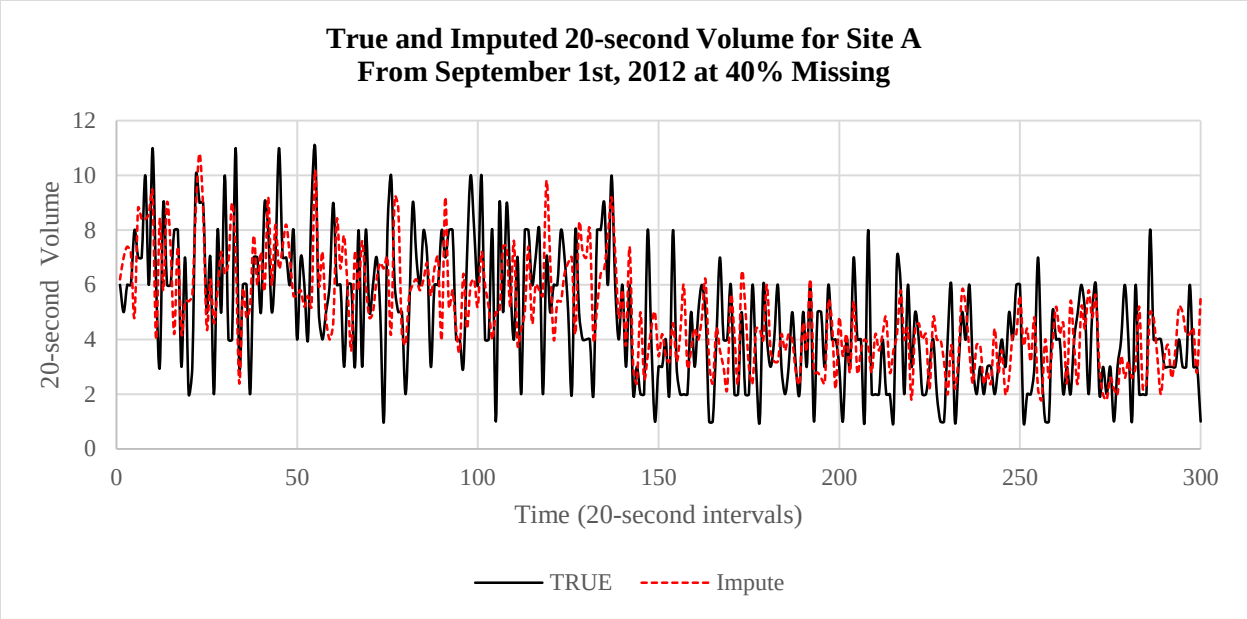

Figure 13: Results for Site A, from September 1st, 2012, 20-second level, 40% missing

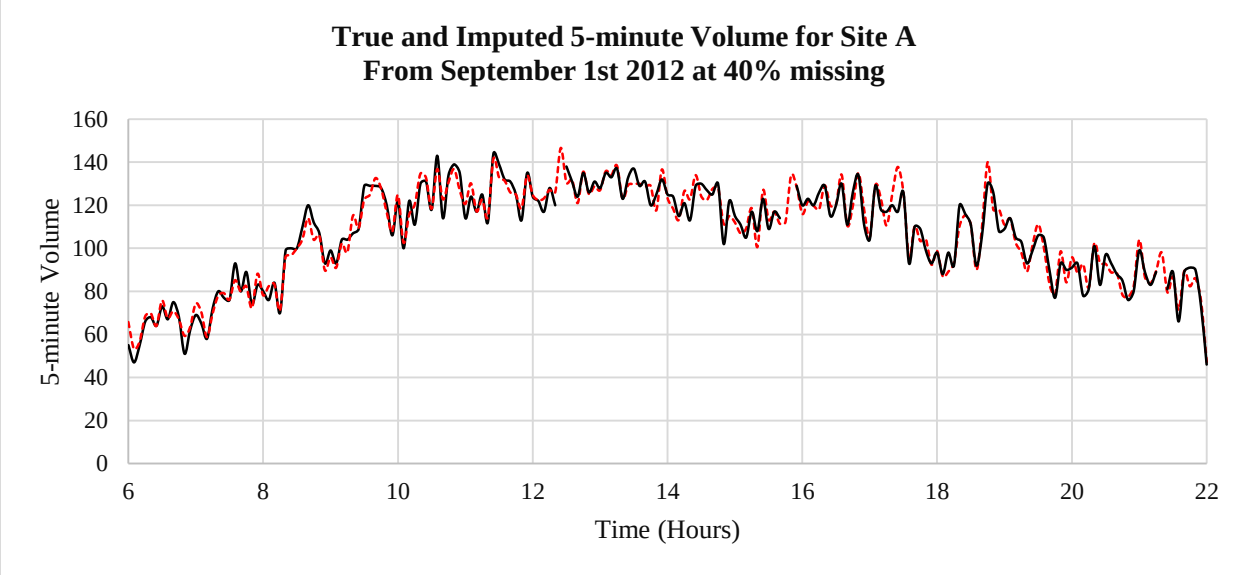

Figure 14: Results for Site A, from September 1st, 2012, 5-minute level, 40% missing

From Figure 15 below it is clear that the more extreme volume values are underrepresented at the 20-second level, and that the mid-range values are somewhat overrepresented. However, when post imputation aggregation is performed, the distribution of true and imputed values are

quite similar as shown in Figure 16. In any case, referring to Table 13, even in the worst case (i.e. missing entire months) any bias present is not consistently positive or negative from month to month, which will result in little bias in AADT and other long term volume measures.

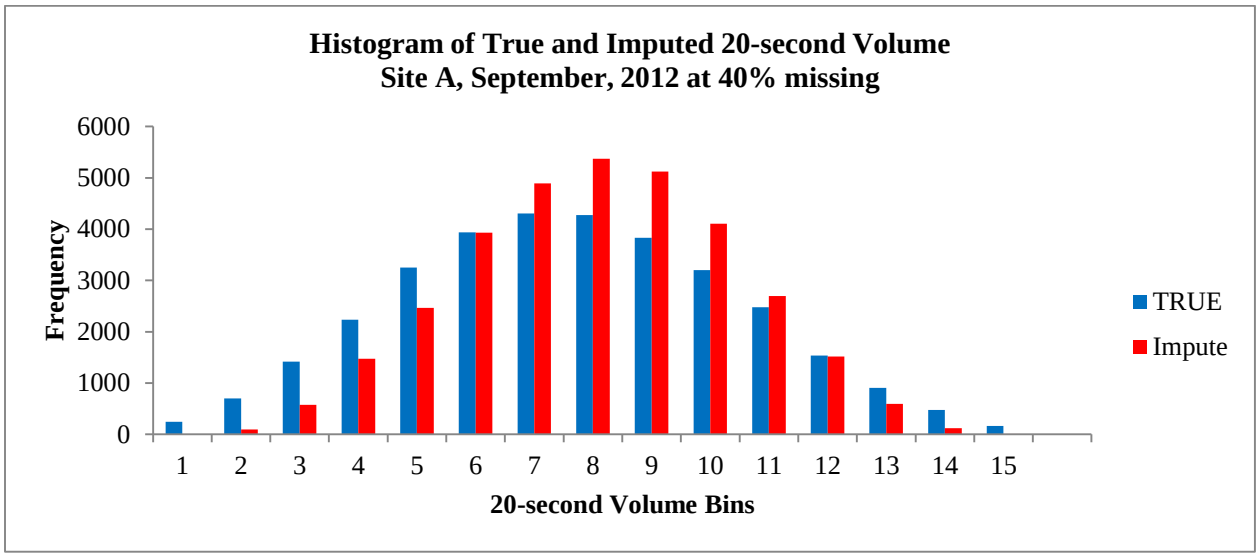

Figure 15: Histogram for Site A, September, 2012, 20 second level at 40% missing

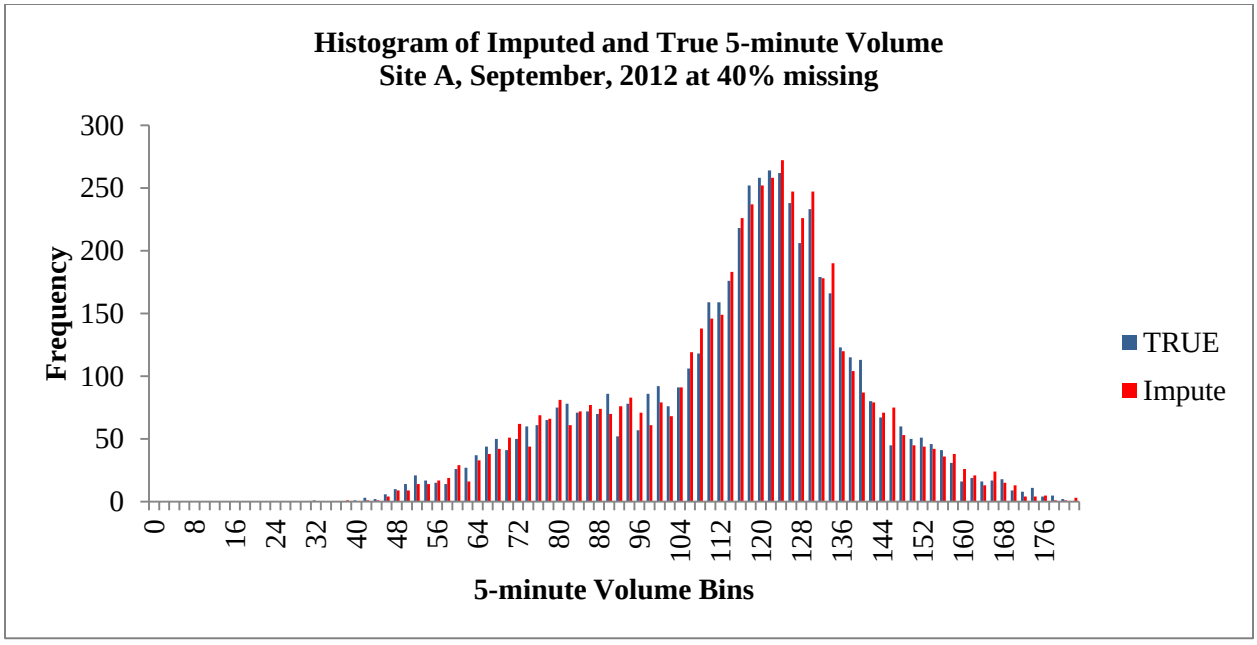

Figure 16: Histogram for Site A, September, 2012, 5-minute level at 40% missing

Confidence intervals for 5-minute volume estimates were developed similarly to those shown for CART imputation. In this case, almost exactly 95.0% of all values fell within the 95% confidence bounds for 5-minute volume. Again, this demonstrates the feasibility of estimating the uncertainty of imputed values with reasonable accuracy.

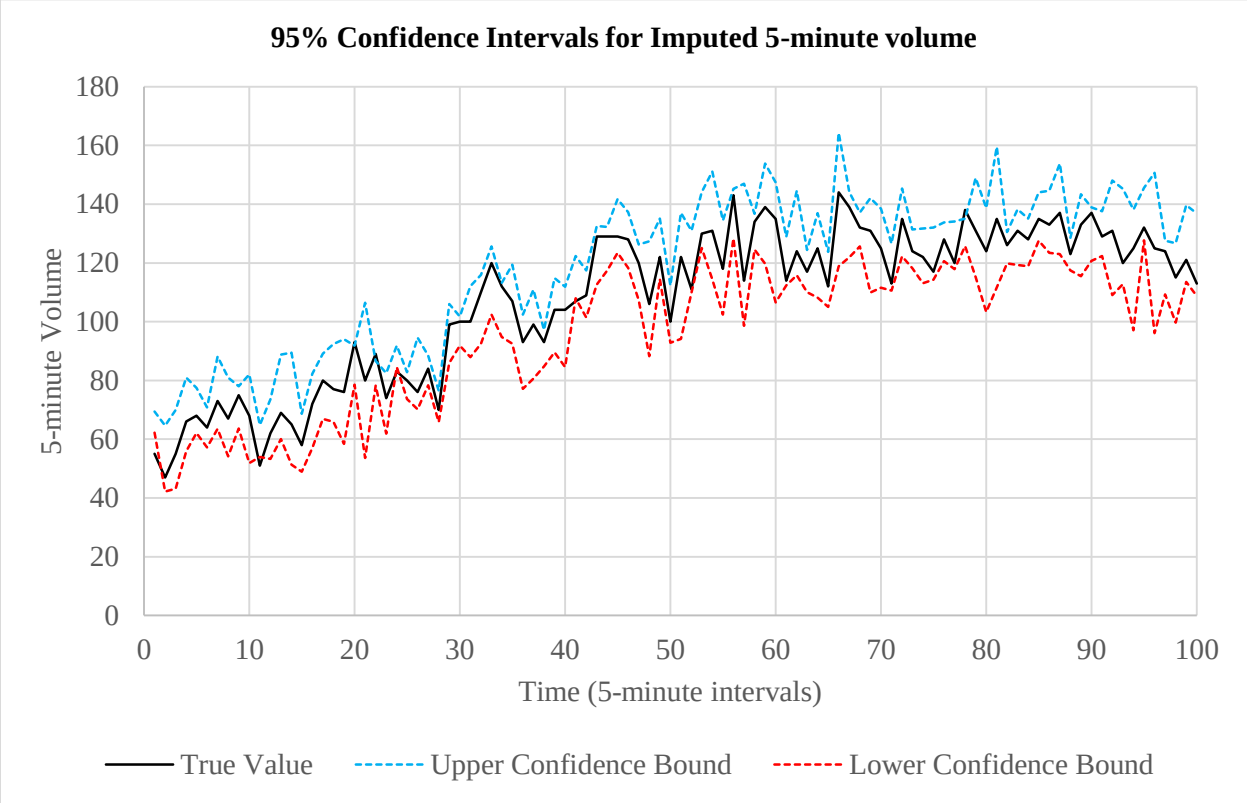

Figure 17: 95% Confidence Bounds for 5-minute Volume, September 2012, 40% missing

Chapter 8: Conclusions

Typically, the purpose of a multiple imputation procedure is to allow some complete data analysis that would not be possible in the presence of missing data. To do this, the desired analysis is performed for each of the multiply imputed datasets, and the results pooled using the guidelines provided by Ruben (1987) or other methods. This is to say, the objective of the process is not to simply provide an imputed replacement value for each of the missing values. Instead, the purpose is to allow analysis that would not otherwise be possible, and do so in a way that realistically represents the added uncertainty introduced by the missing values. Thus, multiple imputation is usually done for one of two purposes:

1. To complete some analysis in the presence of missing data. In this case, multiple imputation is performed by the same person or group of people who will conduct the analysis. The individual(s) will choose the imputation methodology to suit the needs of the specific analysis.
2. Multiple imputation is performed on a dataset that is to be made public. In this case, the dataset is released as $m > 1$ imputed datasets. The importance of "mindless" imputation is critical here, because the objectives and methods of the analysis are not known *a priori*.

Note that the purpose of the methods described in this work do not really fit into either of these two groups. Unlike most applications of multiple imputation, the objective here is not to produce and use multiple imputed datasets in analysis. Rather, the objective is to create and make available a single complete dataset containing enough information for users of the dataset to decide on an acceptable level of uncertainty, and apply the data which fits the needs of a specific application.

The reason for this is that, due to the scope of loop detector datasets (many hundreds of millions of rows), it is not feasible to create and store multiple copies of the data. Likewise, for performance reporting and many other applications of traffic data, repeated analysis is not meaningful or in some cases even computationally feasible. The reality is that a significant fraction of most traffic sensor datasets are missing or erroneous, and that a great number of decisions will be made based on this data regardless of the imputation method applied. Thus, as opposed to dwelling on the notion of a "perfect" imputation and analysis scheme, the policy adopted in this research is one of balancing the combined goals of accuracy, accessibility, and the ability to quantify uncertainty.

The results shown in the previous section demonstrate the effectiveness of the proposed imputation methods in terms of imputation accuracy, as well as in the ability to accurately quantity the uncertainty in the imputed values. Both the volume and volume/occupancy imputation methods preserve the relationships between neighboring lanes, and work well even under challenging missing data patterns. The resulting completed dataset with informative metadata can then be made available for engineers and decision makers. By incorporating this research into the DRIVE Net online data analysis, quality control, and visualization platform, engineers and decision makers will have access to preprocessed, complete, and high quality traffic sensor datasets, along with enough information to understand the level of uncertainty in each value.

8.1. Future Work

A direct and intuitive extension of this work is to apply the MICE procedure to other traffic datasets, for example, probe vehicle speed data. While different temporal and spatial correlation structures are present in such data compared to loop detector data, the primary difference will be simply in the data structure and the model used for imputation. Additional work could also be

put into other predictors that may help to describe the missing data mechanisms, or other methods for imputing NMAR data. Though weather variables such as temperature and precipitation were investigated for this purpose and determined to be of little value, additional work may provide insight into alternative ways of incorporating such predictors in the model. Finally, additional work is needed to automate this work and incorporate it into the DRIVE Net e-science platform developed at the University of Washington STAR Lab. The objective is to create a self-contained, transparent, and semi-automated process for comprehensive data quality control. This work is a key step in achieving this goal.

Additional Acknowledgements

This work was supported in part by the Pacific Northwest Transportation Consortium (PacTrans) and the Washington State DOT.

Bibliography

Al-Deek, H. M., C. Venkata, and S. R. Chandra. New algorithms for filtering and imputation of real-time and archived dual-loop detector data in I-4 data warehouse. Transportation Research Record: *Journal of the Transportation Research Board* Vol. 1867, pp. 116-126, 2004.

Asif, M. T., N. Mitrovic, L. Garg, J. Dauwels, and P. Jaillet. Low-dimensional models for missing data imputation in road networks. *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on,* pp. 3527-3531. IEEE, 2013.

Azur, M. J., E. A. Stuart, C. Frangakis, and P. J. Leaf. Multiple imputation by chained equations: what is it and how does it work?. *International journal of methods in psychiatric research*, Vol. 20, pp. 40-49, 2011.

Burgette, L. F., and J. P. Reiter. Multiple imputation for missing data via sequential regression trees. *American journal of epidemiology* Vol. 172, no. 9, pp. 1070 – 1076, 2010.

Van Buuren, S., and K. Groothuis-Oudshoorn. MICE: Multivariate imputation by chained equations in R. *Journal of statistical software* Vol. 45, no. 3, 2011.

Chang, H., D. Park, Y. Lee, and B. Yoon. Multiple time period imputation technique for multiple missing traffic variables: nonparametric regression approach. *Canadian Journal of Civil Engineering*, Vol. 39, no. 4, pp. 448-459. 2012.

Chen, C., J. Kwon, J. Rice, A. Skabardonis, and P. Varaiya. Detecting errors and imputing missing data for single-loop surveillance systems. Transportation Research Record: *Journal of the Transportation Research Board,* Vol. 1855, pp. 160-167, 2003.

Chen, C., K. Petty, A. Skabardonis, P. Varaiya, and Z. Jia. Freeway performance measurement system: mining loop detector data. Transportation Research Record: *Journal of the Transportation Research Board,* Vol. 1748, pp. 96-102, 2001.

Coifman, B. and S. Kim. Speed estimation and length based vehicle classification from freeway single-loop detectors. *Transportation Research Part C,* Vol 17, pp. 349 – 364, 2009.

Doove, L. L., S. Van Buuren, and E. Dusseldorp. Recursive partitioning for missing data imputation in the presence of interaction effects. *Computational Statistics & Data Analysis*, Vol. 72, pp. 92-104, 2014.

Hastie, T., R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning; Data Mining, Inference, and Prediction, second ed*. New York, NY: Springer Verlag,

Haworth, J., and T. Cheng. Non-parametric regression for space–time forecasting under missing data. *Computers, Environment and Urban Systems,* Vol. 36, no. 6, pp. 538 – 550, 2012.

Horton, N. J., and S. R. Lipsitz. Multiple imputation in practice: comparison of software packages for regression models with missing variables. *The American Statistician*, Vol. 55, no. 3, pp. 244 – 254, 2001.

Ishimaru, J. M., and M. E. Hallenbeck. *Flow evaluation design technical report*. No. WA-RD 466.2, Washington State Department of Transportation, 1999.

Friedman, J., T. Hastie, and R. Tibshirani. Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, Vol. 33, no. 1, pp. 1-22, 2010.

Li, L., Y. Li, and Z. Li. Efficient missing data imputing for traffic flow by considering temporal and spatial dependence. *Transportation Research Part C: Emerging Technologies*, Vol. 34, pp. 108 – 120, 2013.

Little, R. J. Missing-data adjustments in large surveys. *Journal of Business & Economic Statistics*, Vol. 6, no. 3, pp. 287-296, 1988.

Morgan, J. N., and J. A. Sonquist. Problems in the analysis of survey data, and a proposal. *Journal of the American statistical association*, Vol. 58, no. 302, pp. 415 – 434, 1963.

Ni, D., and J. D. Leonard II. Markov chain monte carlo multiple imputation using bayesian networks for incomplete intelligent transportation systems data. Transportation Research Record: *Journal of the Transportation Research Board*. Vol. 1935, pp. 57 – 67, 2005.

Ni, D., J. D. Leonard, A. Guin, and C. Feng. Multiple imputation scheme for overcoming the missing values and variability issues in ITS data. *Journal of transportation engineering*, Vol. 131, no. 12, pp. 931 – 938, 2005.

Qu, L., L. Li, Y. Zhang, and J. Hu. PPCA-based missing data imputation for traffic flow volume: a systematical approach. *Intelligent Transportation Systems, IEEE Transactions on*, Vol. 10, no. 3, pp. 512 – 522, 2009.

R Core Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2014.

Rajagopal, R., and P. P. Varaiya. Health of California's Loop Detector System (UCB-ITS-PRR-2007-13). California PATH Program, Institute of Transportation Studies, University of California at Berkeley, 2007.

Revolution Analytics and S. Weston. doParallel: Foreach parallel adaptor for the parallel package. R package version 1.0.8. 2013

Revolution Analytics and S. Weston. foreach: Foreach looping construct for R. R package version 1.4.2, 2014.

Ripley, B. and M. Lapsley. RODBC: ODBC Database Access. R package version 1.3-10, 2013

Rubin, D. B. Inference and missing data. *Biometrika*, Vol. 63, no. 3, pp. 581 – 592, 1976.

Rubin, D. B. *Multiple imputation for nonresponse in surveys*. New York, NY, John Willey and Sons, 1987.

Schafer, J. L. *Analysis of incomplete multivariate data*. Boca Raton, FL. CRC press, 2010.

Schafer, J. L., and J. W. Graham. Missing data: our view of the state of the art. *Psychological methods*, Vol. 7, no. 2, pp. 147 - 177, 2002.

Schafer, J. L. Multiple imputation: a primer. *Statistical methods in medical research*, Vol. 8, no. 1, pp. 3 – 15, 1999.

Smith, B. L., W. T. Scherer, and J. H. Conklin. Exploring imputation techniques for missing data in transportation management systems. Transportation Research Record: *Journal of the Transportation Research Board*, Vol. 1836, pp. 132 – 142, 2003.

Tan, H., G. Feng, J. Feng, W. Wang, Y. Zhang, and F. Li. A tensor-based method for missing traffic data completion. *Transportation Research Part C: Emerging Technologies*, Vol. 28, pp. 15 – 27, 2013.

Therneau, T., B. Atkinson, and B. Ripley. rpart: Recursive Partitioning and Regression Trees. Version 4.1-8, 2014.

Tibshirani, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 58, no. 1, pp. 267 – 288, 1996.

Tibshirani, R. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, Vol. 73, no. 3, pp. 273 – 282, 2011.

van Buuren S, K. Groothuis-Oudshoorn mice: Multivariate Imputation by Chained Equations. R package version 2.9, 2011.

van Buuren, S., and K. Groothuis-Oudshoorn. Flexible multivariate imputation by MICE. Leiden, The Netherlands: TNO Prevention Center, 1999.

Wang, Y. and N. L. Nihan. Can Single-Loop Detectors Do the Work of Dual-Loop Detectors? *Journal and Transportation Engineering,* Vol 129, no. 2, pp. 169 – 176, 2003.

Wang, Y., and N. L. Nihan. Freeway traffic speed estimation with single-loop outputs. Transportation Research Record: *Journal of the Transportation Research Board*, Vol. 1727, no. 1, pp. 120-126, 2000.

White, I. R., P. Royston, and A. M. Wood. Multiple imputation using chained equations: issues and guidance for practice. *Statistics in medicine*, Vol. 30, no. 4, pp. 377 – 399, 2011.

Wright, D. R., and J. M. Ishimaru. *Data Quality Handling Approach of TRACFLOW Software (WA-RD 679.1).* Washington State Transportation Center Technical Report, 2007.

Appendix: Large dataset Analysis in R

With single threaded architecture and in-memory native data structures, at first glance the R statistical computing language does not seem inherently well suited for working with large datasets. However, because of its flexibility and wide array of sophisticated analytical tools, it has become one of the dominant toolsets for data mining and analytics. This section describes some methods that can be used to increase the efficiency of R when working with large datasets, and which become unavoidable as the scope and complexity of the analysis increases.

Database connectivity

With limited memory resources, it is necessary to have a repository for temporary and permanent storage of raw data and analysis results. One way that this can done in R is to establish a connection to a remote SQL-based database, and query data only as needed. The RODBC package in R (Ripley & Lapsley, 2013) provides access to most t-SQL commands, and can import query results in the form of in-memory tables or dataframes. Database commands including queries, updates, and stored procedures can be executed from the R command line using T-SQL syntax, making database connectivity both fast and intuitive. In this work, complex queries are saved in a Microsoft SQL Server database as stored procedures, thereby limiting the T-SQL code to a single line in R.

Multi-core processing

R is single threaded in its basic implementation, but several packages are available to utilize multiple CPU cores. In a windows system, the doParallel package (Revolution Analytics & Weston, 2014a) essentially launches multiple R instances (each running on a single thread), allowing several processes to be run in parallel on a multi-core workstation. This can speed up

processing time immensely for processes that are amenable to parallelization. In conjunction with the forEach package (Revolution Analytics & Weston 2014b), embarrassingly parallel processes can be run similar to a FOR loop and the results combined either by aggregation or subjoin. For example, this might be used when the same algorithm must be run multiple times with different random seeds. In this work, the doParallel and foreach packages were used in combination to run multiple separate imputation processes simultaneously, thereby decreasing the computation time by nearly a factor of the number of cores used (using between 3 and 6 CPU cores). Efficiency improvements are typically most dramatic for large and intensive processes, as there is some overhead associated with data transfer.

Consider also that, depending on hardware configuration, operations on very large datasets are often constrained primarily by memory instead of CPU. This work was completed using a desktop i7 4790 (8 logical cores) with 24GB DDR3 RAM, and in no case did a 6-core parallel process use more than half the available memory.