

UAFS: Uncertainty-Aware Feature Selection for Problems with Missing Data

Andrew Becker^{1,2} and James P. Bagrow^{1,2,*}

¹Department of Mathematics & Statistics, University of Vermont, Burlington, VT, United States

²Vermont Complex Systems Center, University of Vermont, Burlington, VT, United States

*Corresponding author. Email: james.bagrow@uvm.edu, Homepage: bagrow.com

April 2, 2019

Abstract Missing data are a concern in many real world data sets and imputation methods are often needed to estimate the values of missing data, but data sets with excessive missingness and high dimensionality challenge most approaches to imputation. Here we show that appropriate feature selection can be an effective preprocessing step for imputation, allowing for more accurate imputation and subsequent model predictions. The key feature of this preprocessing is that it incorporates uncertainty: by accounting for uncertainty due to missingness when selecting features we can reduce the degree of missingness while also limiting the number of uninformative features being used to make predictive models. We introduce a method to perform uncertainty-aware feature selection (UAFS), provide a theoretical motivation, and test UAFS on both real and synthetic problems, demonstrating that across a variety of data sets and levels of missingness we can improve the accuracy of imputations. Improved imputation due to UAFS also results in improved prediction accuracy when performing supervised learning using these imputed data sets. Our UAFS method is general and can be fruitfully coupled with a variety of imputation methods.

1 Introduction

Nearly all real-world data sets that are collected have some amount of missingness [1, 2]. Many methods have been developed to handle missingness, and many statistical and predictive methods are robust to some degree of missingness in data. In the case where an analysis method that is not robust to missingness is being used, either deletion of rows or columns with missingness must be done, or the data must be filled in. The process of filling in missing data with new values is called imputation. There are many different methods of imputation, but multiple imputation has become one of the more widely used methods for dealing with missing data over the last several years [3]. Multiple imputation is a process where many imputations are made for each missing value in order to get a sense of the variation in the imputed values. In cases where sample size is limited, the amount of missingness makes listwise deletion remove a significant portion of the sample, or where pairwise comparisons cannot be used to make the needed analyses, imputation methods

and multiple imputation are frequently used.

There has been literature with information regarding how to analyze data post-imputation and deal with integration of the many multiply imputed data sets, along with selection of features (also known as variables or predictors) for the imputation model. [4, 5]. These guidelines suggest that features be kept if they are to be included in the final data model, are part of interactions to be included within the final data model, or are informative regarding the imputation of these interaction terms or the main effects. [6]. Additionally, these authors suggest you keep those features that are most likely to satisfy the assumption of Missing At Random (MAR). These guidelines are designed with data sets that have only a few features with missingness in them, and revolve around attempting to build a data model for those particular features. However, much of the time when there is missingness within a data set it is not confined to a single feature and building data models for each feature can become very complex. The guidelines proposed above do not account for situations where there are large numbers of features that all have missingness, and the relationships between those features are unclear.

When most features have missingness, following the guidelines in [6] results in a cascade of additions of features to the imputation model. When you follow the third step, adding features that are informative regarding features already in the imputation model, additional features with missingness are added. Each of these new features then is being imputed and requires that you add features that are important regarding it's imputation. While many of these features that are important are already in the imputation model, having any new features added can result in many new features being added. In many cases this will result in a long process of adding features again and again that ends in nearly the entire data set being used for imputation. If imputation needs to be performed prior to model building then building an imputation model in this fashion is not possible and other methods must be used to create consistent objective imputation models.

We propose *Uncertainty-aware feature selection* (UAFS), a method for preprocessing data sets with missingness to improve imputation of missing values, especially for data sets containing over 50% of fields missing. UAFS is motivated by the following observations: (1) not all features within most data sets are useful at increasing the accuracy of imputation, (2) features that may be useful for imputation are not necessarily also useful for prediction, and (3) we can determine what features are useful for imputation prior to performing imputation to a high degree of accuracy. This allows us to work with large and complex data sets where the distinction between features useful for imputation and those that are not may not be clear. Using UAFS also reduces dimensionality of the data set prior to imputation and can remove features that may be confounding the imputation model. Additionally, the reduced dimensionality reduces the processing time spent on each imputation and the reduced set of features makes model building a simpler task.

In this work we show that applying UAFS to a variety of data sets results in an increase in imputation accuracy

and model predictive accuracy. This holds for synthetic data, synthetic missingness in real data, and in real data sets with missingness. Increased imputation and model accuracy is particularly apparent in data sets with a large number of features and a large amount of missingness.

The rest of this work is organized as follows. In Sec. 2 we detail the UAFS method for imputation and prediction problems, and provide a motivating theoretical analysis (Sec. 2.1). Section 3 describes the materials and methods we use to understand the performance of UAFS on a corpus of representative data sets. We present the results of our UAFS experiments in Sec. 4. Finally, we conclude with a discussion in Sec. 5.

2 Uncertainty aware feature selection

We propose that in order to reduce error in our analyses when imputing and performing supervised learning we should use only those features $X \in \{X_i\}_{i=1}^k$ that are significantly related to our outcome variable Y . Of course, identifying and focusing upon the most informative features is the standard motivation behind using feature selection and dimensionality reduction when building predictive models. However, when there is missingness in X_i and we wish to determine the strength of the relationship between X_i and Y (as is standard practice, we assume there are no missing values in Y), we need to specifically account for the uncertainty in our measurement of that relationship's strength that is introduced by the missing observations within X_i . To the best of our knowledge feature selection prior to imputation and then supervised learning has not been explored.

UAFS can be added to an imputation and supervised learning workflow using the following steps:

1. Determine strengths of pairwise relationships between Y and the features X_i .
2. Retain any X_i with significant relationships with Y accounting for the uncertainty in our measurements of those relationships due to missingness in X_i .
3. Impute the data set containing Y and the selected X_i using an appropriate imputation method.
4. Conduct analyses on the imputed data set as needed.

UAFS can be applied whenever imputation is appropriate. This means that MCAR data can always be imputed in this way, and MAR data can also be imputed. UAFS can also be used with MNAR data, when the missing data mechanism is adequately accounted for. It is important to make sure that you know what type of missingness you have and that it is being appropriately imputed after applying UAFS. We show that MCAR data is easily handled using UAFS, but it is extensible into cases which are MAR and MNAR as long as the data imputation is handled appropriately.

Our UAFS method proceeds as follows. To determine the relationship between Y and X_i we can only compare

where we have shared values. Each X_i can be partitioned into observed and missing portions, X_i^{obs} and X_i^{mis} with N_i points in X_i^{obs} and the remaining $N - N_i$ points missing. Instead of attempting to impute the values of X_i^{mis} , we first calculate the correlation $r(X_i^{\text{obs}}, Y)$ between X_i^{obs} and Y using the subset of Y corresponding to the observed values of X_i . Yet increased missingness in X_i will affect our measure of r . We therefore need to generate a confidence interval for r , for example by using a Z-transformation. This provides a confidence of the form $CI = r(X_i^{\text{obs}}, Y) \pm Z_{1-\alpha} SE_i$ with standard error

$$SE_i = \frac{1}{\sqrt{N_i - 3}} \quad (1)$$

We then select those X_i with confidence intervals that exclude $r = 0$, i.e.,

$$X_i^{\text{UAFS}} = \left\{ X_i \mid \text{sgn} \left(r(X_i^{\text{obs}}, Y) + Z_{1-\alpha} SE_i \right) = \text{sgn} \left(r(X_i^{\text{obs}}, Y) - Z_{1-\alpha} SE_i \right) \right\}.$$

Following convention, we use $\alpha = 0.025$ giving 95% confidence intervals for performing uncertainty aware feature selection. Once UAFS is used to select features, imputation can be performed using an imputation method of the researchers choosing.

This UAFS procedure results in a natural preference towards those X_i with lower levels of missingness. This is beneficial to the imputation process. Having less missingness in the data set we want to impute reduces the difficulty of the computation and typically increases the accuracy of imputation.

One reason for selecting a subset based on non-zero absolute correlation is because multiple regression coefficients are a *partition* of the simple Pearson correlation coefficient [7]. Because of this, when a feature has a true correlation of zero we would also expect that feature to have a true multiple regression coefficient of zero. This also applies for logistic regression, where the logistic regression coefficients are also partitions of the simple Pearson correlation coefficient. In this classification case, it also makes sense to determine whether a feature is possibly useful by selecting those which we believe to have a nonzero absolute correlation with the outcome variable. If the observed data support the true correlation coefficient being zero, we also expect that any contribution to any imputation model from that feature would simply be a result of noise in the data. It follows that any contribution to a predictive model from such features would also simply be a result of noise. As such, removing them from the data set prior to imputation should result in increases in imputation accuracy and predictive model accuracy.

Most feature selection methods for complete data test subsets of features for their relationship with the outcome variable. Indeed, it is often desired for one to have features that are individually most predictive of the outcome but least correlated with one another. Our approach described above considers simple pairwise relationships between one feature and the outcome variable and, further, does not attempt to find features that are uncorrelated from one another. We do this because retaining uncorrelated features may harm the imputation models, as missing values of one feature

will be predicted using observed values of other features. One can always apply another, traditional subset selection after UAFS and imputation are completed to achieve the latter property if desired, although we do not examine doing so in this work.

There are other feature selection methods that we could use before imputation, yet their properties are not as beneficial as the approach described above. One option is to select those features with an observed correlation greater than some cutoff value, for example, $r > C = 0.05$. We could set this cutoff at another level, and depending on your purposes you may select a larger or smaller cutoff. For a second cutoff based purely on correlation we select those features where $r > C = 0.10$. The three criteria for choosing features are outlined below in Eq. (2).

$$\begin{aligned}
 C_{UAFS} &= \tanh\left(\frac{1.96}{\sqrt{n-3}}\right) \\
 C_{\text{constant}_1} &= 0.05 \\
 C_{\text{constant}_2} &= 0.10
 \end{aligned}
 \tag{2}$$

The cutoff for the C_{UAFS} comes from using an $\alpha = 0.05$ confidence level in the hypothesis test $H_0 : \rho = 0$ vs. $H_0 : \rho \neq 0$, which is equivalent to making a 95% confidence interval around the observed ρ . This can be transformed back through the Fisher Z Transformation to get a cutoff in terms of the sample correlation, r_{xy} .

To compare these selection cutoffs, we simulate pairs of features that are truly correlated, remove data, and compare these methods as the amount of data remaining approaches zero. In order to do this we simulate a data set which contains a single outcome variable, Y , and 1000 X_i . Each of these X_i is truly correlated with Y at the 0.05 level. In Fig. 1 we present the probability in our simulations for a feature to be selected as a function of how much data are missing for that feature. UAFS results in slight selection bias towards those features with more data and against features with high levels of missingness. Interestingly, we also see that the constant cutoff of 0.05 or 0.10 is biased towards features with higher levels of missingness. We would expect that since the variance in sample correlation increases as n decreases we would be less likely to select any features with very low sample sizes, but we find that as we reduce sample sizes below a threshold, the proportion of those features selected increases.

This slight bias as compared to a constant threshold which is biased towards features with a high degree of missingness is very important. Selecting features with a lower amount of missingness means that we can better estimate relationships between those features. Better estimates of relationships between features should result in better predictions of missing data points. Despite this small bias we believe using this threshold to be the best way to select features based on their pairwise correlation. UAFS uses a selection criteria that does not depend on setting an arbitrary cutoff at which to accept features into a data set. If an arbitrary cutoff is used, you can pick a cutoff with very poor properties accidentally, as can be seen with using the criteria to pick those features where $r > C = 0.10$. UAFS has

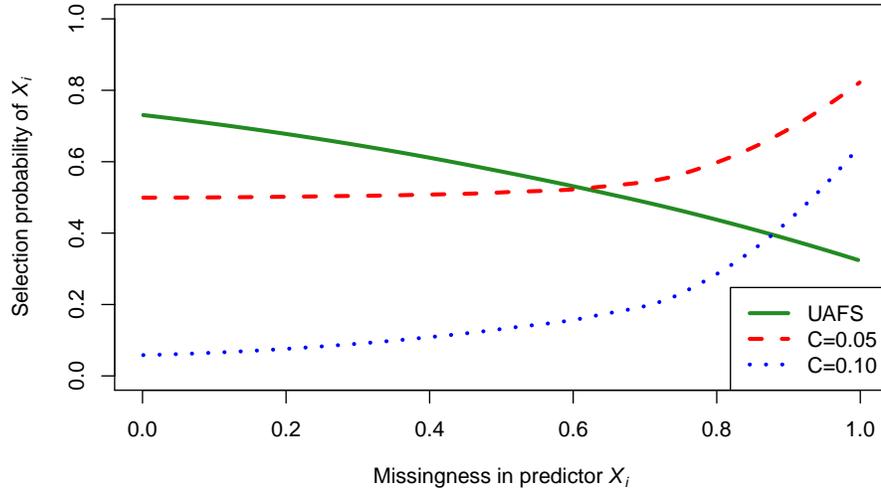


Figure 1: Selecting features for use in imputation using the p-values from a Fisher Z Transformation, as is done in UAFS, results in low bias overall and is slightly biased towards those features with more data.

beneficial properties for imputation and does not rely upon setting arbitrary levels that can be misspecified. Instead significance can be set higher or lower in order to get more or less features in the data set to be imputed, and typically we set this level at $\alpha = 0.05$ as is convention.

Beyond the overall selection probabilities shown in Fig. 1, it is also important to examine the rate of correctly selecting features that should truly be in a model versus those that should not. These same selection criteria are examined in Fig. 2, where we try to select the important features in a data set that contains one informative feature to the outcome, and one uninformative feature. We see that UAFS selects more informative features almost all of the time, yet it performs slightly worse than using the correlation cutoffs at extremely high levels of missingness, as seen in Fig. 2(a). However, at the same time UAFS selects uninformative features much less often than the correlation cutoffs even as we approach much lower levels of missingness, as seen in Fig. 2(b).

This idea of testing for pairwise significance can be extended beyond correlation. This can be useful if we have categorical features with a large number of categories. In the case of two categorical features a chi-squared test of homogeneity can be performed. This test has similar properties to the Fisher Z Hypothesis test, although the lower limit on the number of shared observations is higher. For a situation with a categorical feature paired with a continuous feature an F-test can be performed on the significance of the relationship between those two features. Again, this has the same features as the Fisher Z Hypothesis test we focus on and can be used for feature selection in the same fashion. One reason removing these uninformative features can result in more accurate imputations is explained below.

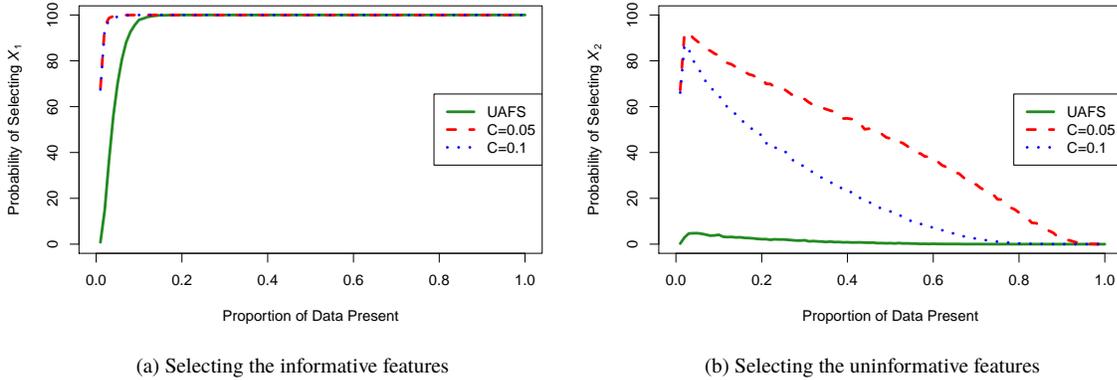


Figure 2: In a model containing only two features, one that is truly predictive of the outcome, and one that is uncorrelated, UAFS selects the truly predictive feature in nearly 100% of the data sets and the uncorrelated feature in less than 5% of the data sets. The accuracy is reduced as the proportion of data present is reduced. In the two other cases shown in Eq. 2 the percentage of the time X_1 is selected remains high, but they also select X_2 with a very high frequency in comparison to UAFS.

2.1 Theory

Here we motivate UAFS by showing that the inclusion of non-informative features when imputing the data set results in inflated estimates of imputation regression coefficients and the variances of those coefficients and that this inflation increases with increased missingness. Of course, preventing error due to use non-informative features is one of the primary motivations for performing feature selection, yet to the best of our knowledge feature selection has not been considered in the context of imputation. Our later experiments (Sec. 3) further show that the error in imputation increases with increased numbers of uninformative features and that UAFS can both improve predictive model performance and lead to more accurate imputation of missing values.

In our imputation experiments (Sec. 4) we primarily use the Predictive Mean Matching (PMM) algorithm [8] (see Sec. 3.4 for details on our imputation procedures). PMM is commonly used in imputation software such as MICE in R [9]. PMM is a complex, Bayesian procedure, yet the essence of PMM and related methods is that the entire data matrix is used to generate estimates of regression coefficients, and those regression coefficients are then used to estimate missing values.

To motivate UAFS by demonstrating the inflation of imputation coefficients analytically, we consider the following simplified variant of PMM. We focus on three basic steps within the PMM imputation process, and then iterating upon those steps. First, PMM is initialized with mean imputation of missing values: for all features X_i , any missing points in X_i are replaced with the mean of X_i^{obs} . Second, a linear model of X_i as a function of other features X_j ($j \neq i$) and of Y is built and used to make initial predictions of the values of X_i^{mis} . Third, a final prediction for each point in X_i^{mis} is made by selecting one or more donor points, those points within X_i^{obs} that are closest to the initial prediction, and

a randomly chosen donor point is then selected as the final imputed value. This process is then repeated for the next feature with missingness, until all features are imputed in this way. Finally, after all with missing points are imputed in this fashion the second and third steps are repeated, either for a fixed number of steps or until a convergence criteria is reached.

The first two steps of PMM, mean imputation and then predicting the missing values based on the other features, can be analytically examined to trace the increase in variance, and thus error, induced by imputing missing values. We examine the following simple case: we have one outcome variable, Y , one feature that is truly predictive of Y , X_1 , and a third, uninformative feature X_2 which does not predict Y . We will ask what effect, if any, including the uninformative X_2 will have. Here X_1 and X_2 are independent and identically distributed (iid) normal features with mean 0 and variance 1: $X_1, X_2 \sim N(0, 1)$. The true underlying model is $Y = X_1 + \epsilon$ where $\epsilon \sim N(0, t)$. These properties lead to $Y \sim N(0, 1 + t)$ and $\text{Cov}(X_1, Y) = 1$. Moreover, the covariance matrix Σ of Y , X_1 , and X_2 , which is also the expected sample covariance matrix S , is

$$\Sigma = E(S) = \begin{bmatrix} 1+t & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

The expectations of the sample covariances involving X_2 are 0, i.e., $E(S_{X_2X_1}) = E(S_{X_2Y}) = 0$. However, despite these zero expected sample covariances, there is still an effect on the variance in a linear model when adding X_2 , as we show below.

Suppose that not all values of the features are observed, i.e., there is missingness in X_1 and/or X_2 . For simplicity, we assume each data point of X_1 is available with constant probability p ; otherwise, it is missing with probability $q = 1 - p$, and likewise for X_2 . Further, assume there are no missing data points in Y .

At this point we begin imputation. The initial step is to do mean imputation on all missing data points. Mean imputation replaces the $n(1 - p)$ missing X_i values with the mean $\bar{X}_i = \sum_{j=1}^{np} X_{ij} / (np)$ of the np observed values. This

reduces the expected variance of the imputed feature depending on the amount of imputed data points:

$$\begin{aligned}
\text{Var}(X_i) &= \frac{1}{n-1} \sum_{j=1}^n (\bar{X}_i - X_{ij})^2 \\
&= \frac{1}{n-1} \left(\sum_{j=1}^{np} (\bar{X}_i - X_{ij})^2 + \sum_{j=np+1}^n (\bar{X}_i - \bar{X}_i)^2 \right) \\
&= \frac{1}{n-1} \left(\sum_{j=1}^{np} (\bar{X}_i - X_{ij})^2 + 0 \right)
\end{aligned}$$

$$E(\text{Var}(X_i^*)) = p \sigma_{X_i X_i} \quad (4)$$

Covariances are also scaled by p :

$$\begin{aligned}
\text{Cov}(Y X_i) &= \frac{1}{n} \sum_{j=1}^n (X_{ij} - \bar{X}_i)(Y_j - \bar{Y}) \\
&= \frac{1}{n} \left(\sum_{j=1}^{pn} (X_{ij} - \bar{X}_i)(Y_j - \bar{Y}) + \sum_{j=np+1}^n (\bar{X}_i - \bar{X}_i)(Y_j - \bar{Y}) \right) \\
&= \frac{1}{n} \left(\sum_{j=1}^{np} (X_{ij} - \bar{X}_i)(Y_j - \bar{Y}) + 0 \right)
\end{aligned}$$

$$E(\text{Cov}(X_i^*)) = p \sigma_{X_i Y} \quad (5)$$

These new mean imputed features are termed X_1^* and X_2^* . Once mean imputation is done, the next step of PMM is to fit models between the features present in order to generate further imputations.

Because we have complete data we can use established matrix methods to calculate expected regression coefficients ($\hat{\beta}$) and the expected variance of those regression coefficients to show the error introduced by adding an uninformative feature to the model. We define \mathbf{M}_1 as an $n \times 2$ matrix containing a constant intercept column and Y . Then $\beta = (\mathbf{M}_1^T \mathbf{M}_1)^{-1} \mathbf{M}_1^T X_1^*$ is the vector of regression coefficients for the model containing only Y and X_1 . We also define $\sigma_{\epsilon_1}^2 = \frac{\sum_{i=1}^n (\hat{X}_1 - X_1^{true})^2}{n}$ as the imputation error, with X_1^{true} being the true values of X_1 if all were observed and \hat{X}_1 as the predictions of X_1 from our model. Finally, $\text{Var}(\beta) = \sigma_{\epsilon_1}^2 (\mathbf{M}_1^T \mathbf{M}_1)^{-1}$. As a result of our assumptions, the variances and means of all features are independent. This is a result of Basu's theorem and a property of the normal distribution. We use this throughout the calculations below to simplify the expectations. To compute β and $\text{Var}(\beta)$, we first find

$$\mathbf{M}_1^T \mathbf{M}_1 = \begin{bmatrix} n & \sum_{i=1}^n y_i \\ \sum_{i=1}^n y_i & \sum_{i=1}^n y_i^2 \end{bmatrix} = \begin{bmatrix} n & n\bar{y} \\ n\bar{y} & nS_y \end{bmatrix} \quad (6)$$

and its expectation

$$E(\mathbf{M}_1^T \mathbf{M}_1) = \begin{bmatrix} n & 0 \\ 0 & (1+t)n \end{bmatrix}. \quad (7)$$

Using the inverse of this matrix we calculate the expected vector of regression coefficients, β .

$$E(\beta) = E\left(\left(\mathbf{M}_1^T \mathbf{M}_1\right)^{-1} \mathbf{M}_1^T X_1^*\right) = \frac{1}{(1+t)n^2} \begin{bmatrix} (1+t)n & 0 \\ 0 & n \end{bmatrix} \begin{bmatrix} 0 \\ E(S_{y_i x_i^*}) \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{p}{1+t} \end{bmatrix} \quad (8)$$

Using this we calculate the variance of each of the elements of β .

$$E(\text{Var}(\beta)) = \sigma_{\epsilon_1}^2 E\left(\left(\mathbf{M}_1^T \mathbf{M}_1\right)^{-1}\right) = \begin{bmatrix} \frac{\sigma_{\epsilon_1}^2}{1+t} & 0 \\ 0 & \frac{\sigma_{\epsilon_1}^2}{(1+t)n} \end{bmatrix} \quad (9)$$

This completes our analysis of the one feature model. In Eq. 9 we see that the variance of the regression coefficients depends only on n and the variance of the features involved in the model.

Now, suppose we use both X_1 and X_2 in the model predicting Y , and incorporate X_2 into our imputation model. We define \mathbf{M}_2 as an $n \times 3$ matrix containing \mathbf{M}_1 as the first two columns and X_2^* as the final column. Then $\sigma_{\epsilon_2}^2$ becomes the model error of this new model containing X_2^* . β' denotes the vector of regression coefficients for this model with $\text{Var}(\beta')$ denoting the variance of those coefficients. Each of these is calculated in the same way as was done in the model containing only Y and X_1 , but with \mathbf{M}_1 replaced by \mathbf{M}_2 . In the following equations using the same procedure as above we calculate these new features and their expectations.

$$(\mathbf{M}_2^T \mathbf{M}_2) = \begin{bmatrix} n & n\bar{y} & n\bar{X}_2^{obs} \\ n\bar{y} & nS_y & nS_{yx_2^*} \\ n\bar{X}_2^{obs} & nS_{yx_2^*} & nS_{x_2^*} \end{bmatrix} \quad (10)$$

$$E(\mathbf{M}_2^T \mathbf{M}_2) = \begin{bmatrix} n & 0 & 0 \\ 0 & (1+t)n & 0 \\ 0 & 0 & np \end{bmatrix} \quad (11)$$

$$\begin{aligned}
E(\beta') &= E\left(\left(\mathbf{M}_2^T \mathbf{M}_2\right)^{-1} \mathbf{M}_2^T X_1^*\right) \\
&= \frac{1}{(1+t)n^3p} \begin{bmatrix} (1+t)n^2p & 0 & 0 \\ 0 & n^2p & 0 \\ 0 & 0 & (1+t)n^2 \end{bmatrix} E \begin{bmatrix} nX_1^{\bar{obs}} \\ nS_{y_i x_1^*} \\ nS_{x_1^* x_2^*} \end{bmatrix} \\
&= \frac{1}{(1+t)n^3p} \begin{bmatrix} (1+t)n^2p & 0 & 0 \\ 0 & n^2p & 0 \\ 0 & 0 & (1+t)n^2 \end{bmatrix} \begin{bmatrix} 0 \\ np \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} 0 \\ \frac{p}{1+t} \\ 0 \end{bmatrix}
\end{aligned} \tag{12}$$

Note we get the same values as in Eq. 8. Finally, $E(\text{Var}(\beta'))$ is

$$E(\text{Var}(\beta')) = \sigma_{\epsilon_2}^2 E\left(\left(\mathbf{M}_2^T \mathbf{M}_2\right)^{-1}\right) = \begin{bmatrix} \frac{\sigma_{\epsilon_2}^2}{n} & 0 & 0 \\ 0 & \frac{\sigma_{\epsilon_2}^2}{(1+t)n} & 0 \\ 0 & 0 & \frac{\sigma_{\epsilon_2}^2}{np} \end{bmatrix} \tag{13}$$

We see from Eq. 8 and Eq. 12 that the expectations of the regression coefficients do not change with the addition of X_2 to the model. From this we know that the expectation of $\sigma_{\epsilon_1}^2$ is equal to the expectation of $\sigma_{\epsilon_2}^2$. We then can look at Eq. 9 and Eq. 13. We see that the expected variances of all shared coefficients are the same, yet we also see that the expected variance of the coefficient on X_2^* , $\sigma_{\epsilon_2}^2/(np)$, is non-zero. This indicates that in all real cases the magnitude of the coefficient on X_2^* is non-zero. Since the expected variance of this coefficient increases as p decreases, the magnitude of the coefficient increases as p decreases as well. While the variance of the coefficients may be small, a non-zero magnitude in the coefficient results in non-zero changes in the predictions of X_i when performing this step of PMM.

The increase in variance as a result of including a single uninformative feature informs us that imputation models containing uninformative features are likely to be less accurate than those containing only informative features. The lower accuracy leads to possible changes in the third step of PMM, replacing these newly imputed values with the nearest value in X_1^{obs} to the new prediction. A change in the observation chosen would mean a change in the final imputation, and this is likely to be a less accurate imputation than if X_2 was excluded from the model.

UAFS attempts to remove these features prior to imputation in an objective manner to prevent this additional error from being introduced into the model in the first place. Other imputation methods that are not using linear functions

yet still rely on interactions between features may also benefit from UAFS. When considering imputation using, e.g., random forests [10], we also attempt to predict missingness based on present values. The form of the increase in variance would not necessarily be the same or as much as the linear case but would likely still be present to some extent. Even in these more complex machine learning based imputation methods there is a benefit from reducing the number of uninformative features as much as possible, even if that benefit is just in reduced computational complexity.

3 Materials and Methods

To test the effects of UAFS on imputation accuracy and predictive model performance, we employ three different types of data. We examine real data sets with missing values, real data sets that are complete where we induce synthetic missingness, and synthetic data sets. Here we describe the data sets used in our study, procedures for processing or generating data, how we induce synthetic missingness, as well as the implementation details of our imputation methods and predictive methods, and how we measure imputation and predictive performance both with and without using UAFS. Our experimental results are presented in Sec. 4.

3.1 Data Sets

We collected a corpus of 13 real world supervised learning data sets, summarized in Table 1, on which to study the effects of UAFS. Eleven of these data sets were collected from the UCI Machine Learning Repository [11], one was taken from data.gov, and one was provided by the authors of a previous study [12]. Seven data sets were classifications while six were regressions. Further, eight data sets were complete while five had missing values. This corpus allows us to study UAFS in those data with real world missingness, while also letting us measure imputation performance using synthetic missingness for those data that are complete.

In Table 1 we have a summary of the data sets used in our study. We see the name, source, task, output, and whether or not there is real missing data in the data set. In the last column we see the number of observations, n , by the number of features, k , present in the data set. A detailed description of any preprocessing performed on these data sets prior to our experiment is available in Appendix A. All data sets are summarized after preprocessing. The Crime data set notably has a large number of rows removed as described below.

Two data sets serve as illustrative case studies in Sec. 4:

- *Crime*— The Communities and Crime data set [13, 14, 15, 16, 17] has real missingness in the data set, but for our case study all rows containing missing data were removed. (We also consider these data including the rows with missingness as part of our analysis of data sets with real missingness; see Sec. 4.) Additionally, the first five

Data set	Source	Task	Output	Missingness	Size ($n \times k$)
Antivirus	UCI	Class.	Virus or Non	Synthetic	372×504
Musk 1	UCI	Class.	Musk or Non	Synthetic	476×167
Spectrometer	UCI	Class.	Curve Type	Synthetic	531×102
Urban Land Cover*	UCI	Class.	Land Type	Synthetic	675×148
Res. Building	UCI	Reg.	Sales Profit	Synthetic	372×108
Superconduct.	UCI	Reg.	Crit. Temp.	Synthetic	1000×82
Wine Quality	UCI	Reg.	Quality (0-10)	Synthetic	6497×11
2015 IRS 990	Data.gov	Reg.	Total Assets	Synthetic	1000×244
Arrhythmia	UCI	Class.	Arrhythmia Type	Syn./Real	452×280
Crime*	UCI	Reg.	Crime Per Capita	Syn./Real	319×123
Mouse Protein	UCI	Class.	Mouse Type	Real	1080×78
Pima Indians	UCI	Class.	Diabetes Present	Real	768×8
eMinder	[12]	Reg.	Electricity Usage	Real	877×656

Table 1: Data sets used in this study. Case studies (Sec. 4.1) are indicated with *.

features were removed as they are non-predictive identifiers. This reduced the data set from 1994 observations of 128 features to 319 observations of 123 features. The outcome variable to be predicted is the number of violent crimes per 100,000 people. All features and the outcome were normalized to $[0, 1]$ prior to submission to the UCI Machine Learning Repository.

- *Urban Land Cover*— The Urban Land Cover data set [18, 19] contains $n = 675$ observations and $k = 148$ features. The outcome variable represents the land cover type of the images encoded by the data. There are nine types of land cover in the data which we aggregated into two classes, the first consisting of trees, grass, soil, pools, and shadow (342 observations), while the second class consists of concrete, cars, asphalt, and buildings (333 observations).

The remaining 11 data sets are the Detect Malicious Executable (AntiVirus) data set, the Arrhythmia data set [20], the Musk data set (version 1) [21], the Residential Building data set [22], the Low Resolution Spectrometer data set, the Superconductivity data set [23], the Wine Quality data set [24], the IRS 2015 Form 990 data set from data.gov, the Pima Indian Diabetes data set [25], the Mouse Protein Expression data set [26], and the eMinder Crowdsourcing data set [12].

3.2 Synthetic data

With synthetic data sets we have knowledge of the underlying true model. This allows us to measure how well UAFS picks features that are in the true model versus the uninformative features in the model.

We generate synthetic data as follows. First, nk values are drawn from a $N(0, 1)$ distribution, where n is the number of observations and k is the number of features. These values become the elements of the design matrix \mathbf{M} . Of these k

features, k_t , chosen at random, are features that are truly related to the outcome variable Y , while $k - k_t$ are not related to Y . Then k coefficients, C , are drawn from a $N(0, 1)$ distribution:

$$\begin{aligned} \mathbf{M} &= [M_{ij}], \\ M_{ij} &\sim N(0, 1), \\ C_j &\sim N(0, 1). \end{aligned} \tag{14}$$

The outcome variable, Y , is generated by multiplying the truly informative features by the true coefficients, scaling that Y to a $N(0, 1)$ feature, and then adding error term:

$$\begin{aligned} Y^* &= \mathbf{MTC}, \\ Y &= \frac{Y^* - \bar{Y}^*}{\sigma_{Y^*}} + \epsilon, \end{aligned} \tag{15}$$

where $\bar{Y}^* = E(Y^*)$, \mathbf{T} is a diagonal coefficient selection matrix with $k - k_t$ zeros and k_t ones placed at random along the diagonal, and $\epsilon \sim N(0, 1)$ is an error term. In the case of a single true model coefficient and a single uninformative feature this is the model we explored analytically in Sec. 2.1.

3.3 Synthetic missingness

We use the following procedure to induce missingness in both synthetic data sets (Sec. 3.2) and real data sets without missing values (Sec. 3.1). Missingness is induced at random by sampling from the data matrix without the outcome variable and replacing those sampled values with missing points. Specifically, each point of each feature is sampled (observed) independently with probability p ; otherwise, that point is missing with probability $1 - p$. This is done across a range of p allowing us to see well UAFS performs at different levels of missingness.

3.4 Imputation methods

We used the MICE package (v3.3.0) within R to perform imputation [9]. In MICE we used PMM and CART methods. For both methods, the number of multiple imputations that were performed was the same as the percentage of missingness rounded up [27, 28]; default values were used for all other parameters. In order to assess the change in imputation accuracy we imputed both the data set containing the features selected by UAFS and the data set containing all features. The number of imputations for each instance of missingness is equal to the percentage of data that is missing.

It is possible that the PMM or CART method will fail to converge, especially for data with extensive missingness. If this happens, we use the simpler mean imputation method again (see above). Mean imputation does not gain from

the benefits given by removing uninformative features, but there may be a gain when fitting a predictive model to the reduced data set. In this work, the eMinder data set was the only case where we resorted to mean imputation, a problem also encountered during previous studies on those data [12].

3.5 Predictive methods

Following these imputations, predictive models were built and compared. The models were built using the randomForest package in R [29]. For each of the simulations a random forest model was built for each imputation within that simulation. A test set of 20% of the data set was selected prior to building the model, and drawn from the full unimputed data set to prevent information leaking between the test and training set from the imputation process. Models were built on the remaining 80% of the imputed data set. The predictions were averages across all imputations, and then compared to the true values, for both the imputation containing all features and the imputation containing only those features selected by UAFS.

For real world data sets we apply a similar procedure. Each data set is multiply imputed (with the exception of the eMinder data set), and random forests are built on each imputation. Predictions are then averaged over each of these random forests. We then can compare the accuracy of these predictions to one another.

In the eMinder data set we mean imputed the entire data set and then built a single random forest model on either the subset selected by UAFS or the full data. These were then compared for predictive accuracy.

3.6 Evaluating prediction and imputation performance

We measure predictive model performance using RMS error on the test data:

$$\text{Model } RMSE = \left[\frac{1}{N^{\text{val}}} \sum_{j=1}^{N^{\text{val}}} (Y_j - \hat{Y}_j)^2 \right]^{1/2}, \quad (16)$$

where N^{val} is the size of the validation set and \hat{Y}_i is the model prediction for point X_i within the test set.

For those data with synthetic missingness, the true values of X_i^{mis} are known, and we can compare them with imputed values, again with RMS error:

$$\text{Imputation } RMSE = \left[\frac{1}{(k-1)N^{\text{mis}}} \sum_{i=1}^{k-1} \sum_{j=1}^{N_i^{\text{mis}}} (X_{ij}^{\text{mis}} - \hat{X}_{ij}^{\text{mis}})^2 \right]^{1/2}, \quad (17)$$

where there are $k - 1$ features, $N^{\text{mis}} = \sum_i N_i^{\text{mis}}$ is the total number of missing points, N_i^{mis} is the number of missing points in feature i , and $\hat{X}_{ij}^{\text{mis}}$ is the imputed value for the j -th missing point of the i -th feature. In practice, however, to

compare the imputation performance with and without UAFS, we modified Eq. 17 to only sum over observations (i, j) that were imputed in both data sets. Doing so ensures we are comparing the same set of features, but also that we only consider points where the imputation method actually converged, leading to an even comparison between UAFS and non-UAFS performance. Additionally this prevents a feature that is high in variance from contributing unfairly to the RMSE in the full data set when that feature was not imputed in the data set selected using UAFS.

To measure the effect UAFS has on performance, we apply the above RMSE measures on both the full data set and the reduced data set after applying UAFS. We then report the Relative RMSE (RRMSE):

$$RRMSE = \frac{RMSE_{Full} - RMSE_{UAFS}}{RMSE_{Full}}. \quad (18)$$

This RRMSE gives us the average percentage reduction in RMSE relative to using the full data set as a baseline. This choice of RRMSE has an additional benefit: The RMSE itself will have different scales across different data sets depending on the scales of the features in those data sets, so by using this ratio, the RRMSE measure is normalized and results on different data sets can be compared.

Finally, when synthetic missingness is induced, with a fraction $1 - p$ of feature points missing, we consider the $RRMSE(p)$ as a function of p and we also report the Net RRMSE averaged over simulations of missingness and summed over simulated values of p :

$$\text{Net } RRMSE = E \left[\frac{\sum_p RMSE_{Full}(p) - \sum_p RMSE_{UAFS}(p)}{\sum_p RMSE_{Full}(p)} \right] \quad (19)$$

where $E[\cdot]$ denotes average over simulations and the sums run over the values of p used to induce missing (we used the same values of p across all simulations; see Sec. 3.3).

4 Results

The experiments and simulations described in Sec. 3 garner information about improvement in imputation accuracy and predictive accuracy in both simulated and real world data with missingness. We show that UAFS improves imputation accuracy and predictive accuracy in nearly all cases. Additionally, applying UAFS to data with real missingness results in improved or unchanged predictive accuracy in all cases tested.

Table 2 summarizes imputation performance with UAFS on real data with synthetic missingness. In the final three columns we show the net increase in imputation accuracy, the minimum average increase and the p at which that minimum occurred, and the maximum average increase with the p at which that maximum occurred.

Likewise, Table 3 summarizes model performance (predictive accuracy) with UAFS. Random forest models

Data set	Imput. Method	Imputation improvement		
		Min (p)	Max (p)	Net
Antivirus	CART	-0.5% (0.25)	6.7% (0.40)	-0.5%
Musk 1	PMM	-24.9% (0.75)	46.0% (0.30)	21.0%
Spectrometer	PMM	-1.6% (0.65)	32.8% (0.20)	11.5%
Urban Land Cover	PMM	-27.9% (0.95)	47.1% (0.19)	14.6%
Residential Building	CART	-1.9% (0.85)	1% (0.20)	-0.9%
Superconductor	PMM	-11.7% (0.55)	1.5% (0.20)	-1.5%
Wine Quality	PMM	-6.5% (0.40)	1.6% (0.95)	-3.2%
2015 IRS 990	CART	-7.8% (0.30)	2.8% (0.25)	-5.2%
Arrhythmia	PMM	47.1% (0.25)	73.4% (0.95)	56.4%
Crime	PMM	-1.7% (0.93)	43.3% (0.34)	18.3%

Table 2: Imputation performance on synthetic missingness in 10 data sets.

Data set	Imput. Method	Model improvement		
		Min (p)	Max (p)	Net
Antivirus	CART	-3.2% (0.20)	4.3% (0.25)	-0.1%
Musk 1	PMM	-1.7% (0.75)	11.8% (0.25)	3.8%
Spectrometer	PMM	-2.4% (0.35)	0.7% (0.20)	-0.4%
Urban Land Cover	PMM	0.2% (0.58)	17.7% (0.16)	3.7%
Residential Building	CART	-0.4% (0.90)	4.4% (0.20)	0.7%
Superconductor	PMM	-0.7% (0.56)	1.7% (0.20)	0.2%
Wine Quality	PMM	-2.2% (0.20)	-0.3% (0.90)	-1.1%
2015 IRS 990	CART	-1.7% (0.20)	3.5% (0.55)	0.8%
Arrhythmia	PMM	-8.7% (0.35)	4.6% (0.20)	-0.3%
Crime	PMM	-0.4% (0.90)	15.9% (0.16)	5.4%

Table 3: Model performance on synthetic missingness in 10 data sets.

(Sec. 3.5) were built on data sets with or without first using UAFS, and their validation errors were compared (Sec. 3.6).

In Table 4 we see the effect of imputation on data sets with real missingness. Two of these data sets, the Crime and Arrhythmia data sets, are also in Table 2. As such we can compare the overall effect of various levels of missingness with the fixed percentage of missingness actually present in the data.

Data set	Imput. Method	% Missing	Model Improvement
Arrhythmia	PMM	5.1%	3.7%
Crime	PMM	15%	<.001%
Pima Indians	PMM	7.2%	0%
Mouse Protein	PMM	1.6%	0%
eMinder	Mean	81.7%	1.4%

Table 4: Performance due to UAFS on data sets with real missingness.

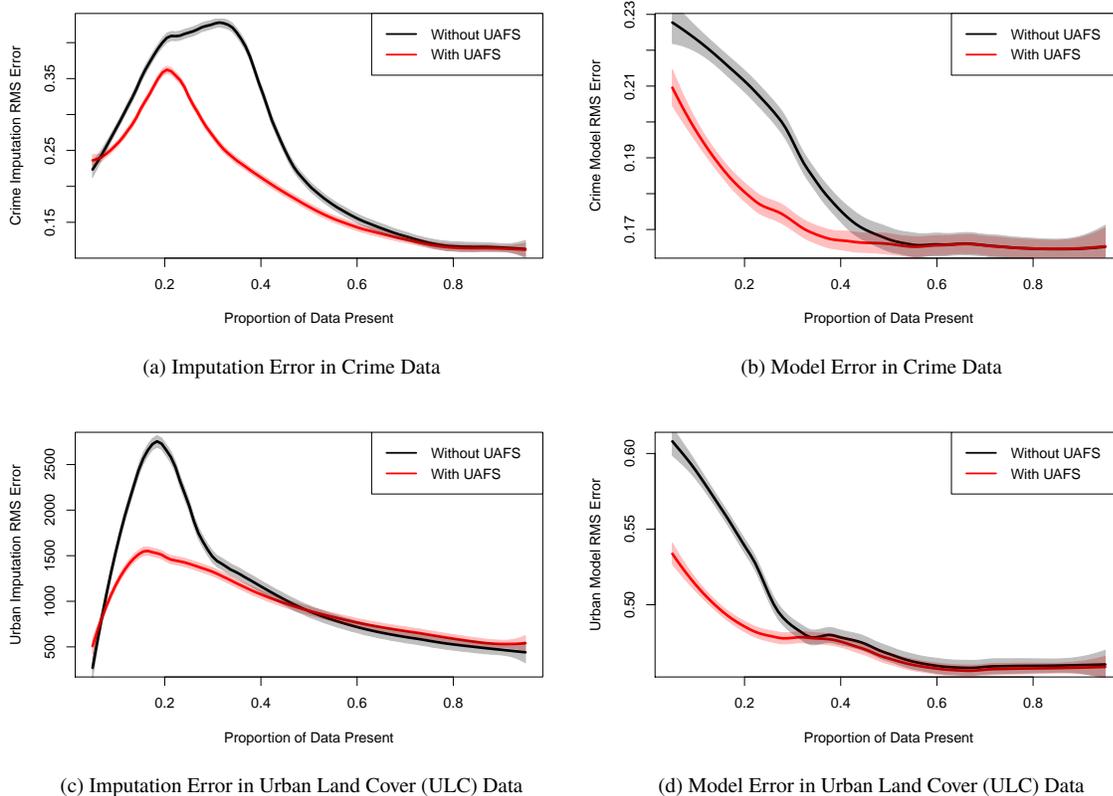


Figure 3: Using UAFS reduces both imputation and model (prediction) error in our two case study data sets. Improvement is especially evident at high levels of missingness (lower values of p). The non-monotonic behaviour for $p < 0.2$ is an artifact of our RMSE measure (Sec. 3) due to imputation convergence issues for (a) and (c). This trend is explored further in Fig. 4.

4.1 Case studies

Two examples of how the difference in imputation accuracy changes over the range of missingness are displayed in Fig. 3. There is a trend in the magnitude as the amount of missingness changes. When the quantity of data present is above 50% the amount of improvement gained from using UAFS in these cases is minimal, but below this threshold there is a large gain in imputation accuracy.

In Fig. 3(a) and (b) we see how an increase in imputation accuracy translates into an improvement in predictive accuracy in a random forest model. Fig. 3b shows the predictive accuracy of a model built to predict violent crimes per capita in the Crime data set. Here, model accuracy follows the same trend as seen in Fig. 3a: there is little gain in model accuracy at less than 50% missingness (but also no loss), while at greater than 50% missing there are significant gains in performance. This is mirrored in Fig. 3(c) and (d).

When looking at the trend in imputation accuracy as missingness is adjusted from $p = 0.05$ (high missingness) to $p = 0.95$ (low missingness) we can see that we gain the most improvement in imputation accuracy when the imputation

without UAFS starts to quickly gain error. In Fig. 3 we can see that this starts to occur around $p = 0.50$. When there is less missingness in the data than this threshold it appears that there is no gain in imputation accuracy from using UAFS. In these cases the gain from UAFS is that the computation takes less time due to fewer features being included in the imputation model. We also see no change in predictive accuracy when we are in the range where we do not see much improvement in imputation accuracy. Looking at Fig. 3 we can see that even in the cases where we do not gain imputation accuracy our model accuracy remains the same. We suspect this is because we did not lose highly predictive features using UAFS. As we see in Fig. 2 we expect to correctly select nearly all of the truly predictive features when using UAFS on data sets containing greater than 15% of data in each column, which we are well above in most cases.

Below this threshold of approximately 15% we begin to see apparently strange trends in imputation accuracy. When looking at Fig. 3(a) and (c) we begin to see overall error drop back down after this point. The reason for this change is because UAFS is selecting fewer features to include in the imputation overall, and no longer selecting many features that were used at lower levels of missingness. In Fig. 4 we see the effective selection rate by UAFS. This shows that once we reach these threshold rates of missingness around 15% we are effectively using a very small proportion of the features, and this accounts for the non-monotonic trend in imputation accuracy.

The way we are comparing imputation accuracy only includes the values that are imputed in both data sets. This means that as we remove more and more data from our data sets we reach a point where we have very little data that is actually being imputed, and the data that is being imputed is the data that we're most clear about the relationship, and thus can more accurately impute. We can see in Fig. 3 that as we approach these exceedingly high levels of missingness that imputation accuracy goes down, but model error continues to rise. With UAFS we still reduce the model error when compared to simply imputing the entire data set, but the results are similar to having 10% more data in the data set in terms of error reduction.

4.2 Synthetic data

Recall Fig. 2 shows how feature selection accuracy changes in simulated data with missingness ranging from $p = [0.1, 0.99]$. We can see that selection accuracy peaks and remains approximately the same once 15% of values are observed. The selection of features that are not truly informative remains at approximately 5% of the time throughout the full range of missingness. This indicates that UAFS is doing well in selection, but this does not tell us the actual effect this has on imputation.

In order to see the effect of UAFS on imputation we perform a sweep of both missingness, as described above, and we sweep across the number of uninformative features. In other words, for a given level of missingness p , we fix the value of informative features at 5 and sweep from 5 uninformative features to 95 uninformative features. We keep the

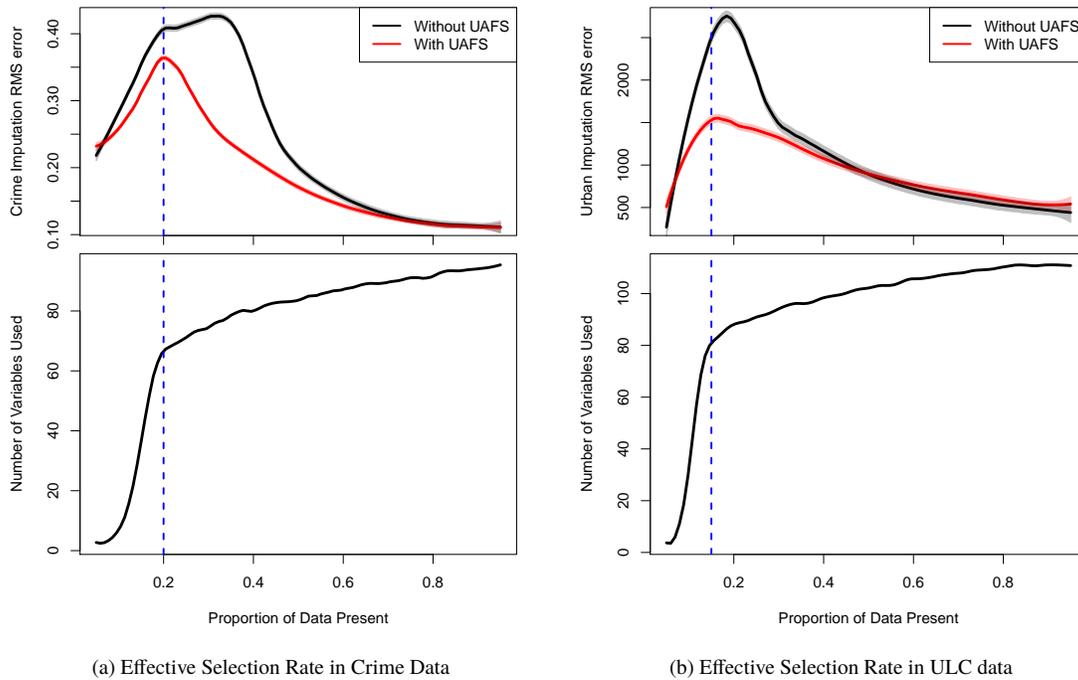


Figure 4: Imputation error and feature selection. The imputation error begins to drop for high missingness precisely when the number of features used begins to drop precipitously.

number of observations constant at 200. Each of these data sets is generated independently and 100 different levels of missingness are induced for each data set.

Results for the synthetic data with 200 observations, 5 informative features, and a varying number of uninformative features, are displayed in Fig. 5. In the upper plot we see the trends at we see the effect averaged across all levels of missingness. We can see that there is an increase in the improvement in using UAFS as the number of uninformative features increases. The black linear regression line serves to guide the eye. We see that this net imputation improvement linearly increases with the number of uninformative features, meaning that UAFS offers higher improvements when data sets contain more uninformative features. In the lower plot of Fig. 5 we see the trends at three different levels of missingness. The lines in these plots are LOESS fits to the data with a span of 0.25 and a 95% confidence interval around them. All levels of missingness benefit from performing UAFS prior to imputation, but those data with more missingness benefit more from UAFS.

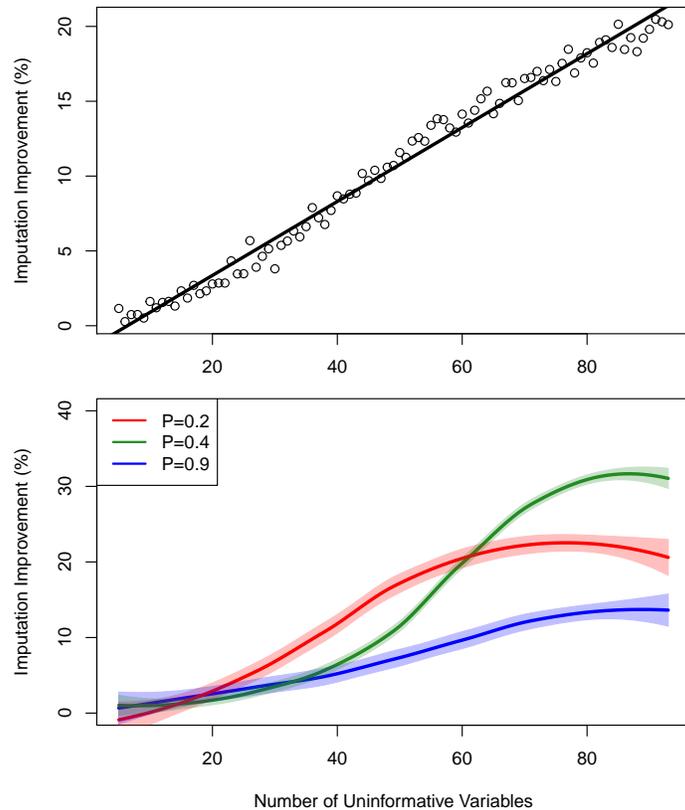


Figure 5: For a fixed set of 5 informative variables and an increasing set of n uninformative variables, UAFS improves imputation performance. While we would expect the presence of uninformative features to have little or no effect on the imputation because they are uninformative of both the outcome and the informative features, instead we see that removing them leads to a significant improvement in imputation accuracy.

5 Discussion

The experiments described above show that using UAFS as a processing step prior to imputation consistently results in increased accuracy in imputation. The increase in imputation accuracy usually results in increased predictive accuracy when generating models from this newly imputed data, but this is not always true. Looking at the last three columns of both of these tables we see that at some levels of missingness there is actually a reduction in performance. These results can be seen clearly when we look at Tables 2 and 3. We note that in most cases the largest reduction occurs at high levels of missingness. Our recommendation is that UAFS should be applied in cases with large numbers of features relative to the number of observations and in cases where missingness exceeds 40% of the data set. It also can be applied in most cases with less than 40% missingness or fewer features to speed up the time it takes to impute the data set, although this is not guaranteed to improve predictive power. The maximum reduction in predictive power across these ten data sets was 8.7% in the Arrhythmia data set with 65% of the data missing.

One outstanding data set in terms of lack of increase in prediction is the Wine Quality data set. The Wine Quality data set contains 11 features and nearly 6500 observations. The gain in predictive accuracy from removing one of these features is likely to be low at best because even at high levels of missingness there are still many shared observations and therefore the true model can be estimated well. Yet, even in this case we only lose 1.1% predictive accuracy overall. Looking at Table 2 we see that at some levels of missingness there is actually an increase in overall accuracy: at $p = 0.95$ there was an increase of 1.6%. This indicates that even in cases where UAFS is applied inappropriately the loss of information is low, and there can still be an increase in accuracy.

In the data sets with real missingness (Table 4), none had a reduction in predictive accuracy due to applying UAFS, and three had measurable improvement. The two cases where no model improvement was observed was the Pima Indian Diabetes data set and the Mouse Protein data set. In the Pima Indian Diabetes data set all features were selected by UAFS. In the Mouse Protein data set the exact same classifications were made in both models. We believe that some improvement or no change will occur in nearly all cases, but care should be taken to compare models as opposed to assuming model improvement has occurred.

Through our simulations (Fig. 3b and d.) we find that UAFS can enable predictive model on data with nearly 70% missingness without losing predictive power in a case with only 319 observations, extending the amount of missingness that can be tolerated without loss of accuracy by an additional 10–15%. This same increase in predictive accuracy does not occur in all cases, but this pattern is common. When looking at the data sets that benefit most from this method, as shown in Table 1 and Table 2, we see that the features that benefit most contain a high ratio of features to observations. As the number of features increases relative to the number of observations we gain a greater relative amount of accuracy, likely due to a reduction in the effectiveness of multiple imputation with all features. Using UAFS to reduce these features prior to imputation helps solve this problem.

In Fig. 5 we can see that our simulations show that the gain in imputation accuracy should increase as the number of features increases, and this gain is large and mostly linear. We suspect that much of the difference between these simulations and the real data is covariance present between variables which we do not simulate. [30] suggests that the removal of auxiliary features prevents bias in regression estimates. This seems to coincide with our methods. Hardt suggests that there should be at most one feature for every three observations in the imputation model, and those data sets that benefit the most from UAFS have greater than one feature for every three observations in the model. This can be seen by comparing the dimensions of the data set in Table 1 to the imputation performance in Table 2. This needs further exploration in order to fully understand exactly what is occurring.

In Table 4 we see the performance on five data sets with real missingness. Most of these missingness values are fairly low, so the model improvement that we see is also fairly low. Of note is the Arrhythmia data set, which sees

3.7% model improvement at a missingness of only 5.1%. The Arrhythmia data set also had the highest net gain in imputation performance, and one of the higher net gains in model performance, as can be seen in Table 2 and Table 3. The Arrhythmia data set has a different pattern in imputation accuracy gain than other data sets. It has the highest level of imputation gain when there is not much missingness, gaining 73.4% accuracy over base when only 5% of the data set is missing. This may be the result of large amounts of confounding in this data set. The 3.7% increase is not strange considering the results of the simulated missingness.

The eMinder data set also had a large increase in model improvement, but perhaps not as large as we would expect given that it has 81.7% missingness. This is because the eMinder data set has such extensive missingness that applying PMM or even CART imputation fails. Instead of applying either of these methods mean imputation was used. Mean imputation does not gain accuracy from UAFS in the same way other imputation methods do. The gain in model accuracy that we see in eMinder comes from a smaller set of features leading to a more concise and accurate random forest model. This model is also not being confounded by features with large amounts of observations that are the same number as UAFS is biased towards features with more observations.

UAFS uses a straightforward method to select features for the imputation model, but other selection methods may also be used within the same framework. There are some LASSO methods that are able to build models with missing data may also be used to reduce dimensionality prior to imputation similar to UAFS. These methods, such as HMLASSO [31], could also be used to select features in a similar way to UAFS prior to imputation. The benefit of these methods needs to be explored further.

One possible extension of UAFS is using Spearman's ρ instead of Pearson's r as a measure of relatedness between features. In many cases the rank correlation is a better measure of relatedness than the simple correlation. A benefit of this would be that X_i that are related to Y in a nonlinear fashion would be more likely to be detected and included in the imputed data set. When using Spearman's ρ , imputation methods other than PMM may need to be used to take full advantage of the nonlinear relations of the features being imputed. CART imputation, as was used in the Antivirus, Residential Building, and IRS data sets, may take advantage of these nonlinear but monotonic relationships.

UAFS can also be extended into unsupervised learning problems. Instead of picking a single feature which we use to supervise, we can apply the same algorithm to each feature separately, but with a more stringent p-value at which to select features. We can build individual imputation models for each feature using this method, and then recombine this data back into a single data set. More experiments need to be done in order to determine the benefits of this method relative to simply imputing the entire data set prior to standard unsupervised learning methods being applied.

In summary, UAFS increases predictive accuracy and imputation accuracy in many cases. In real world data sets UAFS can be used to maintain the same level of imputation accuracy at even higher levels of missingness than could

previously be attained for a whole data set. UAFS improves imputation the most for data sets with high dimensionality and high levels of missingness. Even in other data sets, UAFS can be used to reduce computational time and keep predictive power at the same level. With UAFS we can further push back the threshold of missingness at which we can be confident that our data analysis is providing reasonable results.

6 Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. IIS-1447634.

References

- [1] S. F. Messner, “Exploring the consequences of erratic data reporting for cross-national research on homicide,” *Journal of Quantitative Criminology*, vol. 8, no. 2, pp. 155–173, 1992. [1](#)
- [2] J. Scheffer, “Dealing with missing data,” *Research Letters in the Information and Mathematical Sciences*, vol. 3, pp. 153–160, 2002. [1](#)
- [3] P. H. Rezvan, K. J. Lee, and J. A. Simpson, “The rise of multiple imputation: a review of the reporting and implementation of the method in medical research,” *BMC Medical Research Methodology*, vol. 15, Jul 2015. [1](#)
- [4] M. J. Azur, 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, no. 1, pp. 40–49, 2011. [2](#)
- [5] S. V. Buuren and K. Groothuis-Oudshoorn, “MICE: Multivariate imputation by chained equations in R,” *Journal of Statistical Software*, vol. 45, no. 3, 2011. [2](#)
- [6] L. M. Collins, J. L. Schafer, and C.-M. Kam, “A comparison of inclusive and restrictive strategies in modern missing data procedures,” *Psychological Methods*, vol. 6, no. 4, p. 330–351, 2001. [2](#)
- [7] R. N. Davidson, “The decomposition of correlation coefficients into causal effects,” *Environment and Planning A: Economy and Space*, vol. 9, no. 7, pp. 805–812, 1977. [4](#)
- [8] S. van Buuren, *Flexible Imputation of Missing Data*. Chapman and Hall/CRC, 2012. [7](#)
- [9] S. van Buuren and K. Groothuis-Oudshoorn, “mice: Multivariate imputation by chained equations in r,” *Journal of Statistical Software*, vol. 45, no. 3, pp. 1–67, 2011. [7](#), [14](#)
- [10] L. Breiman, “Random forests,” *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001. [12](#)
- [11] D. Dheeru and E. Karra Taniskidou, “UCI machine learning repository,” 2017. [12](#)
- [12] M. D. Wagdy, J. C. Bongard, J. P. Bagrow, and P. D. H. Hines, “Crowdsourcing predictors of residential electric energy usage,” *IEEE Systems Journal*, 2017. [12](#), [13](#), [15](#), [26](#)
- [13] M. Redmond and A. Baveja, “A data-driven software tool for enabling cooperative information sharing among police departments,” *European Journal of Operational Research*, vol. 141, no. 3, p. 660–678, 2002. [12](#)
- [14] F. B. of Investigation, “Crime in the united states,” 1995. [12](#)
- [15] B. of the Census, “Census of population and housing 1990 united states: Summary tape file 1a & 3a,” 1990. [12](#)

- [16] B. of Justice Statistics, “U.s. department of justice, bureau of justice statistics, law enforcement management and administrative statistics,” 1992. [12](#)
- [17] B. of the Census, “U.s. department of commerce, bureau of the census producer, washington, dc and inter-university consortium for political and social research ann arbor, michigan,” 1992. [12](#)
- [18] B. Johnson and Z. Xie, “Classifying a high resolution image of an urban area using super-object information,” *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 83, p. 40–49, 2013. [13](#)
- [19] B. A. Johnson, “High-resolution urban land-cover classification using a competitive multi-scale object-based approach,” *Remote Sensing Letters*, vol. 4, no. 2, p. 131–140, 2013. [13](#)
- [20] H. A. Guvenir, B. Acar, G. Demiroz, and A. Cekin, “Supervised machine learning algorithm for arrhythmia analysis,” *Computers in cardiology*, pp. 433–436, 1997. [13](#)
- [21] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, “Solving the multiple instance problem with axis-parallel rectangles,” *Artificial intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997. [13](#)
- [22] M. H. Rafiei and H. Adeli, “A novel machine learning model for estimation of sale prices of real estate units,” *Journal of Construction Engineering and Management*, vol. 142, no. 2, p. 04015066, 2015. [13](#)
- [23] K. Hamidieh, “A data-driven statistical model for predicting the critical temperature of a superconductor,” *arXiv preprint arXiv:1803.10260*, 2018. [13](#)
- [24] P. Cortez, A. Cerdeira, F. Almeida, T. Matos, and J. Reis, “Modeling wine preferences by data mining from physicochemical properties,” *Decision Support Systems*, vol. 47, no. 4, pp. 547–553, 2009. [13](#)
- [25] R. S. Johannes, “Using the adap learning algorithm to forecast the onset of diabetes mellitus,” *Johns Hopkins APL Technical Digest*, vol. 10, pp. 262–266, 1988. [13](#)
- [26] C. Higuera, K. J. Gardiner, and K. J. Cios, “Self-organizing feature maps identify proteins critical to learning in a mouse model of down syndrome,” *PloS one*, vol. 10, no. 6, p. e0129126, 2015. [13](#)
- [27] T. E. Bodner, “What improves with increased missing data imputations,” *PsycEXTRA Dataset*, Oct 2008. [14](#)
- [28] P. Royston and I. White, “Multiple imputation by chained equations (MICE): Implementation in Stata,” *Journal of Statistical Software*, vol. 45, no. 4, 2011. [14](#)
- [29] A. Liaw and M. Wiener, “Classification and regression by randomforest,” *R News*, vol. 2, no. 3, pp. 18–22, 2002. [15](#)
- [30] J. Hardt, M. Herke, and R. Leonhart, “Auxiliary variables in multiple imputation in regression with missing x: a warning against including too many in small sample research,” *BMC medical research methodology*, vol. 12, no. 1, p. 184, 2012. [22](#)
- [31] M. Takada, H. Fujisawa, and T. Nishikawa, “Hmlasso: Lasso for high dimensional and highly missing data,” *arXiv preprint arXiv:1811.00255*, 2018. [23](#)

A Data set preprocessing

All 13 data sets used in this study were processed after retrieving them from their sources. In this appendix we give a closer view of how we did this processing for each of these data sets.

In all data sets columns containing unique identifiers were removed. The communities and crime data set as well as the urban land cover data set were processed as described in 3. The communities and crime data set received no processing prior to PMM imputation when looking at the effect of UAFS on the model accuracy with real missingness.

The eMinder data set as we received it from [12] needed the removal of several columns of the data. These columns were questions proposed that were not accepted into the online survey, and thus contained no observations. Other than the removal of these columns no preprocessing was done prior to mean imputation.

The Pima Indian diabetes data set has zeros in place of NaN in the columns of data. These zeros were replaced with NaN prior to PMM imputation. No other preprocessing was done to this data set. The Pima Indian diabetes data set is no longer available from the UCI machine learning data base, but is available from other sources such as Kaggle.

The Arrhythmia data set is unclear on which values are missing and which are intentionally zeros. For the synthetic missingness study it was assumed that the only missing values were the ones marked as NaN in the data set. Any columns that had no variance, such as those which only contained one value, were removed. After this any rows that still had missingness were removed. For the real missingness study it was assumed that for columns numbered 6,16,17,18,19,20,21,28 through 167, and 170 through 279 the presence of a zero meant that there had not been an observation made. We removed these zeros and replaced them with NaN prior to PMM imputation.

Six of the data sets had no cleaning needed prior to simulation studies. These are the residential building data set, the wine data set, the spectrometer data set, the virus data set, the musk 1 data set, and the mouse protein data set.

The IRS and Superconductor data set were not used in their entirety. Instead, a sample of size 1000 was taken from them. This was done using a seed of 2018 for reproducibility. The size of the full original data sets, 307,483 observations and 21,263 observations, respectively, meant that doing simulation studies using the full data set would have taken a prohibitively large amount of computational resources. In the IRS data set all categorical features were converted into sets of 0/1 dummy variables.