Prediction-Oriented Methods in Handling Missing Data

By

CYNTHIA MASCARENHAS
THESIS

Submitted in partial satisfaction of the requirements for the degree of

MASTER OF SCIENCE

in

Computer Science

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Norman Matloff, Chair

Mohammad Sadoghi

Miles Lopes

Committee in Charge

2026

ACKNOWLEDGEMENTS

This Thesis would not have been possible without Dr. Norman Matloff's immense support and guidance. Dr. Norm initially came up with this idea and working with him was so exciting! He has shaped not just my contributions to this work, but my understanding of what it means to live the research life and my love for it. I would love to thank my teammates – Billy and Alekya for all their important contributions, I could not have done this without them! A heartfelt thank you to Prof. Mohammad Sadoghi and Prof. Miles Lopes for agreeing to serve on my thesis committee, generously giving their time to review this work, and offering thoughtful comments that proved invaluable in making it better. Thank you to the GGCS Advising Staff for answering my long list of logistics questions with patience.

Thank you to my partner Technical Sergeant Jason Willam-Lentz Bel Nova for being the best ally and for always lifting my spirits during moments of burnout, especially when I was being too hard on myself over outcomes that, in hindsight, were never short of remarkable. Finally, thank you to my younger self for surviving those years so that I can now live and finally experience what happiness feels like.

STATEMENT OF CONTRIBUTION

The work presented in this thesis is based on a collaborative research project conducted under the supervision of Dr. Norman Matloff. While the underlying research and its documentation were carried out jointly in the form of a paper, the thesis itself was independently drafted by the author of this thesis, Cynthia Mascarenhas.

Cynthia Mascarenhas contributed to the implementation, experimental aspects and writing of the joint paper. Specific contributions include implementing and refining the `lm_prefill` function for CC, MICE, Amelia, and missForest, as well as the `lm_tower` function. Conducted end-to-end testing of the package, including the bootstrap function, and documented identified bugs. For the paper, authored Sections 2 and 8, edited the manuscript in its entirety, and addressed reviewer comments. The thesis manuscript was subsequently drafted independently by the author, drawing upon and expanding the joint paper.

Billy Ouattara contributed to the implementation and experimental aspects of the project. Implemented and refined the `lm_ac` components and developed the bootstrap functions used for benchmarking. Conducted experiments across multiple datasets to compare the methods introduced in the paper. Contributed to Sections 1, 4, 5, 7, and 9, participated in revision, and prepared and submitted the `mvPred` package to CRAN.

Alekya Veluri contributed to the experimental and implementation aspects of the package, specifically the refinement and testing of the `lm_prefill` function. Conducted testing of the package and bootstrap function across multiple datasets and identified and resolved bugs. Authored Sections 6.3 and 6.4 and participated in revision.

Dr. Norman Matloff conceived the project idea, supervised the research, and guided the methodology and analysis. Provided feedback throughout the development of both the package and the paper.

DEDICATION

To Dr. Norman Matloff, for believing in me, to all the resilient problem solvers out there and anyone who has carried the team without any title.

TABLE OF CONTENTS

List of Tables

Abstract

Missing data is a ubiquitous problem in predictive modeling, but the majority of current methods are designed for statistical estimation rather than prediction, leaving the comparative behavior of imputation-based and non-imputational strategies in prediction-oriented settings poorly understood. This paper introduces `mvPred`, an open-source R program for prediction under missing data in supervised regression. The package includes three complementary paradigms: imputation-based prefilling (`lm_prefill`), pairwise available-case estimation (`lm_ac`) and the Tower technique (`lm_tower`), which in turn is based on the Tower Property of conditional expectation and aims to predict accurately. `lm_prefill` supports four techniques, including complete-case analysis, MICE, Amelia and the missForest algorithm, ranging from list-wise elimination to model-based and nonparametric imputation approaches. We perform a rigorous empirical evaluation on five benchmark datasets with different sample sizes, missingness rates and properties of the response variable, using 5-fold cross-validation and four conventional predictive metrics. Our evaluation shows that the performance of a method depends heavily on the dataset characteristics and no single approach universally dominates all dataset combinations, thus motivating the necessity for a uniform benchmarking methodology. Practical guidelines for selecting a method depending on dataset features are also presented.

The package is publicly available at https://github.com/matloff/mvPred

1. Introduction

Predictive modeling requires complete and high-quality data to identify underlying patterns and generalize unforeseen instances. Missing values are a prevalent problem in real-world datasets, which reduces the effective sample size and might introduce bias. Conventional methods like complete-case (CC) analysis remove observations with missing features, therefore losing potentially critical variability and degrading the model performance. Training on datasets that are smaller or not representative leads to a higher risk of both overfitting and underfitting and eventually reduces prediction reliability[8]. Several imputation-based strategies have been proposed to alleviate the effect of missing data. Multiple imputation by chained equations (MICE) is an iterative approach where each variable with missing values is modelled as a function of the other variables, creating numerous completed datasets to capture imputation uncertainty [2]. Amelia produces imputations using a bootstrap based Expectation-Maximization algorithm [6] under the multivariate normal assumption and uses a joint modeling framework. While these methods are beneficial for statistical inference, they are mostly preprocessing tools and do not provide support for prediction when missing characteristics are present for new instances.

Prediction-oriented approaches aim to forecast from partial data without explicitly imputing, unlike imputation-based methods. One such approach is *towerNA* which is a non-imputation approach based on regression averaging, which uses the Tower Property of conditional expectation. This property allows towerNA to provide predictions for observations with missing predictor values without explicit imputation. The tower technique can thus help to lessen the potential bias introduced by imputation models. However, its utility still depends on the data structure and the pattern of missingness [10]. This method is particularly useful in prediction contexts when imputation is difficult, or when imputation models may contribute extra complexity or bias, especially when the imputation model is intrinsically ambiguous. Prediction-oriented approaches like towerNA are especially effective for prediction tasks when the goal is to give credible predictions rather than statistical inference, and computational simplicity is a primary priority [3]. Another popular prediction-based method is the available-cases (AC) regression. The AC regression estimates the model parameters based on the available data for each case and does not impute missing values. AC regression retains some information in partially observed settings by calculating the normal equations of regression models from pairwise-available observations. This works best when the missingness in the dataset is sparse and the missing values are highly correlated, allowing the model to use incomplete data without losing predictive performance [16]. This approach has been shown to perform well in situations where the missing data follow a *Missing at Random* (MAR) mechanism, in which correlations between the missing and observed data serve to retain the predictive ability of the model [15]. All these

approaches have their own advantages. The existing literature has rather been focused on the evaluation of methods in a vacuum, and often in a statistical inference framework rather than in predictive modeling. Especially, the comparison of missing data approaches in real-world predictive workflows is underexplored. Moreover, there are no integrated tools to directly apply and benchmark these algorithms for predictive tasks. This leaves the practitioners with the problem of identifying and comparing the most effective missing data management methods in the context of predictive modeling. This gap refers to the need of embedding software tools that allow users to apply, evaluate and assess a multitude of missing data techniques in a seamless fashion, thus increasing the robustness and reliability of prediction models.

1.1. Contributions

This work shifts the focus from traditional approaches that prioritize parameter recovery under missingness to an emphasis on predictive performance. In particular, we assess the generalization performance of models trained with different missing-data techniques on new observations. We compare these strategies for continuous outcomes in a linear regression model. We seek to give practitioners a detailed grasp of the trade-offs of alternative missing-data management approaches through this evaluation. Moreover, our `mvPred` R package supports decision making by providing a single tool for applying, comparing and choosing adequate missing-data techniques by predictive accuracy.

2. Related Work

For a long time, the management of missing-data in predictive modeling has been a challenge in the fields of statistics and machine learning. Several methods have been developed, ranging from the simple deletion to sophisticated imputation and prediction-oriented tactics. This section reviews notable articles that have influenced the subject, with an emphasis on their contribution to predictive modeling.

In [8], the authors provided a comprehensive treatment of missing data mechanisms -- Missing Completely at Random (MCAR), Missing at Random (MAR) and Missing Not at Random (MNAR) and analyzing their impacts on estimation and inference. The authors demonstrated that simplistic strategies such as deletion (e.g. complete-case analysis) may lead to biased estimators and loss of statistical efficiency when data are not MCAR. Their work established theoretical foundations that guide the development and evaluation of more advanced techniques and highlights the importance to align missing-data approaches with the fundamental mechanisms that drive missingness in predictive contexts.

The authors of [20] introduced Multiple Imputation by Chained Equations (MICE), a widely adopted approach that generates multiple imputed datasets by iteratively modeling each incomplete variable conditional on others. The method is particularly flexible in that it may be applied to a number of types of variables (continuous, binary, ordinal) and complex data structures. The authors showed that MICE outperforms single imputation for parameter estimation and uncertainty quantification. MICE is traditionally employed in inferential studies, yet its use in prediction workflows has increased and many applications use MICE imputations as input to subsequent predictive models.

In [11], a detailed experimental survey of 19 imputation algorithms, including classical statistical methods (e.g., mean/mode imputation), machine learning models (e.g. k-NN, missForest) and deep learning techniques (e.g. neural network-based imputers) are conducted. They conduct extensive experiments on 15 real-world benchmark datasets under varying missingness mechanisms, proportions of missing data, and data types. Additionally, the study evaluates not only imputation accuracy but also the impact of imputation on downstream predictive performance. This provides practical insights into which methods yield robust predictions. Their findings demonstrate that algorithm performance varies markedly across settings, motivating tailored method selection for specific predictive tasks.

In [4] the authors present a survey of missing data strategies in machine learning context, extensively comparing old imputation methods and modern predictive approaches. They discuss approaches including k-Nearest Neighbors (k-NN) imputation, matrix factorization, and ensemble-based imputers, highlighting their benefits, shortcomings, and common use cases.

They also point out issues such as computational cost, model sensitivity to the pattern of missingness, and the need for handling heterogeneous data types. This survey emphasizes that there is no one best imputation approach, but rather that performance is dependent on dataset properties and modeling objectives, thus further motivating the need for comparative evaluation tools.

In [12], the authors systematically investigate how missing data are handled and reported in machine-learning based clinical prediction model studies. Their review reveals that a significant proportion of published models rely on complete-case analysis or poorly documented imputation practices, with many studies failing to justify their approach to missing data. The authors show that inadequate handling and reporting can undermine model validity and reproducibility, particularly in high-stakes domains like healthcare. Their work highlights a gap between methodological best practices and real-world usage, emphasizing the necessity for clearer guidelines and standardized evaluation in predictive-modeling research.

In [13], the authors benchmark missing-data strategies in the context of predictive modeling in health databases, looking at workflows focused on imputation and models that are inherently missing-data tolerant. The authors compare the traditional imputation followed by model training with methods where models (e.g., tree-based learners) handle missing values directly during training and prediction. Their results show that algorithms with built-in missingness handling can outperform or be competitive with sophisticated imputation pipelines in terms of predictive accuracy and computational efficiency, in particular on large, heterogeneous datasets. This result indicates that prediction-focused techniques that incorporate missing-data handling in the learning algorithms can be beneficial for real-world predictive problems.

3. Background

Throughout the following sections, we present several key concepts that will be used throughout the rest of the work to ensure clarity, consistency and a shared understanding of the terminology employed.

3.1. Missing Data Types

In this subsection, we classify and describe the many types of missing data that are commonly found in datasets. Understanding these distinctions is important since the missing-data mechanism can significantly affect the choice of analytical procedures and the interpretation of results.

Let Y be the entire data vector and R be the missingness indicator, where $R_i = 1$ if Y_i is observed and $R_i = 0$ if Y_i is missing. The three canonical forms of missingness are defined as follows.

3.1.1. Missing Completely at Random (MCAR)

The missingness of a data point is independent of both observed and unobserved data:

$$P(R \mid Y) = P(R)$$

MCAR assumes that the occurrence of missing values is completely random. Analyses conducted on the observed data alone remain unbiased.

3.1.2. Missing at Random (MAR)

In this case, missingness may depend on observed data but not on the missing values themselves:

$$P(R \mid Y) = P(R \mid Y_{\text{observed}})$$

Under MAR, valid inference is possible by appropriately conditioning on the observed data. Many standard imputation and likelihood-based methods assume MAR.

3.1.3. Missing Not at Random (MNAR)

The probability of missingness depends on the unobserved, missing values themselves:

$$P(R \mid Y) \text{ depends on } Y_{\text{missing}}$$

For MNAR, ignoring the missing-data mechanism can introduce bias, and explicit modeling of the missingness is required for valid inference.

3.2. Linear Regression

The proposed R package is designed for linear regression and therefore, it is essential to establish the fundamental concepts underlying this methodology.

Linear Regression constitutes a statistical framework for representing the relationship between a dependent variable Y and one or more independent variables $X_1, X_2, \dots, X_p$. The model assumes a linear relationship of the form:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \epsilon,$$

where $\beta_0, \beta_1, \dots, \beta_p$ corresponds to the regression coefficients, and ϵ represents the error term, typically assumed to be normally distributed with zero mean and constant variance.

The regression coefficients are estimated via the Ordinary Least Squares (OLS) method, which minimizes the sum of squared residuals:

$$\hat{\beta} = \arg\min_{\beta} \sum_{i=1}^{n} \left(Y_i - \beta_0 - \sum_{j=1}^{p} \beta_j X_{ij} \right)^2$$

In R, linear regression is implemented using the `lm()` function, which provides estimates of the regression coefficients, associated statistical significance, and goodness-of-fit metrics. A typical usage is illustrated below:

```
model <- lm(Y ~ X1 + X2 + X3, data = dataset)

summary(model)
```

This formulation establishes the theoretical and computational foundation upon which the proposed R package operates.

4. Methodology

4.1. Problem Setup

Let $(\mathrm{x}, y) \sim P(\mathrm{x}, y)$ over $\mathbb{R}^p \times \mathbb{R}$ denote the population-level data-generating process, with unknown conditional mean,

$$f^*(\mathrm{x}) = \mathbb{E}_P[y \mid \mathrm{x}]$$

In practice, f^* is estimated from n independent observations drawn from P, yielding a design matrix $\mathrm{X} \in \mathbb{R}^{n \times p}$ and response vector $\mathrm{y} \in \mathbb{R}^n$. Missing entries in X are encoded by $\mathrm{M} \in \{0,1\}^{n \times p}$, where $M_{ij} = 1$ denotes X_{ij} unobserved. The objective is to estimate $\hat{f}$ from the observed sample such that $\hat{f}(\mathrm{x}_{\text{new}})$ accurately predicts y_{new} for a new observation $\mathrm{x}_{\text{new}} \sim P(\mathrm{x})$, even in the presence of missing entries.

4.1.1. Regression Task

Given an observed sample $\{(x_i, y_i)\}_{i=1}^n$, the aim is to develop an estimator $\hat{f}$ which can make accurate predictions while some of the predictor values are absent. Depending on the approach, the missing entries in X may be either imputed or directly handled in model training and prediction. The estimator $\hat{f}$ is obtained by minimizing empirical risk over the observed (or converted) data:

$$\hat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \ell\left(y_i, f(\mathbf{x}_i)\right),$$

where $\ell(\cdot,\cdot)$ is a chosen loss function (e.g., squared error).

4.1.2. Assumptions

a) **Sampling:** The observed data $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ are independent draws from a fixed unknown population distribution $P(\mathbf{x}, y)$, ensuring that sample-based estimates generalize beyond the observed data.

b) **Missingness:** Missing entries in $\mathbf{X}$ occur under arbitrary patterns at the sample level. No assumptions are made on the missingness mechanism a priori.

c) **Linearity:** The conditional mean is assumed linear in the predictors:

$$y_i = \tilde{\mathbf{x}}_i^{\top} \boldsymbol{\beta} + \epsilon_i, \quad \epsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2),$$

where $\beta \in \mathbb{R}^p$ is estimated from the completed sample $\tilde{\mathrm{X}}$.

d) **Predictive Objective:** The focus is on predictive accuracy on new observations rather than unbiased recovery of β.

4.2. Proposed Analysis

The package includes both imputation-based and prediction-oriented approaches. Imputation-based approaches include methods like Amelia, MICE, and the missForest algorithm. They impute the missing entries before model fitting. In contrast, the prediction-oriented methodology uses methods such as available cases and the tower technique, which are less dependent on imputation and use partially observed data for predictive modeling.

4.2.1. Linear Model with Available Cases (lm_ac)

The `lm_ac` technique uses linear regression based on the *available-cases* (AC) concept, a specific type of pairwise deletion approach for handling missing data without explicit imputation. Rather than deleting complete observations with missing entries, `lm_ac` builds the normal equations term-by-term using all available information.

Specifically, let X denote the design matrix and y the response vector. In case of missing values, the standard ordinary least squares estimator based on the complete cross-product matrices cannot be computed directly. The `lm_ac` method instead constructs normalized available-case estimates of these quantities:

$$\hat{\beta}_{AC} = \left(\widehat{X^\top X}\right)^{-1} \widehat{X^\top} y,$$

where,

$$\left(\widehat{X^\top X}\right)_{jk} = \frac{1}{N_{jk}} \sum_{i \in \mathcal{O}_{jk}} x_{ij}\, x_{ik},$$

and

$$\left(\widehat{X^\top y}\right)_{j} = \frac{1}{N_{jy}} \sum_{i \in \mathcal{O}_{jy}} x_{ij}\, y_i.$$

Here, $\mathcal{O}_{jk}$ denotes the set of observations for which both x_{ij} and x_{ik} are observed, and $N_{jk} = \left|\mathcal{O}_{jk}\right|$. Similarly, $\mathcal{O}_{jy}$ denotes the set of observations for which both x_{ij} and y_i are observed, with $N_{jy} = \left|\mathcal{O}_{jy}\right|$.

The (j,k) entry of $\widehat{X^\top X}$ is computed by averaging $x_{ij}x_{ik}$ across all observations i for which both x_{ij} and x_{ik} are observed. Similarly, each component of $\widehat{X^\top} y$ is the average of $x_{ij}y_i$ over all observations for which both x_{ij} and y_i are known. These available-case averages are then merged to construct estimated system matrices and the regression coefficients are obtained by solving the normal equations that correspond. This approach allows `lm_ac` to take advantage of partially

observed data more efficiently than the complete-case approach, which estimates only on the basis of observations without any missing values among all predictors. The available-cases technique can be biased for some missingness mechanisms but it is simple and computationally cheap and constitutes a prediction-oriented alternative to imputation-based methods, especially where the goal is predictive performance rather than parameter recovery.

Although the available-cases approach allows prediction using partially observed data, it also introduces important numerical challenges. In particular, the pairwise construction of the normal equations matrix means that distinct elements of $\widehat{X^\top X}$ are calculated from separate subsets of data. This means that the matrix that is obtained is no longer guaranteed to be positive *semidefinite*, a property that is satisfied if all cross-products are calculated from a common sample. This loss of structure can cause numerical instability in highly dimensional or connected data. This may lead in some cases to a singular or nearly singular system matrix. The difficulty can be owing to redundancy among predictors, or lack of variability in particular subsets of data. In contrast, complete-case regression creates $X^\top X$ from one coherent set of observations. This problem is typically solved by deleting columns implicitly. Thus, compromising invertibility for the sake of rejection of partially observed data.

4.2.2. *lm_tower*

The `lm_tower` method utilizes the `toweranNA` package that is based on the Tower technique. This package is specifically designed for prediction issues with missing values. The Tower technique does not attempt to impute missing values of predictors, unlike imputation-based methods such as MICE or Amelia. Instead, it directly makes predictions while letting missing values stay in both the training data and new cases.

The technique is based on the Tower Property of the conditional expectation, which says that:

$$E[E(Y \mid U, V) \mid U] = E(Y \mid U).$$

This condition means that the marginal regression function given only the observed variables is the average of the whole regression function over the variables that are missing. In practice, the Tower approach fits the regression model (e.g., `lm`) on the complete-cases in the training data. If a new observation has missing values for predictors, the prediction is obtained by averaging the fitted regression function over training observations that have observed predictor values similar to those of the new case. The averaging is regulated by a tuning parameter k, which defines how many neighboring observations are used. This determines how many nearest neighbors from the training data are used in the averaging step.

Within the `lm_tower`, the function `makeTower()` from the `toweranNA` package is used to construct the Tower model using the training data and response variable. The Tower technique is flexible but not assumption-free. Firstly, it relies on the existence of a sufficiently large and

representative complete-case subset, since the initial model is fitted using only fully observed data. Second, the prediction step assumes that for any partially observed test instance, there are training observations with similar values in the observed feature space, so that nearest-neighbor averaging provides meaningful predictions. This effectively demands some overlap between observed and missing patterns in the dataset. Third, the method implies that the conditional relationship acquired from complete cases can be extended to incomplete situations, which may not be true under sophisticated missingness mechanisms such as MNAR. Therefore, although the method might be applied in many situations of missingness, its success depends on the structure of the data, the sample size and the distribution of the missingness, rather than being able to deal with arbitrary missingness without any constraint.

4.2.3. *lm_prefill*

In the `lm_prefill` strategy, missing values are dealt with by a preprocessing step in which the incomplete entries are first imputed and then a linear model is fitted on the filled data. `lm_prefill` is a wrapper that plugs existing imputation algorithms to a predictive modeling pipeline.

Given a dataset containing missing values, the `lm_prefill` first applies a user-specified imputation algorithm to produce completed datasets. A linear regression model is then fitted using the specified response variable and the other predictors. The procedure is slightly different depending on the imputation method chosen.

a) **Complete-Cases (CC):** The complete-case method removes any observation that has at least one missing value. Let X be the prediction matrix and y the response variable. We build a smaller dataset (X_{cc}, y_{cc}) by choosing only the rows with all observed elements. Then (X_{cc}, y_{cc}) are used to fit a regression model. This is easy to implement but can lead to a large amount of data loss. Thus, unbiased estimates are available only under the MCAR assumption. This corresponds to the case where no imputation is performed in the `lm_prefill` framework.
b) **MICE:** Multiple Imputation by Chained Equations (MICE) imputes missing values by repeatedly modeling each variable with missing entries conditional on the remaining variables. The program iterates multiple times over these conditional models and generates different completed datasets. `lm_prefill` creates m imputed datasets using the MICE package. Then we fit a linear regression model to each of the datasets and average the predictions of the obtained models to get the final prediction.
c) **Amelia:** Amelia uses a joint modeling approach for multiple imputation. The approach assumes a multivariate normal distribution for the data. The approach estimates model parameters using an Expectation-Maximization (EM) algorithm with bootstrap resampling to generate several completed datasets. `lm_prefill` uses the Amelia package to

produce m imputed datasets. For each finished data set, a linear regression model is fit, and the predictions are averaged over the resulting models.

d) missForest: The missForest algorithm (implemented in the missForest package) imputes missing values using random forest models trained on the observed portions of the data. For each variable with missing entries a random forest model is fitted using the other variables as predictors, and missing values are iteratively updated until convergence. In contrast to model-based techniques like MICE or Amelia, missForest is non-parametric and does not make any explicit distributional assumptions. Instead, it uses the predictive capability of random forests to capture complicated, sometimes non-linear relations in the data. In doing so it is able to implicitly take use of patterns of missingness through the observed features. It is more flexible than strictly assumption-driven imputation approaches, while it still depends on the observed variables having sufficient information to correctly anticipate missing values. The missForest technique gives one completed data set and the imputed data are then used directly to train a linear regression model

4.3. Computational Considerations

The methods implemented in `mvPred` are intended for low-to-moderate-dimensional regression settings representative of standard applied analysis. The complete-case (CC) and available-case (AC) estimators inherit the computational complexity of base ordinary least squares fitting. CC regression scales approximately as $O(np^2)$ under QR decomposition, while AC requires $O(np^2)$ work to construct the pairwise cross-product system followed by an additional $O(p^3)$ normal-equation solving. The five datasets employed in this study span sample sizes from $n = 398$ (Auto MPG) to $n = 9{,}627$ (Wine) and dimensionality from $p = 4$ (NHkids) to $p = 13$ (English), all of which fall comfortably within the tractable regime for CC and AC methods. The PREFILL family of methods incurs additional overhead determined by the choice of imputation procedure: single-pass completion introduces negligible cost, whereas iterative schemes such as MICE [20] and missForest [17] exhibit runtime that grows with p, as each requires fitting a separate predictive model per variable per iteration. Missingness in the evaluated datasets ranges from 0.19% (Auto MPG) to 14.77% (NHkids), with per-column missingness reaching as high as 49.91% in the English dataset, representing a moderately challenging imputation setting within the practical scope of these methods.

The computational cost of the TOWER estimator is governed by the number of distinct missingness patterns present in the data, which in the worst case grows combinatorially as $O(2^p)$; given that missingness is concentrated in a small subset of columns across all five datasets (at most 4 of 13 columns), pattern proliferation is not expected to pose a practical concern here. Furthermore, all methods are evaluated within the k-fold cross-validation procedure

implemented in `bootstrap()`, introducing a multiplicative factor of k on total runtime. Nonetheless, a systematic empirical characterization of wall-clock time and memory consumption across a broader factorial grid of (n, p) configurations, extending beyond the dataset sizes considered here, remains an avenue for future investigation (see Section 10).

5. Package Setup

The methods developed in this work are bundled into an R package, `mvPred`. This package provides a unified interface for prediction tasks involving missing data. Packaging the implementation in this form provides users with different methods for handling missing data. The package is available at https://github.com/matloff/mvPred. The package depends on a number of existing R packages, including MICE, Amelia, missForest, toweranNA, regtools, and qeML, as well as the foundation R packages `stats` and `utils`. These relationships are the basis for the imputation, prediction and assessment procedures employed in our solution. Additional function-level documentation is provided in the package manual pages, available in the repository's `man` directory at https://github.com/matloff/mvPred/tree/main/man.

Table I below summarizes the main package functions and their roles.

TABLE I: Main functions provided in the `mvPred` package and their roles in the prediction pipeline.

Function	Goal
`lm_ac`	Fit prediction models using the available-case approach for missing data
`lm_prefill`	Apply imputation-based prefill methods prior to model fitting
`lm_tower`	Interface with tower-based prediction methods for missing-data settings
`qeMLna`	Extend missing-value handling to predictive methods implemented in `qeML`

6. Experimental Setup

6.1. Datasets

For our study, we consider datasets with continuous targets and sample sizes ranging from approximately 300 to $\sim$ 10,000 observations, including Auto MPG[14], TAO, and Wine from the `VIM` package[7], NHkids, and the English dataset from Wordbank. These datasets contain missing values in one or more predictors. The missingness levels ranges from low to substantial (approximately 1.5% to 50%; see Table II below). The datasets include both purely numeric predictors and mixtures of numeric and categorical features. Thus, allowing us to evaluate performance across diverse design structures.

Table II: Summary of datasets used in the empirical study. Here, y denotes the target variable, n denotes the number of observations and p the number of variables. Missing cells (%) is the proportion of all data entries that are missing, Columns w/ NA is the number of variables containing at least one missing value and Max col miss (%) is the maximum percentage of missing values in any single variable.

Dataset	y	n	p	Missing cells (%)	Columns w/ NA	Max col miss (%)
Auto MPG	mpg	398	8	0.19	1/8	1.51
TAO	sea.surface.temp	736	8	3.01	3/8	12.64
NHkids	weight	2519	4	14.77	3/4	46.65
English	vocab	5498	13	13.50	4/13	49.91
Wine	price	9627	9	1.74	2/9	15.43

6.2. Baselines

We start with a complete-case (CC) regression as a baseline. The results from CC are used as a reference for comparison with the available-cases regression, tower, and the prefill-based approaches. The same train/test splits and performance measures used for the other methods are used to evaluate CC regression, to guarantee a fair comparison.

6.3. Evaluation Metrics

We evaluate and compare the performance of the proposed techniques with multiple measures.

a) **Mean Squared Error (MSE)** is the average of the squared errors between the expected and the actual values:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^{2.}$$

Because errors are squared, larger errors have a bigger impact on the MSE. This makes the MSE vulnerable to outliers.

b) **Root Mean Square Error (RMSE)** is the square root of MSE:

$$\text{RMSE} = \sqrt{\text{MSE}}.$$

RMSE has the same units as the response variable and is therefore easier to grasp in the context of the problem.

c) **Mean Absolute Error (MAE)** is the mean of the absolute value of the difference between the prediction and the actual data:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

MAE does not square the errors as MSE does. MAE gives equal weight to all errors and is less vulnerable to outliers.

d) **Coefficient of Determination ($\boldsymbol{R^2}$)** is the fraction of the variance in the response variable that is explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2},$$

where $\bar{y}$ denotes the mean of the observed answers. R^2 varies from 0 to 1 and larger values indicate a better match.

Together, these measures give a holistic view of the predictive performance since they represent multiple aspects of prediction accuracy and error behavior.

6.4. Experimental Protocol

We evaluated the performance of the methods using a k-fold cross-validation procedure. In k-fold cross-validation, the dataset is randomly partitioned into k groups (folds). In each iteration one fold is used for the test set and the other $k-1$ folds are used for training.

In our experiments, we set $k = 5$ and used a random seed of 42 to ensure reproducibility. To prevent data leakage, all preprocessing steps were performed independently within each training fold. We implemented four distinct modeling pipelines to handle missing data: $\mathrm{PREFILL(CC)}$, AC, TOWER, and $\mathrm{PREFILL(MICE, AMELIA, MISSFOREST)}$. The same k-fold cross-validation setup was applied consistently across all four pipelines and evaluated on each of the five datasets.

7. Results

The prediction performance of the competing approaches was compared on our five datasets. The results of our findings are shown in Table III. To have a clear view of the predictive performance, we have employed a collection of metrics such as RMSE, MSE, MAE and R^2. The $lm\_prefill$ methods generally showed the strongest prediction performance across datasets. Among them, MICE achieved the lowest RMSE and MAE values, especially on the Auto MPG and English datasets. The complete-case regression remained a strong baseline and was competitive on several datasets (TAO, NHkids). Although CC ignores observations with missing values, its simplicity and stability enabled it to reach similar performances in some circumstances. AC regression showed more mixed results. Although, the pairwise estimation strategy uses more observations than CC, it can lead to instability in the regression coefficients. This could have a negative impact on predictive accuracy. The Tower approach exhibited dataset-dependent performance. Its performance was not as strong on some datasets, but on others, such as the Wine dataset, it performed well. The performance of this strategy might be contingent on the structure of the missingness and the nature of the dataset. Overall, the results imply that imputation-based approaches, such as $lm\_prefill$, provide the most consistent predictive performance across datasets, although simpler approaches such as complete-case regression remain competitive in some circumstances.

Table III: Predictive performance of the evaluated methods across all datasets. The table reports the average MSE, RMSE, MAE, and R^2 obtained from 5-fold cross-validation for each dataset–method combination.

Dataset	Method	MSE	RMSE	MAE	R^2
Auto MPG	PREFILL (CC)	10.5571	3.2046	2.4619	0.8197
	AC	10.8399	3.245	2.5	0.8153
	TOWER	18.584	4.2905	3.2892	0.6737
	PREFILL (MICE)	10.5331	**3.2014**	2.4643	0.8201
	PREFILL (Amelia)	10.5517	3.2043	2.4657	0.8198
	PREFILL (missForest)	10.5472	3.2035	2.4657	0.8198
TAO	PREFILL (CC)	0.0927	**0.3031**	0.2265	0.9836
	AC	0.4088	0.6364	0.4869	0.9267
	TOWER	0.232	0.4812	0.3787	0.9589
	PREFILL (MICE)	0.1162	0.3407	0.2526	0.9794
	PREFILL (Amelia)	0.1171	0.3421	0.2548	0.9792
	PREFILL (missForest)	0.1259	0.3547	0.2591	0.9777
Wine	PREFILL (CC)	3717.018	59.696	28.5153	0.2708
	AC	3816.585	60.6378	31.6599	0.243
	TOWER	3716.657	**59.6932**	28.5117	0.2708
	PREFILL (MICE)	3723.175	59.7346	28.0011	0.2702
	PREFILL (Amelia)	3725.618	59.7468	27.8284	0.2702
	PREFILL (missForest)	3731.455	59.7999	27.848	0.2687
English	PREFILL (CC)	20285.98	142.3098	112.9916	0.5177
	AC	21575.74	146.8525	119.9773	0.4873
	TOWER	20490.3	143.0485	113.9011	0.5129
	PREFILL (MICE)	20244.52	**142.1605**	112.467	0.5187
	PREFILL (Amelia)	20397.65	142.6948	113.3441	0.5151
	PREFILL (missForest)	20633.51	143.4917	113.2123	0.5095
NHkids	PREFILL (CC)	134.0559	**11.5312**	8.4521	0.7659
	AC	194.9366	13.9107	10.2862	0.6619
	TOWER	135.5537	11.5925	8.4987	0.763
	PREFILL (MICE)	135.3094	11.585	8.3055	0.7639
	PREFILL (Amelia)	135.0964	11.5721	8.3036	0.7645
	PREFILL (missForest)	137.6803	11.6825	8.3744	0.7601

8. Discussion

The results on all five datasets are consistent and interpretable, as shown in III. In this section, we explore the predictive behavior of each class of methods, provide mechanistic explanations for the differences in performance, and place the findings in the context of the broader literature on missing-data handling for supervised regression.

8.1. Bias Implications of Complete-Case Analysis

Complete-Case Analysis (CCA) is unbiased only when MCAR holds, which assumes the probability of missingness is independent of observed and unobserved data [15, 8]. When this assumption is violated, as is typical for MAR or MNAR mechanisms, listwise deletion results in a training set that is not a representative sample of the population distribution $P(\mathrm{x}, y)$, and the obtained estimator $\hat{\beta}$ is afflicted with a systematic bias that does not vanish with increasing sample size [18]. Even under MCAR, CCA incurs a precision penalty: the rejection of incomplete observations reduces the effective sample size, resulting in wider confidence intervals and increased estimator variance [8].

The empirical results reflect both failure modes in a nuanced way. In TAO, where missingness is low (3.01%) and the complete-case subsample is likely large and representative, CCA performs best overall (MSE = 0.093, R^2 = 0.984), consistent with near-MCAR conditions.

On Auto MPG and English, where the missingness is higher and the mechanism is not verified, the best performing imputation variant (PREFILL with MICE) performs at par or marginally better than CCA. However, for NHkids and Wine, CCA is still the best approach, suggesting that imputation does not always provide an advantage and that dataset-specific characteristics such as the degree and structure of missingness substantially influence results. It is worth mentioning that the missingness mechanism is not formally tested across datasets and thus the observed performance differences are treated as empirical evidence of CCA's context-dependent fragility under realistic missing-data conditions rather than as a definitive proof of MAR or MNAR.

8.2. Prediction Coverage Under Missing Test Data

$Prediction\ Coverage$ or the ability of a fitted model to provide a prediction for every new observation, including those with missing predictor values, is an important but understudied aspect of the evaluation of missing-data techniques. This characteristic directly affects each strategy's practical utility, however not all of the solutions looked at in this study guarantee it.

Table IV: Percentage of missing values in the training and testing sets for each dataset.

Dataset	Variable	Training (%)	Testing (%)
Auto MPG	horsepower	0.94	3.80
TAO	Air.Temp	10.90	9.59
	Humidity	12.61	11.64
Wine	taster_twitter_handle	15.46	15.46
English	birth_order	49.67	50.86
	ethnicity	49.47	50.86
	mom_ed	49.67	50.86
	sex	25.91	26.02
NHkids	AgeMonths	47.05	45.80
	height	10.64	13.80

8.2.1. Coverage Properties of Each Method

a) **Complete-Case Analysis (CC)**: Using fully observed training observations, the CC estimator fits a linear model and generates predictions using the formula $\hat{y} = \tilde{\mathrm{x}}^{\mathrm{T}}\hat{\beta}$. At test time, this inner product requires all p predictor values to be present and any test observation with one or more missing entries is therefore unpredictable under CC without an auxiliary strategy. In the 5-fold cross-validation protocol employed here, test folds are drawn from the same dataset as training folds and thus share the same missingness distribution. On datasets with high per-column missingness, such as NHkids (46.65%) and English (49.91%), a non-trivial fraction of test observations contain missing predictors and cannot be scored by CC directly. When such observations occur during deployment, the practitioner has to either label the observation as unpredictable, return a backup prediction like the training mean, or impute on the spot. Standard cross-validation measures do not account for the unique bias-variance tradeoff associated with each option.

b) **Imputation-Based Methods (PREFILL):** MICE, Amelia and missForest all produce a completed design matrix $\tilde{\mathrm{X}}$ prior to model fitting, enabling prediction on any test

observation regardless of its missingness pattern, provided that the imputation model is applied to the test instance at inference time. To ensure complete prediction coverage across all observations, the PREFILL framework handles test-time imputation by applying the training imputation model to each held-out fold. However, when the percentage of missing predictors in a particular test observation rises, imputation quality deteriorates. When a test instance is missing the majority of its predictors, imputed values are driven almost entirely by marginal means rather than conditional relationships, and the resulting prediction may be poorly calibrated.

c) **Available-Cases (AC):** From pairwise available instances, the `lm_ac` estimator generates a coefficient vector $\hat{\beta} \in \mathbb{R}^p$. At test time, prediction via $\hat{y} = \mathrm{x}^\top \hat{\beta}$ again requires full observation of x; partially observed test instances cannot be scored without substituting values for missing entries. AC therefore shares the same coverage limitation as CC at prediction time, despite using more data during training.

d) **TOWER:** Unlike all imputation-based and available-case approaches, TOWER is the only method in this evaluation that natively handles missing predictor values at prediction time. Using only the observed feature subspace, TOWER retrieves the closest neighbor from the complete-case training set for each test instance with missing entries. The fitted value of that neighbor is then returned as the forecast. Without the need for an imputation phase, this design ensures complete prediction coverage for any test observation, regardless of which or how many predictors are missing [10]. Aggregate cross-validation measures do not fully convey this qualitative advantage over all other tested approaches.

8.2.2. Handling Unpredictable Holdout Observations

If a holdout observation cannot be natively scored as is the case under `CC` and `AC` when test instances contain missing values, three practical fallback strategies are available, each with different trade-offs:

a) **Mean imputation fallback:** Missing test predictors are replaced with their training-set column means prior to prediction. However, this approach maintains coverage at the expense of attenuation bias, as mean imputed predictions regress toward the mean of the response distribution and underpredict variance in extreme cases [8].

b) **Available-predictor submodel:** A second model is trained on the subset of predictors that are observed in the test instance. This preserves conditional signal but requires fitting $2^p - 1$ possible submodels at training time -- computationally infeasible for moderate p -- or retraining on-the-fly at inference, which is impractical in production settings.

c) **Abstention with flagging:** The model abstains from prediction and flags the observation for manual review or routing to an alternative pipeline This is appropriate in high stakes

settings, where an incorrect prediction is more costly than no prediction, but reduces effective coverage and may introduce selection bias if flagged observations are systematically different from predictable ones.

In this evaluation context, `lm_prefill` implicitly approximates strategy a) by applying training imputation models to test instances. However, with high missingness at test time, the imputed values are driven more by marginal means than conditional relationships, thus reducing the predictive quality. This limitation is most severe for English and NHkids, where test-set missingness is as high as 50.86% and 45.80%, respectively (Table IV), such that a significant fraction of test observations are effectively scored with mean-driven imputations. `TOWER` avoids this problem altogether by design, working directly over the observed feature sub-space without any imputation step. Its native coverage guarantee then stands as a meaningful practical advantage in deployment settings where test-time missingness is to be expected. This advantage, however, is not reflected in the aggregate R^2 and RMSE values reported in Table III. This point underscores a more general methodological issue: future comparative assessments of missing-data methods should incorporate coverage rates in addition to standard measures of accuracy, since aggregate performance metrics alone do not fully characterize method utility under realistic deployment conditions.

8.3. Robustness Across Imputation Backends

The three `PREFILL` variants, MICE, Amelia and missForest, yield very converging results across the five datasets, with pairwise differences in MSE and R^2 being of negligible magnitude. For example, on Auto MPG, MSE values range over a range less than 0.02 (10.533 to 10.552) and R^2 values are effectively the same (≈ 0.820). On English, `PREFILL(MICE)` achieves the lowest MSE among all evaluated methods (20244.52 vs. 20285.98 for `CC`), while on Auto MPG all three variants marginally outperform `CC`. On Wine and NHkids, however, `CC` achieves lower error, and `PREFILL` variants remain competitive without surpassing it. This pattern suggests that the benefit of principled imputation is dataset-dependent, modulated by the degree and structure of missingness, rather than universally dominant.

The convergence across imputation backends is itself a substantive finding. All three variants impute missing values prior to model fitting using principled statistical estimators, preserving the covariance structure necessary for reliable OLS estimation. The choice of specific algorithm has negligible downstream effect on predictive accuracy. This is in line with prior work by van Buuren [19] and Stekhoven and Bühlmann[17], who found that multiple imputation methods tend to converge in predictive accuracy under typical MAR conditions. Practically, this means that

practitioners using the PREFILL framework can choose any of the three supported imputation backends, knowing that the predictive results will be largely the same and that the choice of method can be driven by computational budget or interpretability requirements.

8.4. Complete Cases Outperform Available Case Estimation

Contrary to the expectation that the use of all available observations should lead to better model fit, CC always outperforms AC for all the five datasets. For example, on TAO, CC achieves an R^2 of 0.9836 vs 0.9267 for AC, almost six points difference, although AC uses strictly more observations during training. There are two complementary explanations for this pattern.

The first is the mechanics of the AC estimator. Instead of listwise deletion, `lm_ac` constructs the normal equations directly from pairwise available cases, maximizing data utilization at the element level. This results in a basic inconsistency: each entry of the normal equation matrix is obtained from a possibly different subset of observations. If the missingness is not MCAR, these subsets have different effective composition and the assembled matrix does not correspond to any one coherent joint distribution[8]. Such inconsistent system of equations leads to biased and possibly unstable estimates of coefficients, degrading the predictive performance.

The second explanation is a selection effect inherent to CC. CC is trained on a smaller but internally consistent subset of the data by restricting model fitting to fully observed instances. In the case of MAR or MNAR mechanisms, complete cases are a purposefully selected subpopulation that might differ in significant ways from the overall data set [8, 18]. In the 5-fold cross-validation setting used here, training/test folds are selected from the same dataset and this selection effect is consistent across folds, and CC may be leveraging a structural regularity in the complete-case subpopulation rather than truly generalizing to the full data distribution. Thus, the apparent better predictive performance may be partly due to overfitting to a narrow, homogeneous subset.

In practice, the observed performance difference is likely to be due to a combination of both factors, compounded by the small sample sizes of the datasets being analysed. This is a direction for future work that would require controlled experiments under known missingness mechanisms (MCAR, MAR, and MNAR) with systematically varied sample sizes and missingness rates to disentangle these effects.

8.5. TOWER is Sensitive to Sample Size and Missingness Pattern

Of the methods evaluated, TOWER is theoretically well-suited for prediction with missing data and one would expect it to perform competitively across held-out observations with missing predictors. But the empirical results tell a more nuanced story. TOWER performs substantially worse on Auto MPG (MSE $= 18.58$ $vs.\backslash \sim 10.53$ for `PREFILL`). TOWER also performs worse than both `CC` and `PREFILL` on TAO and English, but performs on par with `CC` on Wine and is competitive on NHkids. This mixed behavior warrants a careful examination of TOWER's internal mechanism.

As implemented in the `toweranNA` package [10], TOWER operates in two phases. It fits a global linear model on the complete-case subset during training, and gets a vector of fitted values on that subspace. At prediction time, TOWER uses only observed features to compute the Euclidean distance in the reduced feature subspace and finds the nearest neighbor in the complete-case training set for each test instance with missing values, returning that neighbor's fitted value as the prediction. At inference time, the principal mechanism for handling missingness in TOWER is a nearest-neighbor lookup over fitted values, with $k = 1$ by default. The reliability of such a lookup depends critically on the density of the complete-case training set.

For small datasets, the complete-case subset may be too sparse for any test instance to find a meaningfully close match. This problem is exacerbated when several features are missing in a test instance, as the neighbour search is performed in a lower dimensional subspace, increasing the chances of finding a structurally dissimilar neighbour whose fitted value is a poor approximation of the true response. Since TOWER fits its global model on complete cases only — identical in this respect to `CC` — any performance advantage over `CC` must arise entirely from the prediction-time lookup. If the complete-case training set is sparse or the missingness pattern is complex, this lookup is not beneficial enough to outweigh the variance it adds, and TOWER consequently performs worse than imputation-based methods that reconstruct the full feature vector before prediction.

In the Wine result, TOWER achieves the best performance together with `CC` among all methods, partially supporting TOWER's theoretical promise. This means that TOWER achieves its design goal when the full-case training set is sufficiently dense and the missingness pattern is such that neighbor lookup in the observed feature subspace reliably finds close matches. Its poor performance in other settings can therefore be interpreted as a result of the small sample size and complex missingness patterns, rather than a fundamental failure of the algorithm. The comparative advantages of TOWER might be more evident in future evaluation on larger datasets and with different values of k.

8.6. Diminishing Returns of Imputation Under Low Missingness

CC outperforms all PREFILL variants on three datasets: TAO, NHkids and Wine, with R^2 scores of 0.9836, 0.7659 and 0.2708 respectively, compared to PREFILL (MICE)'s 0.9794, 0.7639 and 0.2702. The TAO result is the easiest to interpret: with only 3.01% of cells missing in two variables and a maximum missingness rate of 12.61% per column (Table IV), the complete-case subset preserves the vast majority of the 736 available observations. Under such conditions, multiple imputation offers no meaningful informational advantage over listwise deletion, as the missing entries contribute little additional signal that imputation can recover, while the imputation model itself introduces a small but non-negligible source of additional variance. When the complete-case subsample already represents the great majority of the data's predictive structure, this imputation overhead—the stochasticity introduced by estimating and filling missing values—becomes the main limitation on performance. This is in line with the theoretical result that CC estimators are asymptotically unbiased under MCAR [8] and TAO's low, seemingly unsystematic missingness pattern is consistent with such a mechanism.

The missingness rate alone does not easily explain the NHkids outcome, as AgeMonths is missing in nearly half of all observations (Table IV). Nevertheless, the complete-case set appears to retain sufficient predictive structure for CC to be competitive, suggesting that the observed variables still contain strong predictive signal after the listwise deletion. The Wine result similarly reflects a single- variable missingness pattern (taster_twitter_handle at 15.46%), where the complete-case subset remains largely representative of the full data distribution.

It is important to note that the missingness mechanism cannot be formally verified for any of these datasets without additional diagnostic testing. The prevalence of CC is compatible with MCAR but not conclusive. Even when missingness is low, CC estimates may be biased with MAR or MNAR [18]. As discussed in Section 8.4, the observed performance superiority may be a reflection of overfitting to the complete-case subpopulation. We thus interpret these results as indicating that when missingness is low and the complete-case subsample is large, the practical benefit of imputation is minor - a finding of immediate relevance to practitioners considering whether the overhead of multiple imputation is merited in their particular setting.

8.7. Prediction Difficulty Overshadows Imputation Strategy

The Wine dataset warrants separate examination, as all methods converge on comparatively low R^2 values in the range of 0.24 - 0.27, irrespective of the missing data strategy employed. With only 1.74% of cells missing across two variables (Table IV), Wine is both the largest dataset and the least affected by missingness in our benchmark, and the near-uniform performance across methods cannot be attributed to differences in missing data handling. The dominant constraint is instead the inherent difficulty of predicting wine price from the available feature set under a linear model. Wine price is known to be highly right-skewed, driven by a small number of prestige bottles commanding disproportionately high prices, and is shaped by factors such as vintage, critic reputation, and regional appellation [1], many of which interact nonlinearly and are only partially captured by the available predictors. The residual variance that OLS cannot explain within this feature space is therefore large and irreducible under a linear assumption, placing a hard ceiling on achievable R^2 regardless of how missing values are handled. The right-skewed price distribution further violates the homoscedasticity assumption of OLS, introducing systematic error that no missing data strategy can remedy.

The marginal separation between methods - CC and TOWER achieving the highest R^2 of 0.2708 against $\mathrm{AC}'s$ 0.2430 - reflects the negligible role of imputation when missingness is this low. The Wine results therefore serve as an informative negative control: they demonstrate that $prefill$'s imputation step introduces no meaningful distortion even under a difficult, skewed prediction task, and that the framework degrades gracefully when the binding constraint is irreducible response variance rather than missing data. More broadly, they delineate a practical boundary condition in which the choice of missing data strategy is of negligible consequence, and model improvement must instead come from richer feature representations or non-linear modeling approaches.

8.8. Practical Suggestions for Method Selection

The empirical findings across five benchmark datasets collectively motivate a set of practical guidelines for selecting an appropriate missing data strategy within a linear regression pipeline. While no single method universally dominates, the results reveal clear and interpretable patterns that can inform method selection based on dataset-specific properties, summarised in Table V below.

Table V: Recommended missing data strategy based on dataset characteristics.

Condition	**Recommended Method**	**Evidence**
Low missingness (<5%), large n, no prior evidence of MNAR	CC	TAO
Moderate–high missingness, non-normal or continuous variables	PREFILL (missForest)	Auto MPG, English
Moderate–high missingness, mixed variable types or custom imputation required	PREFILL(mice)	Auto MPG, English
Large n with dense complete-case subset and low test-time missingness	TOWER	Wine
Small n, any missingness rate	PREFILL (MICE or missForest)	Auto MPG
High irreducible response variance	Any	Wine

When missingness is low and the complete-case subsample is large, listwise deletion (CC) is a defensible and computationally efficient default. As demonstrated by TAO, where only 3.01% of cells are missing and CC achieves the best R^2 of 0.9836, the overhead of multiple imputation is not justified when the complete-case subsample already captures the vast majority of the data's predictive structure. Practitioners can apply CC with confidence in such regimes, provided there is no strong prior evidence of a MAR or MNAR mechanism that would render the complete-case subsample systematically unrepresentative [8].

When missingness is moderate to high and sample size is sufficient, multiple imputation prior to model fitting is the recommended strategy. On Auto MPG and English, all three `prefill` variants --- MICE, Amelia, and missForest --- match or outperform competing methods with near-identical predictive performance across backends, confirming that the choice of imputation algorithm is less consequential than the act of principled imputation itself. When secondary considerations do apply, missForest is preferable for mixed variable types or non-normal distributions, as it requires no parametric assumptions [17]; MICE offers greater flexibility through variable-specific conditional models; and Amelia is most appropriate when multivariate normality is plausible and computational efficiency is a priority.

TOWER is most appropriate when the complete-case training set is sufficiently dense to support reliable nearest-neighbor retrieval. On Wine, TOWER achieves the best RMSE among all methods, and it remains competitive on NHkids, suggesting that its prediction-time lookup performs well when sufficient structure is preserved in the observed feature subspace. Conversely, TOWER should be used with caution on smaller datasets or when many predictors are simultaneously missing at test time, as the neighbor search degrades in substantially reduced feature subspaces. Its theoretical advantages are also most likely to materialize in non-parametric settings; within a linear regression pipeline, as evaluated here, its benefits are constrained [5, 9].

When the response variable exhibits high irreducible variance or a strongly non-linear relationship with its predictors, the choice of missing data strategy is of secondary importance. As illustrated by Wine, where all methods converge on $R^2 \approx 0.24$--0.27 despite a large sample size ($n = 9{,}627$) and low missingness (1.74%), the dominant constraint is the expressive capacity of the linear model rather than missing data handling. Practitioners in such settings should prioritize feature engineering, response transformation, or non-linear modeling over imputation method selection.

These recommendations are derived from evaluation on five datasets under a linear regression framework with 5-fold cross-validation, and their generalizability to non-linear models, high-dimensional feature spaces, or extreme missingness rates remains an open question. Practitioners should therefore treat these guidelines as informed defaults rather than prescriptive rules and validate method selection empirically on their specific dataset --- precisely the workflow that the `mvPred` package is designed to facilitate.

9. Limitations

The `mvPred` package offers a uniform approach to missing data handling in predictive modeling. However, there are several restrictions that should be mentioned.

First, the available-case linear modeling approach implemented in `lm_ac` can fail when the estimated cross-product matrix is singular or nearly singular. In such cases, the normal equations cannot be solved reliably. This issue is more likely to arise when predictors are highly collinear, when the effective amount of observed data is small, or when missingness substantially reduces the information available for certain variable pairs.

Another important point is that not all the methods in this package rely on the same assumptions about how the data is missing. In practice, the performance of a method is highly dependent on the nature of missingness i.e. MCAR, MAR or MNAR. Imputation-based approaches such as MICE and Amelia are generally justified under MAR, while Amelia further relies on approximate multivariate normality. Although missForest is more flexible, it is still dependent on the observed variables containing sufficient information to predict the missing values, hence is closer to a weak MAR assumption than a totally assumption-free solution. The tower-based method is less susceptible to explicit imputation assumptions and can be used under MCAR, MAR and some MNAR settings, but it does not give unbiased predictions. The approach also depends on sufficient full observations and significant overlap in observed characteristics, which may limit effectiveness in the case of sparse or complicated missingness.

Furthermore, the bundle is mostly targeted at predictive performance. Therefore, estimated coefficients may be biased. Standard errors or confidence intervals, if computed, may not be reliable. These methods should therefore not be used for inferential interpretation or parameter estimation without additional assumptions.

Another limitation is related to evaluation. In practice, some methods require dropping observations with missing predictors at prediction time. As a result, performance may only be assessed on subsets of complete data. This may overestimate real-world performance, where new observations often contain missing values.

Finally, imputation-based methods like `mice`, `Amelia` and the `missForest` algorithm can be computationally intensive, particularly with big datasets or when numerous imputations are needed. These techniques are frequently good at prediction, but their computational cost may restrict their practical usage in resource-limited environments.

10. Conclusion and Future Work

This paper introduced `mvPred`, an R package that provides a unified framework for prediction under missing data in supervised regression settings. The package implements three complementary strategies for handling missing values --- imputation-based prefill approaches (`lm_prefill`), available-case regression (`lm_ac`), and the Tower method (`lm_tower`). Within `lm_prefill`, four strategies are supported: complete-case analysis (CC), MICE, Amelia, and missForest. All methods are evaluated under a consistent 5-fold cross-validation protocol across five benchmark datasets spanning a range of sample sizes, missingness rates, and response variable characteristics.

Several operational inferences can be drawn from the empirical results. First, the `lm_prefill` framework provides the most consistent predictive performance across datasets. Among its variants, MICE, Amelia, and missForest yield near-identical results in most settings, suggesting that the act of principled imputation matters more than the specific algorithm chosen. Complete-case analysis, the simplest `lm_prefill` strategy, remains a competitive and computationally efficient option, particularly when missingness is low, and the complete-case subsample is large and likely representative, as demonstrated on the TAO dataset. Second, the available-case estimator (`lm_ac`) consistently underperforms relative to all `lm_prefill` variants, due to the inconsistency introduced by assembling normal equations from different row subsets under non-MCAR mechanisms and is not recommended as a general-purpose strategy in linear regression pipelines. Third, the Tower method (`lm_tower`) exhibits condition-dependent performance: it can underperform on smaller datasets or under complex missingness patterns but achieves the best results on Wine, where the large and dense complete-case training set enables reliable prediction-time neighbor retrieval. Fourth, when the response variable exhibits high irreducible variance --- as in the Wine dataset, where price is driven by nonlinear market dynamics not captured by the available predictors --- the choice of missing data strategy is of negligible consequence, and model improvement must come from richer feature representations or non-linear modeling approaches. Taken together, these findings suggest that no single method universally dominates across all datasets and model configurations, reinforcing the practical value of a unified benchmarking framework such as `mvPred` that enables systematic and fair comparison of multiple strategies under a common evaluation protocol.

Several directions remain open for future work. In the context of Scalability Benchmarking, the datasets employed in this study are modest in scale, spanning $n <= 9{,}627$ observations and $p <= 13$ predictors and consequently do not stress-test the computational limits of the methods implemented in `mvPred`. Future work should conduct a systematic empirical evaluation of wall-clock time and peak memory consumption across a factorial grid of (n, p) configurations — for

example, $n \in 10^3, 10^4, 10^5$ and $p \in 10, 50, 100, 500$ - under controlled missing-completely-at-random (MCAR) conditions at varying missingness rates. Such a benchmark would establish concrete practical thresholds for each method, quantify the runtime penalty introduced by iterative imputation schemes (MICE, missForest) relative to simpler alternatives (CC, AC) and characterize the sensitivity of the TOWER estimator to missingness pattern proliferation as p increases. Additionally, benchmarking under missing-at-random (MAR) and missing-not-at-random (MNAR) mechanisms would provide a more complete picture of both statistical and computational performance across realistic data conditions. Second, the current evaluation is restricted to linear regression; extending `mvPred` to non-parametric and machine learning downstream models such as random forests or gradient boosting --- would be a natural next step and may reveal more favorable conditions for the Tower method, whose theoretical guarantees are not specific to linear models. Third, the datasets evaluated here span missingness rates from approximately 0.19% to 49.91%; controlled experiments under known and varied missingness mechanisms (MCAR, MAR, and MNAR) with systematically varied sample sizes would allow cleaner attribution of performance differences to specific data-generating conditions. Fourth, the `lm_ac` estimator currently encounters singularity issues under high collinearity or sparse observation patterns; incorporating regularization or rank-deficiency diagnostics into the solver would improve its robustness. Fifth, currently the `lm_prefill` pipeline supports three imputation backends besides CC. The coverage of the framework might be further expanded by adding other techniques such as deep learning-based imputation [11] or gain-based approaches. Finally, the extension of the software to classification tasks and to non-continuous response variables would greatly increase its application to real-world predictive modeling challenges.

The code for the `mvPred` package is publicly available at https://github.com/matloff/mvPred, and we encourage contributions of additional methods, datasets and evaluation procedures by the community to further enhance the framework.

REFERENCES

[1] Orley Ashenfelter, David Ashmore, and Robert Lalonde. 12 - bordeaux wine vintage quality and the weather. In Stephen Satchell, editor, Collectible Investments for the High Net Worth Investor, pages 233–244. Academic Press, Boston, 2009.

[2] Melissa J. Azur, Elizabeth A. Stuart, Constantine Frangakis, and Paul J. Leaf. Multiple imputation by chained equations: What is it and how does it work? International Journal of Methods in Psychiatric Research, 20(1):40–49, 2011.

[3] Leo Breiman. Random forests. Machine Learning, 45(1):5–32, 2001.

[4] Tlamelo Emmanuel, Thabiso Maupong, Dimane Mpoeleng, Thabo Semong, Banyatsang Mphago, and Oteng Tabona. A survey on missing data in machine learning. Journal of Big Data, 8, 10 2021.

[5] Xiao Gu and Norman Matloff. A different approach to the problem of missing data. In Proceedings of the Joint Statistical Meetings (JSM), 2015. arXiv:1509.04992.

[6] James Honaker, Gary King, and Matthew Blackwell. Amelia ii: A program for missing data. Journal of Statistical Software, 45(7):1–47, 2011.

[7] Alexander Kowarik and Matthias Templ. Imputation with the R package VIM. Journal of Statistical Software, 74(7):1–16, 2016.

[8] Roderick J. A. Little and Donald B. Rubin. Statistical Analysis with Missing Data. Wiley, 3rd edition, 2020.

[9] Norman Matloff. Statistical Regression and Classification: From Linear Models to Machine Learning. Chapman and Hall/CRC, 2017.

[10] Norman Matloff. toweranNA: A Method for Handling Missing Values in Prediction Applications, 2023. R package version 0.2.0.

[11] Xiaoye Miao, Yangyang Wu, Lu Chen, Yunjun Gao, and Jianwei Yin. An experimental survey of missing data imputation algorithms. In IEEE Access, 2023. IEEE, 15 benchmark datasets evaluated, state-of-the-art imputation algorithms.

[12] Swj Nijman, E. R. Groenwold, and T. P. Moons. Missing data is poorly handled and reported in prediction model studies using machine learning: a literature review. Journal of Clinical Epidemiology, 142:218–229, 2022.

[13] Alexandre Perez-Lebel, Michael A. Koyejo, and Brandon J. Marlin. Benchmarking missing-values approaches for predictive models on health databases. GigaScience, 11:giac013, 2022.

[14] R. Quinlan. Auto MPG. UCI Machine Learning Repository, 1993. DOI: https://doi.org/10.24432/C5859H.

[15] Donald B. Rubin. Multiple Imputation for Nonresponse in Surveys. John Wiley & Sons, Inc., New York, 1987

[16] Joseph L. Schafer. Analysis of Incomplete Multivariate Data. Chapman and Hall/CRC, 1 edition, 1997.

[17] Daniel J. Stekhoven and Peter B¨uhlmann. MissForest—non-parametric missing value imputation for mixed-type data. Bioinformatics, 28(1):112–118, 2012.

[18] Jonathan A. C. Sterne, Ian R. White, John B. Carlin, Michael Spratt, Patrick Royston, Michael G. Kenward, Angela M. Wood, and James R. Carpenter. Multiple imputation for missing data in epidemiological and clinical research: Potential and pitfalls. BMJ, 338:b2393, 2009.

[19] Stef van Buuren. Flexible Imputation of Missing Data. CRC Press, Boca Raton, FL, 2nd edition, 2018.

[20] Stef van Buuren and Karin Groothuis-Oudshoorn. Mice: Multivariate imputation by chained equations in r. Journal of Statistical Software, 45(3):1–67, 2011