Comprehensive Study of Data Imputation Techniques For Machine Learning Models

Vaibhav Tummalapalli

Atlanta, USA
vaibhav.tummalapalli21@gmail.com

Abstract:
Missing data is a significant challenge in machine learning, particularly in the development of propensity models where accurate predictions depend on complete and reliable data. This paper provides a comprehensive exploration of various imputation techniques tailored for machine learning workflows, specifically in the context of propensity modeling. Each technique is categorized by its applicability to different types of data and scenarios of missingness. The goal is to equip practitioners with the tools and knowledge to effectively handle missing data, ensuring robust and accurate propensity models.

Keywords: Imputation, Machine Learning, K-Nearest Neighbors, Distance metrics, Iterative Imputation, Weight of Evidence, Propensity Models.

I. INTRODUCTION

In real-world datasets, missing values are inevitable and can significantly impact analysis and model performance. Missing data arises from various factors such as system errors, data entry mistakes, or respondent non-responses in surveys. To address this, imputation methods are employed to estimate and replace missing values, ensuring the integrity of the dataset. Understanding the nature of missingness—whether **MCAR (Missing Completely at Random)**, **MAR (Missing at Random)**, or **MNAR (Missing Not at Random)**is critical for selecting the appropriate imputation technique [1] [4].
Handling missing data effectively requires understanding its nature, which can typically be categorized into three types:

A. MCAR (Missing Completely at Random)

- **Definition**: The probability of a value being missing is unrelated to both observed and unobserved data.
- **Characteristics**: Introduces no systematic bias into the analysis. The missingness is purely random, meaning the likelihood of missing data is the same for every observation.
- **Example**: During a survey, some respondents accidentally skip a question due to an interface glitch, regardless of their demographics or responses to other questions. In a dataset of car sales, missing entries for customer phone numbers occur purely due to random data entry errors.
- **Handling**: Dropping rows or columns with missing values is often acceptable because the missingness does not depend on the data itself. Simple imputation techniques like replacing with the mean, median or mode work effectively [5].

B. MAR (Missing at Random)

- **Definition**: The probability of missingness is related to observed data but not to the missing data itself.
- **Characteristics**: Requires identifying relationships between the observed data and the missingness pattern. Can be addressed using imputation techniques that account for the relationship between observed variables.
- **Example**: In a healthcare dataset, blood pressure values might be missing more often for younger patients because they are less likely to undergo regular checkups. In a marketing dataset, income might be missing for respondents who have lower education levels, as they may be less willing to disclose financial information.
- **Handling**: Use predictive imputation techniques like regression imputation or Multiple Imputation by Chained Equations (MICE) [2], [3], which model the missing values based on other observed variables.

Creating missing indicators can also help capture the relationship between observed data and missingness for use in machine learning models.

C. MNAR (Missing Not at Random)

- **Definition**: The probability of missingness is directly related to the value of the missing data itself.
- **Characteristics**: Introduces bias into the dataset because the missingness depends on the unobserved data. Requires domain expertise to properly address the issue.
- **Example**: In a survey, individuals with higher incomes might be less likely to disclose their earnings, resulting in missing income values. In a churn dataset, customers who are dissatisfied might be less likely to provide feedback, making dissatisfaction-related fields missing more frequently.
- **Handling**: Domain knowledge is essential to model the underlying mechanisms of missingness. Techniques include creating proxy variables or indicators for missingness, Using advanced imputation strategies like Bayesian modeling [1],[8] or latent variable modeling to estimate the missing values, and remove the variable entirely from the analysis if its missingness significantly affects the model's accuracy.

II. IMPUTATION TECHNIQUES

A. Simple Imputation

- **Mean Imputation**: For a variable X, the mean imputed value X_{imputed} is:

$$X_{\text{imputed}} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

- **Median Imputation**: Median imputation replaces missing values with the **median** of the non-missing values in the variable. The **median** of a variable X is the value that separates the dataset into two equal halves when sorted in ascending order.

Let $\{x_1, x_2, \ldots, x_n\}$ represent the observed (non-missing) values of the variable X. Sort these values in ascending order. If n (number of observed values) is odd, the median is:

$$\text{Median}(X) = x_{\left(\frac{n+1}{2}\right)}$$

If n is even, the median is:

$$\text{Median}(X) = \frac{x_{\left(\frac{n}{2}\right)} + x_{\left(\frac{n}{2}+1\right)}}{2}$$

Replace each missing value in X with Median(X).

Example: If the observed values of X are {3,7,1,9,5}, the sorted values are {1,3,5,7,9}, and the median is 5. Replace all missing values with 5.

- **Mode Imputation:** Mode imputation replaces missing values with the **mode** of the non-missing values in the variable. The **mode** of a variable X is the value that occurs most frequently among the observed values.

Let $\{x_1, x_2, \ldots, x_n\}$ represent the observed (non-missing) values of X. Count the frequency of each unique value in $\{x_1, x_2, \ldots, x_n\}$. Identify the value $\hat{x}$ with the highest frequency:

$$\hat{x} = \arg\max_{x} \text{Frequency}\,(x)$$

Replace each missing value in X with $\hat{x}$

Example: If the observed values of X are {3,5,3,7,9,5,3}, the mode is 3 (appears 3 times). Replace all missing values with 3

Advantages of Simple Imputation

- **Ease of Implementation**: Simple imputation methods like mean, median, mode, or constant value are straightforward to implement and do not require complex algorithms or computations.
- **Low Computational Cost**: These methods are computationally inexpensive and work well for small to medium-sized datasets.
- **Baseline Method for Comparison**: Simple imputation serves as a baseline against which more advanced imputation techniques can be compared.
- **Works Well for MCAR**: Simple imputation is reasonably effective when data is **Missing Completely at Random (MCAR)**, as there is no bias introduced by the missingness pattern.

Limitations of Simple Imputation

- **Loss of Variability**: Imputed values like the mean or median are fixed and do not capture the variability of the data, potentially leading to biased estimates and reduced variability in the dataset.
- **Introduction of Bias**: When data is **Missing Not at Random (MNAR)** or **Missing at Random (MAR)**, simple imputation may introduce bias because it does not account for relationships between variables.
- **Distorted Data Distribution**: Mean or median imputation can alter the underlying distribution of the data, particularly when the proportion of missing data is high. **Example**: Imputing the mean for a highly skewed dataset will artificially reduce skewness, distorting the data distribution.
- **Inappropriate for Complex Relationships**: Simple imputation does not leverage multivariate relationships between variables, leading to suboptimal imputations for datasets with complex interactions.
- **Impact on Downstream Models**: Imputed values may lead to biased or overfitted models, particularly for machine learning algorithms sensitive to data distribution (e.g., regression models).
- **Does Not Handle Correlations**: Simple imputation does not account for correlations between features, potentially leading to inconsistent imputations.

B. KNN Imputation

K-Nearest Neighbors (KNN) imputation is a powerful technique for handling missing data, leveraging the similarity between instances in the dataset. The algorithm works by identifying the k-nearest neighbors for each missing value based on the available features and imputes the missing values using the data from those neighbors.

For a missing value x_i , the imputation is:

$$x_i = \frac{1}{k}\sum_{j \in N_k} x_j$$

where N_k represents the k-nearest neighbors.

Steps to impute value using KNN:

- Compute pairwise distances. The most used metric is the **Euclidean distance**, but other metrics like **Manhattan distance**, **Minkowski distance**, or even domain-specific metrics can be used.
- For two rows (instances) A and B with n features (x_1, x_2, …, x_n), the **Euclidean distance** is defined as:

$$d(A,B) = \sqrt{\sum_{i=1}^{n} w_i (x_{iA} - x_{iB})^2}$$

- x_{iA} , x_{iB} : Values of feature i for instances A and B, respectively.
- w_i: weight for feature i. Typically, w_i =1 if the feature is used in the distance calculation and 0 if it is missing for either instance.
- The summation includes only features where both x_{iA} and x_{iB} are non-missing.
- Identify the k-nearest neighbors.
- Impute using mean/mode of the neighbors.

Note that k is a hyperparameter and needs to be tuned. Try different values of k and check how your model performs. Start with small values and gradually increase the value of k. You could also use error metrics like Mean squared error or mean absolute error especially if you have the ground truth values of the columns.

Advantages of KNN Imputation

- **Handles Both Numerical and Categorical Data**: KNN can impute missing values for both numerical (using mean/median) and categorical variables (using mode).
- **Leverages Multivariate Relationships**: Unlike simpler imputation techniques (e.g., mean, median), KNN uses relationships across multiple features to provide contextually meaningful imputations.
- **Preserves Data Distribution**: Imputed values are drawn from the dataset itself, preserving the distribution of the data.
- **No Assumptions About Data Distribution**: KNN does not assume any specific data distribution, making it versatile for a wide range of datasets [7].

- **Handles Complex Patterns**: Can model complex interactions between variables when finding similar instances, resulting in more accurate imputations.

Limitations of KNN Imputation

- **Computationally Expensive**: KNN requires pairwise distance calculations between all rows for each missing value, making it slow for large datasets. High memory usage as the algorithm needs to store all data points.
- **Sensitive to Outliers**: Outliers in the data can skew distance calculations, resulting in poor imputations.
- **Feature Scaling is Crucial**: Numerical features with large ranges can dominate distance calculations unless scaled properly.
- **Choice of k (Hyperparameter)**:
  - Selecting an appropriate number of neighbors (k) is non-trivial and often requires experimentation.
  - Small k may lead to overfitting; large k may over smooth the imputation.
- **Handling Missing Data in Neighbors**: When multiple features are missing, finding neighbors becomes challenging as distance metrics exclude missing values.
- **Imputation Bias for Sparse Data**: For datasets with high levels of missing data, KNN may struggle to find similar neighbors, leading to biased imputations.
- **Domain Knowledge Required**: While KNN can impute missing values, it may not account for underlying biases or domain-specific nuances (e.g., MNAR data).
- **Not Ideal for High-Dimensional Data**: The "curse of dimensionality" reduces the effectiveness of distance-based methods like KNN. As dimensions increase, the distance between points becomes less meaningful.
- **Deterministic Nature**: By default, KNN produces deterministic imputations that may not capture the variability inherent in the data.

C. Regression/Iterative Imputation

Iterative imputation, commonly implemented using regression models, is a powerful approach for imputing missing values by leveraging relationships between variables. Here's an expanded explanation with practical insights and steps:

For variable Y with missing values:

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

Overview

- Each variable with missing values is modeled and predicted using the remaining variables in the dataset.
- The process is repeated in cycles, refining the imputed values as the algorithm progresses.
- The method continues until the imputed values stabilize (convergence) or a predefined stopping criterion is met.

Assume you have 4 variables in your data X1, X2, X3 and X4 for which you want to perform imputation

Steps for Iterative Imputation

- **Initialization:** All missing values are initialized with simple estimates (e.g., mean, median, or constant values). This provides a starting point for the iterative process.
- **First Iteration:** For each variable with missing values:
- **Impute X1**: Build a regression model using X2, X3, X4 as predictors. Predict the missing values in X1 using this model.
- **Impute X2**: Build a regression model using X1 (imputed), X3, and X4.Predict the missing values in X2.
- Repeat for all variables (X3, X4).
- **Subsequent Iterations**: For each variable, use the most recent imputed values of other variables to rebuild the regression model and refine predictions. The algorithm cycles through all variables in each iteration, progressively improving the imputations.

- **Convergence:** The process continues until the imputed values stabilize, meaning the changes between successive iterations are negligible or a predefined maximum number of iterations is reached.

D. Weight of Evidence

Weight of Evidence (WOE) imputation is a method primarily used in binary classification problems, where the goal is to replace missing values in a way that preserves the relationship between features and the target variable. This approach transforms variables into bins and assigns a weight to each bin based on its predictive power for the target variable.

Steps for WOE Imputation

- **Bin the Data:** Divide the feature into discrete intervals or bins. This can be done manually (using domain knowledge) or automatically (using techniques like quantile binning or decision tree splits). For missing values, create a dedicated "missing" bin to ensure they are handled separately.
- **Calculate WOE for Each Bin:** For each bin, compute the Weight of Evidence using the formula.

$$WOE = \ln\left(\frac{\text{Proportion of Events in Bin}}{\text{Proportion of Non-Events in Bin}}\right)$$

- **Impute Missing Values**: Assign the WOE of the "missing" bin to all missing values in the feature. For non-missing values, replace them with the WOE of their respective bins.
- **Use Transformed Features**: The feature with imputed WOE values can now be used in the machine learning model.

Advantages of WOE Imputation

- **Preserves Target Variable Relationship:** WOE imputation ensures that the imputed values retain a meaningful relationship with the target variable, improving model performance [6].
- **Handles MNAR Effectively:** By creating a separate bin for missing values, WOE imputation accounts for the possibility that missingness is related to the target variable.
- **Improves Interpretability:** WOE-transformed variables are easy to interpret in logistic regression and other linear models.
- **Mitigates Outlier Influence:** By binning the data, WOE reduces the impact of outliers.

Limitations of WOE Imputation

- **Requires a Binary Target:** WOE is inherently designed for binary classification and is not directly applicable to multi-class or regression problems without modification.
- **Sensitive to Small Bins:** Small bins with few observations can lead to unstable WOE values. Using a minimum bin size is crucial.
- **Loss of Information:** Binning the data reduces granularity, potentially losing detailed patterns.
- **Manual Effort for Bin Creation:** If automatic binning methods are not used, creating bins requires domain expertise and significant manual effort.

E. Handling MNAR (Missing Not at Random)

Missing Not at Random (MNAR) occurs when the likelihood of missingness depends on the missing values themselves. This is a challenging scenario, but there are strategies to handle MNAR effectively:

Introducing a Missing Indicator Variable

- Create a binary variable to indicate whether a value is missing (1 for missing, 0 for non-missing).
- This variable can be used as an additional feature in the model to capture patterns associated with missingness.
- Helps the model explicitly account for the missingness pattern, especially if the missingness is related to the target variable (as is common in MNAR).

Use Domain-Specific Knowledge

- **Leverage Expertise**: Use domain knowledge to determine why data is missing and make informed decisions about imputation or exclusion. Example: In a survey, if high-income individuals are less likely to report income, domain knowledge can inform adjustments to correct for this bias.
- **Custom Imputation Rules**: Impute missing values using logical constraints or business rules based on what is known about the data. Example: For an MNAR variable like "credit limit," missing values may be set to a conservative default based on regulatory or business norms.

III. CONCLUSION

Imputation is a foundational step in data preprocessing that significantly impacts the downstream performance of predictive and Machine learning models. By understanding the nature of missingness and applying appropriate techniques, data practitioners can ensure robust and reliable analyses.

REFERENCES:

1. R. J. A. Little and D. B. Rubin, *Statistical Analysis with Missing Data*, 2nd ed. Hoboken, NJ: Wiley, 2002.
2. 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.
3. J. L. Schafer, *Analysis of Incomplete Multivariate Data*. London: Chapman & Hall, 1997.
4. P. D. Allison, *Missing Data*. Thousand Oaks, CA: Sage Publications, 2002.
5. J. W. Graham, "Missing data analysis: Making it work in the real world," *Annual Review of Psychology*, vol. 60, pp. 549–576, 2009.
6. T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York: Springer, 2009.
7. J. Honaker and G. King, "What to do about missing values in time-series cross-section data," *American Journal of Political Science*, vol. 54, no. 2, pp. 561–581, 2010.
8. J. A. C. Sterne et al., "Multiple imputation for missing data in epidemiological and clinical research: potential and pitfalls," *BMJ*, vol. 338, p. b2393, 2009.