Dhirubhai Ambani
Institute of Information and Communication Technology

IT462: Exploratory Data Analysis

Group - 11

Assignment 2: Missingpy Package

Dataset: Flight Delay and Cancellation

Group Details	
Name	Student ID
Akshat Kadia	202203029
Gunjan Sethi	202312112
Jaydeep Darji	202411029

Github Link

Contents

1 Introduction

The missingpy library is a Python package designed to handle missing data in machine learning, with a focus on advanced imputation techniques. It is particularly useful when dealing with datasets where missing values cannot simply be dropped, and where simple imputation methods like mean or median imputation might not be effective. missingpy implements two primary imputation algorithms: K-Nearest Neighbors (KNN) imputation and MissForest imputation, both widely considered for their performance in predictive tasks involving incomplete data.

2 When to Use missingpy

2.1 Complex Missing Data Patterns:

Simple imputation techniques like filling with the mean or median may work in some cases, but missingpy is designed to handle more complex relationships between variables.

2.2 Mixed Data Types:

If your dataset contains continuous and categorical variables, MissForest imputation is particularly useful because it can simultaneously impute both types of data.

2.3 Predictive Modeling:

If we are working on a predictive modeling task and want to preserve the relationships between features even when data is missing, the KNN or MissForest imputers can help maintain these relationships by considering the structure of the entire dataset.

3 How missingpy Works

3.1 Data Reprocessing:

Before applying missingpy, it's essential to reprocess the data appropriately. The missing values should be represented using NaN in a NumPy array or a Pandas DataFrame. Both algorithms in missingpy handle missing values internally.

3.2 Fitting the Model:

The imputation classes in missingpy use a fit() or fit_transform() method similar to other machine learning libraries like scikit-learn. The method trains the imputer on the provided dataset and fills the missing values accordingly.

3.3 Imputation Process:

Depending on the algorithm chosen (KNN or MissForest), the missing values are either filled based on the nearest neighbors' values or predicted using Random Forests.

3.4 Post-Imputation:

Once the missing values are imputed, you can use the filled dataset for further analysis, such as training machine learning models. The imputed data will be returned in the same format as the input.

4 K-Nearest Neighbors (KNN) Imputation

The K-Nearest Neighbors (KNN) imputation method provided by the missingpy package is a technique for filling in missing data based on the values of similar (or "neighboring") data points.

4.1 What Is KNN Imputation?

- KNN imputation fills in missing data by finding the k-nearest neighbors (rows in the dataset that are closest to the row with the missing value) based on some distance metric (e.g., Euclidean distance for numerical data). It then imputes the missing value by either:
  - Taking the mean or median (for continuous data) of the neighbors' values for the feature with missing data.
  - Taking the most frequent value (for categorical data).

4.2 How KNN Imputer from missingpy Works

missingpy.KNNImputer is used to perform KNN-based imputation for missing values. It works for both continuous (numerical) and categorical (discrete) data. The imputer has the following steps

1. **Identify Missing Values:** The algorithm looks for rows with missing values in the dataset.
2. **Calculate the Distance:** The distance between rows is calculated using a distance metric (by default, Euclidean distance) across the non-missing columns of the rows.
3. **Impute Missing Value:**The missing value is imputed based on the values of the nearest neighbors:
   - For numerical data: It computes the mean of the neighbors' values for that feature.
   - For categorical data: It assigns the most frequent category among the neighbors.

4.3 Pros and Cons of KNN Imputation

4.3.1 Pros:

- Works for both categorical and continuous data.
- KNN works well when there are complex, non-linear relationships between features.
- Unlike methods like mean or median imputation, KNN doesn't assume anything about the data distribution.

4.3.2 Cons:

- KNN imputation involves calculating distances between rows, which can be slow for large datasets.
- Outliers can heavily influence imputation, especially if n_neighbors(number of neighbors to use for imputation) is small.
- If the dataset is not representative or there are significant differences between neighbors, the imputation might be biased.

5 MissForest Imputation

- **Decision tree:** A decision tree is a machine-learning model that makes decisions based on a series of conditions. It can be used for both classification and regression.
- **Random Forest:** A Random Forest is a connection of many decision trees. It's a type of connection learning method, where multiple decision trees are combined to improve the performance and reduce the risk of overfitting.

5.1 What is MissForest?

MissForest is an imputation method based on Random Forests. The core idea behind MissForest is to use Random Forests to predict missing values for each feature based on the non-missing values of other features.

- For each feature with missing data, a Random Forest regressor or classifier is trained on the observed data.
- The missing values in that feature are then predicted using this trained model.
- This process is performed iteratively, meaning the imputation is refined over several iterations.

5.2 How MissForest Works

5.2.1 Initial Imputation:

- First, it fills in the missing values using an initial guess. The common approach is to fill in the mean or the most frequent category.

5.2.2 Iterative Process:

1. For each feature with missing values, a Random Forest model (either a regressor or classifier) is built using the non-missing data.
2. The model is then used to predict the missing values for that feature.
3. The imputed values from this iteration replace the missing values from the previous step.

5.2.3 Convergence:

- The process is repeated iteratively until the imputations stabilize, i.e., when the difference between successive iterations becomes small or meets a stopping criterion. The algorithm checks if the imputed values are changing significantly with each iteration, and if not, it stops.

5.2.4 Imputation for Continuous and Categorical Data:

- **Continuous Features:** Random Forest regressors are used, and the imputed value is the predicted mean from the Random Forest model.
- **Categorical Features:** Random Forest classifiers are used, and the imputed value is the predicted category from the classifier.

5.3 Pros and Cons of MissForest

5.3.1 Pros:

- MissForest is capable of imputing both continuous and categorical data, making it more versatile than methods that handle only one type of data.
- MissForest can capture non-linear relationships between features, which simpler methods like mean imputation or KNN imputation may miss.
- It refines the imputed values over iterations, ensuring that the final result is based on the best available information. Also, MissForest doesn't make strong assumptions about the data distribution.

5.3.2 Cons:

- MissForest can be slow, especially on large datasets with many missing values, as Random Forests are computationally expensive, and the iterative nature increases the workload.
- Random Forest models require a lot of memory, so MissForest may be impractical for very large datasets or when running on systems with limited resources.
- While MissForest performs well in capturing complex relationships, it may still introduce some bias in the imputed values, particularly if the number of missing values is large.

6 Conclusion

- In the 'missingpy' library, the **KNNImputer** and **MissForest** functions offer two robust methods for imputing missing data.
- The KNNImputer fills in missing values by identifying the nearest neighbors in the dataset and averaging or selecting the most common value from them, making it effective for datasets with local similarities.
- MissForest uses Random Forest models to iteratively predict missing values, leveraging the power of decision trees to handle both categorical and continuous data, and is particularly useful for capturing complex, non-linear relationships in the data.
- Both methods are more advanced and accurate than simple imputation techniques, with MissForest being better suited for more complex datasets.