Data and Information Quality Project

2022 – 2023

Project ID: 7

Project Number: 1

Assigned Dataset: ecoli.csv

Assigned Task: Classification

Student: Andrea Chizzola (10614212)

POLITECNICO
MILANO 1863

Index

1. Introduction

Data quality is an important aspect of developing successful machine learning models. The quality of the data used to train a model can have a major impact on the model's ability to accurately make predictions and perform well on unseen data. Poor quality data can lead to a variety of problems, including overfitting and biased results. Hence, it is essential to ensure that the data used to train a machine learning model is accurate, consistent, and free of errors and biases.
It is often required to deal also with missing data. In many cases, datasets will contain missing or incomplete values, which can impact the performance of a machine learning model. This is because many machine learning algorithms are not designed to handle missing values, and as a result, they may produce unreliable or inaccurate predictions.
To address this problem, it is important to identify and replace missing values in a dataset before training a machine learning model. This can be done by using a variety of imputation techniques which aim to fill in the missing values in a way that does not negatively impact the model's performance. In particular, the following pages will provide a description of how Median Imputation and MICE (Multiple Imputation by Chained Equations) Imputation can affect the performance of different classification algorithms, namely Logistic Regression and Support Vector Machines (SVM).

2. Dataset Preprocessing

The data provided for the project is the Ecoli[1] dataset, which contains information about protein localization sites. By analyzing the content of the dataset, the first column proved to contain a different value for each row. Hence it was dropped, as it wouldn't have provided any useful information for the classification task. Moreover, given the strong class imbalance showed in Fig.1, the classes 'omL', 'imS' and 'imL' were deleted from the dataset as the number of samples belonging to each of them was not sufficient to allow proper generalization capabilities.
Before applying the selected classification algorithms, the dataset was given as input to a preprocessing function (*dirty_completeness.injection*) which created five different versions of the initial dataset, each containing 50%, 60%, 70%, 80% and 90% of correct values respectively, while setting the remaining values as missing (*NaN*).

[1] Dataset: https://archive.ics.uci.edu/ml/datasets/ecoli

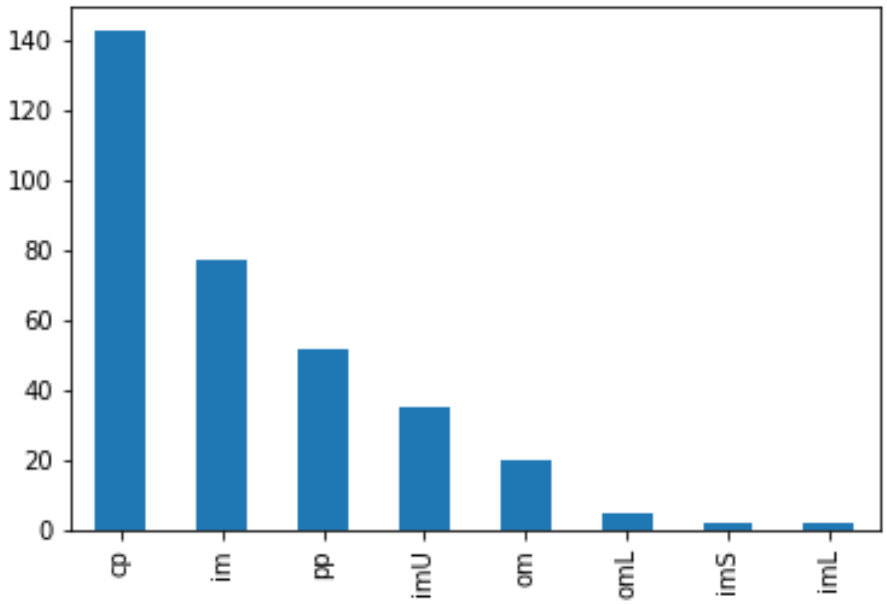

Fig. 1

3. Imputation Algorithms

A possible approach to deal with missing data could be to drop the rows containing a missing value. Despite its simplicity, this should however be avoided as it may excessively reduce the size of the dataset and cause the loss of valuable information.
Instead, in order to replace the missing values injected in the preprocessing step, two imputation techniques were used: Median and MICE Imputation.

3.1 Median Imputation

Median imputation works by replacing a missing value with the median of the corresponding column. It has some disadvantages as it may not be appropriate for data that is not normally distributed, and it may not accurately represent the central tendency of the data in all cases. However, it also has some good qualities as it is robust to outliers and its simplicity makes it easier and more efficient to compute and use.

The Root Mean Squared Error (RMSE) was used to evaluate the imputation performance, and as showed in Fig.2, it provided a similar behavior in all the different versions of the dataset.

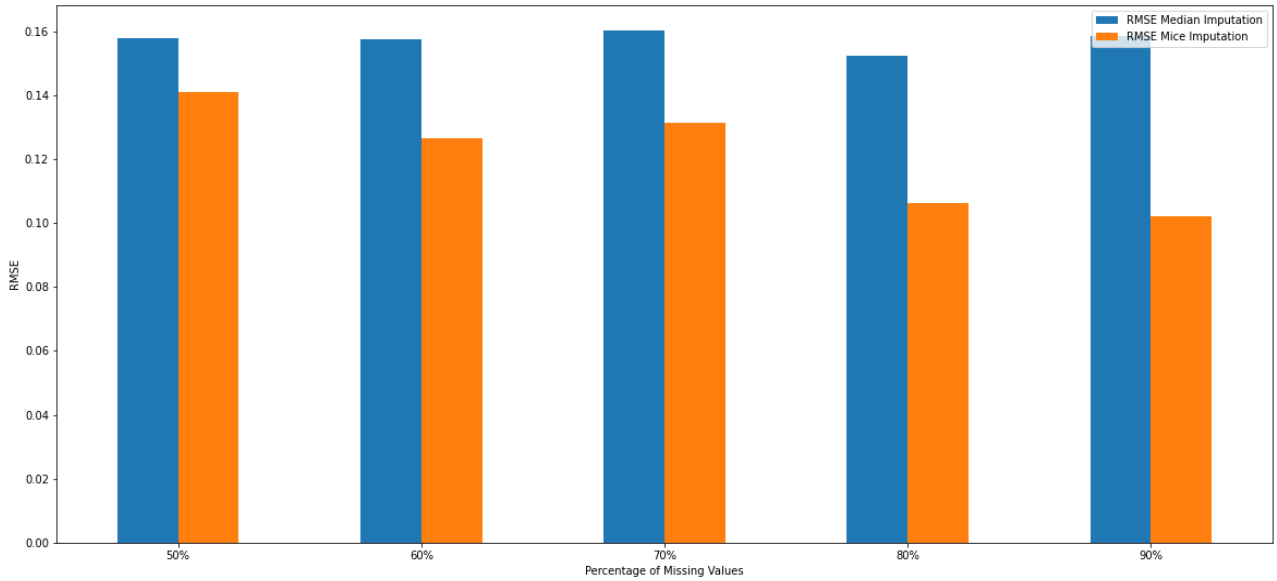

Fig. 2

3.2 MICE Imputation

MICE operates under the assumption that given the variables used in the imputation procedure, the missing data are Missing At Random (MAR), which means that the probability that a value is missing depends only on observed values and not on unobserved values. By creating multiple imputations, as opposed to single imputations, it also accounts for the statistical uncertainty in the imputations. Moreover, differently from single imputation algorithms, it imputes the missing values by looking at the data of the other columns. The chained equation process can be broken down into four general steps:

- **Step 1:** A simple imputation is performed for every missing value in the dataset. These imputations can be thought of as “place holders”.
- **Step 2:** The “place holder” imputations for one variable (“var”) are set back to missing.

- **Step 3:** The observed values from the variable "var" in Step 2 are regressed on the other variables in the imputation model. Hence, "var" is the dependent variable in a regression model and all the other variables are independent variables in the regression model.
- **Step 4:** The missing values for "var" are then replaced with predictions (imputations) from the regression model. When "var" is subsequently used as an independent variable in the regression models for other variables, both the observed and these imputed values will be used.
- **Repeat steps 2–4** for each variable that has missing data. The cycling through each of the variables constitutes one iteration. At the end of one cycle all the missing values have been replaced with predictions from regressions that reflect the relationships observed in the data.

In the actual implementation, the imputations were performed using a KNN Regressor considering 15 neighbors.

Like with Median imputation, RMSE was used to evaluate its performance. As showed in Fig.2, as the percentage of missing values decreases, the error of the imputation becomes smaller, reaching overall better results than Median imputation.

4. Classification Algorithms

Classification algorithms are a type of machine learning algorithm used to predict the class or category to which a given data point belongs. These algorithms are trained using a labeled dataset, where the correct class for each data point is known. The goal of a classification algorithm is to accurately predict the class of new, unseen data. This is done by finding patterns and relationships in the training data and using these patterns to make predictions on new data.

In the project, two classification algorithms were selected: Logistic Regression and Support Vector Machines (SVM).

4.1 Logistic Regression

Logistic Regression is an algorithm typically used for binary classification. It performs a classification analysis that predicts the probability of an event occurring, such as the likelihood that a data point belongs to a certain class.

To estimate this probability, also denoted as *posterior class probability*, it uses the *logistic sigmoid function* as follows:

$$p(C_i|\varphi) = \sigma(w^T\varphi)$$

Where φ is the feature vector (or a nonlinear transformation of it) and w is the weight vector containing the parameters of the model. The Maximum Likelihood approach is then applied to estimate the parameters, which tries to maximize the probability of predicting the correct class.

Logistic regression can also be used for multiclass classification, where the dependent variable can take on more than two possible values. In this case, the model will estimate the probabilities for each class, and the predicted class will be the one with the highest probability, selected after applying a *softmax* transformation.

4.2 Support Vector Machines (SVM)

Support Vector Machines (SVM) is another supervised learning algorithm used for classification tasks. The main goal of SVM is to find the optimal line (in two dimensions), or hyperplane (in multiple dimensions), that maximally separates the different classes in the training data. This line or hyperplane is called *decision boundary*, and the points that are closest to it are called *support vectors*. The SVM approaches this problem through the concept of *margin*, which is defined to be the smallest distance between the decision boundary and any of the samples. In support vector machines the decision boundary is chosen to be the one for which the margin is maximized.
One of the main features of SVM is that they use a kernel function to map the data into a higher-dimensional space, where it becomes linearly separable. This is also known as "kernel trick" and it allows SVM to work well with data that is not linearly separable in the original feature space.

4.3 Classification Evaluation

The confusion matrix was used to evaluate the classification performance. It is obtained by computing the number of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN).
More specifically, in a binary classification problem, a predicted value is said to be a:

- True Positive if it correctly predicts Positive class sample.
- False Positive if it wrongly predicts a Positive class sample.
- True Negative if it correctly predicts a Negative class sample.
- False Negative if it wrongly predicts a Negative class sample.

The obtained values are then used to calculate some of the metrics typically used for classification models:

- Accuracy is the ratio of the number of correct predictions to the total number of predictions:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

- Recall is the ratio of true positives to all the positives in the dataset:

$$Recall = \frac{TP}{TP + FN}$$

- Precision is the ratio of the correct positive predictions to the total number of positive predictions:

$$Precision = \frac{TP}{TP + FP}$$

- F1-score is the harmonic mean of Precision and Recall:

$$F1 - score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

These measurements were then adapted to the multiclass classification case by applying a *macro average*.

The results obtained when using the datasets filled with Median and MICE imputation are showed in Fig.3 and Fig.4 respectively. As expected, as the percentage of missing values decreases, the performance of both Logistic

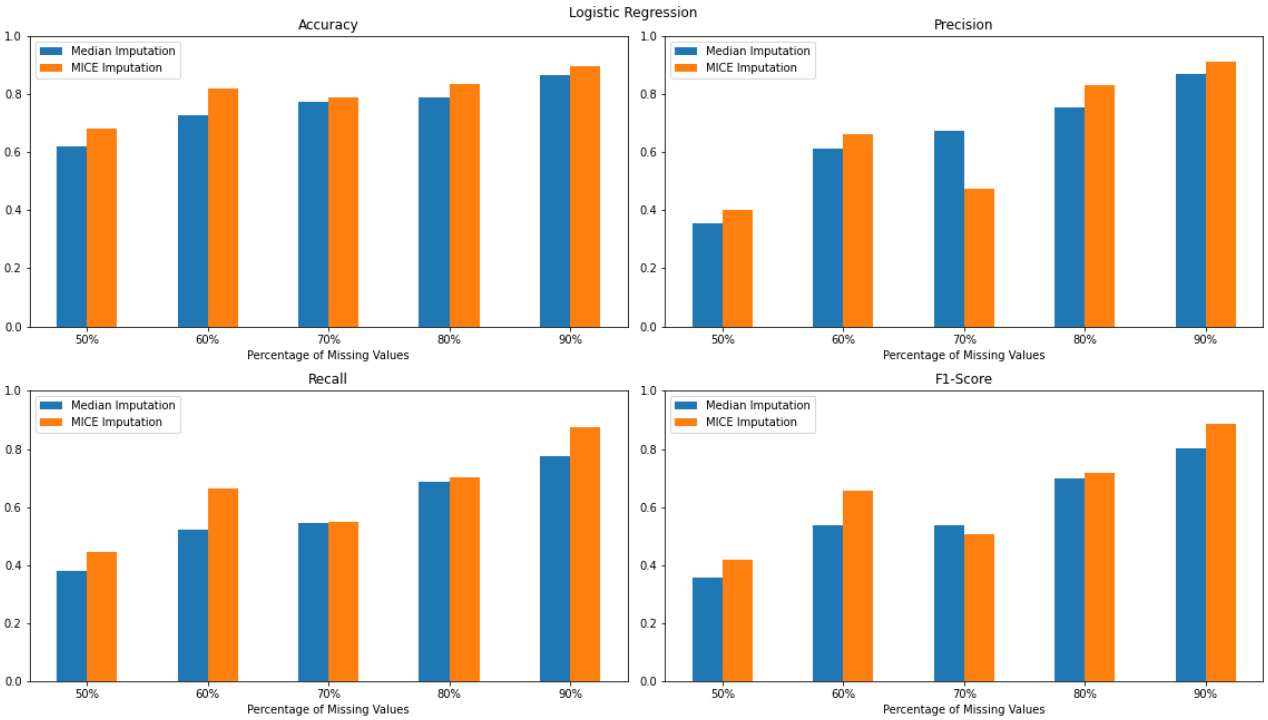

Fig. 3 Classification with Logistic Regression

Regression and SVM increases. Moreover, it can be seen how a more accurate imputation method, such as MICE, leads to better classification results, as the previously mentioned metrics obtained higher results compared to the Median imputation case.

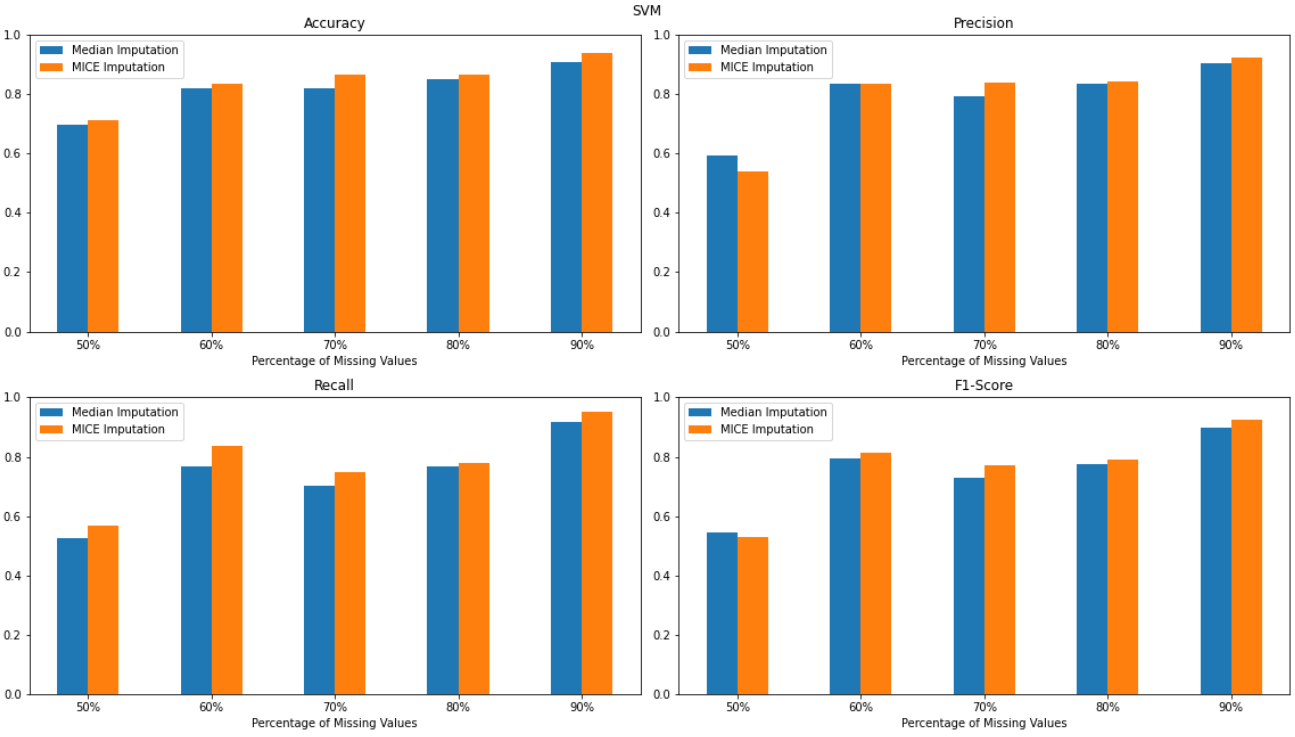

Fig. 4 Classification with SVM