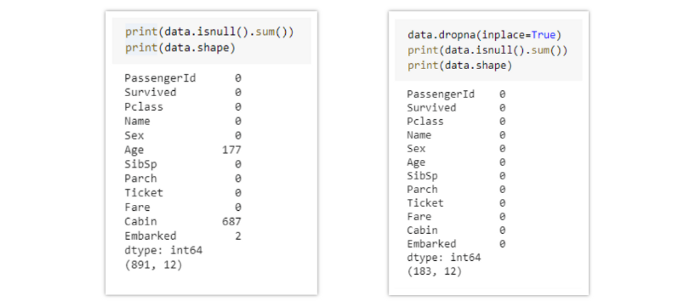
<https://towardsdatascience.com/7-ways-to-handle-missing-values-in-machine-learning-1a6326adf79e>

This article covers 7 ways to handle missing values in the dataset:

1. Deleting Rows with missing values
2. Impute missing values for continuous variable
3. Impute missing values for categorical variable
4. Other Imputation Methods
5. Using Algorithms that support missing values
6. Prediction of missing values
7. Imputation using Deep Learning Library — Datawig

Delete Rows with Missing Values:

Missing values can be handled by deleting the rows or columns having null values. If columns have more than half of the rows as null then the entire column can be dropped. The rows which are having one or more columns values as null can also be dropped.



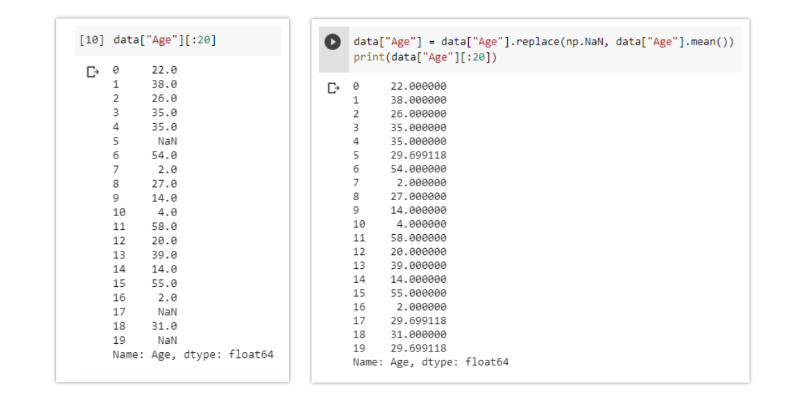
Pros:

* A model trained with the removal of all missing values creates a robust model.

Cons:

* Loss of a lot of information.
* Works poorly if the percentage of missing values is excessive in comparison to the complete dataset.

Impute missing values with Mean/Median:



Pros:

* Prevent data loss which results in deletion of rows or columns
* Works well with a small dataset and is easy to implement.

Cons:

* Works only with numerical continuous variables.
* Can cause data leakage
* Do not factor the covariance between features.

Imputation method for categorical columns:

When missing values is from categorical columns (string or numerical) then the missing values can be replaced with the most frequent category. If the number of missing values is very large then it can be replaced with a new category.

Pros:

* Prevent data loss which results in deletion of rows or columns
* Works well with a small dataset and is easy to implement.
* Negates the loss of data by adding a unique category

Cons:

* Works only with categorical variables.
* Addition of new features to the model while encoding, which may result in poor performance

Other Imputation Methods:

Depending on the nature of the data or data type, some other imputation methods may be more appropriate to impute missing values.

For example, for the data variable having longitudinal behavior, it might make sense to use the last valid observation to fill the missing value. This is known as the Last observation carried forward (LOCF) method.

For the time-series dataset variable, it makes sense to use the interpolation of the variable before and after a timestamp for a missing value.

Using Algorithms that support missing values:

All the machine learning algorithms don’t support missing values but some ML algorithms are robust to missing values in the dataset. The k-NN algorithm can ignore a column from a distance measure when a value is missing. Naive Bayes can also support missing values when making a prediction. These algorithms can be used when the dataset contains null or missing values.

The sklearn implementations of naive Bayes and k-Nearest Neighbors in Python do not support the presence of the missing values.

Another algorithm that can be used here is RandomForest that works well on non-linear and categorical data. It adapts to the data structure taking into consideration the high variance or the bias, producing better results on large datasets.

Pros:

* No need to handle missing values in each column as ML algorithms will handle them efficiently.

Cons:

* No implementation of these ML algorithms in the scikit-learn library.

Prediction of missing values:

In the earlier methods to handle missing values, we do not use the correlation advantage of the variable containing the missing value and other variables. Using the other features which don’t have nulls can be used to predict missing values.

The regression or classification model can be used for the prediction of missing values depending on the nature (categorical or continuous) of the feature having missing value.

Pros:

* Gives a better result than earlier methods
* Takes into account the covariance between the missing value column and other columns.

Cons:

* Considered only as a proxy for the true values