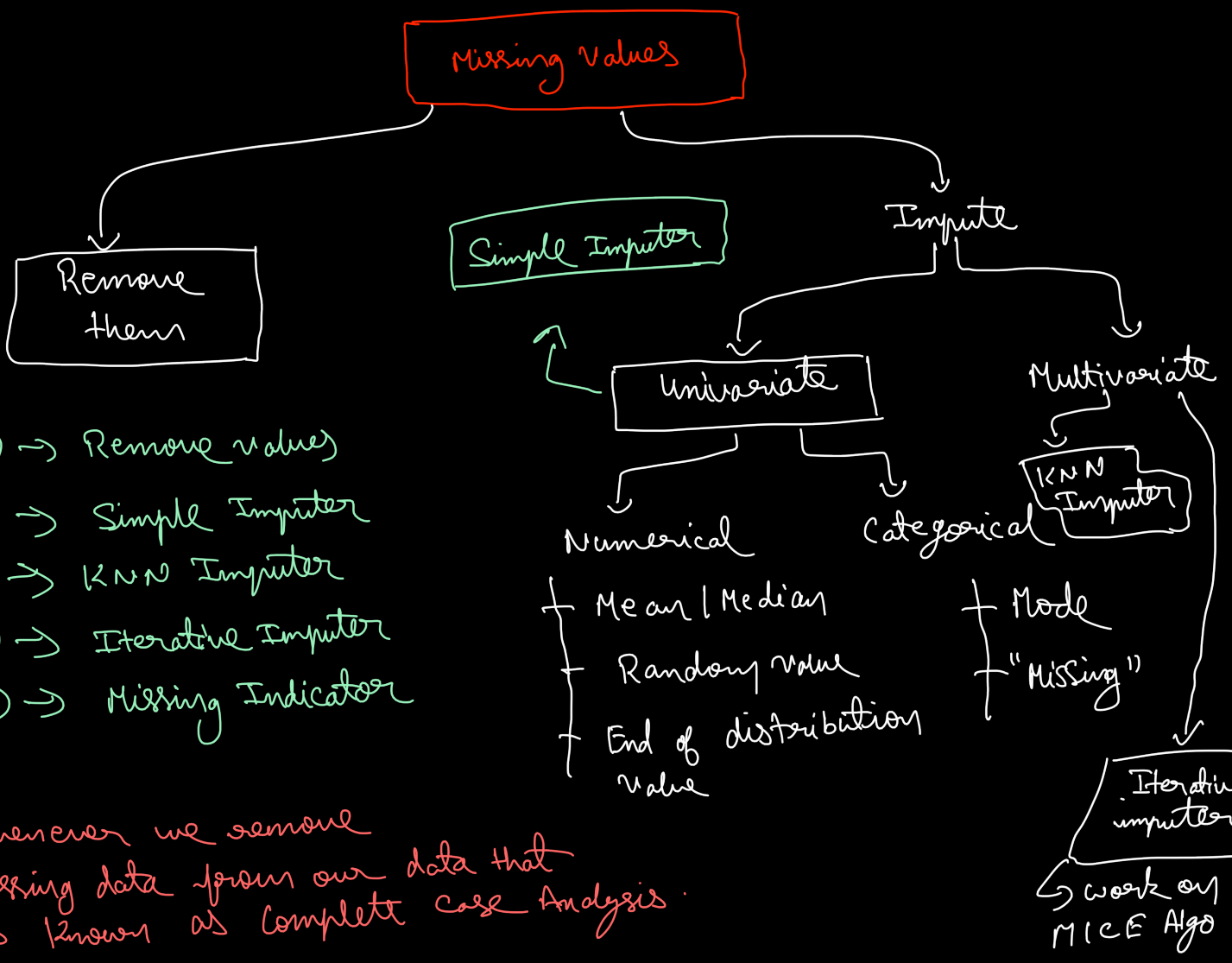
Handling Missing Data



→ ① → Remove values
→ ② → Simple Imputer
→ ③ → KNN Imputer
→ ④ → Iterative Imputer
→ ⑤ → Missing Indicator

* whenever we remove missing data from our data that is known as Complete case Analysis.

Complete case Analysis : Complete-case analysis (CCA), also called "list-wise deletion" of cases, consists in discarding observations where values in any of the variables are missing.

Rows
Columns

⇝ Complete case Analysis means literally analyzing only those observations for which there is information in all of the variables in the dataset.

Assumption for CCA

1) Missing completely at Random

Advantage / Disadvantage

Advantage :

1. Easy to implement as no data manipulation required.
2. Preserves variable distribution (if data is MCAR, then the distribution of the variables of the reduced dataset should match the distribution in the original dataset.

MCAR: Missing Completely At Random

Disadvantage :

1. It can exclude a large fraction of the original dataset (if missing data is abundant)
2. Exclude observations could be informative for the analysis (If data is not missing at random)
3. When using our models in production, the model will not know how to handle missing data.

When to use CCA ?

1. Data should be MCAR.
2. Missing Data should less than or equal to 5%.
   * also this is not rule, it depends
   * Sometime if more missing values are there, instead of removing records we can remove that particular column.

Handing Missing data | Numerical Data | Simple Imputer

→ Mean / Median Imputation

Age
27
32
Na → Mean/Median
27

Benefit

i) Simple to implement

Disadvantage : 1. change in distribution shape
2. outliers occurs
3. variance, Covariance / Co-relation with other variables changes.

When to use : When Data is MCAR
Missing Data < 5%.

→ Arbitrary value Imputation

Benefit
i) Easy to apply

Disadvantage
Same as above

(i) when dat is not MCAR.

→ End of Distribution Imputation (Extension of arbitrary value Imputation)

* In arbitrary value imputation, it is difficult to find best random value to impute. that's why end of distribution imputation comes into picture. where we pick value from end of distribution.

If data is normally distributed then : we replace missing values with (mean + 3σ) or (mean − 3σ)

If data is Skewed: (IQR Proximity)
$Q_1 - 1.5\,IQR$
$Q3 + 1.5\,IQR$

$Q_3 - Q_1 = IQR$

Benefits : easy to use Disadvantages : Same as above

↳ When to use : when data is not MCAR.

→ Random Sample Imputation

→ Automatically select best imputation technique

Handling categorical Missing Data

most frequent

"Missing"

Data should MCAR

→ Mean / Median → for numerical

missing less than 5%.
Data

→ Mode
→ frequency

Benefit : Easy to use

Disadvantages : Same as above

"Missing"

Sometime we make a new category such as "missing" and fill NA with that category.

So our ML algo come to know that this is missing and learns how to handle.

we do this when missing data is not MCAR.

And missing data > 10% or more.

→ Here after this Randomness Comes into data because we are not imputing here we are creating a different category.

Missing Indicator | Random Sample Imputation |

Random Imputation : Impute by any random value of that particular Column.

Age
- 26
- 32
- 56
- 41
- NA

→ Random numbers

=

NA

Benefit

⇝ Preserves the variance of the variable (But why?) and shape of distribution. And it good for linear algorithm.

Why: Data points which are present large in numbers, there probability to get selected randomly is high that's why shape of distribution and variance get preserved.

⇝ well suited for linear models as it does not distort the distribution, regardless of the % of NA

⇝ No impact over outliers

Disadvantages

→ Memory heavy for deployment, as we need to store the original training set to extract values from and replace the NA in coming observations.

⇝ Covariance with other variables get disturbed because of randomness.

→ If more data is missing then it is not good technique. In case of categorical variable, this can change the ratio of categories or even in case of numerical variable also variance can be changed.

Missing Indicator

↳ Make completely new column for every column which has missing data. In which we keep two values True and False. True for those record which has value and False for missing record.

Age	Fare	Age_Na
22	32	F
41	35	F
Na	41	T
62	32	F

* There is assumption with this technique that Machine Learning algorithm learns to differentiate between missing values and non-missing values. And because of this algorithm performs better some time.

* Atleast we should try it once.

Automatically select value for Imputation

↳ By Grid Search CV
↳ Go to Code

Univariate Imputation : when we fill missing values by taking help of that particular variable only.

Multivariate Imputation : when we fill missing values of any column by taking help of others variables in data.

KNN Imputer | Multivariate Imputation

↳ use KNN algorithm

↳ To calculate distance : It uses nan Euclidean distance

→ Imputation for Completing missing values using k-Nearest Neighbors.

→ Each sample's missing are imputed using the mean value from n_neighbors nearest neighbors found in the training set. Two samples are close if the features that neither is missing are close.

Nan Euclidean distances : Calculate the euclidean distances in the presence of missing values.

When calculating the distance between a pair of samples, this formula ignores feature coordinates with a missing value in either sample and scales up the weight of the remaining coordinates :

dist(x, y) = Sqrt (weight * Sq distance from present coordinate)

where, weight = Total number of coordinates / Number of present coordinates.

for example, the distance between [3, na, na, 6] and [1, na, 4, 5] is

$$\sqrt{\frac{4}{2}\left[(3-1)^2 + (6-5)^2\right]}$$

4 = Total number of coordinates

2 = number of present coordinates.

* If all the coordinates are missing or if there are no common present coordinates then NaN is returned for that pair.

Advantages & Disadvantages

- More accurate
- More number of calculation
- Prediction slow [every time when we do prediction, to calculate missing values we have to use the training data kept on server, that is also a problem. (Storage problem)]

Iterative Imputer / MICE

MICE stands for Multivariate Imputation by Chained Equations

Assumptions

category of missing data

- MCAR
- MAR
- MNAR

$\Rightarrow$ Data should Missing At Random.

This is assumption

Advantage & Disadvantages

Advantage
- accurate

Disadvantages
- Slow
- need to store training data on server to impute when we do prediction

$\Rightarrow$ How it works?

- Fill all the NaN values with mean of respective Column.
- Remove all Col1 missing values (starting from left)
- Predict the missing values of Col1 using other Columns By using any machine learning algorithm. Treat Col1 missing as output feature and rest as input features.
- do same for all Columns in which missing values are there.

→ Then find difference find data and data with mean imputed. our task is to reduce this difference close to zero.

→ After one iteration, we take first iteration final output as base and repeat the all above mentioned steps. Till we get difference close to zero.

→ Go to code for more understanding or read some blog over google.