Data Imputation

Imputation Techniques

1. Mean imputation
2. Median imputation
3. Mode imputation
4. Regression imputation
5. K-nearest neighbor (KNN) imputation
6. Multiple imputation
7. Hot deck imputation
8. Cold deck imputation
9. Stochastic regression imputation
10. Bayesian imputation
11. Random forest imputation
12. MICE (Multivariate Imputation by Chained Equations) imputation / Iterative Imputer
13. EM (Expectation-Maximization) imputation
14. Deep learning-based imputation
15. Matrix factorization-based imputation

Iterative Imputation

Iterative imputation is a technique for imputing missing values in a dataset by using a sequence of predictive models. The technique involves imputing missing values for each feature based on the other features in the dataset and updating the model with the new imputed values until convergence is reached. This technique can account for correlations between features and capture complex patterns in the data, but it can be computationally expensive and requires careful tuning of the model hyperparameters.

KNN Imputer

- KNN imputer uses NaN – Euclidian distance

SN	Column 1	Column 2	Column 3	Column 4
1	33	NaN	67	21
2	NaN	45	68	12
3	23	51	71	18
4	40	NaN	81	NaN
5	35	60	79	NaN

KNN Imputer

KNN imputer uses NaN – Euclidian distance

SN	Column 1	Column 2	Column 3	Column 4
1	33	NaN	67	21
2	NaN	45	68	12
3	23	51	71	18
4	40	NaN	81	NaN
5	35	60	79	NaN

$$dist(Point1, Point2) = \sqrt{Weight \times (Square\ of\ Disatance\ of\ Present\ Coordinate)}$$

$$Weight = \frac{Total\ Number\ of\ coordinate\ pairs}{Number\ of\ Coordinate\ pair\ without\ NaN\ value}$$

KNN Imputer

KNN imputer uses NaN – Euclidian distance

SN	Column 1	Column 2	Column 3	Column 4
1	33	NaN	67	21
2	NaN	45	68	12
3	23	51	71	18
4	40	NaN	81	NaN
5	35	60	79	NaN

$$Weight = \frac{Total\ Number\ of\ coordinate\ pairs}{Number\ of\ Coordinate\ pair\ without\ NaN\ value} \longrightarrow Weight = \frac{4}{2}$$

$$dist(Point1, Point2) = \sqrt{Weight \times (Square\ of\ Disatance\ of\ Present\ Coordinate)}$$

$$\hookrightarrow dist(Point1, Point2) = \sqrt{\frac{4}{2} \times [(67 - 68)^2 + (21 - 12)^2]} \longrightarrow 12.80$$

KNN Imputer

KNN imputer uses NaN – Euclidian distance

SN	Column 1	Column 2	Column 3	Column 4
1	33	NaN	67	21
2	NaN	45	68	12
3	23	51	71	18
4	40	NaN	81	NaN
5	35	60	79	NaN

Let’s impute the value in second row, to impute that value we need to calculate distance of second row with all other rows using previous formula
For KNN implementation let’s take value of K = 2
Now number of nearest neighbour K = 2, therefore we will look for 2 rows that are close to row number 2

Now after calculation we would know row number 3 and 4 is near to row number 2, so to impute the value in column1, row2 we will take average of value in column1 row3 and column1 row4

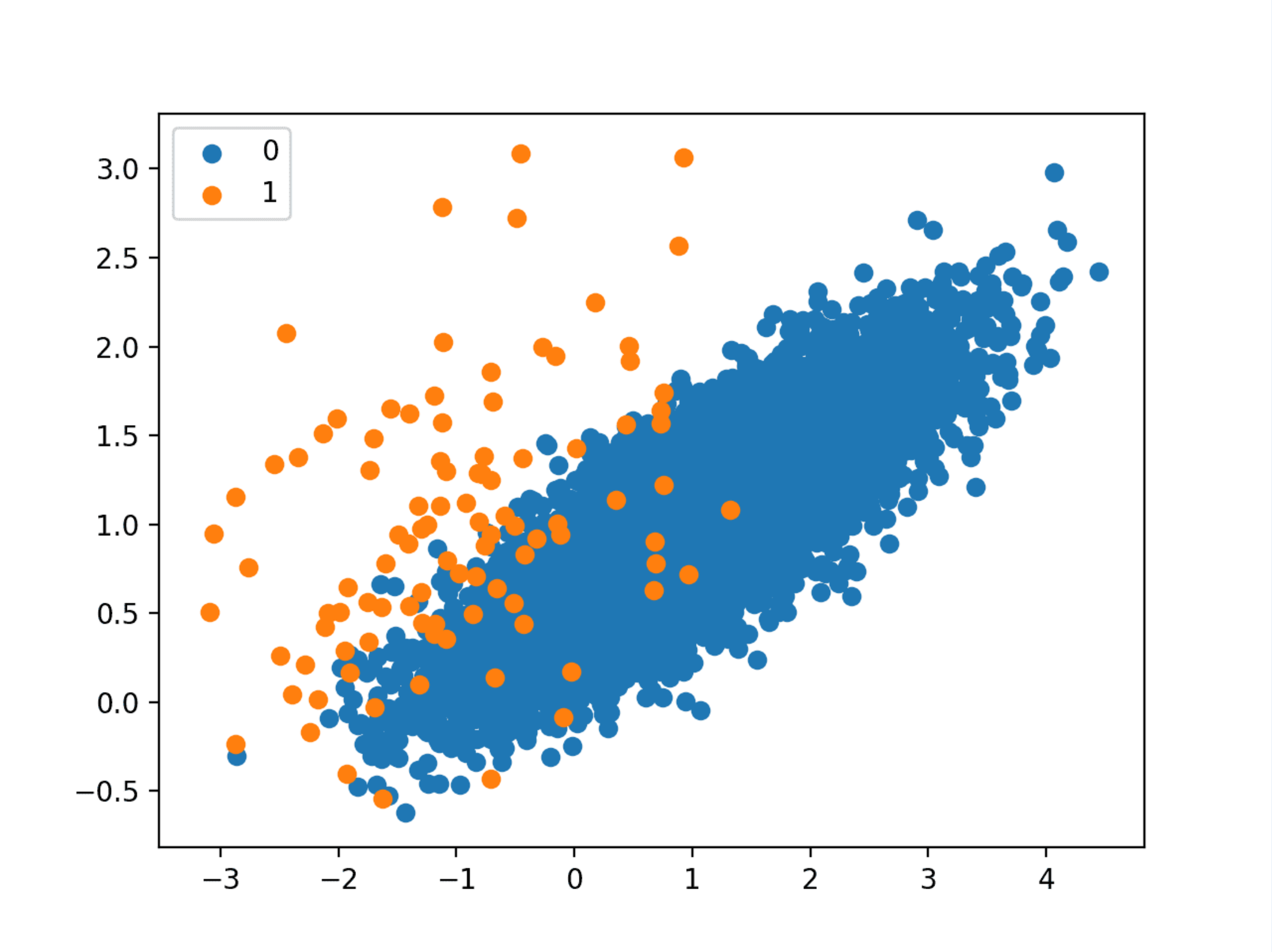
Advantages of KNN Imputer

1. Non-parametric method: The KNN imputer is a non-parametric method, which means it does not require any assumptions about the distribution of the data. It can be used with any type of data, including continuous, categorical, and binary data.
2. Flexibility: The KNN imputer can handle missing values in both numeric and categorical data. It is also flexible enough to handle different distances measures.
3. Can handle complex relationships: KNN imputer can handle complex relationships in the data that may be difficult to capture with other imputation methods.
4. Retains original data distribution: The KNN imputer imputes missing values based on the values of the nearest neighbors, so it can retain the original data distribution.

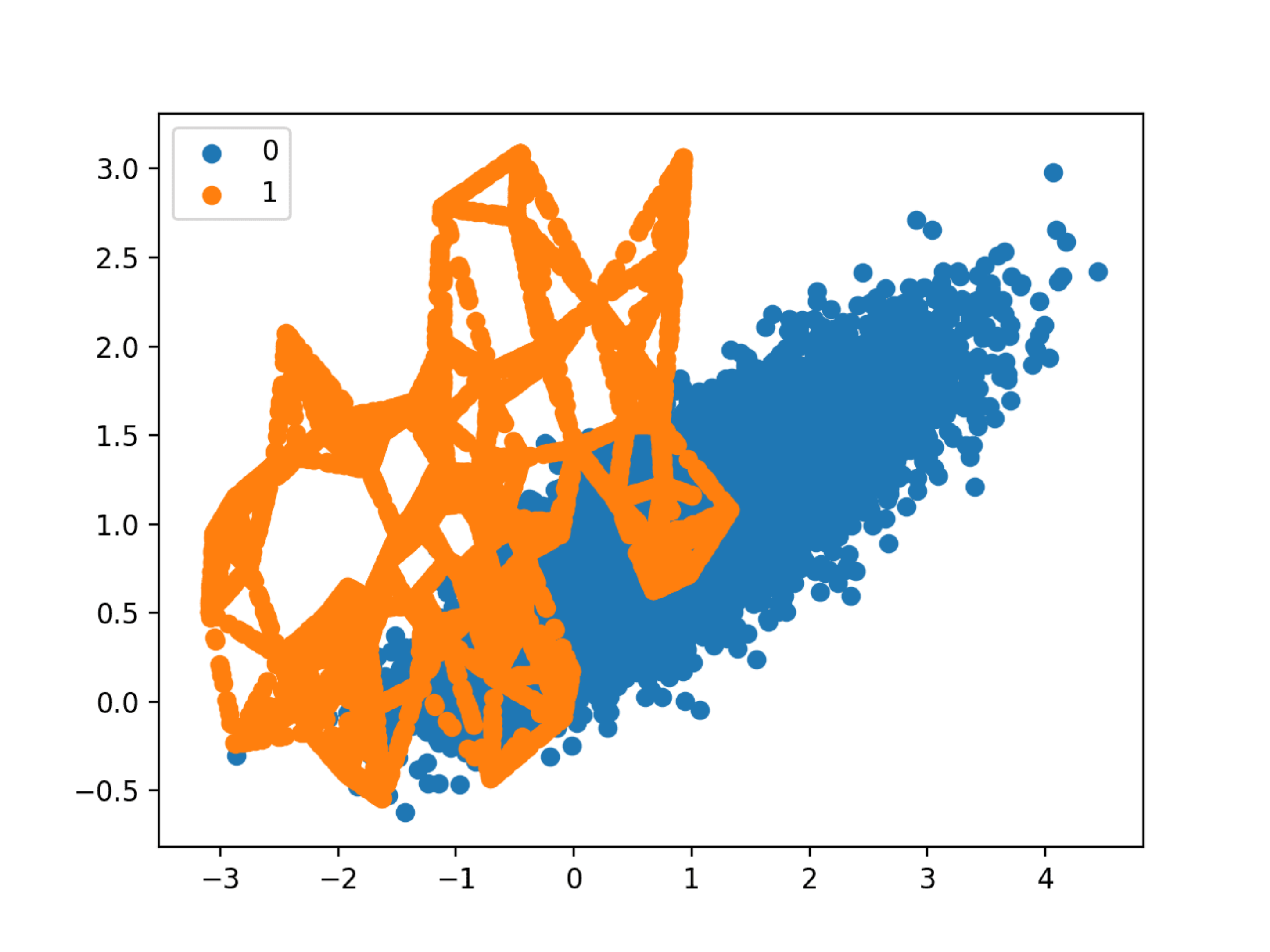
Disadvantages of KNN imputer

1. Sensitive to the number of neighbors (K): The KNN imputer requires the user to specify the number of nearest neighbors to use in imputing missing values. The performance of the imputer can be sensitive to the value of K, and selecting the optimal value can be challenging.
2. Computationally expensive: The KNN imputer is a computationally expensive method, especially for large datasets. The time required to compute the nearest neighbors increases with the size of the dataset.
3. Bias in imputed values: The KNN imputer can introduce bias in the imputed values if the data has a high level of missingness or if there are systematic differences between the missing and non-missing values. In such cases, the imputed values may not accurately reflect the true values.
4. Can't impute missing values outside of range: The KNN imputer can only impute missing values within the range of the existing values. If the missing value is outside of this range, the KNN imputer may not be able to impute a reasonable value.

SMOTE Animation



SMOTE Animation



0
1
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−3
−2
−1
0
1
2
3
4