# Data:

### Source:

|  |
| --- |
| **Abstract**: Data taken from the Blood Transfusion Service Center in Hsin-Chu City in Taiwan -- this is a classification problem. |

<https://archive.ics.uci.edu/ml/datasets/Blood+Transfusion+Service+Center> (UCI Repository)

### Features:

**Continuous Feature 1:** Monetary quantitative c.c. blood Input

**Continuous Feature 2:** Time quantitative Months Input

**Categorical Feature:** Whether he/she donated blood in March 2007 binary 1=yes 0=no

**Instances:** 600

# Methods:

### Imputation Methods:

**KNN:**

In *k-NN classification*, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small).

In *k-NN regression*, the output is the property value for the object. This value is the average of the values of *k* nearest neighbors.

**1NN:**

This algorithm is same as K-NN algorithm but If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor.

**Weighted KNN:**

In this algorithm the neighbors are found for the imputing point and the inverse of each distance is taken. The sum of each distance is considered and then divided by each inverse value, the result is the distance from the imputing point. The one with lowest distance is then imputed as the original value.

### Feature Scaling Methods:

**Standard Scaler:**

Standardize features by removing the mean and scaling to unit variance.

The standard score of a sample x is calculated as:

z = (x - u) / s

Where u is the mean of the training samples or zero if with mean=False, and s is the standard deviation of the training samples or one if with\_std=False.

**MinMax Scaler:**

Transforms features by scaling each feature to a given range.

This estimator scales and translates each feature individually such that it is in the given range on the training set, e.g. between zero and one.

X\_std= (X -X.min (axis=0)) / (X.max (axis=0) -X.min (axis=0))

X\_scaled=X\_std\* (max-min) +min

### Imputation Accuracy Methods:

The Accuracy is measured after comparing the original value and the predicted value. If the difference between the two is zero then the values match and accuracy is 100%. If the values don’t match then the count of each matched value is divided by total number of values and the accuracy percentage is calculated.

In some of the cases of continuous features the accuracy was found to be zero as the mean imputed value did not exactly match with the original integer.

### Distance Methods:

**Euclidean Distance:**

The Euclidean distance between two items is the square root of the sum of the squared differences of coordinates.

**Manhattan Distance:**

The Manhattan distance between two items is the sum of absolute differences of coordinates.

# Tools:

Python (version 3.7) language along with Spyder IDE(Anaconda 4.7.11)

Import scikit using command pip install sklearn

Import numpy using pip install numpy

# Results:



# Comparative analysis of imputation: