Report : Big Data Assignment 1

# Dataset

**Source** : <https://www.kaggle.com/dansbecker/aer-credit-card-data>

**Description** : A small credit card dataset for simple econometric analysis,

### **Content**

* **card:** Dummy variable, 1 if application for credit card accepted, 0 if not
* **reports:** Number of major derogatory reports
* **age:** Age n years plus twelfths of a year
* **income:** Yearly income (divided by 10,000)
* **owner:** 1 if owns their home, 0 if rent
* **selfempl:** 1 if self employed, 0 if not.
* **months:** Months living at current address

# Methods

## Imputation methods used :

1. **1 Nearest Neighbour**: Finds the nearest neighbour and imputes the value with neighbour’s value.

1. **K Nearest Neighbour** (KNN where k=7) : find the nearest K (here K=7) neighbours and impute with average value of the K neighbours in case of numerical attribute. But, in case of categorical attribute impute with the **Mode** of class.
2. **Weighted KNN** – here I have used distance weighted KNN where if have modified the calculation for categorical feature to incorporate its contributions as well.

## Distance measure used:

1. **Euclidian Distance (numerical) :** The distance between two points defined as the square root of the sum of the squares of the differences between the corresponding coordinates of the points;
2. **Chi-squared Distance (numerical) :** The distance between two points defined as the half of sum of the squares of the sum between the corresponding coordinates of the points divided by difference between the corresponding coordinates.
3. **Hamming Distance (categorical):** The distance between two points defined as the sum of the absolute differences between the corresponding coordinates of the points;

## Feature Scaling methods used:

1. **Z-score Scaling :** z = (X - μ) / σ

where z is the z-score, X is the value of the element, μ is the population mean, and σ is the standard deviation

1. **Min-Max Scaling:** y= (x-min)/(max-min)

where min and max are the minimum and maximum values in X, where X is the set of observed values of x.

## Accuracy measure:

1. **For Numerical Attributes :** R^2 (coefficient of determination) regression score function.

Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse). A constant model that always predicts the expected value of y, disregarding the input features, would get a R^2 score of 0.0.

1. **For Categorical Attributes :** Accuracy classification score.

In multilabel classification, this function computes subset accuracy: the set of labels predicted for a sample must exactly match the corresponding set of labels in y\_true.

# Tools and Library:

**Programming Language :** Python 3.7

**IDE :** Jupyter Labs/ Jupyter notebook

**Libraries:** Numpy, Pandas, Scikit learn, Scipy

# Results

## Accuracy output for unscaled features







## Accuracy output with Z-score Scaling





## Accuracy using Min Max Scaling







# Observation/Conclusions

* Weighted KNN tends to give more accuracy as compared to KNN.
* When we apply scaling accuracy increases for each feature.