CREDIT RISK ANALYZER

WIT­H AI

**-By Vaibhav Vats**

**Introduction**

**What is Credit Analysis?**

Credit analysis is a type of analysis an investor or bond portfolio manager performs on companies or other debt issuing entities to measure the entity's ability to meet its debt obligations. The credit analysis seeks to identify the appropriate level of default risk associated with investing in that particular entity.

**What type of information is critical?**

Some types of loans require more thorough analysis than others. Larger, long-term loans for fixed assets require more thorough analysis than short-term working capital loans. For individual loans, loan analysis and follow-up visits provide most of the guarantee for the institution and thus the analysis is necessarily more extensive. Group loans transfer most of this responsibility to the clients and therefore do not require detailed analysis.

**Credit Risk Predictive Modelling and Credit Risk Predictive By Machine Learning**

If past is any guide for predicting future events, credit risk prediction by Machine Learning is an excellent technique for credit risk management. Prediction models are developed from past historical records of credit loans, containing financial, demographic, psychographic, geographic information, etc. From the past credit information, predictive models can learn patterns of different credit default/delinquency ratios, and can be used to predict risk levels of future credit loans. It is important to note that statistical process requires a substantially large number of past historical records (or customer loans) containing useful information. Useful information is something that can be a factor that differentially affects credit default/delinquency ratios.

**Objective**

To ensure that loans are made on appropriate terms to clients who can and will pay them back. What analysis is needed and what is the most efficient approach to fulfil that need is primarily determined by the type and nature of the loan.

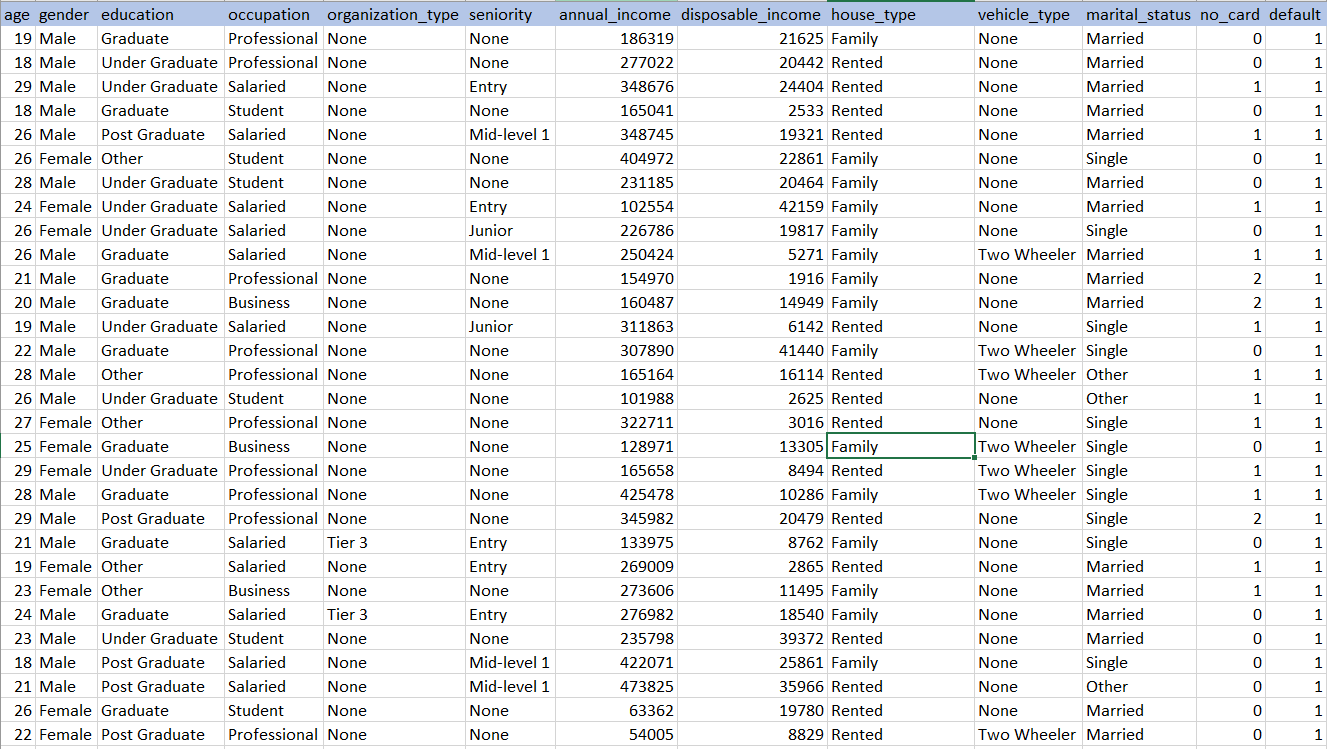
• To place good and appropriate loans - can the loan generate income for repayment and will the client repay

• Determine eligibility of the applicant - is he/she eligible according to the the program criteria

• Training needs and skills - to assess the training needs and develop the financial management skills level of the client. (This is the basic principal of programs that integrate their credit and training methodologies.)

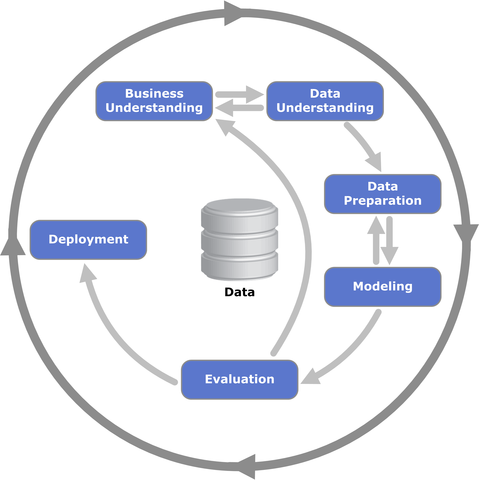
• Program Indicators - loan analysis may also be used to generate the indicators that will be used to evaluate the impact of the loan.

**Dataset**



**Methodology and Techniques Used**

**Cross-industry standard process for data mining**



CRISP-DM stands for cross-industry process for data mining. The CRISP-DM methodology provides a structured approach to planning a data mining project. It is a robust and well-proven methodology. We do not claim any ownership over it. We did not invent it. We are however evangelists of its powerful practicality, its flexibility and its usefulness when using analytics to solve thorny business issues.

**Python Modules Used**

* Scikit-learn (sklearn)
  + DecisionTreeClassifier
  + KFold
  + train\_test\_split
  + accuracy\_score
* Matplotlib
* Pandas
* NumPy

**Decision Tree**

A **decision tree** is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. It is one way to display an algorithm that only contains conditional control statements.

A decision tree is a flowchart-like structure in which each internal node represents a “test” on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes). The paths from root to leaf represent classification rules.

There are three commonly used impurity measures used in binary decision trees: **Entropy**, **Gini index**, and **Classification Error**.

**Entropy** (a way to measure impurity):

Entropy=−Sum (p \* log2p)

**Gini index** (a criterion to minimize the probability of misclassification):

Gini=1−Sum (p \* p)

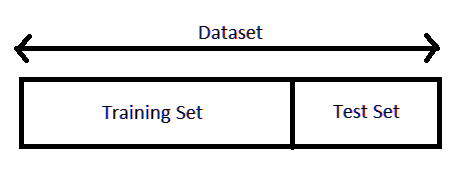
**Classification Error**:

ClassificationError=1−max(p)

where p is the probability of classes.

**Train Test Split**

As we work with datasets, a [**machine learning algorithm**](https://data-flair.training/blogs/machine-learning-algorithm/) works in two stages. We usually split the data around 20%-80% between testing and training stages. Under supervised learning, we split a dataset into a training data and test data in Python ML.



The following section will split the dataset randomly into two groups, training dataset and test dataset. We will use 70% data as training data and remaining 30% as test data.

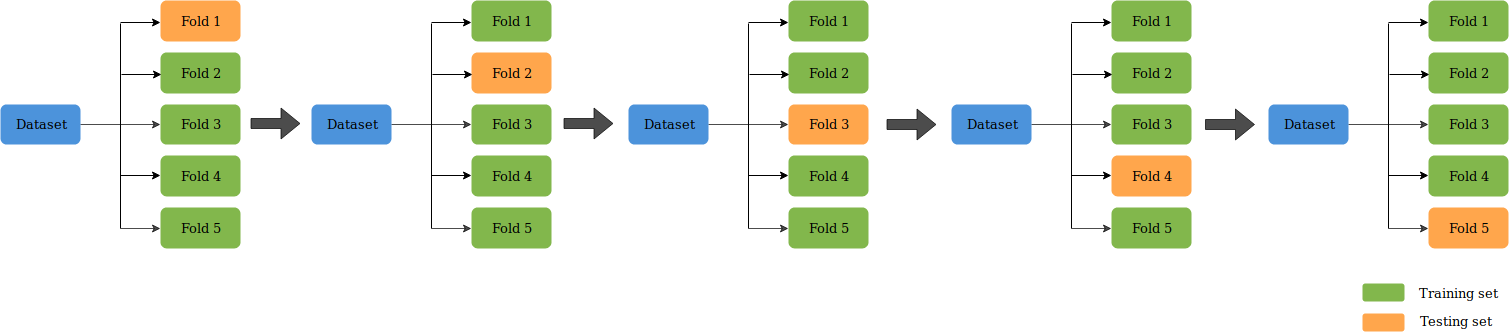
***from sklearn.model\_selection import train\_test\_split***

***x\_train, x\_test, y\_train, y\_test= train\_test\_split (X, Y, test\_size=0.3, random\_state=1234)***

**K-fold Cross Validation**

Cross-validation is a resampling procedure used to evaluate machine learning models on a limited data sample.

The procedure has a single parameter called k that refers to the number of groups that a given data sample is to be split into. As such, the procedure is often called k-fold cross-validation. When a specific value for k is chosen, it may be used in place of k in the reference to the model, such as k=10 becoming 10-fold cross-validation.



In this project value of k is 4.

**Train Decision Tree Model with Gini and Entropy Criterion-**

* K-Fold (4-Fold) Cross validation to attain high accuracy
* Predict data for every fold

Picking average of scores

**Model Summary**

* 84.46% (Gini Criterion with K-Fold)
* 84.29% (Entropy Criterion with K-Fold)
* 84.08% (Gini Criterion with train\_test\_split)
* 83.92% (Entropy Criterion with train\_test\_split)