

IST 340 Final Research Paper

Sango Fon

Claremont Graduate University

**Executive Summary:**

This research project purposes to produce a well-crafted overview of decision tree induction data mining technique and a reasonable comparison of two data mining software (SAS EM vs IBM SPSS DT/Modeler) based on specific criteria. In order to efficiently and effectively evaluate the selected DM Software, the research closely follows the Center for Data Insight (CDI) framework known as “ CDI’s tool evaluation Methodology” by C. Ken & C. Bernard in their article titled “ A Methodology for Evaluating and Selecting Data Mining Software”. Also, the study faces few limitations such a first-hand experiment obtained from building the same algorithm using the same data set and both data mining software for more detailed and reasonable comparison(apple to apple comparison in simple term) .

In addition, upon meticulous comparison of both data mining software, we found that they have some slight differences in their algorithms naming conventions and their approach to decision tree splitting methods as shown in functionality criteria below. Also, we found some similarities and differences based on Performance, Ancillary support, and usability criteria as shown in their respective sections below.

As a brief conclusion, one could say with high confidence, upon reading this paper, that both SAS EM and IBM SPSS DT/Modeler are great data mining software that can efficiently implement DT algorithms. Furthermore, both DM software help to build great models based on various algorithms and allow you to import models from various other sources. They provide a very flexible environments that can be tuned up based on specific business needs.

Table of Contents

[Introduction: 3](#_Toc39760281)

[Objectives of the paper 3](#_Toc39760282)

[List of Two Decision Tree Induction covered in the paper 3](#_Toc39760283)

[Limitation of the paper 3](#_Toc39760284)

[Overview on Decision Induction 3](#_Toc39760285)

[Description of Decision Tree: 3](#_Toc39760286)

[Decision Tree Growth: 4](#_Toc39760287)

[Decision Tree Pruning: 5](#_Toc39760288)

[Decision Tree Evaluation: 6](#_Toc39760289)

[Evaluation Criteria: 6](#_Toc39760290)

[Functionality: 6](#_Toc39760291)

[Ancillary Tasks Support: 7](#_Toc39760292)

[Performance: 7](#_Toc39760293)

[Usability: 8](#_Toc39760294)

[Description of the DT induction Software: 8](#_Toc39760295)

[SAS Enterprise Miner 8](#_Toc39760296)

[Overview 8](#_Toc39760297)

[Functionality Criteria: 9](#_Toc39760298)

[Ancillary Tasks Support: 14](#_Toc39760299)

[Performance Criteria: 14](#_Toc39760300)

[Usability: 15](#_Toc39760301)

[IBM SPSS Decision Tree/ Modeler 15](#_Toc39760302)

[Overview 15](#_Toc39760303)

[Functionality Criteria: 16](#_Toc39760304)

[Ancillary Tasks Support: 17](#_Toc39760305)

[Performance Criteria: 18](#_Toc39760306)

[Usability Criteria: 18](#_Toc39760307)

[Comparative Analysis in terms of the relevant criteria 18](#_Toc39760308)

[Comparison 20](#_Toc39760309)

[What have you learned? 20](#_Toc39760310)

[Conclusion: 20](#_Toc39760311)

[Reference 22](#_Toc39760312)

**Introduction:**

Objectives of the paper

The main goal of this research project is to provide a brief overview of decision tree induction data mining technique. Then, upon giving a brief overview of selected software ( SAS EM & IBM SPSS DT), evaluate them based on four keys criteria(Functionality, Performance, Ancillary Support, and Usability). Then, provide a brief comparative summary and finally, deduce a short conclusion based on the summary analysis of both selected software.

List of Two Decision Tree Induction covered in the paper

SAS Enterprise Miner (SAS EM) and IBM SPSS Decision Trees/ Modeler (SPSS data mining software)

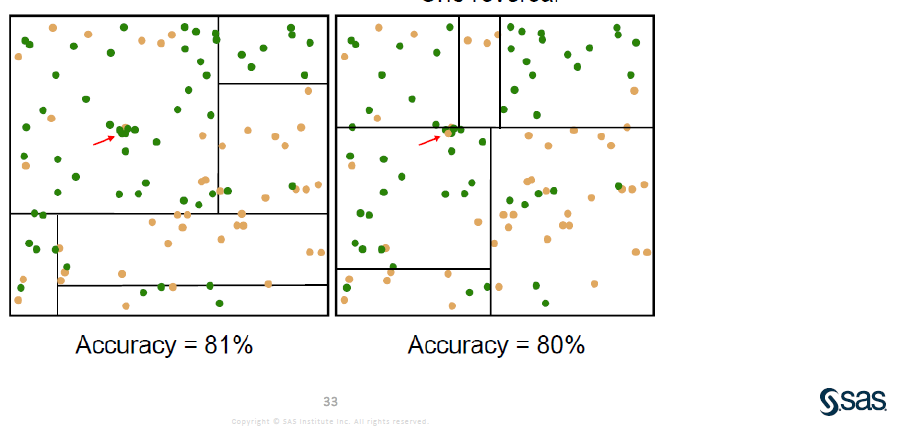
Limitation of the paper

This project is limited (constraint) by the following factors: **Time**: the time assigned to carry out well organized research and produce a well-crafted, and detailed comparison is short. In addition, there is an ongoing Covd-19 outbreak that has heavy tampered the researcher personal’s schedule. **Technical constraints**: no decision tree algorithm will be built from scratch using both software for a better functionalities appraisal. Finally, this project does not provide a full capability of each software; but does provide a concise summary of selected items for a simple and reasonable comparison of both tools.

**Overview on Decision Induction**

## Description of Decision Tree:

Decision tree (DT) are machine language algorithms that successively dissects data sets into smaller data groups based on descriptive characteristic or feature, until they reach sets smaller enough to be described by some label. Since decision tree falls under the category of supervised learning, it uses existing labeled data to predict behavior in new data. In other words, DT requires that one has data that is labelled or tagged for it to label new data based on that preexisting knowledge. Decision tree uses the tree representation to solve: classification problem ( where machines sort data into different classes. Here, the dependent variable is qualitative or categorical), and regression problem ( where machine predicts value. Here, dependent variable is continuous or quantitative). In addition, benefits from a technical viewpoint include: Robustness to outliers(the ability to withstand outlier), the ability to accommodate Missing Values, accommodate different measurement scale(can handle both continuous and categorical data), very simple to interpret, fast training time, simple models deployment into production, ability to consistently give the same accuracy when data changes, wide range of uses including(variable selection, models…etc), and the ability to provide clear indication of which variables are most important for prediction and classification. However, DTs also have many limitations including: possible structural instability with respect to perturbations in data, errors prone when the number of training examples is small, and the fact that once the algorithm makes the decision on which node to split, that decision is never revised which does not make current induction methods optimal. Another disadvantage of decision tree include the fact that a small change in data can result in a totally different looking tree. As shown in the Screen Captured below. In the image, we notice that a slight difference of 1% accuracy shows completely different graphical representation.



## Decision Tree Growth:

According to IBM Knowledge Center, “decision tree growing is done by creating a decision tree from a data set. Splits are selected, and class labels are assigned to leaves when no further splits are required or possible. Furthermore, “the growing starts from a single roots node, where a table that contains a training data set is used as input table. Then, when doing split, each of the created descendant nodes corresponds to the applicable subset of the training data set. Further splits of these nodes result in new nodes that correspond to smaller subsets of data sets, and so on. Also, nodes that are not split further become leaves ”. Elements in the decision tree growth could be briefly described as follows: Root Node (the beginning of a tree), Internal Node (splits into further nodes), Leaf Node (a node that no longer splits), and Branch( the link between nodes). In addition, decision trees apply a top-down approach when it comes to dealing with data. In the sense that for a given data set, they start by grouping data and labeling observations that have similarity between them, and then, seek for the best rules that split the observations that have dissimilarity until they reach certain degree of similarity. There are two types of splitting including binary(which splits each node into at most two sub-groups and try to find the optimum partitioning), and multiway (which splits each node into multiple sub-groups, using as many partitions as existing distinct values). In addition, examples of splitting methods includes: entropy, Gini, and Probchisq. Very briefly, the entropy method selects the best split that maximizes its reduction (entropy reduction); Gini uses the reduction in the Gini index; and then, Probchisq uses Chi-square statistics. So, a rule to thumb states that “better the value of chi-square, higher the statistical significance of the splitting rule”. Detailed information on splitting method for each dt algorithms are given in the Functionality Criteria section below.

Finally, decision tree growing is a repetition of the following operations that include: **Stopping criteria** (this operation determines whether a split is done, or whether a node becomes a leaf because it is not split further), Class label assignment(here, the assignment of class label is done for all nodes), and split selection (this operation assigns minimum-impurity splits to nodes for which the stopping criteria were not satisfied).

## Decision Tree Pruning:

As the number of decision trees splits gets bigger, it becomes difficult for the DTs model to control noises. This situation can negatively impact the performance of the model on new data by means of overfitting. So, in very simple term**,** Decision Trees Pruningrefers to a decision tree technique that help to deal with overfitting issues by reducing the size of the DTs and removing sections that does not add value to new data prediction and classification. IBM Knowledge Center reiterated this definition of pruning when it stated, “ decision tree pruning reduces the risk of overfitting by removing overgrown subtrees that do not improve the expected accuracy on new data”. Pruning is composed of two important phases including Pre-pruning and Post-pruning. So, Post-pruning Phase main goal is to generalize the decision tree that was generated in the growth phase in order to avoid overfitting. There are several strategies that help to cope with post-pruning including: Minimum Error (here, the tree is pruned back to the point where the cross-validated error is a minimum); Smallest Tree(here, the tree is pruned back slightly further than minimum); and None (where no action is taken).

Meanwhile, pre-pruning also known as early stopping, is an alternative method used to prevent overfitting by stopping the DT-building process early before it produces leaves with very small sample. Thus, concerning DT pruning, one could keep in mind that for best accuracy, post-pruning or minimum error pruning without early stopping is commonly considered a good choice. Whereas, pre-pruning is important to produce an even smaller tree or to reduce the running time while allowing accuracy to decrease.

## Decision Tree Evaluation:

The decision tree evaluation is a process of evaluating and comparing predictive models in order to choose the best model for decision making. DT evaluation is composed of many criteria and factors depending on the tools one chooses to use. Examples of the criteria may include performance measures (accuracy, simplicity, lift, stability, and precision …etc.), and the weight and the threshold assigned to those measures. Furthermore, even though there are some benchmark depending on the industry and DTs tools used, the weight and threshold assigned to each performance measures could be subjective depending on the project or evaluators. Also, the criteria of evaluation could also vary depending of the type of data (qualitative or quantitative). With this in mind, one could say with high confidence level that choosing the best decision tree would completely depending on the weight and threshold that are assigned to each performance measure.

**Evaluation Criteria:**

## Functionality:

Here, in depth comparison will be done based on the following set of criteria: Variety of Decision Tree algorithms criterion: this will help to closely look at algorithms provided in the decision tree growth phase that allow users to build different models on the same dataset in order to be able to select the best one. Model refinement techniques criterion: Some model refinement techniques, we will be comparing include for instance: Ensemble (bagging, boosting and others). Splitting criterion: here, we will closely look at different splitting methods such as GINI and Entropy such to cite a few. Also, Model assessment criteria and reporting methods (misclassification rate, lift chart, and confusion matrix) will show how each DT software assesses model and displays result.

## Ancillary Tasks Support:

In this group of criteria, the goal is to show how each software approaches the concept of data manipulation for decision tree algorithms. Based on the CDI frameworks, these criteria would include: data cleansing, value substitution, data filtering randomization, metadata manipulation, record deletion, Randomization and etc. **Data Cleansing criterion**: here, the goal is to look at how the whole data cleansing operation is done in each decision tree supported software. For instance, when comparing SAS EM and IBM SSPS, I will be able to show different steps that each takes in handling “Invalid Values” and /or “Missing Value” upon loading the data set. **Data Filtering criterion**: here, the main goal is to determine if each tool has a filter. If yes, the next step would be to depict the filtering options and how these options differentiate from one software to another. **Metadata manipulation criterion**: Here, the purpose is to explore the metadata tool within each software in order to depict how the tool present data description, data types and etc. Furthermore, this criterion would help to highlight how the tool manipulate metadata and then, briefly extract a clear comparison between both software. So, upon applying the set of criteria listed above, I will be able to come up with a well-crafted summary comparison on both DT tools, based on ancillary task support criteria.

## Performance:

In the set of criteria selected for performance, based on the CDI Framework, will help to come up with a well-crafted comparative summary at a granular level and in a reasonable manner. These criteria will include: **Platform Variety**: in this criterion, I will point out different platforms that support each DT software. **Software Architecture**: here, the objective is to depict the type of architecture used by each DT solution. **Robustness**: The goal here is to point out the software stress test. In other words, this criterion will help to drill down a comparative analysist that will to help to know the performance of each DT under stress( does the tool run consistently without crashing). However, it important to keep in mind that the analysis would solely be based available data on each DT tool not an actual experiment.

## Usability:

When interacting with any type of technology in general, the notion of usability most often prompts the idea of user interface, ease-of -use, and how intuitive the technology is. However, for an objective and deeper comparison of DT algorithm tools, the concept of usability will be dissected into several criteria following the CDI framework. As stated in CDI Framework, this view of usability is “an accommodation of different levels and types of users without loss of functionality or usefulness” by C. Ken & C. Bernard in their paper titled “ A Methodology for Evaluating and Selecting Data Mining Software”. With this in mind, usability criteria would include: **User interface criterion**, User types, Data visualization, Error Report, and Action History criterion. So, in User interface criterion, I will closely look at how user interact with each DT tool, how the navigation tool is displayed, and how easy and intuitive is it to navigate within each DT tool, and how easy it is to navigate the result dashboard.

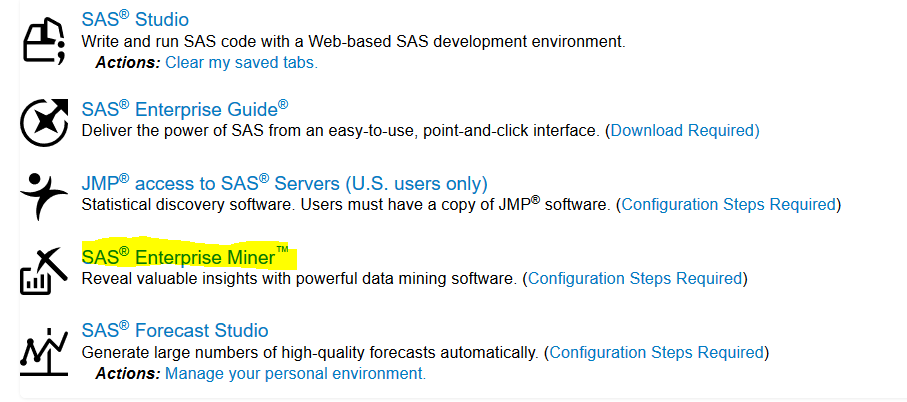
In addition, in the **User Types criterion**, the purpose is to depict the user minimum skills required to efficiently use the software. In other words, this criterion will help to answer the following question: “ is the software designed for beginning, intermediate, or advanced user types. Data Visualization: The purpose is to look at different graphic representation available in each DT software and draw a brief comparison.

**Description of the DT induction Software:**

## SAS Enterprise Miner

## Overview

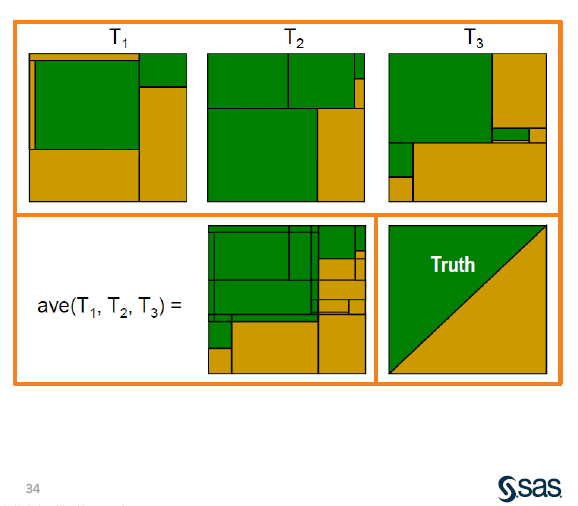
SAS Enterprise Miner is a Machine Learning Data Science Software developed by SAS Institute in 1999. It helps to build accurate predictive and descriptive models on small and large volumes of data across different sources in the organization. According to SAS Institute, “ SAS Enterprise Miner is delivered as a distributed client/server system. This provides an optimum architecture so data miners and business analysts can work more quickly to create accurate predictive and descriptive models and produce result that can be shared and incorporated into business process”. Furthermore, SAS EM is an easy-to-use GUI based software, designed to work seamlessly with other SAS solutions, such as data integration, reporting, and forecasting as shown in the image below.



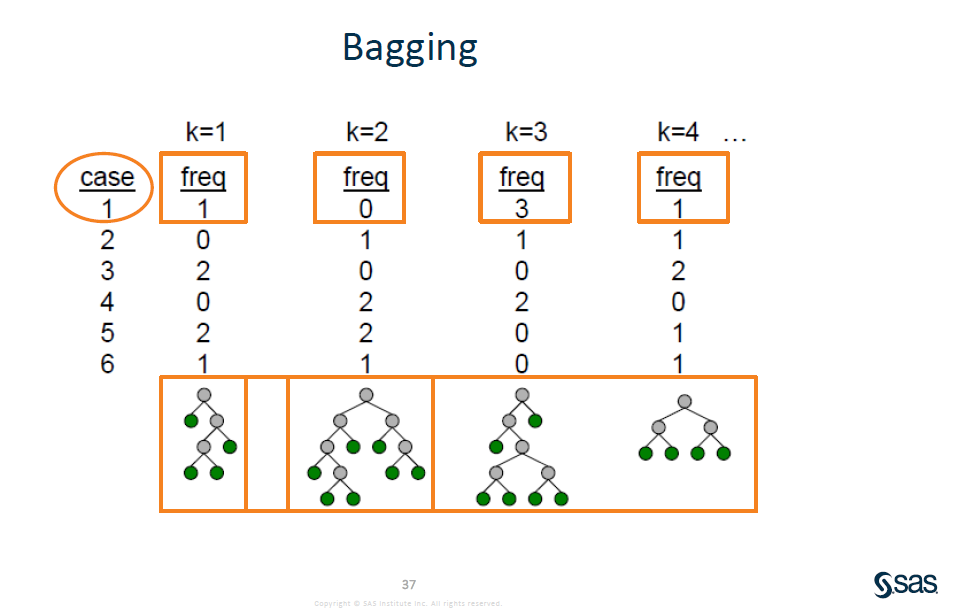
In addition, SAS Enterprise Miner offers several features and functionality that help to model data. Also, it offers a very flexible data management capabilities and data preparation to create models with an interactive, rich visualization and data exploration. It uses in-database , in-memory and grid capabilities for faster response and results. It does offer “a state-of-art predictive analytics and data mining capabilities that enable organizations to analyze very complex data, depict important insights and make sound decision or fact-based decisions”.

## Functionality Criteria:

With no further ado, SAS EM model refinement techniques criterion, includes decision trees methodologies such CHAID, classification and regression trees, bagging, boosting, gradient boosting and bootstrap forest. The general underlying idea behind these methodologies is that “ Two or more predictive models combined to create a potential more accurate model works better when model prediction are uncorrelated” by SAS Institute. Furthermore, according SAS Institute, the model refinement techniques such as ENSEMBLE Trees rely on the concept Perturb and Combined(P&C) methods which help to “generate multiple models by manipulating the distribution of the data or altering the construction method and then averaging the results as shown in the screen captured below”.



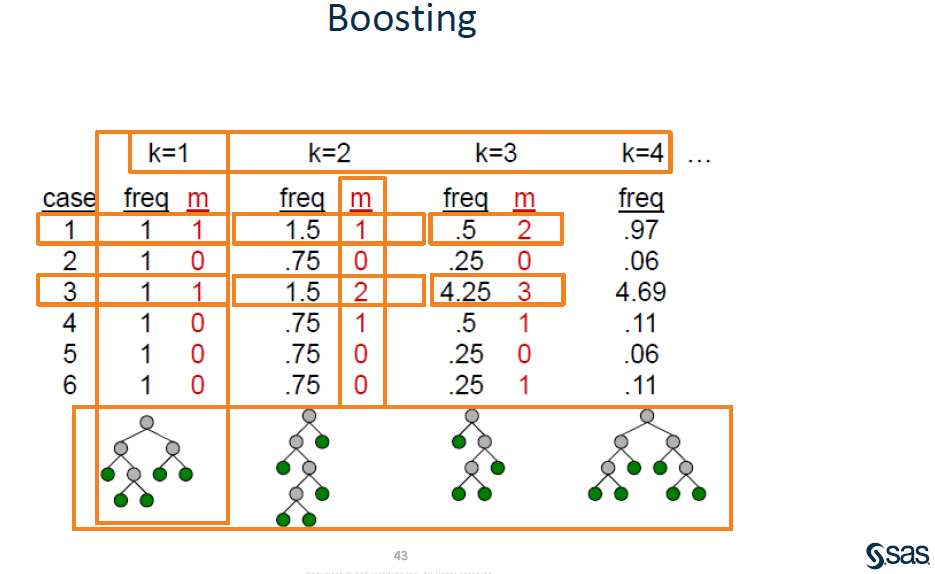
SO, a brief summary of some of the Ensemble tree components include: **Bagging**(bootstrap aggregation): It is the original Perturbed and combined method known as “Breiman”. Note that a bootstrap sample is a “random sample of size n drawn from the training data with replacement”. Here, few observations will be left out of the sample, few others will be represented more than once, the tree will be built on each sample. The following diagram by SAS clearly illustrated bagging and then, show how to build the DT algorithm on SAS EM



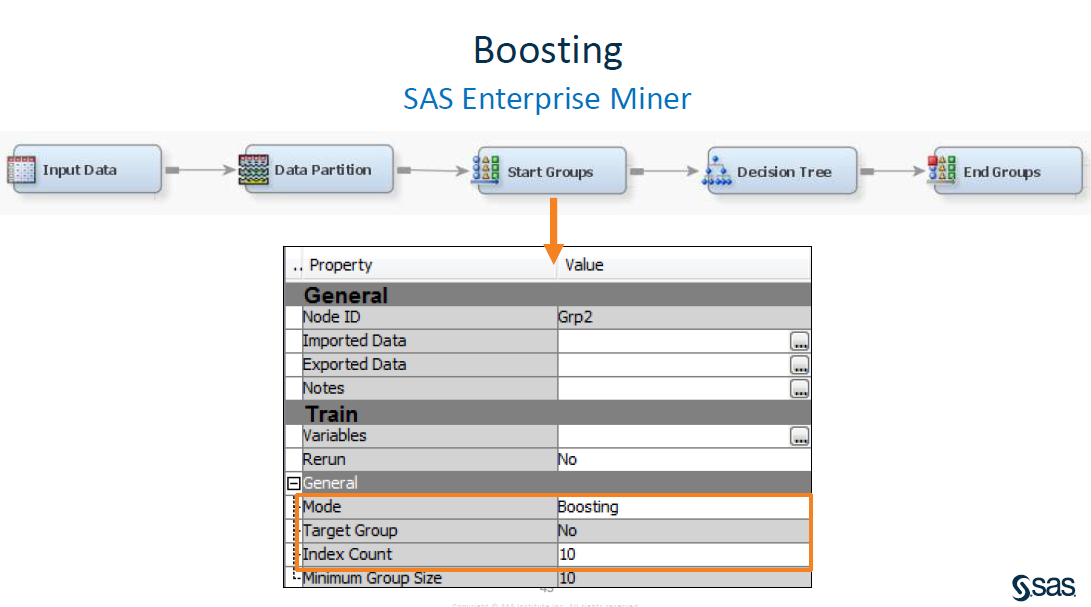


As explicitly shown in the image above, in order to build a Bagging algorithm in SAS EM, you must have a Start Groups and End Groups node. Mode and the number of iterations(index count) required are both configured in the START Groups Node 🡺 Property 🡺 General.

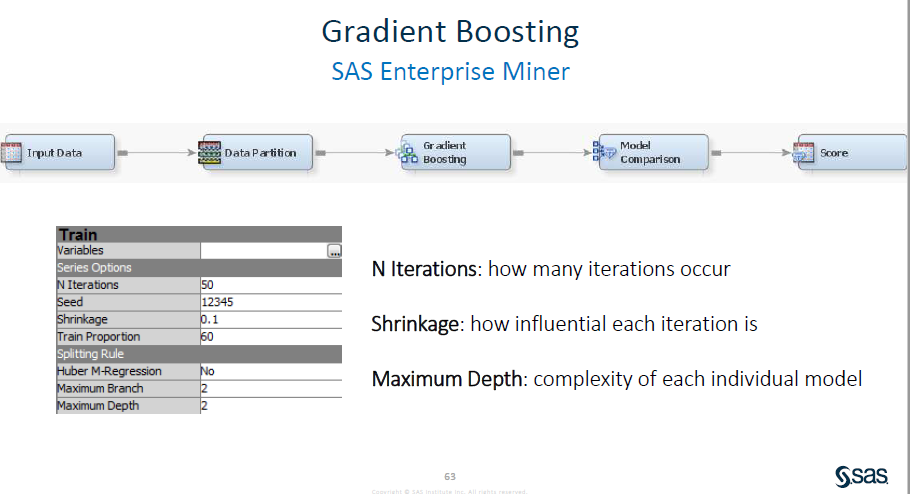
Boosting: is another important component of ENSEMBLE Tree. It is an adaptive resampling and combining (Arcing ) methods that sequentially perturb the training databased on the results obtained from a previous model.



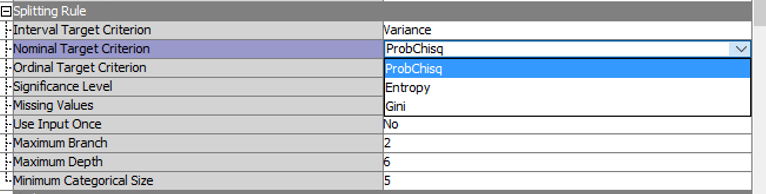
Boosting: refers to building an ensemble of models using the same supervised learning algorithm on data sampled with replacement from a single training data. Building a Boosting algorithm using SAS EM is as simple as the steps depicted in the image below.



Furthermore, gradient boosting could be seen as a combination of several decision trees, consisting of a forest decision trees known as “Shrubs”, “Stumps”. Few advantages of gradient boosting include the following: It helps to handle outliers and missing values, it help to consistently give the same accuracy when data changes. Whereas, as disadvantage, it can cause overfitting and slightly more difficult to deploy the model into production. Below is an example of gradient boosting using SAS EM.



Thus, one should keep in mind that in SAS EM , “Ensemble models are created using the posterior probabilities for class targets from multiple models”. From Dr Yan Li ENSEMBLE, slide11.

In Addition, SAS EM has several splitting methods including Pro Chi-square, Gini, Entropy, and variance reduction as shown in the screen captured below:

Finally, SAS EM include the following Assessment Measurement:

**Decision (Default):** It “Selects the tree that has the largest average profit and the smallest average loss if a profit or loss matrix is defined”. quote From SAS EM Help. Also, “the value of the model assessment is reset in the training process, depending on the measurement level of the target”. So, if the target is categorical the measure should be set to “Misclassification”. Whereas, if the target is interval, the measure should be set to “Average Square Error”.

**Average Square Error**: it does select tree that has the minimum average square error.

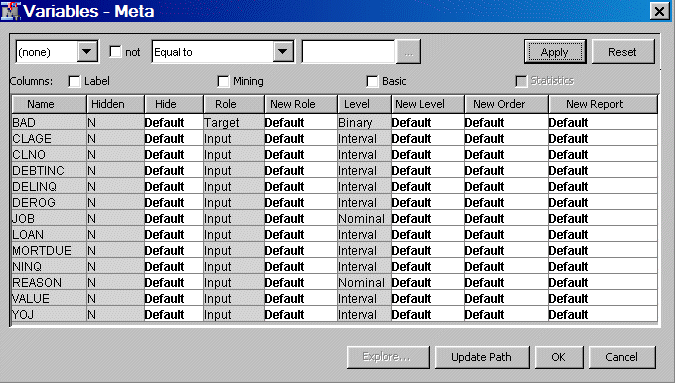
**Misclassification**: which to select the tree having smallest misclassification rate

**Lift**: Which helps to evaluate tree based on the prediction of the top n% of the ranked observations.

Furthermore, Assessment Results are presented as followed: Fit Statistics Table( the fit statistics from the model), Classification Chart( which is a bar chart showing the correct classification of the values of the target variable), Score Rankings Overlay Chart, Score Ranking Matrix, and Score Distribution Plot. Thus, as a brief conclusion of functionality criteria, keep in mind that SAS EM has a lot of tools that help to build and fine-tune a decision tree algorithm. Some of these tools are not explicitly explain in the functionality criteria but could be find in SAS EM Help page.

## Ancillary Tasks Support:

**Data Cleansing**: In SAS EM, observations that contain missing values for target variables are excluded from the analysis. So, it is advised to clean the training data, or to use impute Node to deal with missing values. Impute Node will help to clean the training data by replacing the missing target level values for instance. Data filtering: SAS EM has a Filter Node that is located on the Sample Tab of the SAS EM tools bar. It helps to create and apply filters to the training data set. Also, the filter node allows users to exclude certain observations such as outliers and errant data not needed for the specific analysis. Filtering training data tends to produce better models, partly because the parameter estimates will be more stable upon applying the filter. Metadata manipulation criterion: SAS EM has a Metadata Node that is located in the Utility tab. It helps to Sample, Explore, Modify, and Assess data. For instance, you could modify attributes such as variables roles, order, and measurement levels. Below is a screen captured of a Meta data:



## Performance Criteria:

SAS suite is an open architecture that supports compilers and procedures. It is accessible through the following operating systems: Windows, MAC OS, IBM mainframe, Unix/Linux, and OpenVMS Alpha. Furthermore, SAS EM is integrated with R programming language. So, it is possible to write code in R language inside of SAS EM. Also, you can score SAS Enterprise Miner directly inside Aster Data, DB2, Greeplum, Hadoop, IBM Netezza, Oracle, SAP HANA, SAS Scalable Performance Data Server, and Teradata databases with SAS Scoring Accelerator. In term of Scalability, you can scale SAS EM from a single-user system to very large enterprise solutions having several users, with the Java client and SAS server architecture.

## Usability:

SAS EM is an easy-to-use, drag and drop interface. It could be accessed via different operating systems including Windows, MAC OS, IBM Mainframe, and Unix/Linux just to cite a few. Furthermore, SAS Enterprise miner is built for all type of Users (Beginner, intermediate, and advanced). The software has a very rich, well organized, and informative Help page that can help to build algorithms without ambiguities.

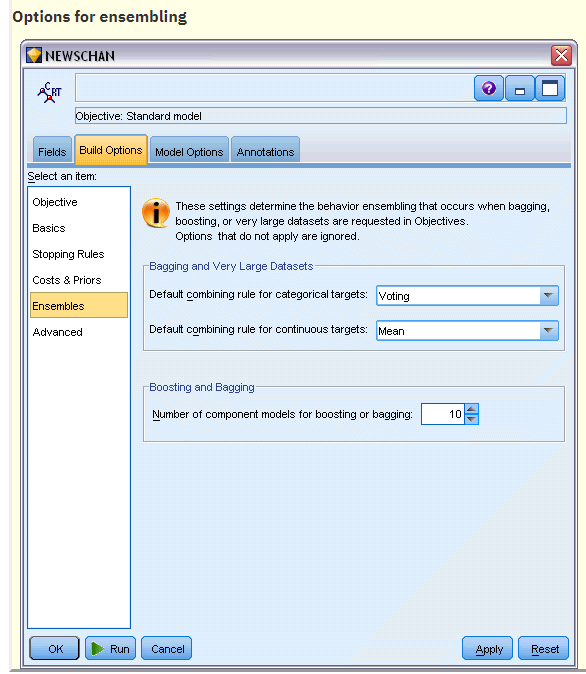
**IBM SPSS Decision Tree/ Modeler**

## Overview

IBM SPSS is a software suite that includes several solutions dealing with Statistics, Data Mining, and Data Science just to cite a few. As a brief history, developed by Norman H. Nie, Dale H.Bent and Hadlai hull in 1968 by SPSS Inc, SPSS was acquired by IBM Inc in 2009 and formally changed the software name to IBM SPSS. In this project, the focus will only be on two components of IBM SPSS that deals with DT algorithms known as SPSS Decision Tree and SPSS Modeler. Briefly, according to IBM Knowledge Center, “ SPSS DT enables to identify groups, discover relationship between them, and predict future events.” Furthermore, it features visual classification and decision trees to help present categorical results and more clearly explain analysis to non-technical audiences. It helps to build classification models for segmentation, prediction, stratification, data reduction and variables. SPSS DT model or component is included in SPSS professional and premium package. Meanwhile, IBM SPSS Modeler “Is a set of data mining tools that enable t quickly build predictive models using business expertise and deploy them into business operations to improve decision making.” Designed around the CRISP-Model, IBM SPSS Modeler supports the entire data mining process. The tool offers a variety of modeling methods taken form machine learning, artificial intelligence, and statistics. Each method has a particular strength that is best suited for a specific type of problems. Some of these particularities will be mentioned in the functional evaluation criteria below. Finally, with SPSS Modeler, you can build accurate predictive models very easily, fastly, intuitively, and without need of a programming skill. In addition, IBM SPSS Decision Trees help to build decision trees and classification in order to identify groups, discover relationship between groups, and make accurate predictions. SPSS Decision Trees allows classification and decision trees for : Segmentation, Prediction, Stratification, Interaction identification, Data reduction and variable screening, Category merging, and Discretizing continuous variables.

## Functionality Criteria:

SPSS Decision Trees includes four varieties of decision tree algorithms: **CHAID** (it is a fast, statistical, multi-way tree algorithm that explores data efficiently and fastly, and then, builds segments and profiles based on the desired outcome. CHAID was created by KASS in 1980). Exhaustive CHAID ( which is a modification of CHAID that examines all possible splits for each predictor. It created in 1991 by Biggs, de Ville, and Suen). Classification and Regression Trees abbreviated as C&RT (is a complete binary tree algorithm that partition data and produces homogenous and accurate subsets. It was created in 1984 by Breiman, Friedman, Olshen, and Stone). QUEST (this algorithm selects variables without bias and builds accurate binary trees efficiently and effective. It was created by Loh and Shih in 1997). In addition, SPSS has some model refinement techniques such Ensemble (bagging and boosting). According to IBM knowledge Center, “The settings determine the behavior of ensemble that occurs when boosting, bagging “. So, when bagging and boosting, you should specify the number of base models to build when the objective is to enhance model accuracy and stability; Thus, for bagging, this will be the number of bootstrap samples . I has to be a positive integer. Also, in SPSS DT “Boosting always uses a weighted majority vote to score categorical targets and a weighted median to score continuous targets”. The screen captured below show an example of bagging using SPSS DT



Furthermore, SPSS DT Model Assessment includes the following: Generic Risk and classification tables, and summary node performance that shows Gains, Index (Lift), Response, Mean, Average Profit, and ROI .Reporting tools include : Evaluations charts, Analysis Node, Cross-validation charts, and others. Finally, concerning the Slipping Method: the algorithms will slightly differ in the criteria used for the splits. IBM Knowledge Center made it clear that when C&R Tree predicts a categorical output, Gini coefficient, which is the default, is used. For continuous targets, Least Square Deviation method is used. Also, CHAID uses a chi-square test, QUEST uses a chi-square test for categorical targets, and Variance for continuous inputs.

## Ancillary Tasks Support:

Concerning data cleansing, missing values can be handled in SPSS DT in the following ways: By assigning a missing value to a category or by impute using surrogate. Also, missing is handled slightly differently depending if you are using C& R tree , QUEST, or CHAID. For instance, CHAID makes the missing values a separate category and allows them to be used in tree building. Whereas, QUEST uses a fractioning method. Also, SPSS has a metadata that describes how data is collected , stored and organized. Metadata include information such as variable names, multiple response variable, and the structure of data just to cite a few.

## Performance Criteria:

SPSS Decision Trees architecture: there is a readily available solution for installation as Client-Only software. However, for scalability and greater performance a server-based version is available. It is possible to increase the software performance by scaling both horizontally and vertically. Also, SPSS DT is supported by the Operating Systems including: Windows, Linux, Unix, and Mac OS X. Finally, SPSS Decision trees/modeler offers a lot of flexibility to integrate it with other vendors’ software in order to increase to increase performance.

## Usability Criteria:

IBM SPSS Decision Tree and Modeler are both interactive GUI. Concerning the type of user, the software is developed for intermediate to advanced users. Without proper training on how to use the solution, it will be very difficult for a novice user, to build a decision tree algorithm intuitively. In addition, IBM SPSS decision is accessible via any Web browser when using the client-server solution. Also, there is single user license that could be installed and run from a PC and laptop (Stand-alone).

**Comparative Analysis in terms of the relevant criteria**

**Summary Table**

|  |  |  |
| --- | --- | --- |
|  | **IBM SPSS DT/ Modeler** | **SAS Enterprise Miner** |
| **Functionality Criteria:** | Splitting Method:  *Gini, Least square deviation, CHAID, QUEST, Variance*  *Ensemble (Bagging, boosting, gradient boosting)*  *Etc…* | Splitting Method:  *Gini, Entropy, Probchisq, Variance*  *Ensemble(Bagging, boosting...)*  *Etc..* |
| **Performance Criteria:** | Software Architecture:  *Procedures*  Platform:  *Stand-Alone or client-only, Server based for big companies with large projects*  Possible Integrations with:  *Can integrate with all IBM products. IT enables users to import models from a variety of sources with the only condition that external model be expressed in the form of* ***DS 1 Code***  Scalability**:**  *Can increase performance by using a server based software*  Compatible OS:  *Windows, Linux, Unix, and Mac OS, IBM Mainframe* | Software Architecture:  *Compilers and procedures*  Platform:  *Server based software or*  *distributed client/server system*  Possible Integrations with:  *IBM Netezza, Oracle, SAP HANA, Aster data, DB2, Greeplum, Hadoop, R language*  *It uses in-database , in-memory and grid capabilities for faster response*  Scalability:  *Can increase Scalability by integration SAS EM with other vendors’ products such as Netezza, Oracle…*  Compatible OS:  *Windows, Linux, Unix, and Mac OS X, IBM Mainframe, OpenVMS Alpha* |
| **Ancillary Tasks Support:** | Data Cleansing:  **Missing Value:** *can be handled in SPSS DT in the following ways: By assigning a missing value to a category or by impute using surrogate*  ***Metadata*** *is available and help to edit variables*  **Filter** is available | Data cleansing:  **Missing value**: *Use Impute Node to deal with.*  ***Metadata*** *is available and help to edit variables*  **Filter** is available |
| **Usability Criteria:** | User Type:  *Intermediate to Advanced users*  Type of Software:  *Graphic User Interface (GUI)*  Documentation :  *Both IBM Knowledge Center and Help page are not easy to follow along.*  *Other comments:*  *IBM SPSS decision is accessible via any Web browser when using the cloud based solution. Also, there is single user license that could be installed and run from a PC and laptop (Stand-alone).*  *During update there is not service interruption* | User Type:  *Beginner, Intermediate, advanced users*  Type of Software:  *Graphic User Interface (GUI) and Console*  Documentation :  *Help page is very rich and easy to follow along.*  *Other comments: the software must be download and run for each work session*  *SAS EM cannot be accessed during schedule update and maintenance that happens every few days.* |

## Comparison

When it comes to Usability’s criteria, we notice that IBM SPSS offers a lot of options in term of how to access and run the software. SPSS DT/Modeler has a cloud version that could be accessed via different supports (Mobile, computers) connected to the internet using any common web Browser. Whereas, SAS EM software must be downloaded, installed, and launched for each work session. In addition, a user has a choice to schedule and update SPSS any time there are new updates required. Whereas, SAS EM is scheduled and done by the vendor (SAS Institute). When it comes to performance criteria, we notice a huge difference between both software. SAS EM is a compiler and procedural software, meaning it can be used as a regular software or a programming language ( you can write code and accomplish tasks). Whereas, SPSS DT/ Modeler for data mining cannot be used as a programming language. It is possible to easily integrate SAS EM with other vendor software such as Oracle, SAP HANA, R language, and Hadoop just to cite a few. Concerning Functionality and ancillary support criteria, please refer back to my remark on their respective sections above.

What have you learned? Throughout this research project, I have discovered several algorithms offered by IBM SPSS for both decision trees algorithms and different splitting methods. Also, I discovered SAS EM compilers and its integration with R language. I will definitely be investigating both tools further in my spare time.

**Conclusion:**

After a meticulous review of both data mining software above, I believe that their approach to dealing with Decision Tree induction data mining technique are different. They have a slightly different naming convention for algorithms and spitting methods. Also, I noticed that SAS EM is designed for all type of users(beginner, intermediate, and advanced user), and is intuitive and easy to follow along using the Help Menu. Whereas, IBM SPSS DT\Modeler is designed around the CRISP-Model, for users with some intermediate to advanced exposures in statistical concepts. In term of scalability, both data mining software offer several venues and tune up options to tackle large data mining project. In addition, when it comes to integrating both DM solutions with other vendors’ software, they offers a lot of flexibility that make it possible to import models from other software. For instance, IBM SPSS DT/Modeler enables users to import models from a variety of sources with the only condition that external model be expressed in the form of DS 1 Code. Meanwhile, SAS EM can be integrated with various other tools such as Oracle, IBM Netezza, Hadoop, and SAP HANA just to cite a few. Because both DM software are dealing efficiently with small, medium, and large data mining project, the choice of one versus another would be very subjective. However, for a novice in data mining, I suggest using SAS Enterprise Miner to build various models. SAS EM is well-designed and has a very rich Help Content easy to follow along.

# Reference

[1]Collier, K., Carey, B., Sautter, D., & Marjaniemi, C. (n.d.). A methodology for evaluating and selecting data mining software. *Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences. 1999. HICSS-32. Abstracts and CD-ROM of Full Papers*. doi: 10.1109/hicss.1999.772607

[2]O’Reilly Online Learning. (2020). *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management, Third Edition*. [online] Available at: https://learning.oreilly.com/library/view/data-mining-techniques/9780470650936/9780470650936c07.xhtml [Accessed 8 Mar. 2020].

[3]Datasciencecentral.com. (2020). *The Complete Guide to Decision Trees*. [online] Available at: https://www.datasciencecentral.com/profiles/blogs/the-complete-guide-to-decision-trees [Accessed 8 Mar. 2020].

[4]Ibm.com. (2020). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/SSFMBX/com.ibm.swg.im.dashdb.analytics.doc/doc/r\_decision\_trees\_pruning.html [Accessed 8 Mar. 2020].

[5]Ibm.com. (2020). *IBM Knowledge Center*. [online] Available at: https://www.ibm.com/support/knowledgecenter/en/SS3RA7\_sub/modeler\_tutorial\_ddita/clementine/clementine\_at\_a\_glance.html [Accessed 8 Mar. 2020].

[6] Institute, SAS. “Building Decision Tree SAS Enterprise Miner.” *Documentation.sas.com*, documentation.sas.com/?docsetId=emgsj&docsetVersion=15.1&docsetTarget=p1hkahr5i5ua1tn1ho0c7uk1h1x3.htm&locale=en.

[7]Collier, K., Carey, B., Sautter, D., & Marjaniemi, C. (n.d.). A methodology for evaluating and selecting data mining software. *Proceedings of the 32nd Annual Hawaii International Conference on Systems Sciences. 1999. HICSS-32. Abstracts and CD-ROM of Full Papers*. doi: 10.1109/hicss.1999.772607