

# Introduction to Machine Learning

**Melodie Rush**

**Customer Success Principal Data Scientist**

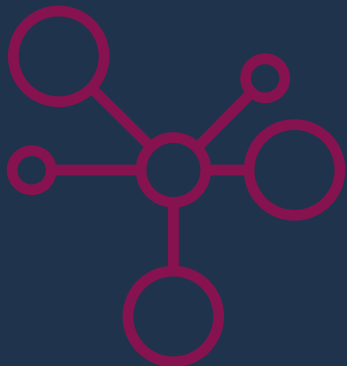
Connect with me:

LinkedIn: <https://www.linkedin.com/in/melodierush>

Twitter: @Melodie\_Rush



# How does SAS support Machine Learning?



## Agenda

- What is Machine Learning?
- Terminology and key characteristics
- Introduction to Decision Trees, Random Forest, Gradient Boosting, Neural Networks, and k-means Clustering
- How you can use machine learning in SAS
- Examples in SAS 9.x and SAS Viya

# Machine Learning



## Definition

- Using iterative processes, machine learning builds models that **automatically adapt** with little or no human intervention.



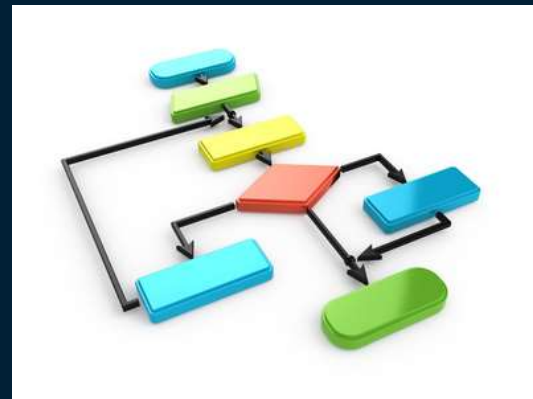
# Why is it so important now?



**Data**



**Computing  
Power**



**Algorithms**

# Terminology

# Terminology

Machine learning terms versus inferential statistics terms

*What are all these archaic, outmoded and confusing terms?*

*What are all these new fangled and confusing terms?*

- Feature
- Input
- Target
- Object



- Variable
- Independent Variable
- Dependent Variable
- Observation

# Terminology

What are Machine Learning terminology?

- In statistics we predict a  $Y$  or a dependent variable.
- In data mining,  $Y$  is called a target.
- In machine learning, a target is called a label.
- In statistics and data mining our inputs are called  $X$ 's.
- In machine learning our inputs are called features.
- In statistics and data mining we transform our  $X$ 's.
- In machine learning we do feature creation.





# Does Machine Learning Work?

Distinguish apple from orange

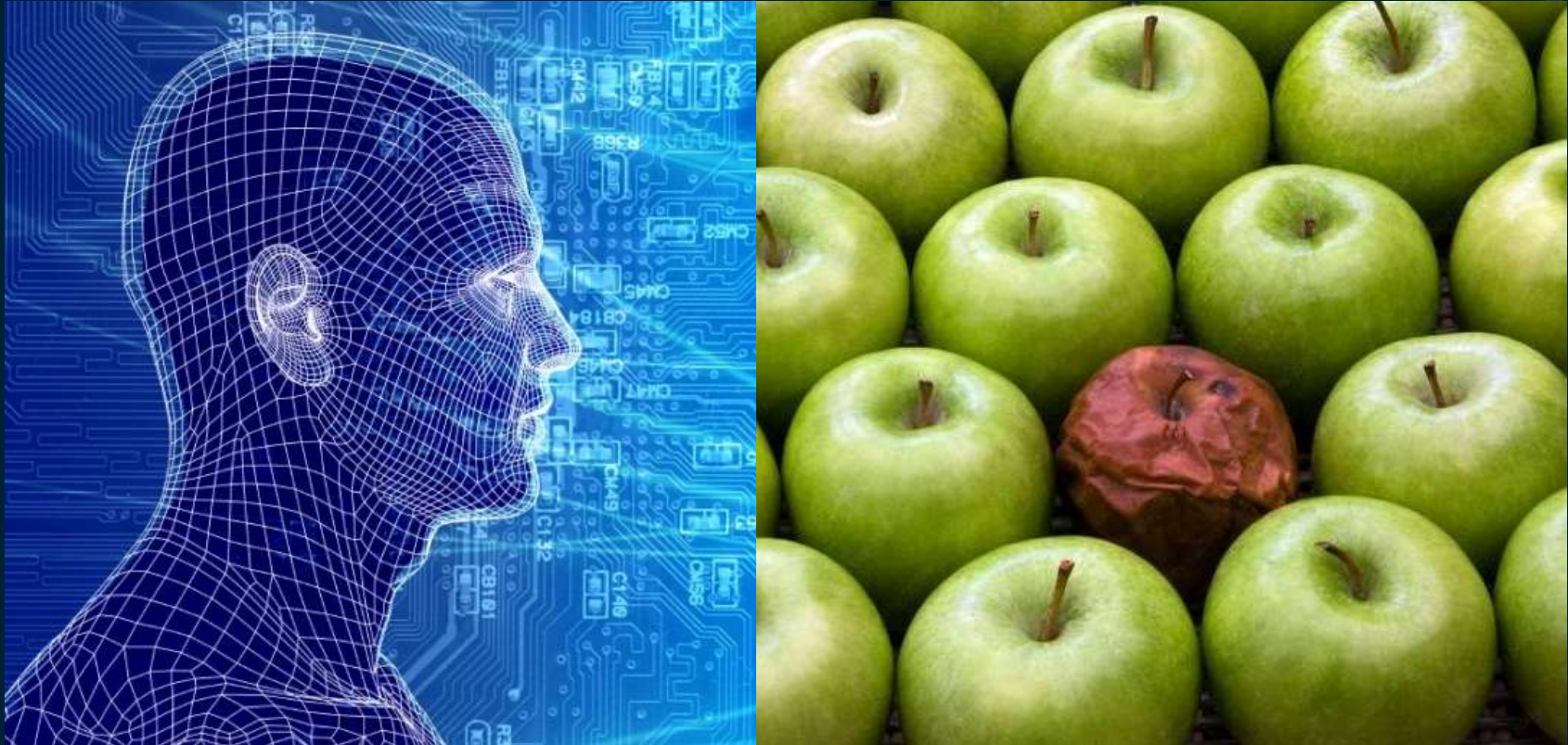






# How Does Machine Learning Work?

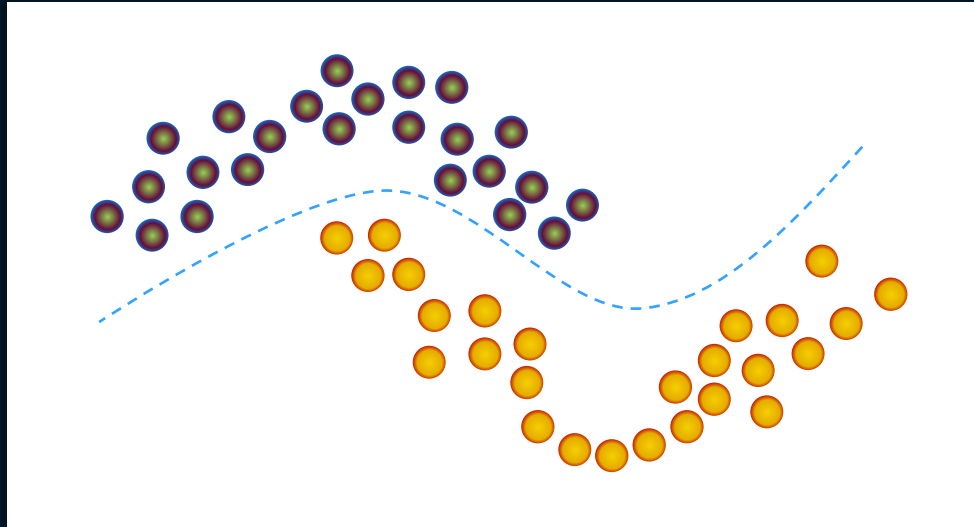
Finding the rotten apple



# How Does Machine Learning Work?

## Supervised Learning

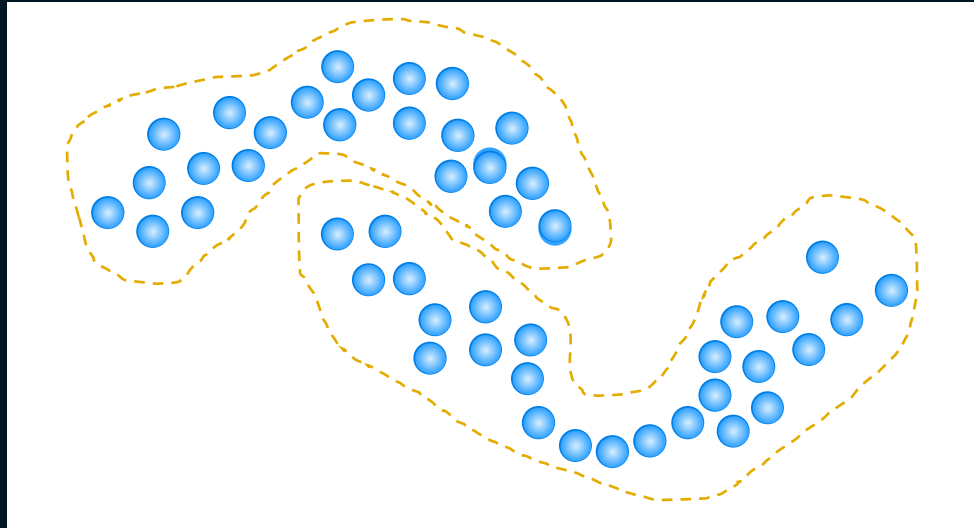
Trained on labeled examples



# How Does Machine Learning Work?

## Unsupervised Learning

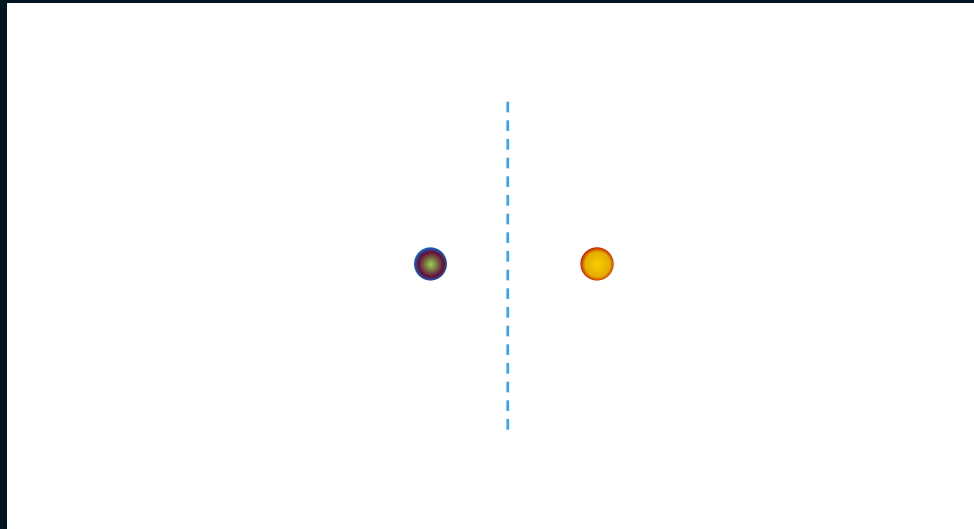
Trained on unlabeled examples



# How Does Machine Learning Work?

## Semi-Supervised Learning

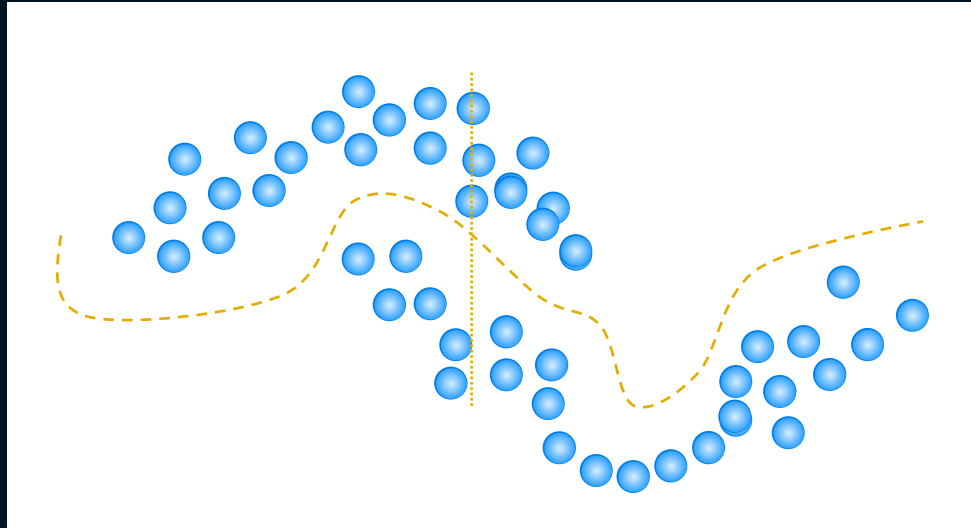
Use labeled and unlabeled observations



# How Does Machine Learning Work?

## Semi-Supervised Learning

Use labeled and unlabeled observations



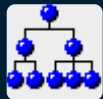
# How Does Machine Learning Work?

Not New for SAS

Machine Learning has been available in both SAS/STAT and SAS Enterprise Miner for decades



Neural Networks



Decision Trees



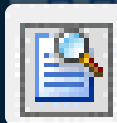
Random Forests



Clustering



Gradient Boosting



Text Analytics



Regression

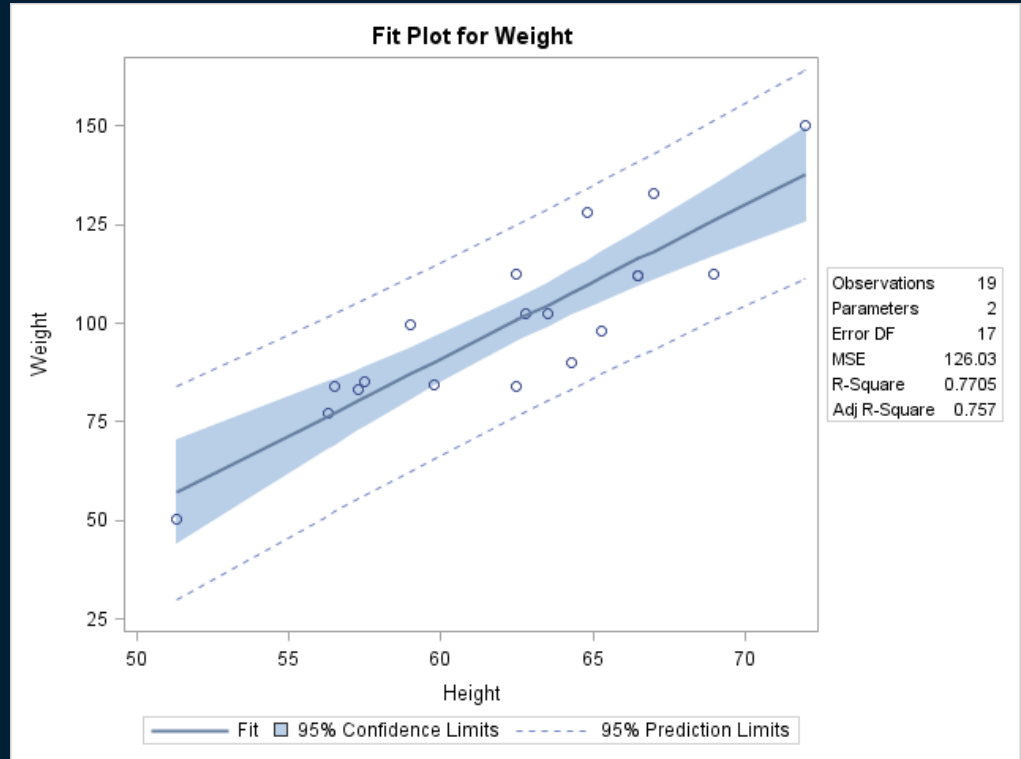


# Machine Learning Algorithms

# Regression

## What Is It?

- Used to identify the relationship between a dependent variable and one or more independent variables
- Many types – linear, logistic, quantile, polynomial, stepwise, ridge, lasso, ElasticNet, etc...
- Oldie but goodie

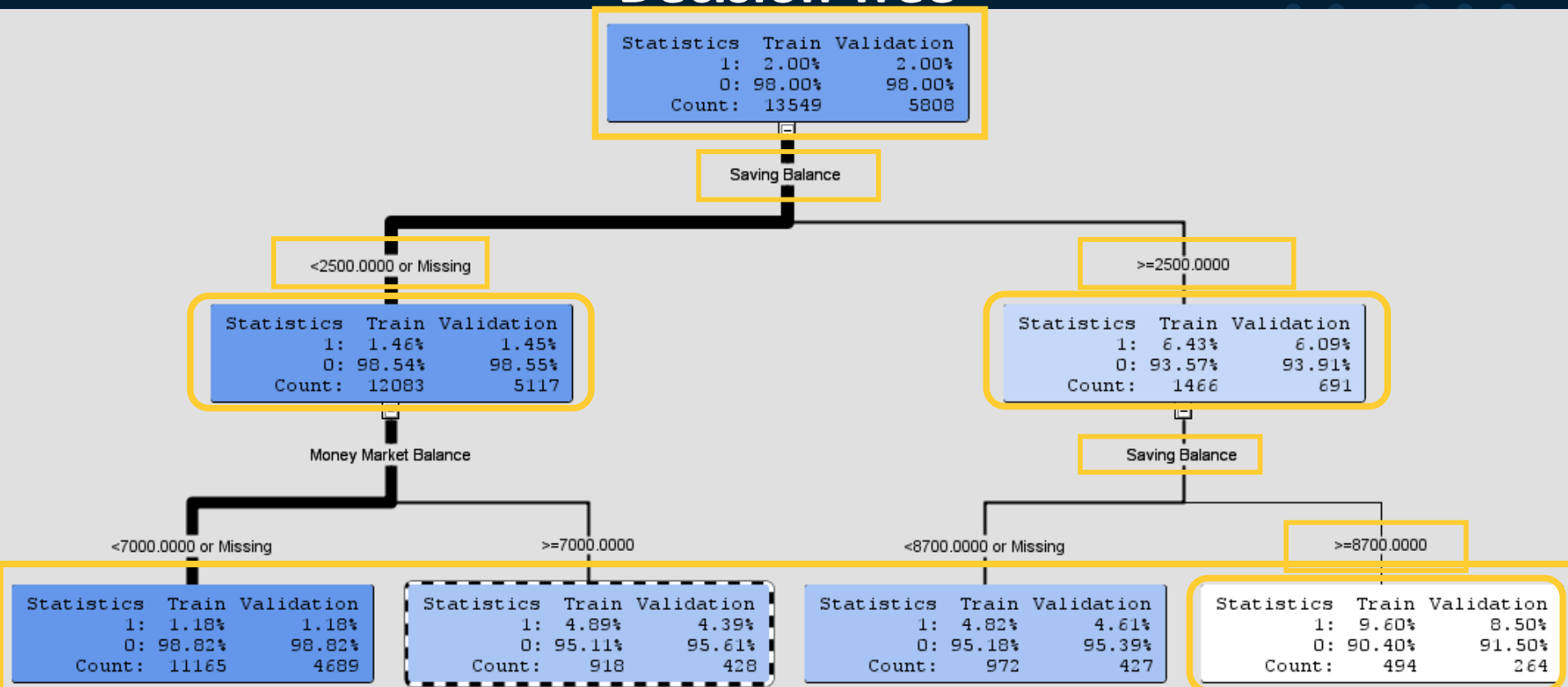


# Decision Trees

## What Is It?

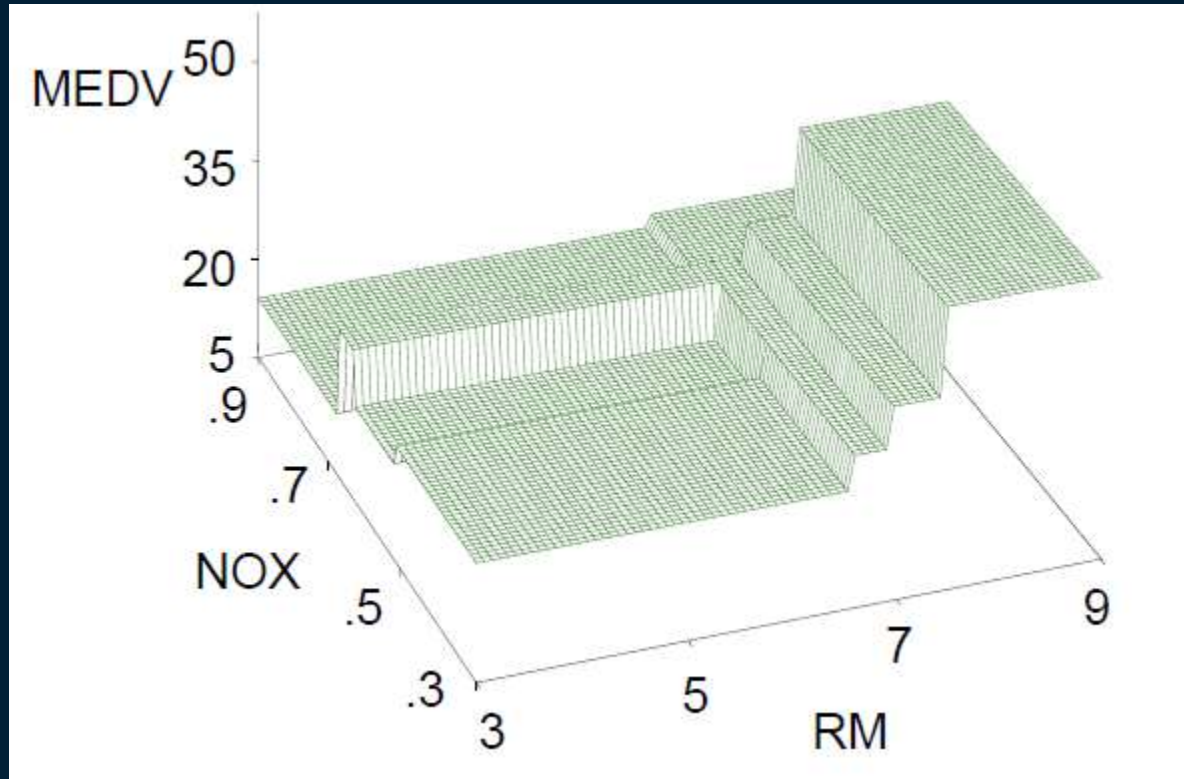
- Linear separation of data using “if then else” logic
- Separation is performed via an exhaustive search of splitting points for each variable.
- Many different architectural variations based on the above architecture
- Users might refer to them as
  - CHAID Trees
  - CART Trees
  - C4.5 Trees
  - C5.0 Trees.
  - Each of the above is simply a variation on the tree architecture.

# Decision Tree



# Decision Trees

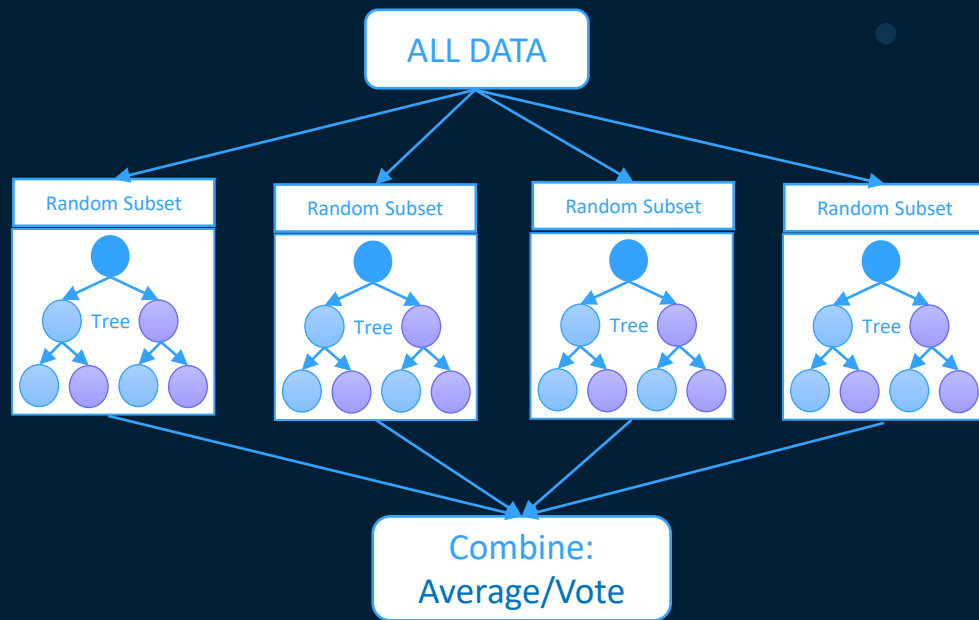
## Multivariate Step Function



# Random Forest

## What Is It?

- A combination of several “decision trees.”
- A random forest consists of a forest of fully trained decision trees.
- The random forest averages the output of all the decision trees in the “forest.”



# Random Forest

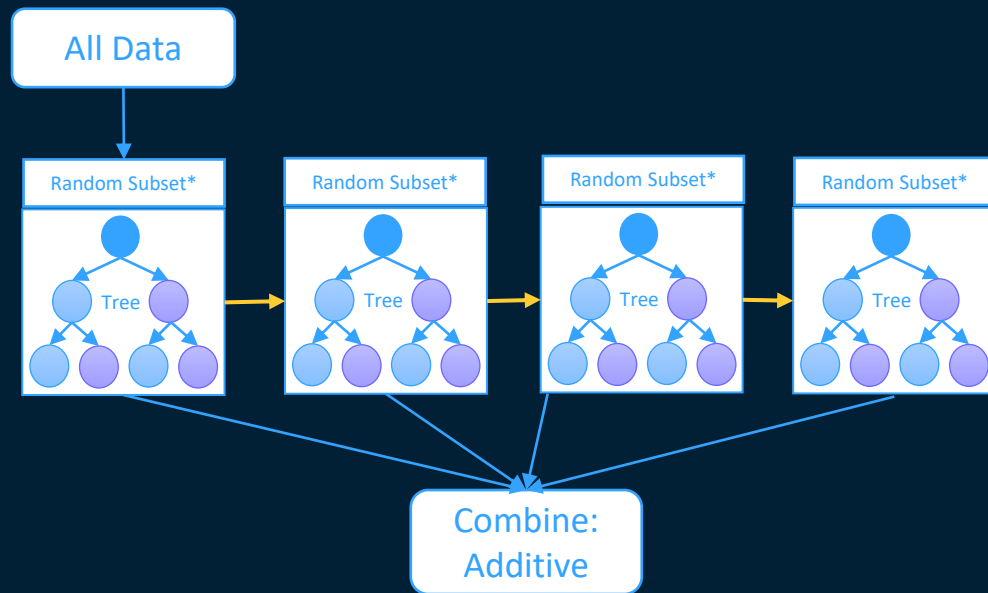
## Algorithm

- Select a number of trees in the random forest.
- For each tree in the forest, use the following split algorithm:
  - Select a random sample of data.
  - Select a random subset of variables.
  - Determine the best split from the sample of data and the sample of variables.
  - Keep selecting random data and random subsets of variables until the maximum number of trees is trained.
- When all the trees are built, the prediction is the average of all trees.

# Gradient Boosting

## What Is It?

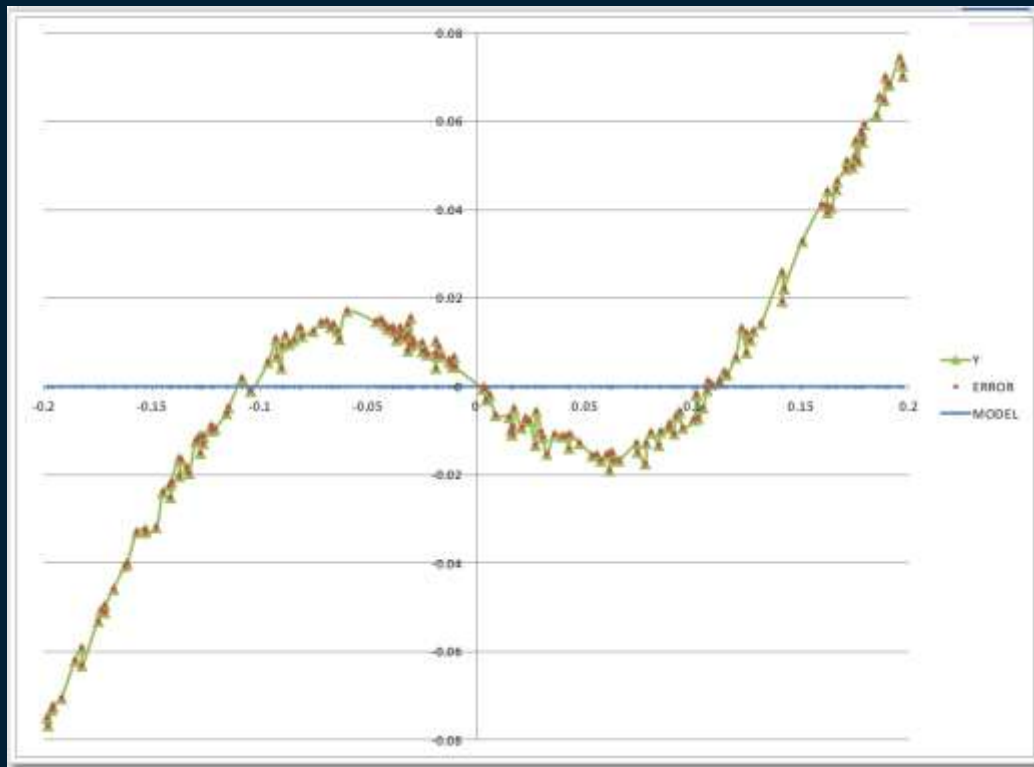
- A combination of several “decision trees.”
- Gradient boosting consists of a **forest** of **small** decision trees (“**shrubs**”, “stumps”).
- Each **shrub** is poor at predicting target, but each subsequent shrub tries to fit the remaining error.
- Eventually converges to good solution.





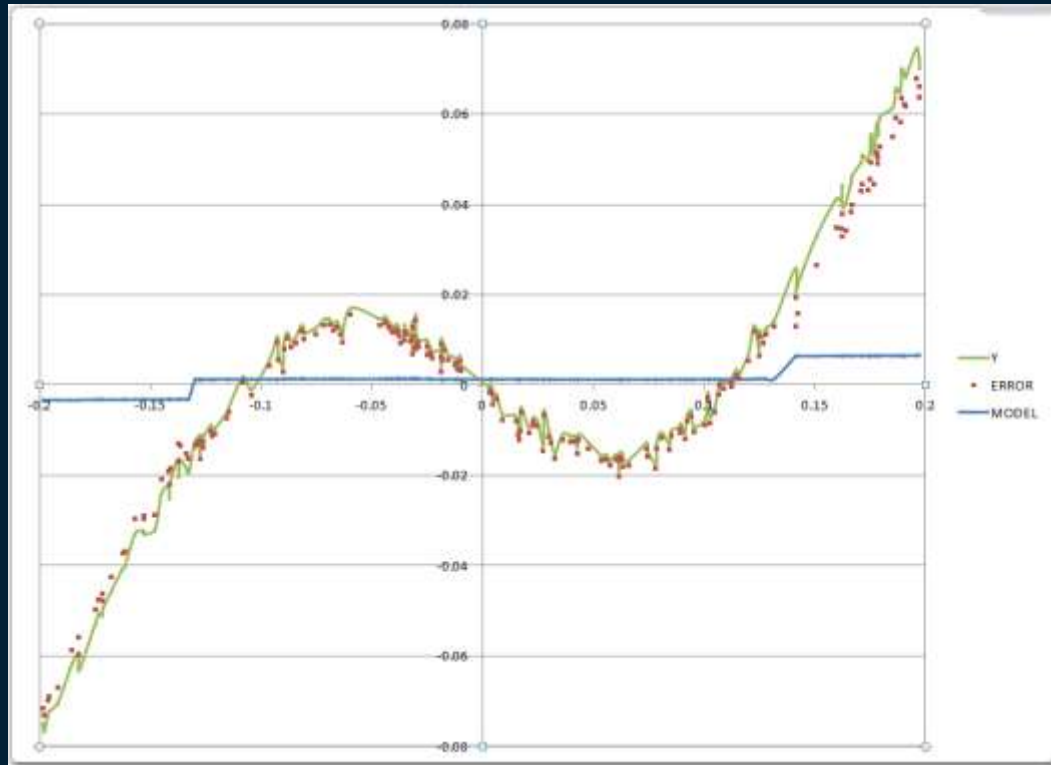
# Gradient Boosting

## Example: Iterations=0



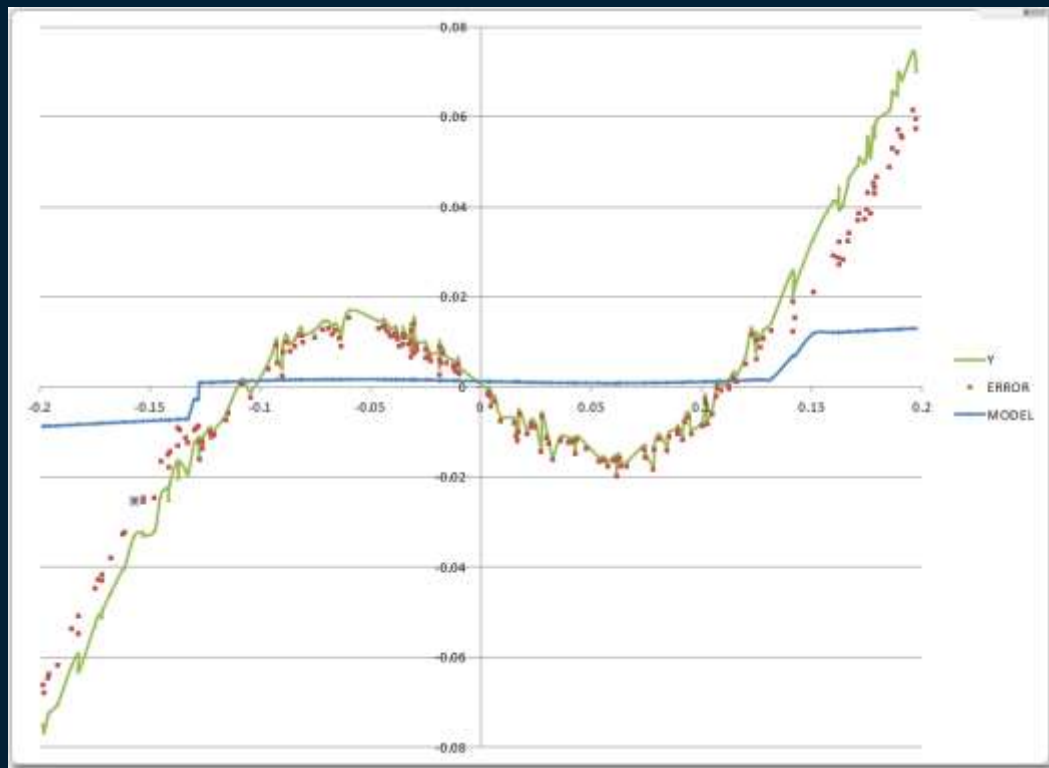
# Gradient Boosting

## Example: Iterations=1



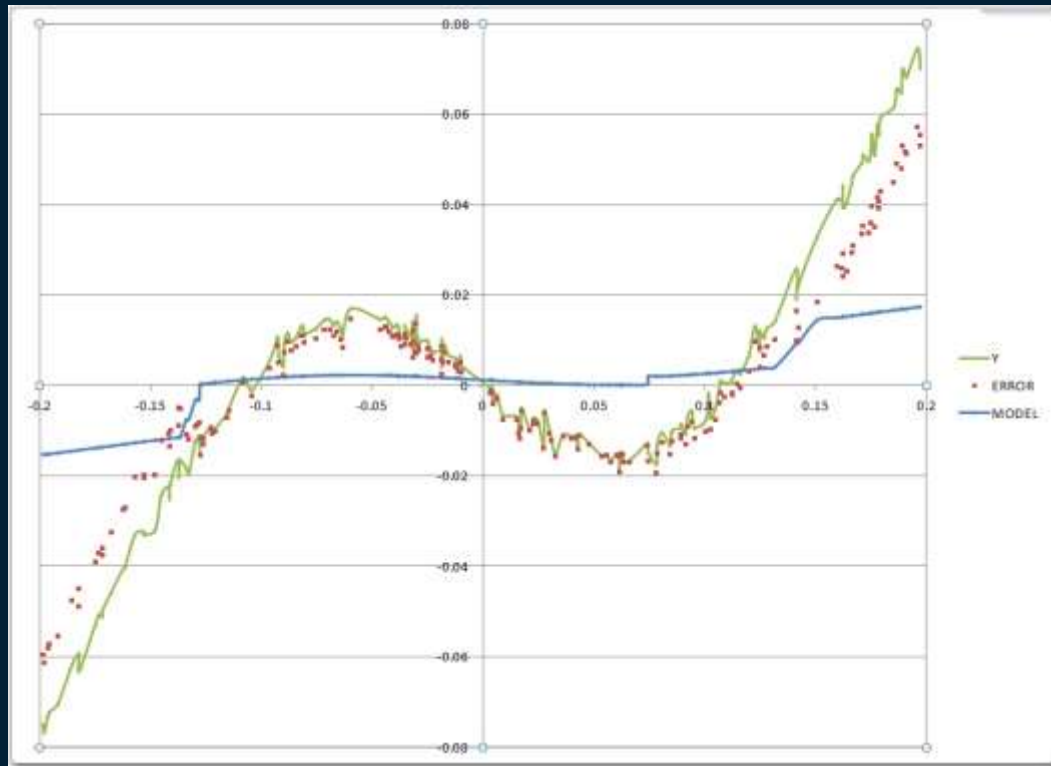
# Gradient Boosting

## Example: Iterations=10



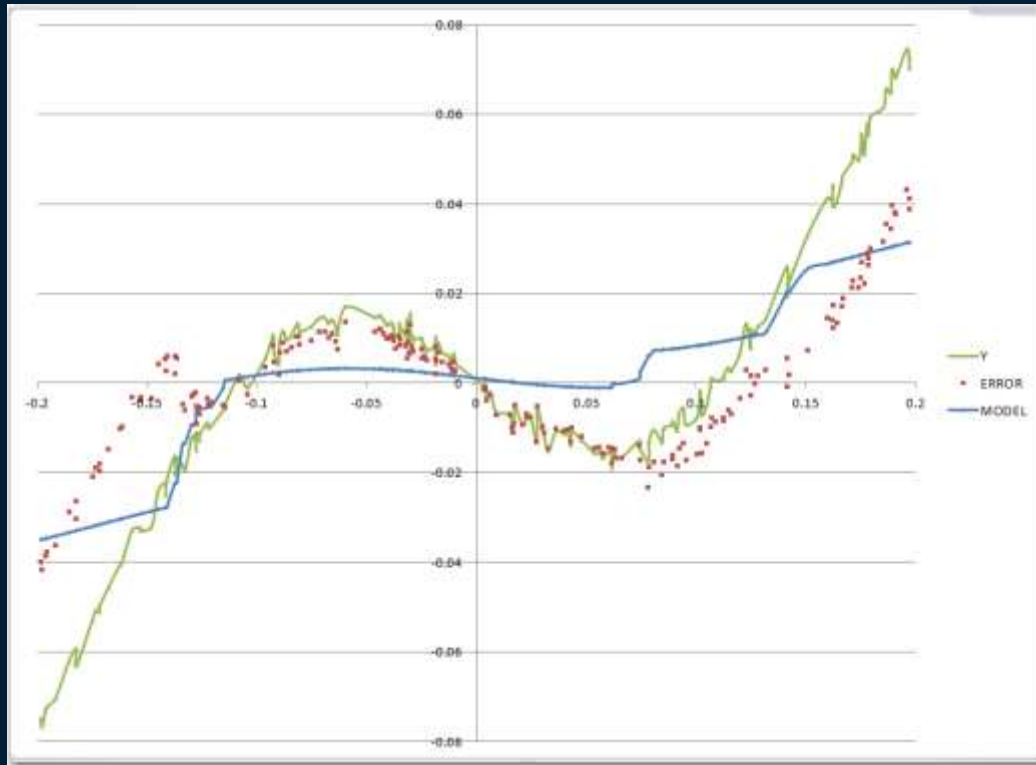
# Gradient Boosting

## Example: Iterations=25



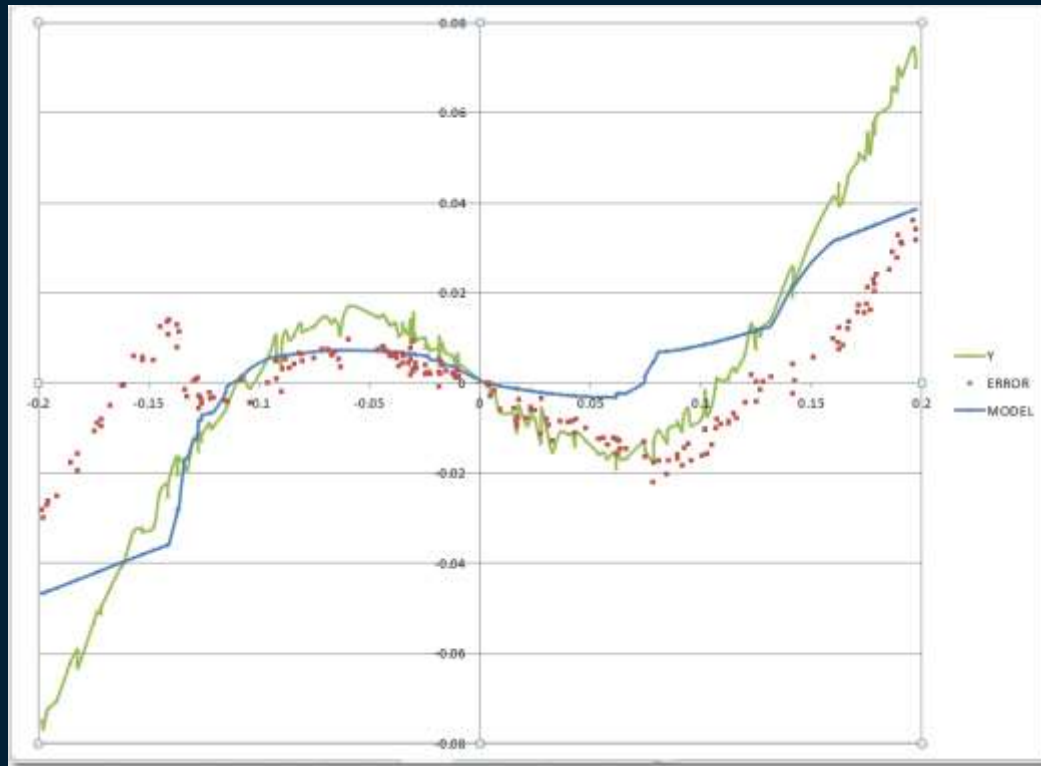
# Gradient Boosting

## Example: Iterations=50



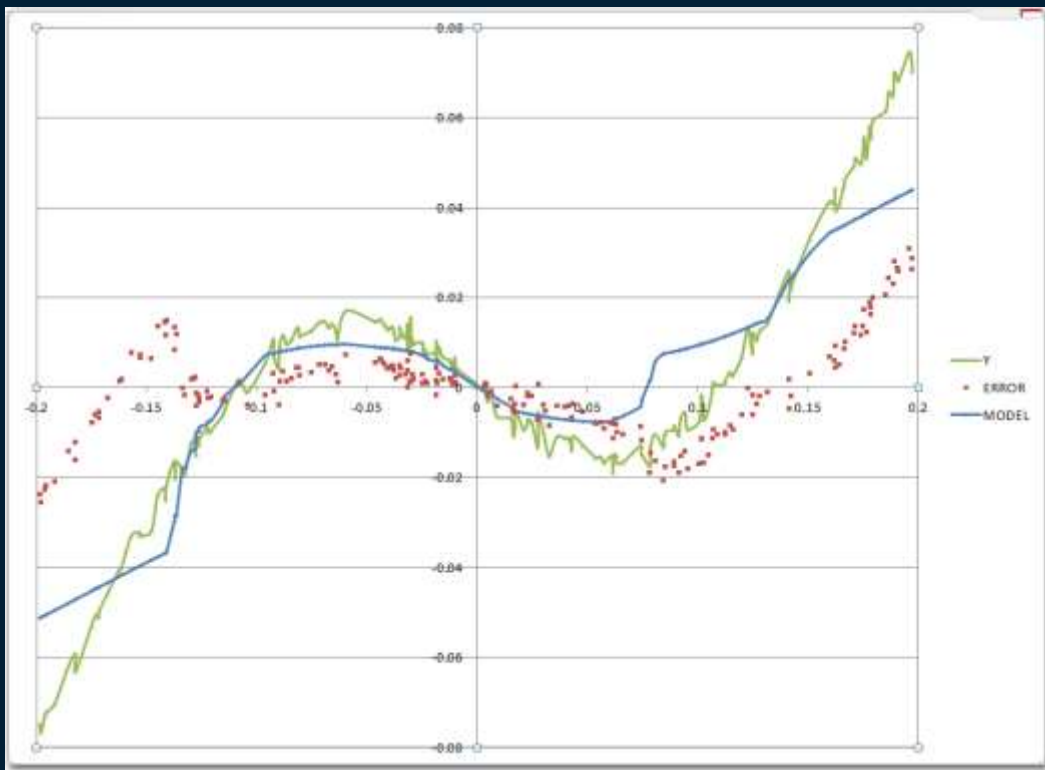
# Gradient Boosting

## Example: Iterations=75



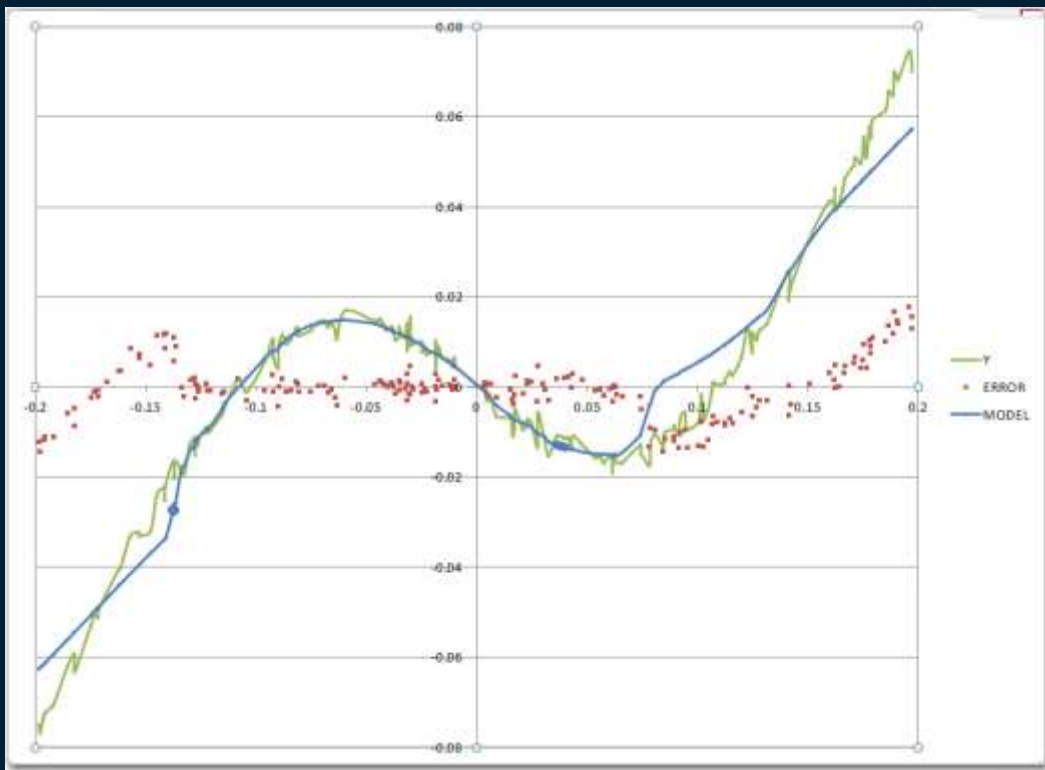
# Gradient Boosting

Example: Iterations=100



# Gradient Boosting

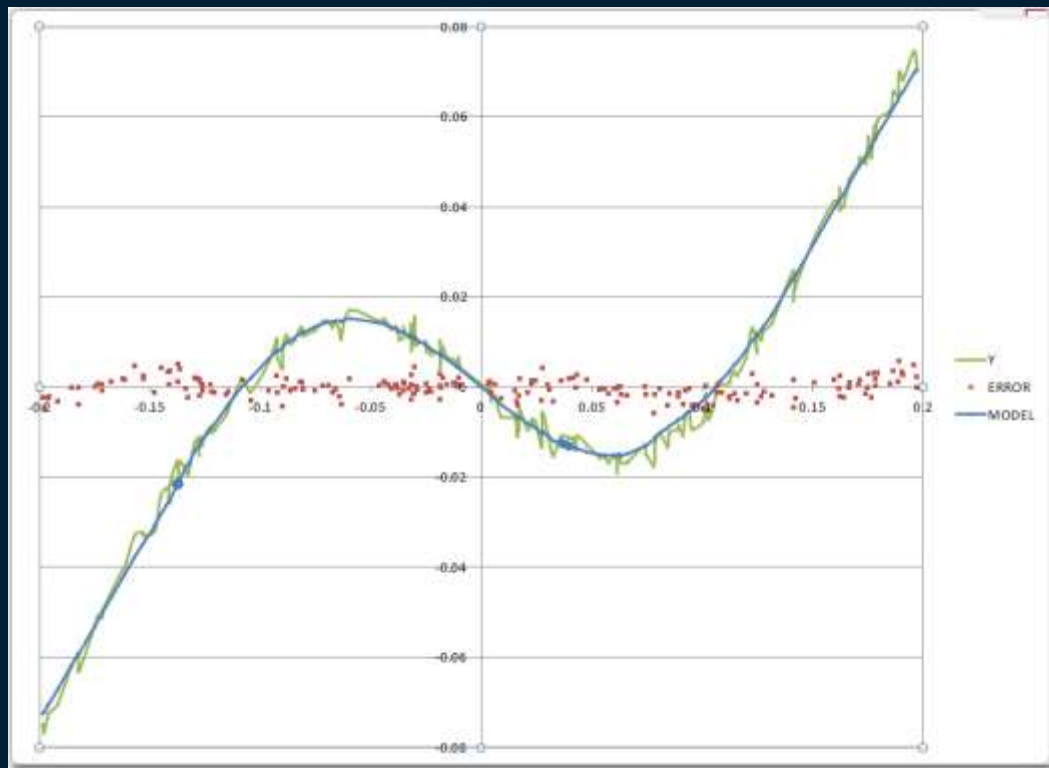
## Example: Iterations=200



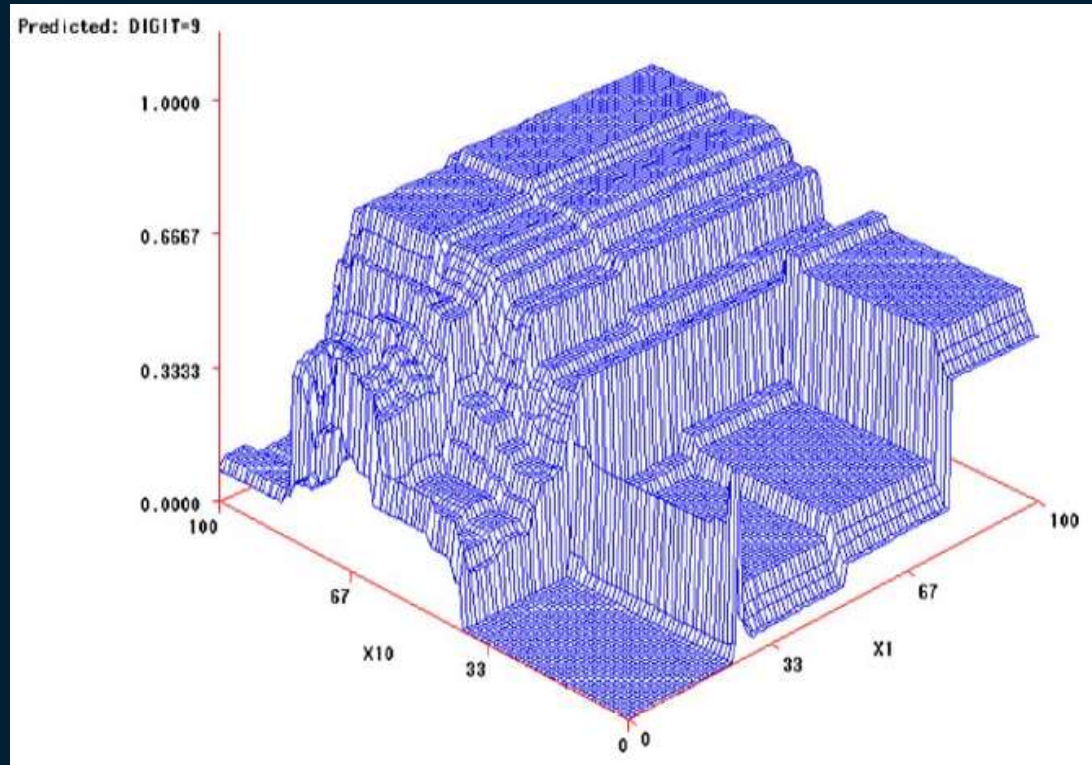


# Gradient Boosting

## Example: Iterations=300



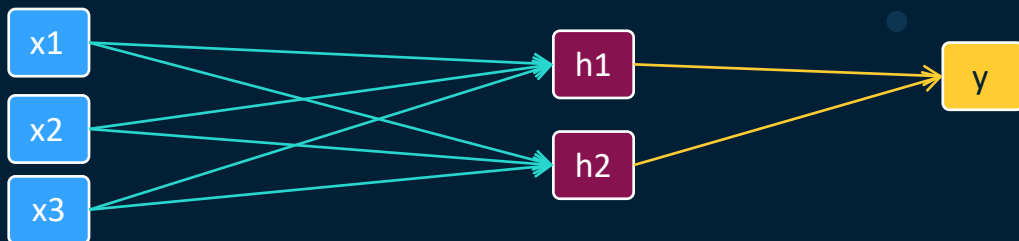
# Gradient Boosting



# Neural Network

## What Is It?

- Non-linear relationship between inputs and output
- Prediction more important than ease of explaining model
- Requires a lot of training data
- Users can specify the number of hidden layers, the number of hidden neurons, and associated activation functions for each layer
- Users can configure Input and Target Standardizations, Target Error, and Activation Functions



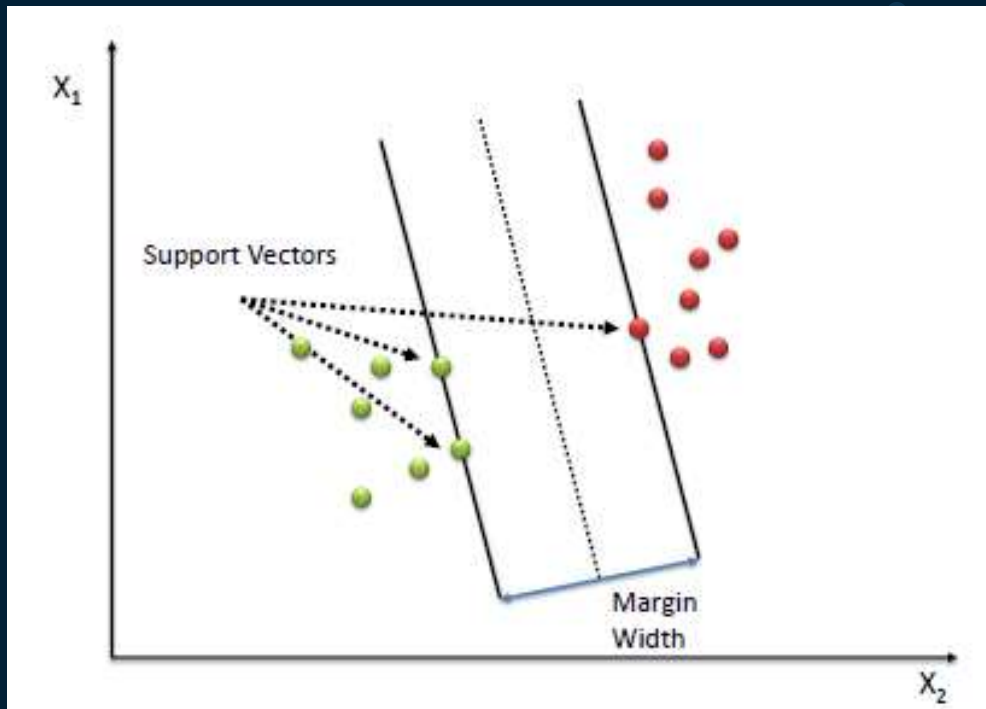
## Many types...

- Feedforward Neural Network
- Radial Basis Function Neural Network
- Multilayer Perceptron
- Convolutional Neural Network (CNN)
- Recurrent Neural Network (RNN)
- Modular Neural Network.
- Sequence-To-Sequence Models

# Support Vector Machines

## What Is It?

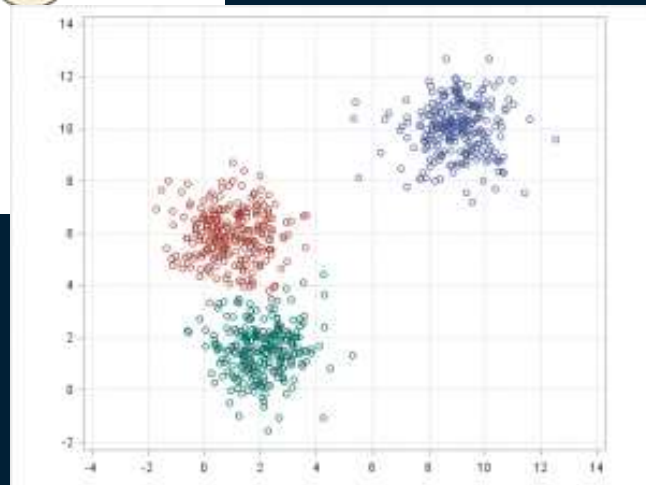
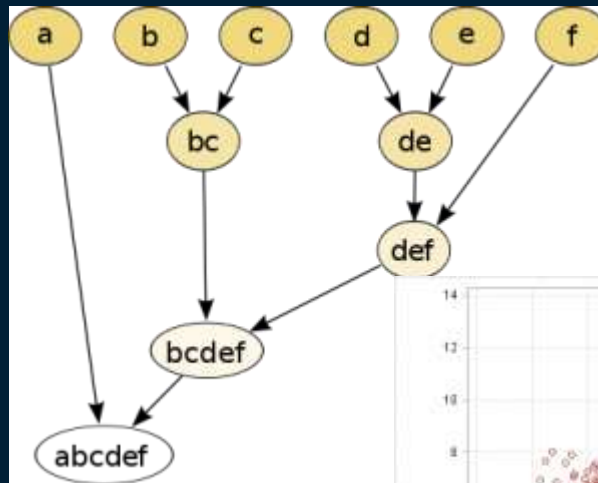
- Enables the creation of linear and nonlinear support vector machine models
- Constructs separating hyperplanes that maximize the margin between two classes
- The vectors (cases) that define the hyperplane are the support vectors
- Enables use of a variety of kernels: linear, polynomial, radial basis function, and sigmoid function. The node also provides interior point and active set optimization methods.



# Clustering

## What Is It?

- Goal: The goal of clustering is to partition data into groups so that the observations within a group are as similar as possible to each other, and as dissimilar as possible to the observations in other groups.
- Many types - Hierarchical, k-means, SOM, etc..



# Ensemble Modeling

## What Is It?

- **Two or more** predictive models **combined** to create a potentially more accurate model
- Works better when model predictions are uncorrelated
- Creates new models by combining the posterior probabilities (for class targets) or the predicted values (for interval targets) from multiple predecessor models.
- 3 Methods
  - Average
  - Maximum
  - Voting



# SAS 9.x

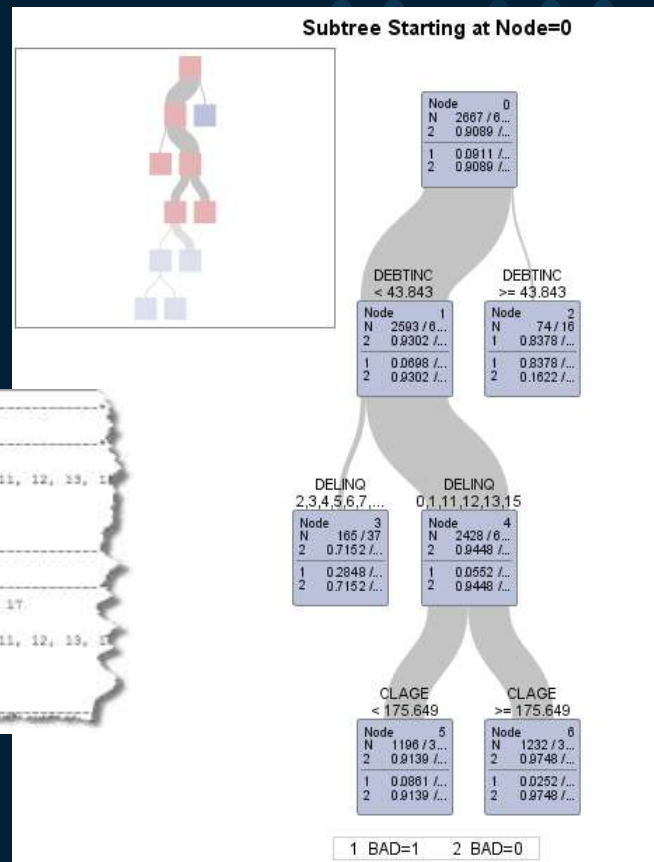
SAS/STAT and SAS Enterprise Miner

# SAS Decision Trees

## HPSplit Procedure

```
proc hpsplit data=sashelp.hmeq maxdepth=7 maxbranch=2;  
  target BAD;  
  input DELINQ DEROG JOB NINQ REASON / level=nom;  
  input CLAGE CLNO DEBTINC LOAN MORTDUE VALUE YOJ  
    / level=int;  
  criterion entropy;  
  prune misc / N <= 6;  
  partition fraction(validate=0.2);  
  rules file='hpsplhme2-rules.txt';  
  score out=scored2;  
run;
```

```
Node = 2  
-----  
DELINQ IS ONE OF 5, 6, 7, 8, 10, 11, 12, 13, 15  
AND DELINQ IS ONE OF 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 15  
PREDICTED VALUE IS 1  
PREDICTED 1 = 0.9342 ( 71/76)  
PREDICTED 0 = 0.0658 ( 5/76)  
-----  
Node = 4  
-----  
NINQ IS ONE OF 4, 5, 6, 8, 9, 10, 11, 12, 13, 14, 17  
AND DELINQ IS ONE OF MISSING, 1, 2, 3, 4  
AND DELINQ IS ONE OF 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 15  
PREDICTED VALUE IS 1  
PREDICTED 1 = 0.8714 ( 61/70)  
PREDICTED 0 = 0.1286 ( 9/70)
```



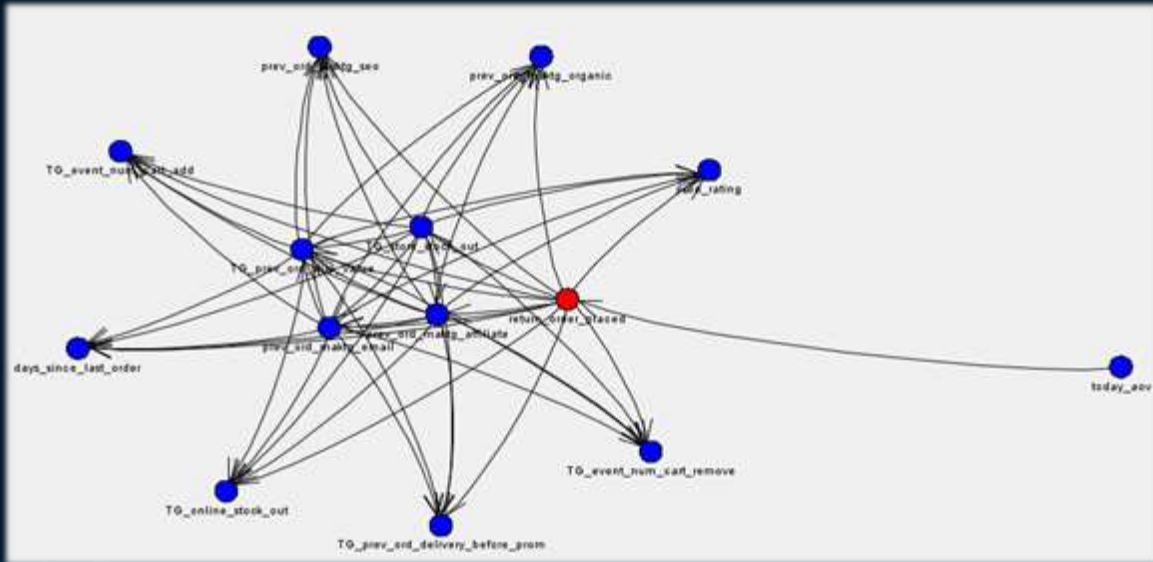
[HPSPLIT Procedure Documentation](#)



# SAS Enterprise Miner

- Algorithms – basic and advanced

- Linear & Logistic Regression
- Decision Trees
- Random Forest
- Gradient Boosting
- Support Vector Machines
- Neural Networks
- Clustering
- Bayesian Networks
- Principal Components
- Open Source Models



# Classification

Our example today

- The dataset is from a financial institution with customer demographics and loan/credit behavior.
- The goal of this modeling exercise is to **predict which people are likely to default on a home equity loan.**
- The data are at the customer-level (subject-level).
- n=5960
- columns = 13

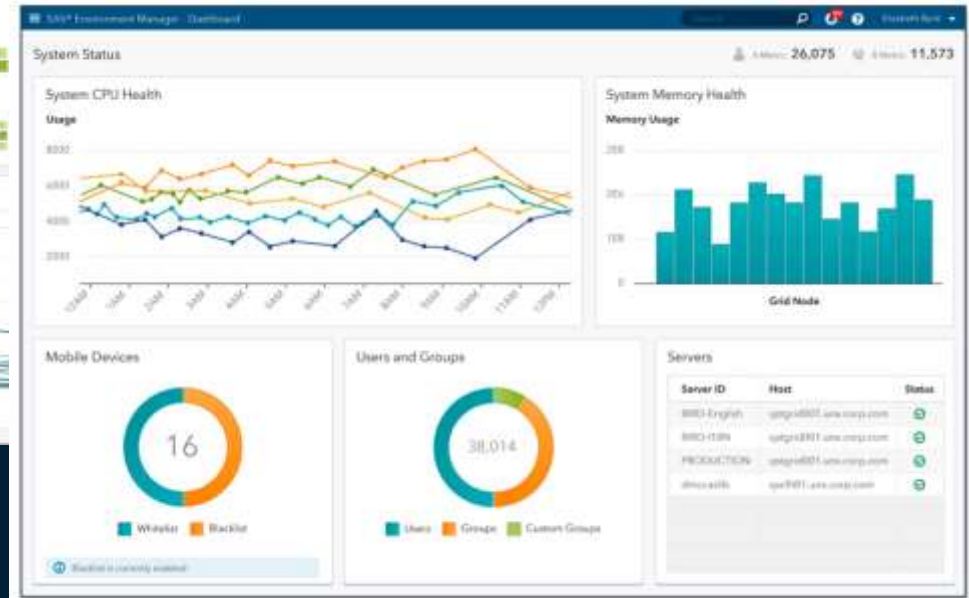
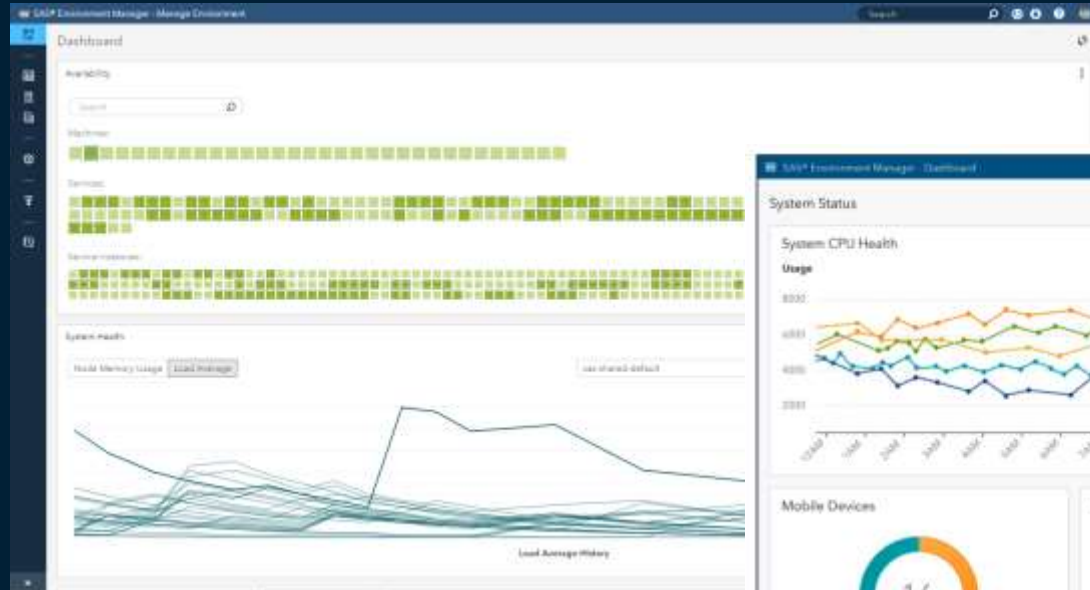
Alphabetic List of Variables and Attributes				
#	Variable	Type	Len	Label
1	BAD	Num	8	Default or seriously delinquent
10	CLAGE	Num	8	Age of oldest credit line in months
12	CLNO	Num	8	No. of trade credit lines
13	DEBTINC	Num	8	Debt to income ratio
9	DELINQ	Num	8	No. of delinquent credit lines
8	DEROG	Num	8	No. of major derogatory reports
6	JOB	Char	7	Prof/Exec/Office/Self/Other
2	LOAN	Num	8	Amount of current loan request
3	MORTDUE	Num	8	Amount due on existing mortgage
11	NINQ	Num	8	No. of recent credit inquiries
5	REASON	Char	7	Home improvement or Debt Consolidation
4	VALUE	Num	8	Value of current property
7	YOJ	Num	8	Years on current job

# SAS Viya

SAS Visual Statistics and  
SAS Visual Data Mining and Machine Learning

# What is SAS Viya?

Viya is a cloud-enabled, in-memory analytics engine that provides quick, accurate and reliable analytical insights.



# SAS Viya Products

SAS Viya takes advantage of a cloud-enabled, open platform. Most offerings include both a coding interface as well a visual interface.

- SAS Visual Analytics
- SAS Visual Statistics
- SAS Visual Data Mining and Machine Learning
- SAS Visual Forecasting
- SAS Visual Text Analytics
- SAS Optimization
- SAS Econometrics
- SAS Model Manager
- SAS Data Preparation
- SAS Visual Investigator
- SAS Business Analytics
- SAS Intelligent Decisioning
- SAS Cybersecurity
- SAS Detection and Investigation
- SAS Event Stream Processing
- And more...

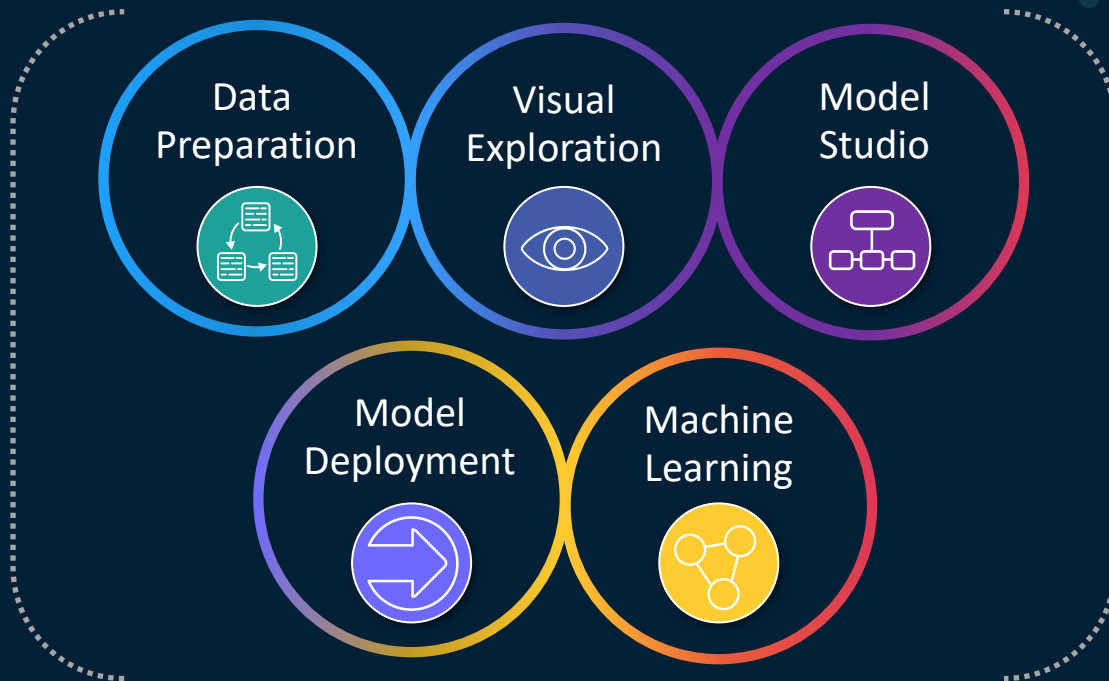


# SAS® Visual Data Mining and Machine Learning

Visual "drag & drop"  
Interface

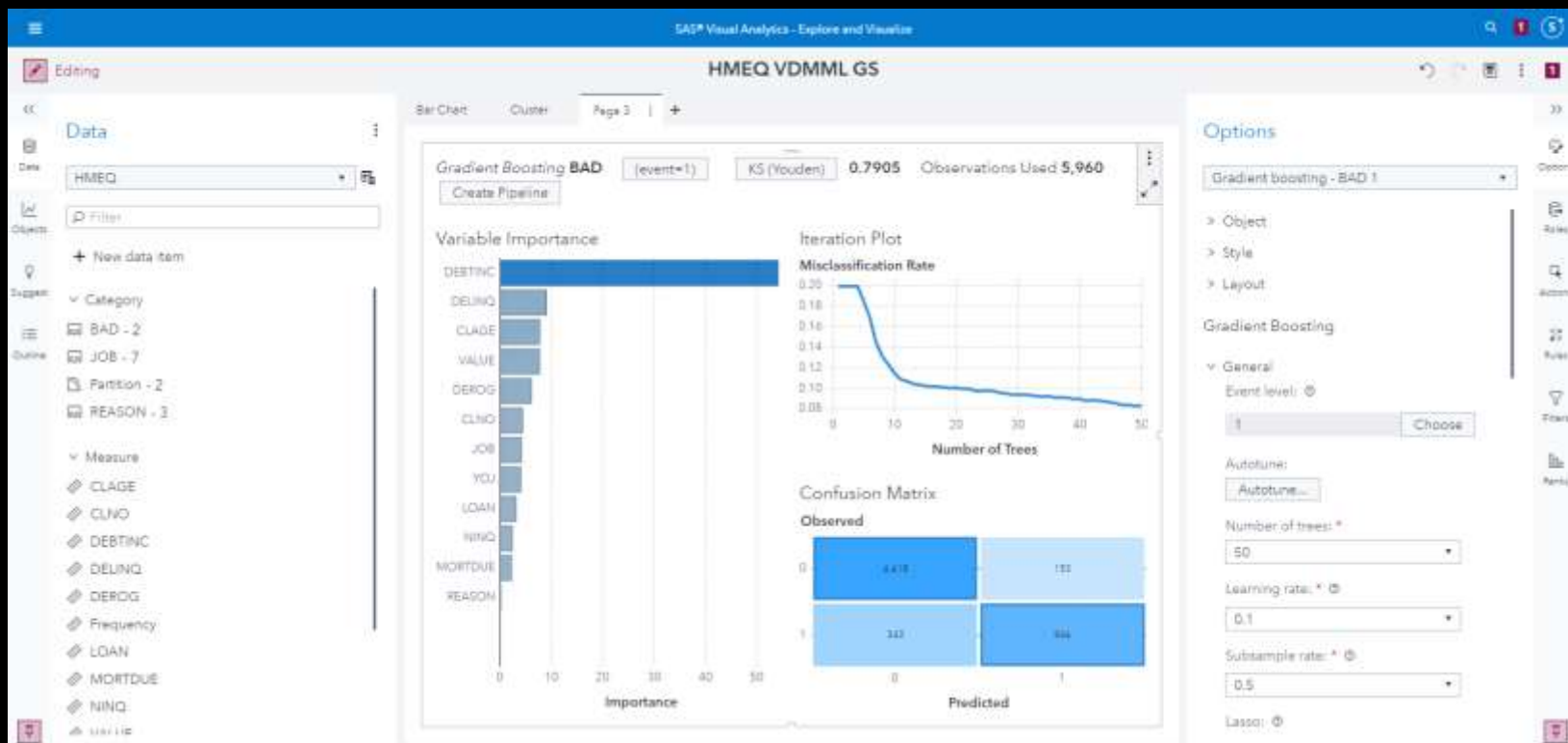


Programming  
Interfaces



# Interfaces

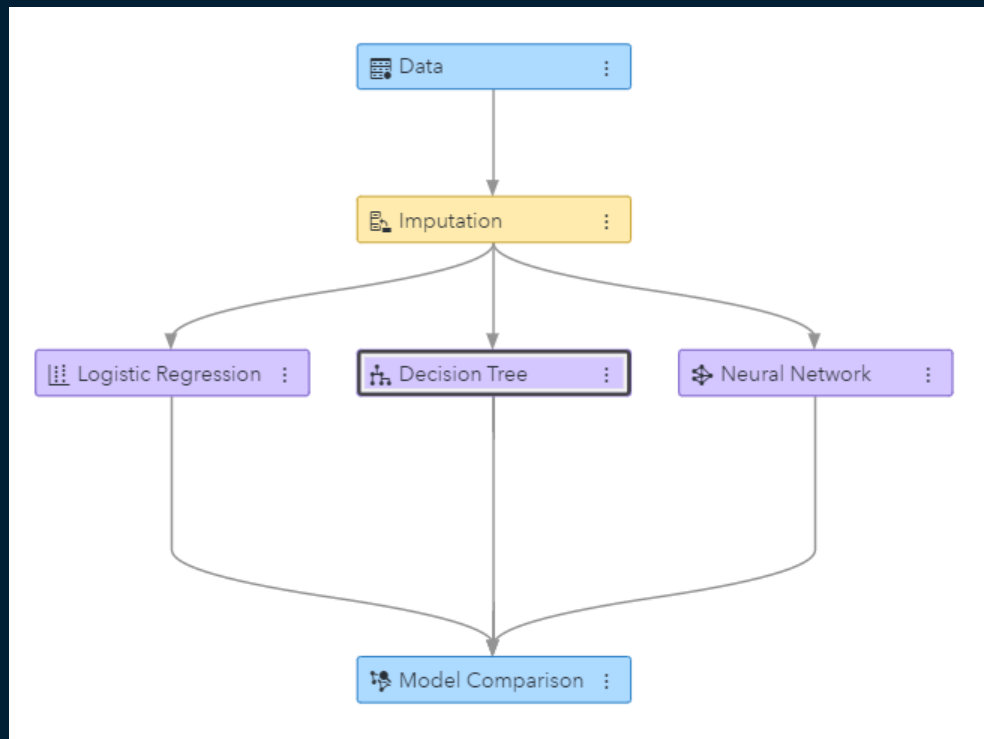
## Building a Model from Scratch in the Visual Reporting Interface



# Interfaces

## Build Models Using Pipelines in Model Studio

- Drag-and-drop pipelines including preprocessing and machine learning techniques
- Customizable and portable nodes and SAS best practice pipelines (Toolbox)
- Support for SAS coding (macro, data step, procs, batch Enterprise Miner) within pipelines
- Collaboration using the “Toolbox” – a collection of SAS Best Practice Pipelines, in addition to user-generated templates

















[Example Code for Pipeline](#)
















# SAS® Visual Data Mining and Machine Learning Pipelines

## ▼ Data Mining Preprocessing

-  Anomaly Detection
-  Clustering
-  Feature Extraction
-  Feature Machine
-  Filtering
-  Imputation
-  Interactive Grouping
-  Manage Variables
-  Reject Inference
-  Replacement
-  Text Mining
-  Transformations
-  Variable Clustering
-  Variable Selection








## ▼ Supervised Learning

-  Batch Code
-  Bayesian Network
-  Decision Tree
-  Forest
-  GLM
-  Gradient Boosting
-  Linear Regression
-  Logistic Regression
-  Model Composer
-  Neural Network
-  Quantile Regression
-  Score Code Import
-  SVM

## ▼ Postprocessing

-  Ensemble

## ▼ Miscellaneous

-  Data Exploration
-  Open Source Code
-  SAS Code
-  Save Data
-  Score Data
-  Scorecard
-  Segment Profile

# Interfaces

## Building a Model Using SAS Studio Tasks

The screenshot displays the SAS Studio interface for developing SAS code. The top menu bar includes 'New', 'Options', 'View', 'Open', and 'Save All'. The left sidebar shows a 'Tasks' panel with a search filter and a list of machine learning tasks. The 'SAS Viya Machine Learning' section is expanded, showing 'Supervised Learning' tasks, with 'Gradient Boosting' selected. The main workspace is divided into three panes. The left pane shows the 'DATA' section with 'PUBLIC.HMEQ' selected. The 'Partition Data' section is expanded, showing 'Validation data' selected, 'Test data' unchecked, and 'Identify partitions' set to 'Specify a sample proportion'. The 'Proportion of validation cases' is set to 0.30. The 'ROLES' section is expanded, showing 'Target' set to 'Use a nominal target' with 'BAD' selected. The right pane shows the generated SAS code, which includes a comment block and a PROC GRADBOOST statement.

**Tasks**

- Scoring
- Register
- SAS Viya Statistics
  - Clustering
  - Principal Component Analysis
  - Linear Regression
  - Logistic Regression
  - Generalized Linear Models
  - Partial Least Squares Regression
  - Quantile Regression
  - Decision Tree
- SAS Viya Machine Learning
  - Automated Machine Learning
  - Unsupervised Learning
  - Supervised Learning
    - Neural Network
    - Forest
    - Gradient Boosting**
    - Factorization Machine
    - Support Vector Machine
    - Bayesian Network

**DATA**

- PUBLIC.HMEQ
- Partition Data
  - Input data contains training data. Include:
    - ☒ Validation data
    - ☐ Test data
  - Identify partitions:
    - Specify a sample proportion
  - Proportion of validation cases: \*
    - 0.30
  - ☐ Random number seed

**ROLES**

- Target
  - ☒ Use a nominal target
  - ☐ Use an interval target
- Nominal target: \*
  - BAD

**Code**

```
1 /*  
2 *  
3 * Task code generated by SAS® Studio 3.2  
4 *  
5 * Generated on '2/9/20, 1:41 PM'  
6 * Generated by 'sasdemo'  
7 * Generated on server 'sasserver'  
8 * Generated on SAS platform 'Linux X64 3.10.0-957.27.1.el7.x86_64'  
9 * Generated on SAS version 'V.03.05P00110619'  
10 * Generated on browser 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/  
11 * Generated on web client 'http://10.96.17.31/SASStudioV/main?locale=en_US'  
12 */  
13  
14 ods noproctitle;  
15  
16 proc gradboost data=PUBLIC.HMEQ;  
17   partition fraction(validate=0.3);  
18   target BAD / level=nominal;  
19   input LOAN MORTGAGE VALUE YOI DEROS DELINQ CLAGE NIMQ CLMO DEBTINC /  
20     level=interval;  
21   input REASON JOB / level=nominal;  
22 run;
```

# Visual Data Mining and Machine Learning

## Programming Tasks in SAS Studio

### ◀ SAS Viya Statistics

- 🌳 Clustering
- 📊 Principal Component Analysis
- 📈 Linear Regression
- 📊 Logistic Regression
- 📈 Generalized Linear Model
- 📈 Partial Least Squares
- 📊 Quantile Regression
- 🌳 Decision Tree

### ◀ SAS Viya Machine Learning

#### ▶ Automated Machine Learning

#### ▶ Unsupervised Learning

#### ◀ Supervised Learning

##### 🌳 Neural Networks

##### 🌳 Forest

##### 🌳 Gradient Boosting

##### 📊 Factorization

##### 📊 Support Vector Machines

##### 🌳 Bayesian Networks

#### ◀ Semi-supervised Learning

##### 🌳 Semi-supervised Learning

#### ◀ Computer Vision

##### 📊 Load Images

### ◀ SAS Viya Machine Learning

#### ◀ Automated Machine Learning

##### 🌳 Automated Feature Engineering

#### ◀ Unsupervised Learning

##### 🌳 Fast k-Nearest Neighbor

##### 📊 Robust Principal Component Analysis

##### 📊 Moving Window Principal Component Analysis

##### 📊 Support Vector Data Description

##### 📊 Market Basket Analysis

# Interfaces

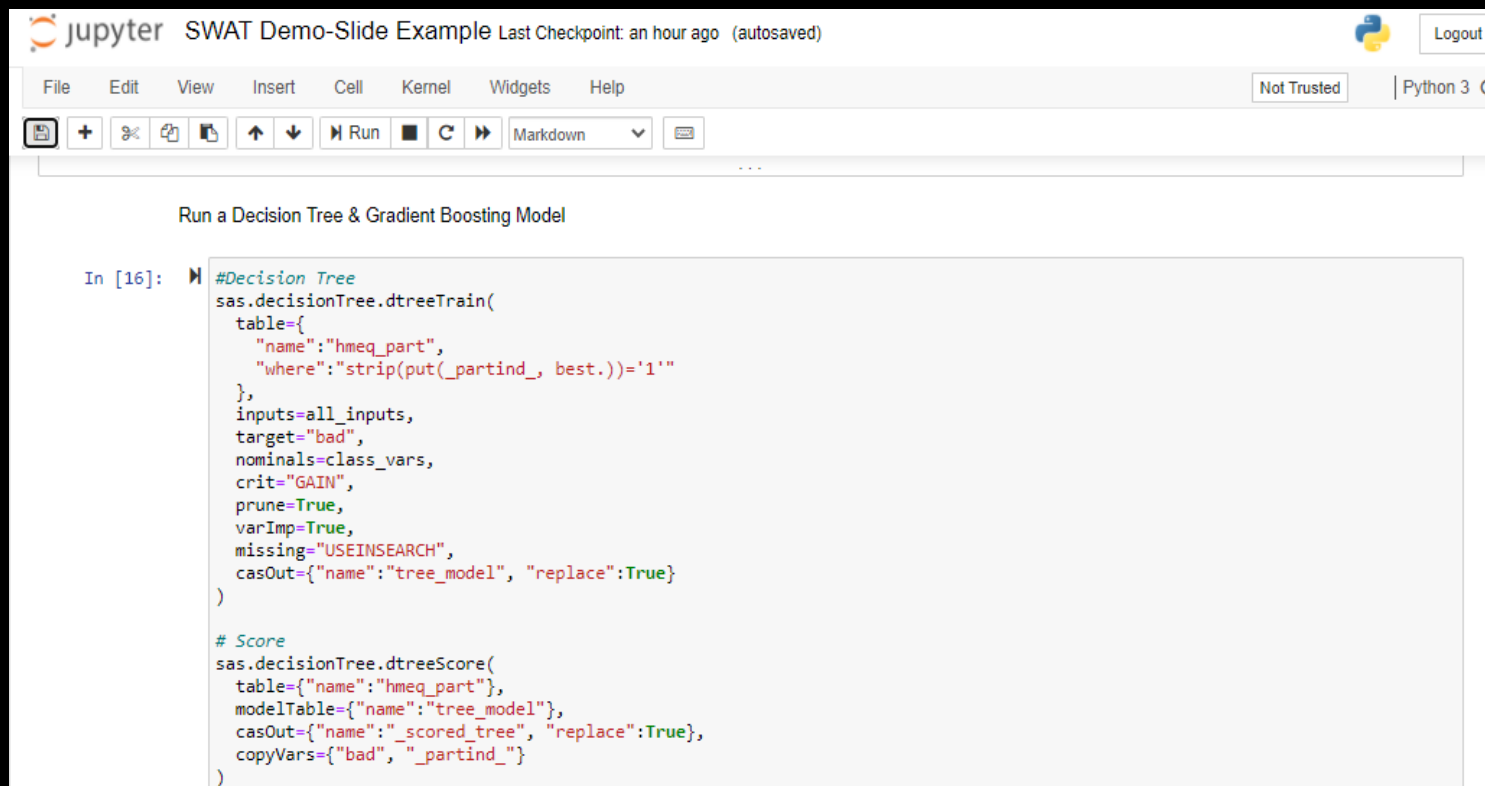
## Building a Model Using SAS Studio Snippets

The screenshot displays the SAS Studio - Develop SAS Code interface. On the left, the Snippets pane is open, showing a tree view of available snippets. The 'SAS Viya Machine Learning' category is expanded, revealing options like 'Load Data', 'Prepare and Explore Data', 'Compare Two ML Algorithms', 'Compare Several ML Algorithms', 'Generalized Linear Models', 'Unsupervised Learning', and 'Supervised Learning'. The 'Supervised Learning' snippet is selected. The main editor area shows a SAS program with the following code:

```
115 by _NAME_;
116 run;
117
118 /* Variance explained by iteration plot */
119 proc sgplot data=out_iter_trans;
120 title "Variance Explained by Iteration";
121 yaxis label="Variance Explained";
122 vbar iteration / response=CGI group=_NAME_;
123 run;
124
125 /* Build a predictive model using Random Forest */
126
127
128 proc forest data=Acaslibname._prepped ntrees=50 nmbin=20 minleafsize=5;
129 input &interval_inputs. / level = interval;
130 input &class_inputs. / level = nominal;
131 target &target / level = nominal;
132 partition rolevar=_partind (train='1' validate='0');
133 code file="&outdir._forest.sas";
134 ods output FitStatistics=fitstats;
135 run;
136
137 /* Score the data using the generated model */
138
139
140 data &caslibname._scored_forest;
141 set &caslibname._prepped;
142 %include "&outdir._forest.sas";
143 run;
```

# Interfaces

## Building a Model Using Open Source



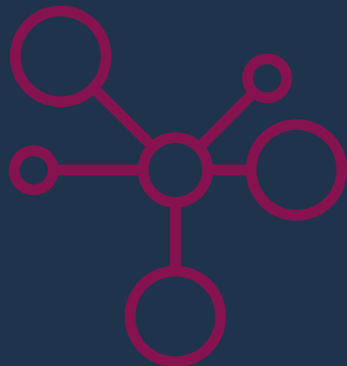
The image shows a Jupyter Notebook interface for a project named "SWAT Demo-Slide Example". The top bar includes the Jupyter logo, the project name, and a status message "Last Checkpoint: an hour ago (autosaved)". On the right, there is a "Logout" button and a Python 3 logo. Below the top bar is a menu bar with options: File, Edit, View, Insert, Cell, Kernel, Widgets, and Help. To the right of the menu bar is a "Not Trusted" warning and a "Python 3" indicator. Below the menu bar is a toolbar with icons for saving, adding new files, opening recent files, running, and other standard Jupyter actions. The main area of the notebook displays a code cell with the following content:

```
Run a Decision Tree & Gradient Boosting Model

In [16]: #Decision Tree
sas.decisionTree.dtreeTrain(
  table={
    "name":"hmq_part",
    "where":"strip(put(_partind_, best.))='1'"
  },
  inputs=all_inputs,
  target="bad",
  nominals=class_vars,
  crit="GAIN",
  prune=True,
  varImp=True,
  missing="USEINSEARCH",
  casOut={"name":"tree_model", "replace":True}
)

# Score
sas.decisionTree.dtreeScore(
  table={"name":"hmq_part"},
  modelTable={"name":"tree_model"},
  casOut={"name":"_scored_tree", "replace":True},
  copyVars={"bad", "_partind_"})
```

# How does SAS support Machine Learning?



## Review

- What is Machine Learning?
- Terminology and key characteristics
- Introduction to Decision Trees, Random Forest, Gradient Boosting, Neural Networks, and k-means Clustering
- How you can use machine learning in SAS
- Examples in SAS 9.x and SAS Viya

# Resources

Where to learn more

# Machine Learning Algorithms Cheat Sheet

### Unsupervised Learning: Clustering

Flowchart for Unsupervised Learning: Clustering:

- Start: Gaussian Mixture Model
- Decision: Prefer Probability (YES/NO)
- Path YES: k-means
- Path NO: Categorical Variables
- Decision: Hierarchical (YES/NO)
- Path YES: Hierarchical
- Path NO: Need to Specify k
- Decision: Hierarchical (YES/NO)
- Path YES: Hierarchical
- Path NO: DBSCAN

### Supervised Learning: Classification

Flowchart for Supervised Learning: Classification:

- Start: Linear SVM, Naive Bayes
- Decision: Data Is Too Large (YES/NO)
- Path YES: Naive Bayes
- Path NO: Explainable
- Decision: Explainable (YES/NO)
- Path YES: Decision Tree, Logistic Regression
- Path NO: Speed or Accuracy
- Decision: Speed or Accuracy (SPEED/ACCURACY)
- Path SPEED: Kernel SVM
- Path ACCURACY: Random Forest, Neural Network, Gradient Boosting Tree

### Unsupervised Learning: Dimension Reduction

Flowchart for Unsupervised Learning: Dimension Reduction:

- Start: Dimension Reduction
- Decision: Have Responses (YES/NO)
- Path YES: Topic Modeling, Probabilistic
- Path NO: Principal Component Analysis, Singular Value Decomposition
- Decision: Probabilistic (YES/NO)
- Path YES: Latent Dirichlet Analysis
- Path NO: Singular Value Decomposition

### Supervised Learning: Regression

Flowchart for Supervised Learning: Regression:

- Start: Predicting Numeric
- Decision: Predicting Numeric (YES/NO)
- Path YES: Speed or Accuracy
- Decision: Speed or Accuracy (SPEED/ACCURACY)
- Path SPEED: Decision Tree
- Path ACCURACY: Random Forest, Neural Network, Gradient Boosting Tree



# Recommended Resources

## An Overview of SAS® Visual Data Mining

<https://support.sas.com/resources/papers/proceedings17/SAS1492-2017.pdf>

## Video - Automated Machine Learning at Scale

[http://www.sas.com/en\\_us/webinars/automated-machine-learning-scale.html](http://www.sas.com/en_us/webinars/automated-machine-learning-scale.html)

## Machine learning - what it is and why it matters (reading)

[http://www.sas.com/en\\_us/insights/analytics/machine-learning.html](http://www.sas.com/en_us/insights/analytics/machine-learning.html)

## Live web and classroom training - Big Data, Data Mining, and Machine Learning

[Big Data course](#)

# Recommended Resources

Machine learning - what it is and why it matters (reading)

[http://www.sas.com/en\\_us/insights/analytics/machine-learning.html](http://www.sas.com/en_us/insights/analytics/machine-learning.html)

Live web and classroom training - Big Data, Data Mining, and Machine Learning

[Big Data course](#)

# SAS Tutorial

## Videos

### How to Choose a Machine Learning Algorithm

<https://youtu.be/-oZcf0QEzYM>

### Transforming variables in SAS

<https://communities.sas.com/t5/SAS-Data-Mining-and-Machine/New-video-Transforming-Variables-in-SAS/m-p/710687#M8553>

# SAS® Visual Data Mining and Machine Learning

[Try it before you buy!](#)

SAS® Visual Data Mining and Machine Learning

[Overview](#)

[Support](#)

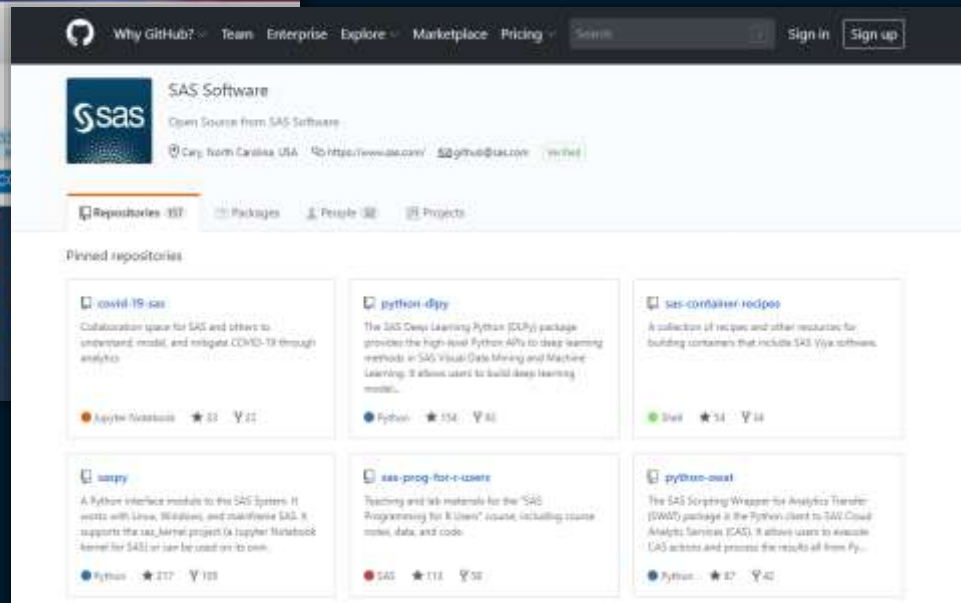
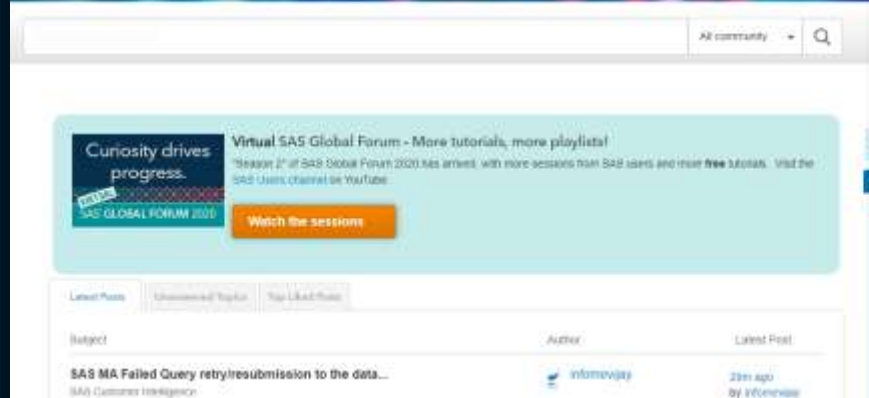
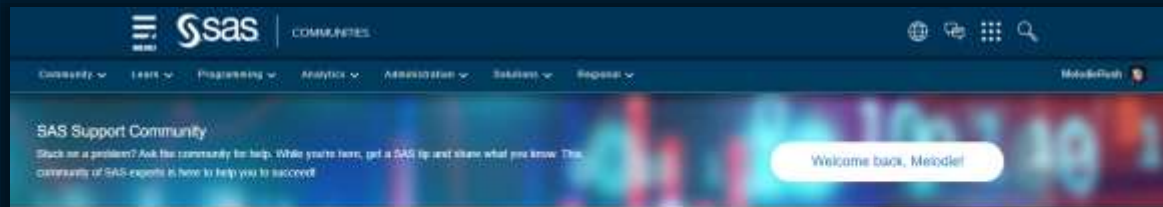
[Free Trial](#)

SAS® VISUAL DATA MINING AND MACHINE LEARNING

Everything you need to solve the most complex analytical problems – in a single, integrated, collaborative solution.

Try it for free

# Communities



[Communities.sas.com](https://communities.sas.com)  
[Github.com/sassoftware](https://github.com/sassoftware)  
[Developer.sas.com](https://developer.sas.com)



# Questions?

# Thank you for your time and attention!

Connect with me:

LinkedIn: <https://www.linkedin.com/in/melodierush>

Twitter: @Melodie\_Rush

sas.com

