SAS® Visual Data Mining and Machine Learning (VDMML)

Getting Started

**Presenter:** Melodie Rush, Principal Data Scientist  
**Q&A:** Twanda Baker, Data Scientist  
**Host:** Lauren Gray, Webinar Host
Goals

• Increase awareness of and comfort with capabilities in SAS® Visual Data Mining and Machine Learning™ (VDMML)
• Share resources for learning more
Presentation Content

• Introduction to SAS® Visual Data Mining and Machine Learning
• Value of SAS® Visual Data Mining and Machine Learning
• Included Algorithms
• Tour of the interfaces
  • Visual
  • Programming
  • Open Source
The volume of inquiry calls from Gartner clients about AI, advanced ML and related topics increased by 200% between 2015 and 2016. The data science platform segment grew 9.0% from 2015 to 2016.

Source: Gartner, July 2017
Lack of architecture to support the analytics life cycle
Need for data science talent
Missing Data
Model performance deterioration
Long time to interactively build analytical models
The SAS Platform
SAS® Visual Data Mining and Machine Learning is an end-to-end machine learning solution on the most advanced analytics platform.
SAS® Visual Data Mining and Machine Learning

Key Benefits

**SPEED**
Get answers fast through integrated advanced analytics workflows in a single environment

**EFFICIENCY**
Boost productivity of your data scientists through access from open source

**PRECISION**
Get accurate answers to your questions by applying modern machine learning and predictive analytics algorithms

**SCALABLE**
Scale environment with your growing needs

*End to end data preparation, analytics, machine learning and model scoring in a single in-memory, scalable environment*
Visual Data Mining and Machine Learning

What does it include?

- Visual Analytics
- Visual Statistics
- Visual Data Mining and Machine Learning

- Baseline Procedures
- VS Procedures
- VDMML Procedures

- Python, Java, Lua, R
- REST APIs

Requires Visual Analytics
Requires Visual Statistics

Baseline Action sets
VS Action sets
VDMML Action sets
Visual Data Mining and Machine Learning

What do you get?

Visualizations
- Forest
- Gradient Boosting
- Neural Networks
- Support Vector Machines
- Factorization Machines
- Bayesian Networks

VDMML PROCs
- FOREST
- GRADBOOST
- NNET
- SVMACHINE
- FACTMAC
- TEXTMINE
- TMSCORE
- BOOLRULE
- ASTORE
- CAS
- NETWORK
- BNET
- FASTKNN
- more...

VDMML CAS action sets
- MLEARNING
- TEXTMINE
- DMMLVISSET
- CRSBOOLRULE
- CRSNEURALNET
- CRSSVM
- CRSTKFACTMAC
- TKCAS
- CRSNETSOC
- CSRNETCOMMON
- CRSASTORE
- CRSCMPTRVSN
- CRSDTREEADVN
- CRSTXTMINADV
- more...
SAS® Visual Data Mining and Machine Learning 8.3
Visual Interface

Machine Learning Techniques
• Forest
• Factorization Machine
• Gradient Boosting
• Neural Network
• Support Vector Machine
• Bayesian Networks

Common Features
• Training-Validation
• Model Assessment
• Model Comparison
• Score Code or Astore Table
• Ability to export model statistics into Excel
SAS Visual Data Mining and Machine Learning 8.3
Programming Tasks in SAS Studio

Includes algorithms in the visual interface plus

• Unsupervised Learning
  • Moving Window PCA
  • Robust PCA
  • Support Vector Data Description
  • Text Parsing and Topic Discovery

• Supervised Learning
  • Quantile Regression
  • Partial Least Squares Regression
What’s Included
SAS® Visual Data Mining and Machine Learning

Visual “drag & drop” Interface

Programming Interface

Data Preparation

Visual Exploration

Model Building

Model Deployment

Machine Learning
Data Preparation

- Access to different data sources
- Training-Validation Data Partitioning
- Feature Engineering (e.g. parameters, interactions)
- Variable selection and missing values
Visual Exploration

- Interactively discover relationships, trends, outliers
- Smart autocharting
- Analytics driven visualizations
- Explore predicted outputs
- Variable transformation
Model Building

- Pipeline of activities
- Drag and drop and access to code
- Nodes are run asynchronously
- Reproducibility
- SAS best practice toolkit
Modern Machine Learning

- Forest
- Neural Network (including Deep Learning)
- Gradient Boosting
- Support Vector Machines
- Factorization Machines
- Bayesian Networks
- Autotuning
Comparison and Deploy

- Model comparison summaries
- Interactively assess models
- Assessment charts for partitioned data
- Publish score code; batch, API call, in-database
Visual Interface

SAS Visual Analytics
A random forest is used primarily when building classification models on large datasets.

It is an ensemble model of many decision trees from slightly different samples of the training data. Specifically:
- Each tree is built on subset of observations (rows)
- The features (variables) available to each splitting node are subset

Forest Business Use Cases:
- Customer Churn. Determining the factors that can cause churn.
- Loan Default. Predicting who will default on their loan using customer behavior patterns.

Forest Competitive Differentiators
- Distributed and massively parallel
- Faster, more memory-efficient, and more scalable algorithm
- Truly deployable
Supervised Learning: VDMML Forest
Supervised Learning: VDMML
Gradient Boosting: PROC GRADBOOST

**Gradient Boosting**

Gradient boosting is used primarily when building classification models on large datasets.

It generates many decision trees sequentially from slightly different subsamples of the training data. In the end, all models are given a weight depending on their accuracy, and the model results are combined into one consolidated result.

GB Business Use Cases:
- Customer Churn. Determining the factors that can cause churn.
- Loan Default. Predicting who will default on their loan using customer behavior patterns.

GB Competitive Differentiators
- Distributed and massively parallel
- Faster, more memory-efficient, and more scalable algorithm
- Truly deployable
Supervised Learning: VDMML

Gradient Boosting
Neural networks are used to solve a wide variety of tasks that are hard to solve using other methods.

The goal of the neural network is to solve problems in the same way that the human brain would. They are comprised of processing elements called units or neurons.

NNET Business Use Cases:
- Computer vision.
- Speech recognition.

NNET Competitive Differentiators
- Distributed and massively parallel
- Faster, more memory-efficient, and more scalable algorithm
- Can be auto-tuned
Supervised Learning: VDMML
Neural Networks
Support vector machine is used to separate or classify data into groups.

The standard SVM model solves binary classification problems that produce non-probability output (only sign +1/-1) by constructing a set of hyperplanes that maximize the margin between two classes.

SVM Business Use Cases:
- Customer survey. Predicting which product a customer might buy next based on previous behavior and survey results.
- Predicting bad debt or loan default.

SVM Competitive Differentiators
- Distributed and massively parallel
- Faster, more memory-efficient, and more scalable algorithm
- Truly deployable
Supervised Learning: VDMML
Support Vector Machines
Factorization machines are used when data has high dimensionality and sparse data.

The most common use case is in recommender engines. “If you like item A, you may also like item B”.

FM Business Use Cases:
- Customer survey. Predicting which product a customer might buy next based on previous behavior and survey results.
- Online real-time recommendations. Surfacing relevant recommendations based on products that the customer views.

FM Competitive Differentiators
- Distributed and massively parallel
- Faster, more memory-efficient, and more scalable algorithm
- Truly deployable
Supervised Learning: VDMML

Factorization Machine
A Bayesian network is a directed, acyclic graphical model in which the nodes represent random variables and the links between the nodes represent conditional dependency between two random variables.

BNET Business Use Cases:
- They can be used for a wide range of tasks including prediction, anomaly detection, diagnostics, automated insight, reasoning, time series prediction and decision making under uncertainty.

BNET Competitive Differentiators
- When you have a lot of missing data, e.g. in medicine, BN's can be very effective since modeling the joint reduces your dependency in having a fully observed dataset.
- When you want to model a domain in a way that is visually transparent, and also aims to capture cause→effect relationships, BN's can be very powerful. Note that the causality assumption in BN's is open to debate though.
Supervised Learning: VDMML

Bayesian Network
Click on applications menu and select Explore and Visualize Data.
Click on Blue Box and Select New Report
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
Visual Interface

Pipelines
SAS® Visual Data Mining and Machine Learning 8.3

Pipelines

- 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 through the use of the “Toolbox” – a collection of SAS Best Practice Pipelines, in addition to user-generated templates

Example Code for Pipeline
Pipelines
SAS® Visual Data Mining and Machine Learning 8.3

Pipelines

• Automated API generation for retraining and scoring
• Ability to deploy models in to databases directly
• Assessment against imported Test datasets
• Integration with Model Manager for versioning, tracking and deployment
• Integration with SAS 9.4 Enterprise Miner score code & Batch Code
• Integration with Open Source (R and Python)
Visual Interface Demo

Pipelines
SAS Programming Interface

SAS Studio
SAS® Visual Data Mining and Machine Learning

Openness for a Non-SAS Programmer

SAS Language  Python, R, Lua & Java  REST APIs
### SAS® Visual Data Mining and Machine Learning

#### Analytical Capabilities

<table>
<thead>
<tr>
<th><strong>Data Manipulation</strong></th>
<th><strong>Statistics</strong></th>
<th><strong>Machine Learning</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>In-Memory Data Step</td>
<td>Cox Proportional Hazards</td>
<td>Bayesian Networks</td>
</tr>
<tr>
<td>Frequency / Crosstab</td>
<td>Decision Trees</td>
<td>Boolean Rules</td>
</tr>
<tr>
<td>Data Transpose</td>
<td>Design Matrix</td>
<td>Factorization Machines</td>
</tr>
<tr>
<td>Variable Binning</td>
<td>General Additive Models</td>
<td>Frequent Item Set Mining</td>
</tr>
<tr>
<td>Variable Cardinality Analysis</td>
<td>Generalized Linear Models</td>
<td>Gradient Boosting</td>
</tr>
<tr>
<td>Variable Summary</td>
<td>K-means and K-modes Clustering</td>
<td>K Nearest Neighbor</td>
</tr>
<tr>
<td>Sampling and Partitioning</td>
<td>Linear Regression</td>
<td>Image Processing</td>
</tr>
<tr>
<td>Missing Value Imputation</td>
<td>Logistic Regression</td>
<td>Market Basket Analysis</td>
</tr>
<tr>
<td>Variable Selection</td>
<td>Nonlinear Regression</td>
<td>Moving Windows PCA</td>
</tr>
<tr>
<td>Model Assessment</td>
<td>Ordinary Least Squares Regression</td>
<td>Network Analytics/Community Detection</td>
</tr>
<tr>
<td>DS2</td>
<td>Partial Least Squares Regression</td>
<td>Neural Networks / Deep Learning</td>
</tr>
<tr>
<td>FedSQL</td>
<td>Pearson Correlation</td>
<td>Random Forest</td>
</tr>
<tr>
<td></td>
<td>Principal Component Analysis</td>
<td>Robust PCA</td>
</tr>
<tr>
<td></td>
<td>Quantile Regression</td>
<td>Support Vector Data Description</td>
</tr>
<tr>
<td></td>
<td>Shewhart Control Chart Analysis</td>
<td>Support Vector Machines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Text Mining</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Variable Clustering</td>
</tr>
</tbody>
</table>
SAS Programming Interface Demo

SAS Studio
Autotuning
Automating Autotuning: Hyperparameters

- Training a model involves using an algorithm to determine model parameters or other logic to map inputs to a target.
- Tuning a model involves determining the algorithm hyperparameters (tuning options) that result in the model which maximizes predictability on an independent data set.

Very data/problem dependent!
Autotuning

Methods

• SAS Visual Data Mining and Machine Learning offers:
  • Random search (highly parallelizable)
  • Latin Hypercube (highly parallelizable)
  • LH + proprietary SAS/OR algorithm (sequential in nature – *named Genetic in Options*)
• All 5 VDMML procs as well as Decision Trees can be autotuned via SAS Visual interface and SAS Studio interface
Autotune Statement
How SAS proprietary tuning is done

Decision tree: PROC TREESPLIT
• Depth of tree
• Splitting criterion
• Number of bins for interval variables
Forest: PROC FOREST
• Number of trees
• Number of levels in each tree
• Bootstrap sampling rate
• Number of inputs used for splitting a node
Gradient Boosting: PROC GRADBOOST
• Number of iterations (trees)
• Sampling proportion
• LASSO (L1) regularization
• Ridge (L2) regularization
• Number of inputs used for splitting a node
• Learning Rate
Neural Networks: PROC NNET
• Number of hidden layers
• Number of neurons in each hidden layer
• L1 regularization
• L2 regularization
• SGD options (annealing rate, learning rate)
Support Vector Machines: PROC SVMACHINE
• Polynomial degree
• Penalty value
Factorization Machine: PROC FACTMAC
• Number of factors
• Step size (learning rate)
• Number of iterations

Uses Standard Grid, Random Search or Latin Hypercube to seed the Genetic algorithm

LOOP until stop criterion (e.g. max time, max models, max iterations, population size etc.)
SAS Programming Interface - AutoTuning Demo

SAS Visual Analytics & SAS Studio
Open Source Interface

Jupyter Notebooks
The New Languages of SAS 9.4 and SAS Viya: A SAS Programmer’s Primer

Open Access

SAS language
APIs
Other programming languages – R, Lua, Python and Java
Developer & user communities

Ask the Expert

The New Languages of SAS 9.4 and SAS Viya: A SAS Programmer’s Primer

Copyright © SAS Institute Inc. All rights reserved.
SAS Viya
An Example

APIs

proc print data = hmeq (obs = 10);
run;

df = s.CASTable('hmeq')
df.head(10)

df <- defCasTable(s, 'hmeq')
head(df, 10)

Translated Command

[table.fetch]
table.name = "hmeq"
from = 1 to = 10
CAS Actions Hierarchies

```plaintext
```
table.attribute <result = results> <status = rc> /

attributes = {
    column = "string",
    * key = "string",
    value = "string" | 64-bit-integer | integer | double | binary-large-object
}, {...}
```
PROC versus CAS Action

```
proc factmac data=mycas.movlens nfactors=10 learnstep=0.15
   maxiter=20  outmodel=mycas.factors;
   input userid itemid /level=nominal;
   target rating /level=interval;
   output out=mycas.out1 copyvars=(userid itemid rating);
run;
```

```
proc cas;
   action factmac result=R / table={name="movlens"},
      outModel={name="factors_out", replace=true},
      inputs={"userid", "itemid"},
      nominals={"userid", "itemid"},
      target="rating",
      maxIter=20, nFactors=10, learnStep=0.15,
      output={casout={name="score_out", replace="TRUE"},
              copyvars={"userid","itemid","rating"}};
run;
```
Open Source Interface Demo

Jupyter Notebooks
Additional Capabilities
Integrating Viya and SAS Enterprise Miner
Viya Code Node
Other Features in SAS VDMML 8.3

Additional Analytical Algorithms and Options

- Tensor Factorization
- Neural Network Autoencoders
- Clustering mixed variables
- Deep forward neural networks (DNNs), convolutional neural networks (CNNs) and recurrent neural networks (RNNs)
- Bayesian Network
- Market Basket Analysis
- Image Processing (only CAS Actions currently)
  - Load images recursively & at random
  - Retrieve Image labels across all folders when importing
  - Convert image table action (wide format)
  - Support image processing with Deep Learning
What’s New in VDMML 8.3
In case you have an earlier version

• SAS Drive replaces SAS Home
• Visual Interface
  • Bayesian Network in Visual Interface
  • Model Interpretability Charts
    - Variable Importance
    - Partial Dependence
    - ICE
    - LIME
  • Create Pipelines or Add Pipelines to existing projects
• Registering models from Visual Interface

• Model Studio
  • Remembers where you were
  • Added to Pipelines
    - Feature Engineering Template
    - Supervised Learning
      - Batch Code
      - Quantile Regression
      - Score Code Import
    - Miscellaneous – Open Source Code
    - Transformation Node – Best Transformation
Resources
Where can I learn more?
SAS® Visual Data Mining and Machine Learning

Try it before you buy!

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
SAS® Visual Data Mining and Machine Learning 
Visual Interface – SAS Visual Analytics

SAS Visual Data Mining and Machine Learning on Visual Interface
https://goo.gl/qDNdCS
SAS® Visual Data Mining and Machine Learning
Programmatic Interface - SAS Studio

SAS Visual Data Mining and Machine Learning on SAS Studio
https://youtu.be/X0AU4gDUc_Y
SAS® Visual Data Mining and Machine Learning
Programming with Open Source

SAS Visual Data Mining and Machine Learning with Python Demo

https://youtu.be/LXoikPWQJ3o
SAS® Visual Data Mining and Machine Learning

Where to learn more?

Video Resources

- Video Tutorials

SAS Visual Data Mining and Machine Learning
https://youtu.be/X0AU4gDUc_Y

SAS Visual Data Mining and Machine Learning with Python
https://youtu.be/LXoikPWQJ3o

‘How do I’ videos
http://support.sas.com/training/tutorial/viya/index.html
Key Resources

- SAS VDMML Product Web Page
- Factsheet
- SAS Viya Brochure
- Documentation
- VDMML SAS Community
Communities
Questions?
Thank you for your time and attention!

Connect with me:
LinkedIn: [https://www.linkedin.com/in/melodierush](https://www.linkedin.com/in/melodierush)
Twitter: @Melodie_Rush