Getting Started with SAS® Visual Data Mining and Machine Learning (VDMML)

Ask the Expert

Melodie Rush

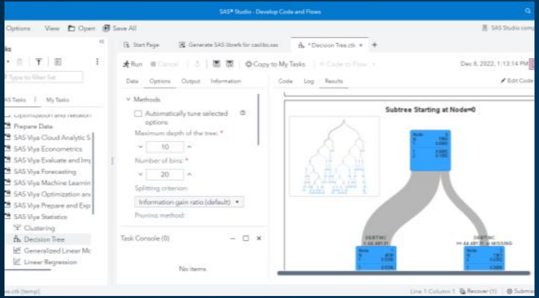
Global Customer Success Principal Data Scientist

Connect with me:
LinkedIn: https://www.linkedin.com/in/melodierush

SAS® Visual Data Mining and Machine Learning

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



Methods

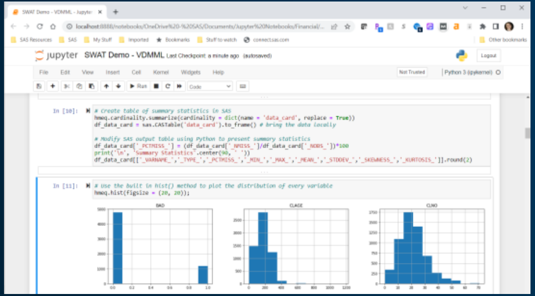
☐ Automatically tune selected options

Maximum depth of the tree: * 10

Number of bins: * 20

Splitting criterion: Information gain ratio (default)

Pruning method:

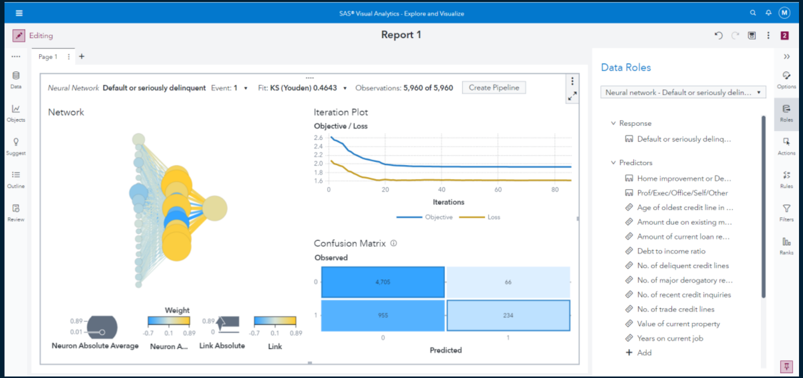


jupyter SWAT Demo - VDMML Last Checkpoint: a minute ago (autosaved)

```
In [10]:  # Create table of summary statistics in SAS
          hmeq.cardinality.summarize(cardinality = dict(name = 'data_card', replace = True))
          df_data_card = sas.CASTable('data_card').to_frame() # bring the data locally

          # Modify SAS output table using Python to present summary statistics
          df_data_card['_PCTMISS_'] = (df_data_card['_NMISS_']/df_data_card['_NOBS_'])*100
          print('\n', 'Summary Statistics'.center(90, ' '))
          df_data_card[['_VARNAME_','_TYPE_','_PCTMISS_','_MIN_','_MAX_','_MEAN_','_STDDEV_','_SKEWNESS_','_KURTOSIS_']].round(2)

In [11]:  # Use the built in hist() method to plot the distribution of every variable
          hmeq.hist(figsize = (20, 20));
```



Neural Network Default or seriously delinquent Event: 1 Fit: KS (Youden) 0.4643 Observations: 5,960 of 5,960

Observed	0	1
0	4,705	66
1	955	234

Data Roles

Neural network - Default or seriously delin...

Response
- Default or seriously delinq...

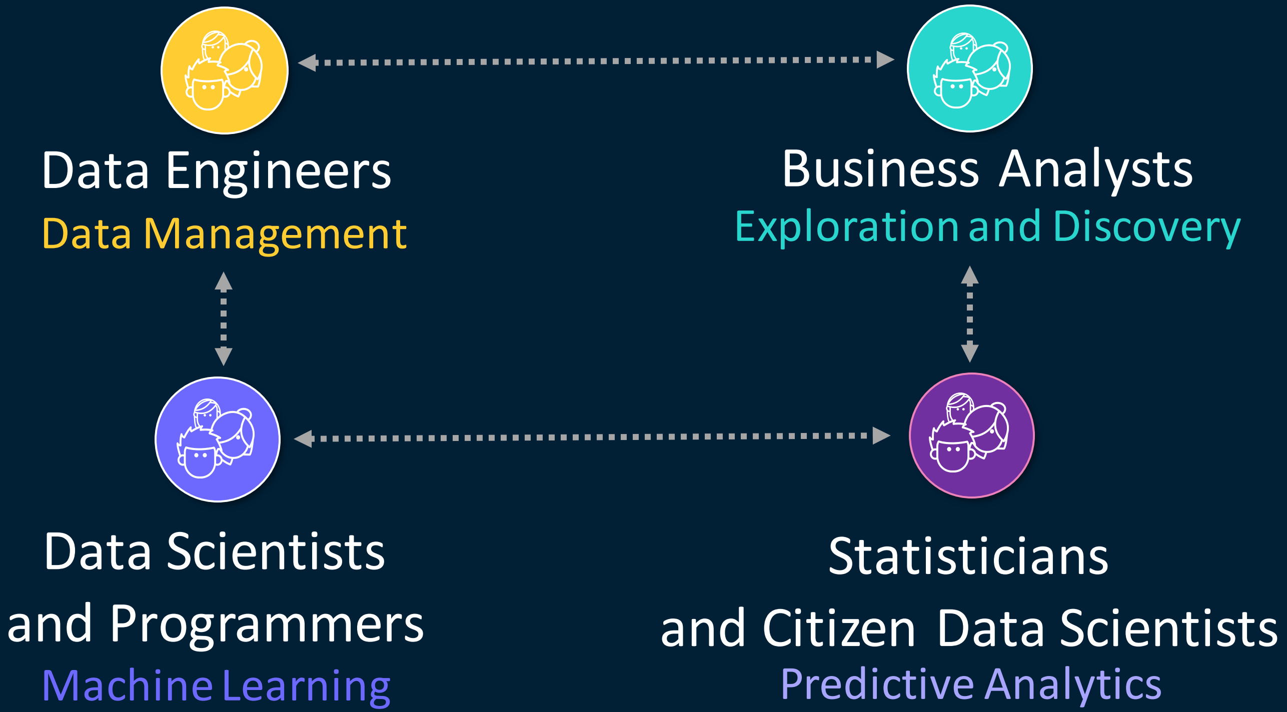
Predictors
- Home improvement or De...
- Prof/Exec/Office/Self/Other
- Age of oldest credit line in ...
- Amount due on existing m...
- Amount of current loan re...
- Debt to income ratio
- No. of deliquent credit lines
- No. of major derogatory re...
- No. of recent credit inquiries
- No. of trade credit lines
- Value of current property
- Years on current job

+ Add

1. Integrated data mining and machine learning
2. Open platform with access from SAS and Open Source
3. Modern machine learning algorithms
4. Cloud-ready and industrial scalable analytics

SAS® Visual Data Mining and Machine Learning is an end-to-end machine learning solution on the most advanced analytics platform.

Collaboration and Personas



Data Engineers

Data Management

Business Analysts

Exploration and Discovery

Data Scientists
and Programmers

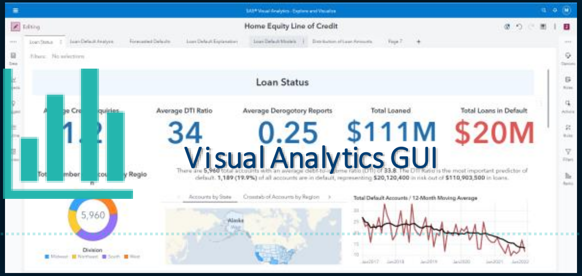
Machine Learning

Statisticians
and Citizen Data Scientists

Predictive Analytics

SAS Visual Data Mining and Machine Learning

What does it include?



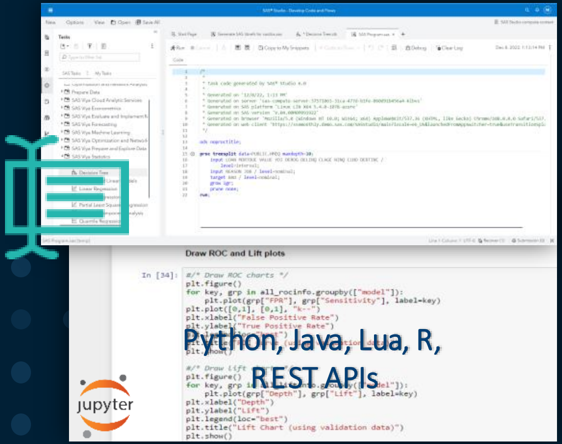
Home Equity Line of Credit

Loan Status

Average DTI Ratio	Average Derogotory Reports	Total Loaned	Total Loans in Default
34	0.25	$111M	$20M

SAS® Visual Analytics - Explore and Visualize
5,960
Division
Total Default Accounts / 12-Month Moving Average

Visual Analytics GUI



```
ods noproctitle;

proc treesplit data=PUBLIC.HMEQ maxdepth=10;
    input LOAN MORTDUE VALUE YOJ DEROG DELINQ CLAGE NINQ CLNO DEBTINC /
        level=interval;
    input REASON JOB / level=nominal;
    target BAD / level=nominal;
    grow igr;
    prune none;
run;
```

Draw ROC and Lift plots

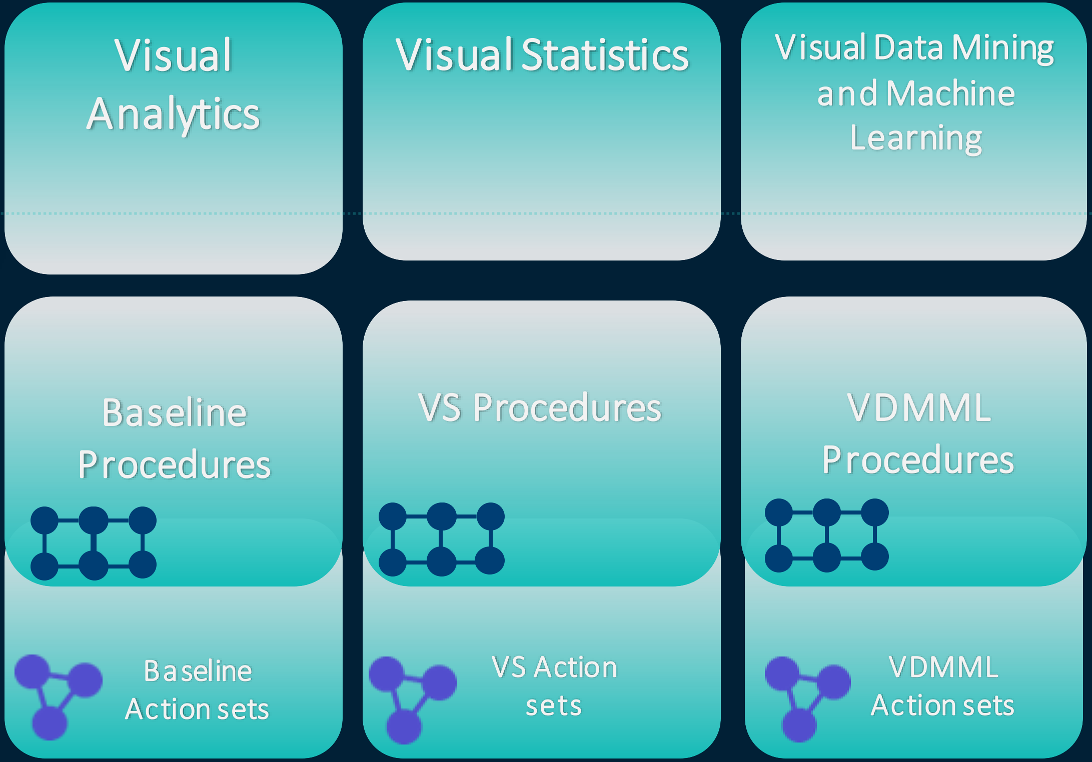
```
In [34]: #/* Draw ROC charts */
         plt.figure()
         for key, grp in all_rocinfo.groupby(["model"]):
             plt.plot(grp["FPR"], grp["Sensitivity"], label=key)
         plt.plot([0,1], [0,1], "k--")
         plt.xlabel("False Positive Rate")
         plt.ylabel("True Positive Rate")
         plt.legend(loc="best")
         plt.title("ROC Curve (using validation data)")
         plt.show()

         #/* Draw Lift charts */
         plt.figure()
         for key, grp in all_liftinfo.groupby(["model"]):
             plt.plot(grp["Depth"], grp["Lift"], label=key)
         plt.xlabel("Depth")
         plt.ylabel("Lift")
         plt.legend(loc="best")
         plt.title("Lift Chart (using validation data)")
         plt.show()
```

Python, Java, Lua, R, REST APIs

Requires Visual Analytics

Requires Visual Statistics



Visual Analytics	Visual Statistics	Visual Data Mining and Machine Learning
Baseline Procedures	VS Procedures	VDMML Procedures
Baseline Action sets	VS Action sets	VDMML Action sets

SAS Visual Data Mining and Machine Learning

What do you get?

Visualizations

- Forest
- Gradient Boosting
- Neural Networks
- Support Vector Machines
- Factorization Machines
- Bayesian Networks
- Interactive Decision Tree

From Visual Statistics

- Regression (linear and logistic)
- Decision Tree
- Clustering
- Nonparametric Logistic
- Generalized Additive Model
- Generalized Linear Model

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…

Visual Interface

SAS Visual Analytics

SAS Visual Data Mining and Machine Learning

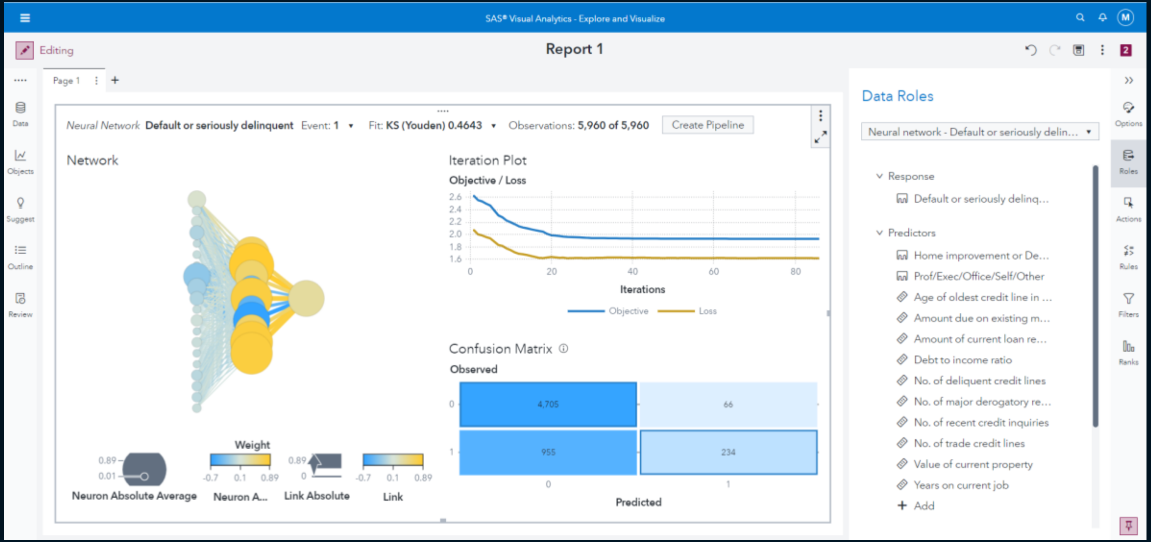
Visual Interface

Machine Learning Techniques

- Forest
- Factorization Machine
- Gradient Boosting
- Neural Network
- Support Vector Machine
- Bayesian Network

Common Features

- Training-Validation
- Model Assessment
- Model Comparison
- Score Code or Astore Table
- Ability to export model statistics



Neural Network **Default or seriously delinquent** Event: **1** ▾ Fit: **KS (Youden) 0.4643** ▾ Observations: **5,960 of 5,960**

Observed \ Predicted	0	1
0	4,705	66
1	955	234

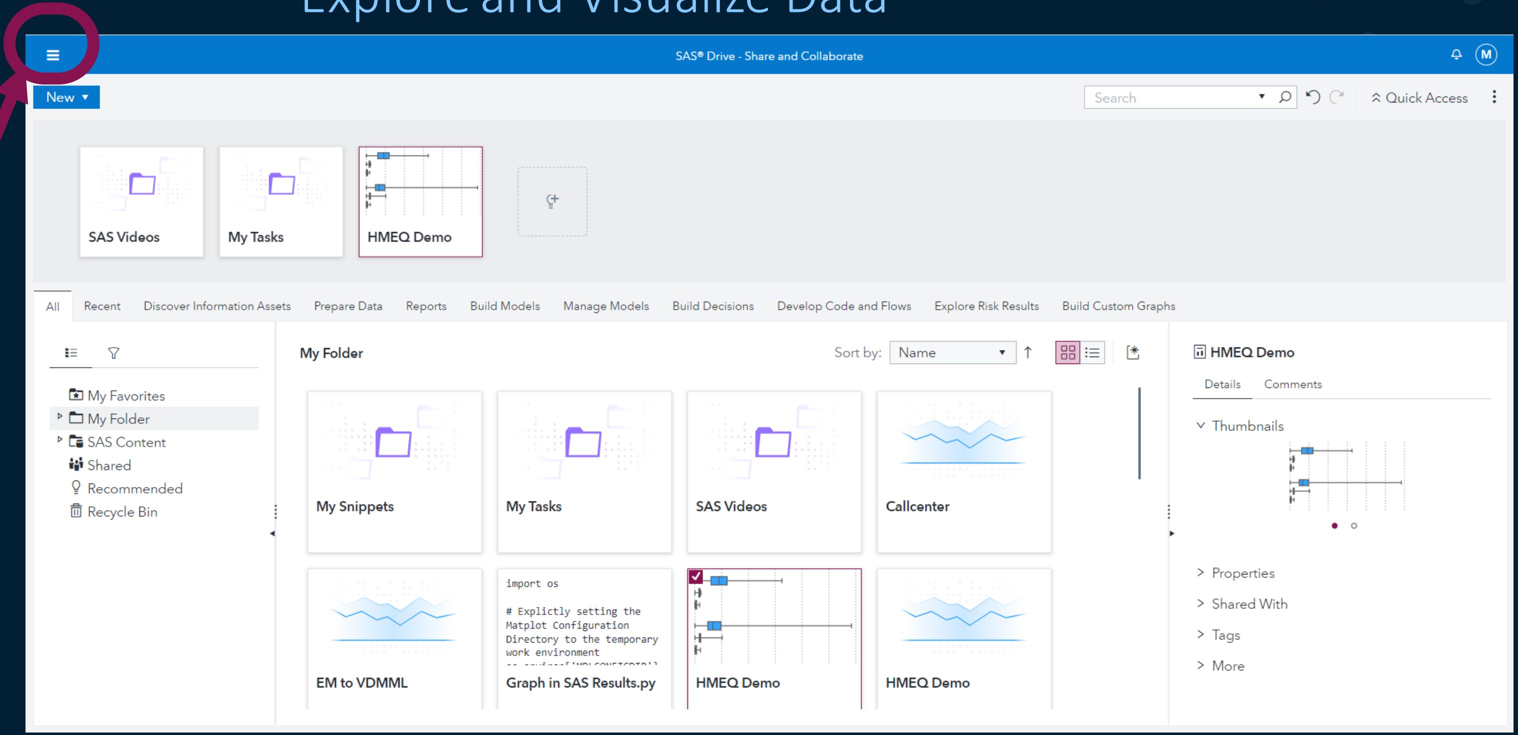
Neural network - Default or seriously delin... ▾

- Response
  - Default or seriously delinq...
- Predictors
  - Home improvement or De...
  - Prof/Exec/Office/Self/Other
  - Age of oldest credit line in ...
  - Amount due on existing m...
  - Amount of current loan re...
  - Debt to income ratio
  - No. of deliquent credit lines
  - No. of major derogatory re...
  - No. of recent credit inquiries
  - No. of trade credit lines
  - Value of current property
  - Years on current job
  - + Add

SAS® Drive

Explore and Visualize Data

Click on applications menu and select Explore and Visualize Data



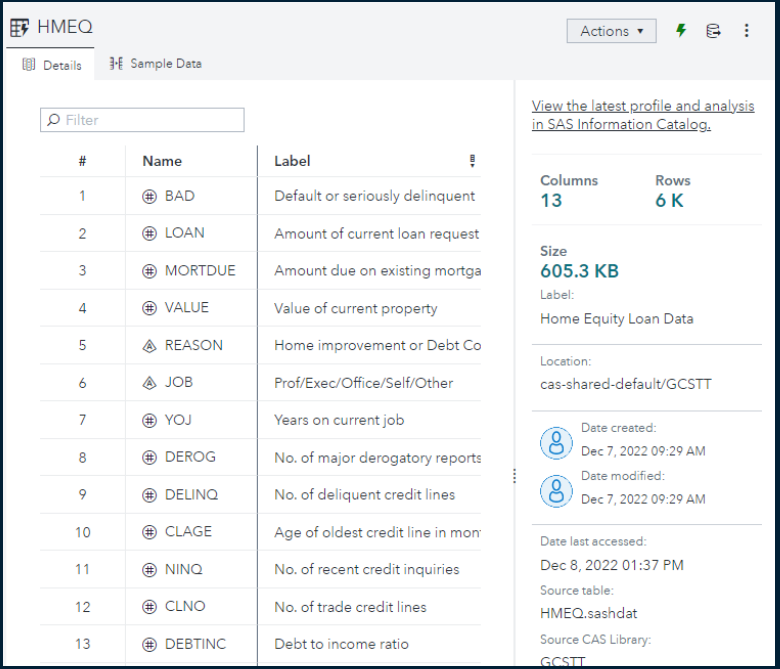
```
import os

# Explictly setting the
Matplot Configuration
Directory to the temporary
work environment
```

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



#	Name	Label
1	BAD	Default or seriously delinquent
2	LOAN	Amount of current loan request
3	MORTDUE	Amount due on existing mortga
4	VALUE	Value of current property
5	REASON	Home improvement or Debt Co
6	JOB	Prof/Exec/Office/Self/Other
7	YOJ	Years on current job
8	DEROG	No. of major derogatory reports
9	DELINQ	No. of deliquent credit lines
10	CLAGE	Age of oldest credit line in mon
11	NINQ	No. of recent credit inquiries
12	CLNO	No. of trade credit lines
13	DEBTINC	Debt to income ratio

View the latest profile and analysis in SAS Information Catalog.

Columns
13

Rows
6 K

Size
605.3 KB

Label:
Home Equity Loan Data

Location:
cas-shared-default/GCSTT

Date created:
Dec 7, 2022 09:29 AM

Date modified:
Dec 7, 2022 09:29 AM

Date last accessed:
Dec 8, 2022 01:37 PM

Source table:
HMEQ.sashdat

Source CAS Library:
GCSTT

Visual Interface Demo

SAS Visual Analytics

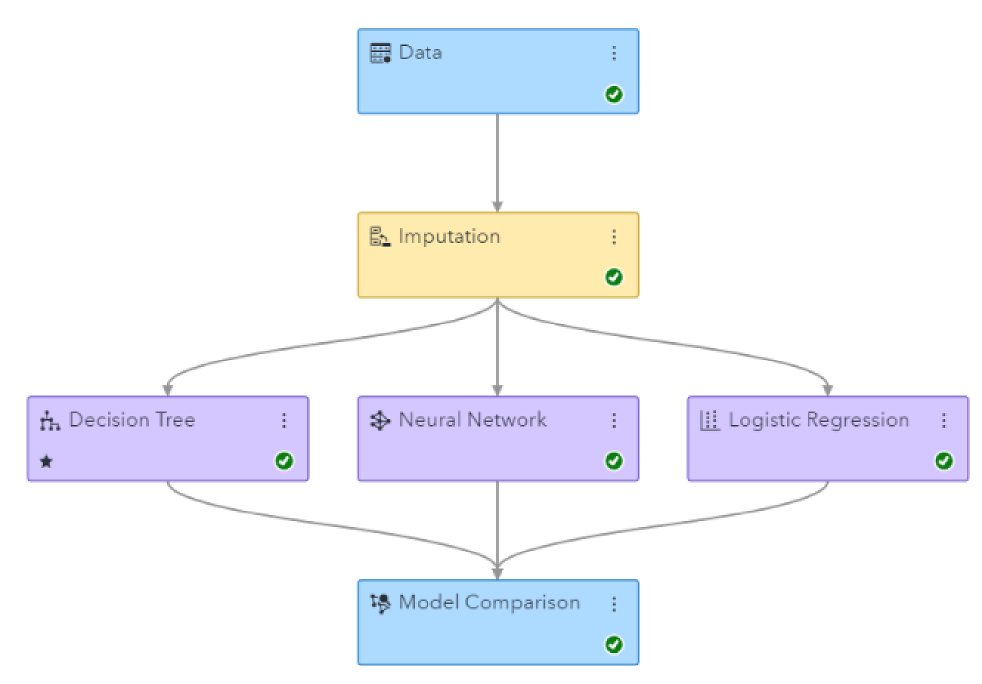
Visual Interface

Model Studio - Pipelines

SAS® Visual Data Mining and Machine Learning

Pipelines

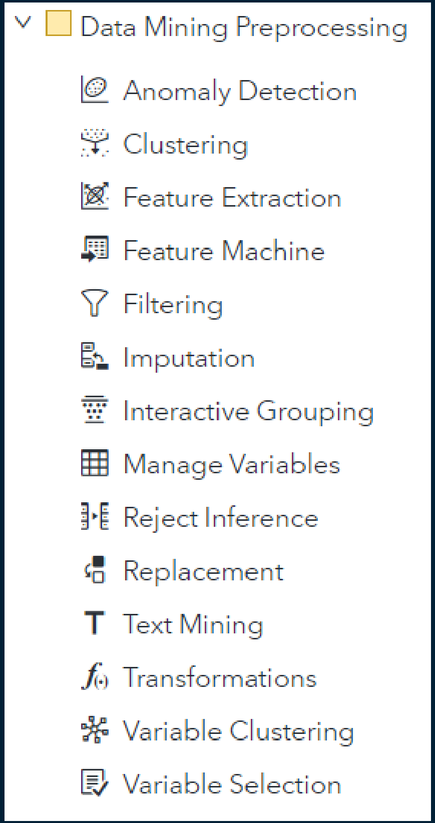
- Drag-and-drop pipelines including preprocessing and machine learning techniques
- Customizable and portable nodes and SAS best practice pipelines (Toolbox)
- Support for SAS coding (macro, data step, procs, batch Enterprise Miner) within pipelines
- Collaboration using the “Toolbox” – a collection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



Example Code for Pipeline

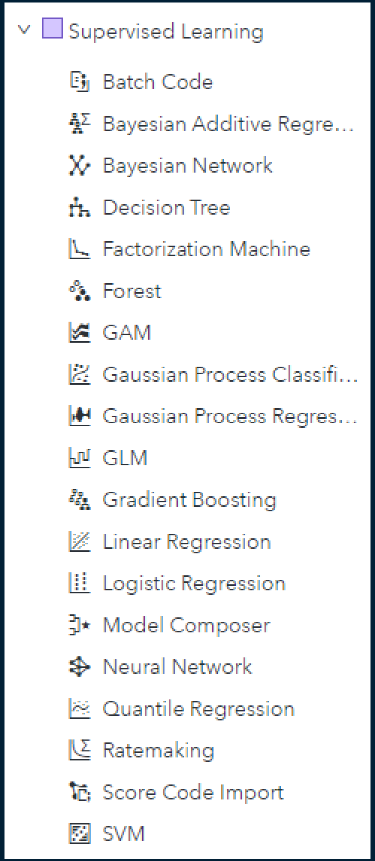
SAS® Visual Data Mining and Machine Learning

Pipelines



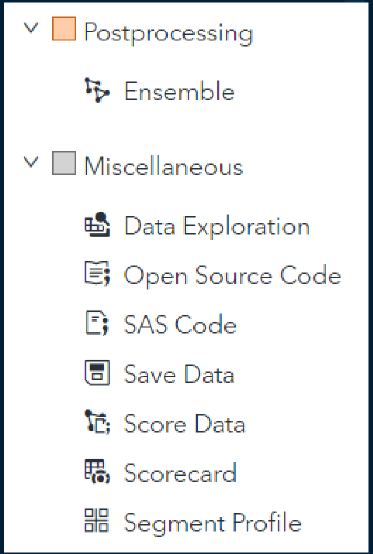
Data Mining Preprocessing

- 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 Additive Regre…
- Bayesian Network
- Decision Tree
- Factorization Machine
- Forest
- GAM
- Gaussian Process Classifi…
- Gaussian Process Regres…
- GLM
- Gradient Boosting
- Linear Regression
- Logistic Regression
- Model Composer
- Neural Network
- Quantile Regression
- Ratemaking
- Score Code Import
- SVM



Postprocessing

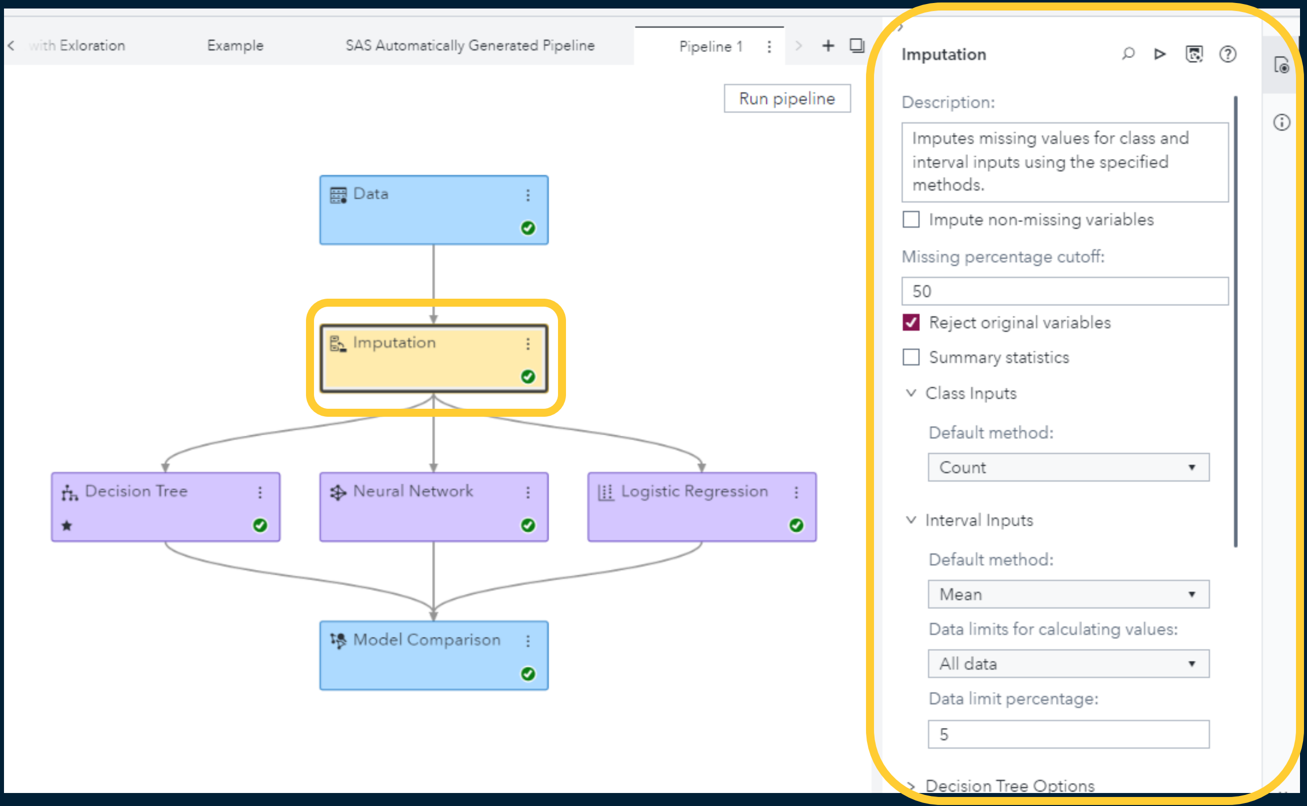
- Ensemble

Miscellaneous

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

SAS® Viya

Pipelines



Imputation

Description:

Imputes missing values for class and interval inputs using the specified methods.

☐ Impute non-missing variables

Missing percentage cutoff: 50

☑ Reject original variables

☐ Summary statistics

Class Inputs

Default method: Count

Interval Inputs

Default method: Mean

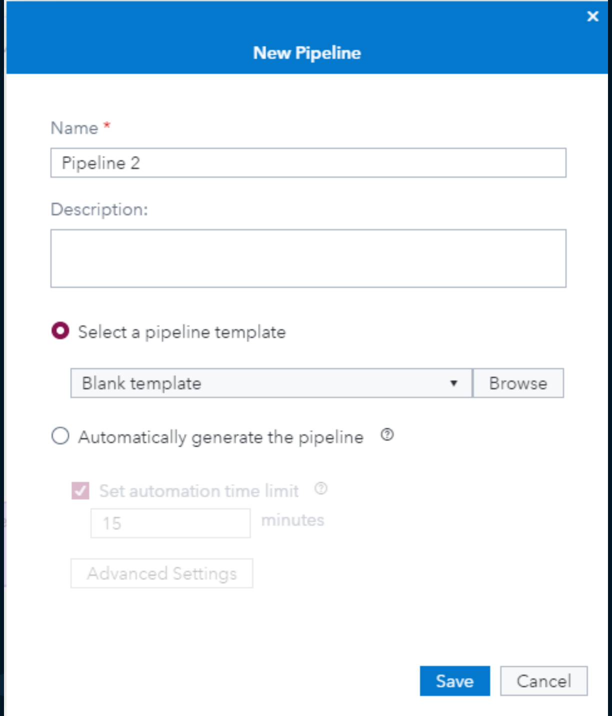
Data limits for calculating values: All data

Data limit percentage: 5

Decision Tree Options

Build Pipelines

Use prebuilt templates or automatically generate the pipeline



New Pipeline

Name *

Pipeline 2

Description:

● Select a pipeline template

Blank template ▾ | Browse

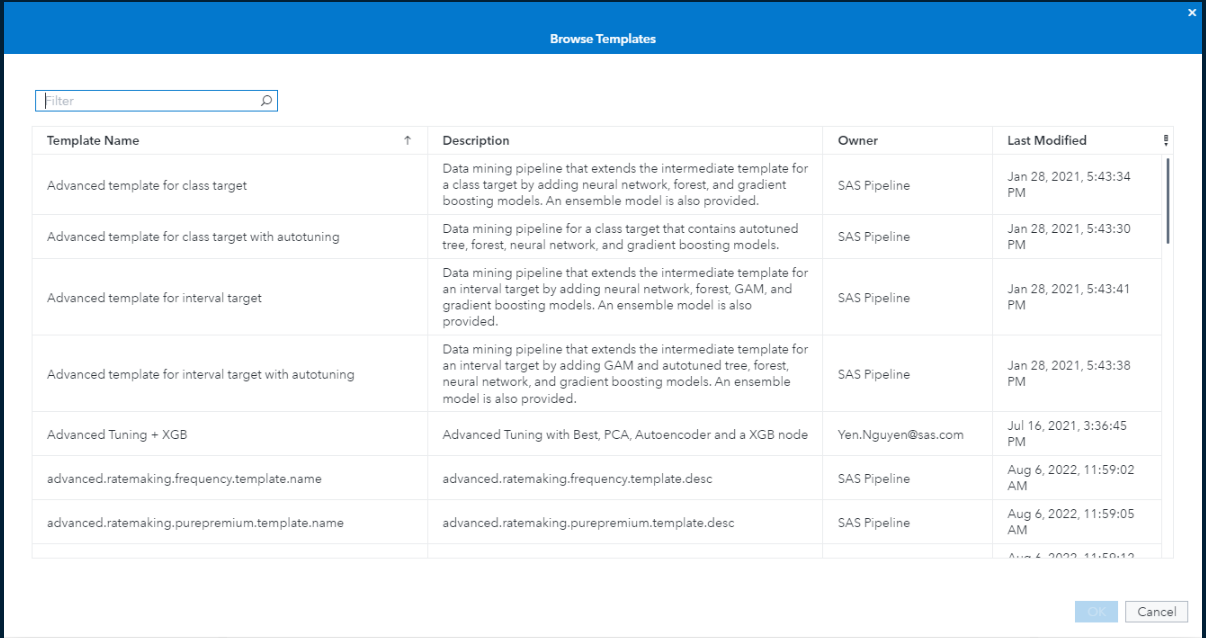
○ Automatically generate the pipeline

☑ Set automation time limit

15 minutes

Advanced Settings

Save | Cancel



Browse Templates

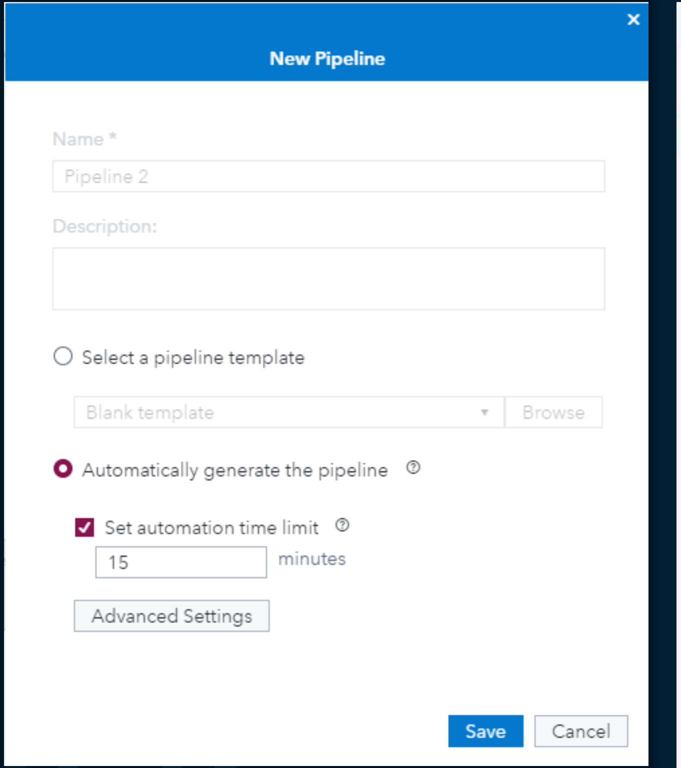
Filter

Template Name ↑	Description	Owner	Last Modified
Advanced template for class target	Data mining pipeline that extends the intermediate template for a class target by adding neural network, forest, and gradient boosting models. An ensemble model is also provided.	SAS Pipeline	Jan 28, 2021, 5:43:34 PM
Advanced template for class target with autotuning	Data mining pipeline for a class target that contains autotuned tree, forest, neural network, and gradient boosting models.	SAS Pipeline	Jan 28, 2021, 5:43:30 PM
Advanced template for interval target	Data mining pipeline that extends the intermediate template for an interval target by adding neural network, forest, GAM, and gradient boosting models. An ensemble model is also provided.	SAS Pipeline	Jan 28, 2021, 5:43:41 PM
Advanced template for interval target with autotuning	Data mining pipeline that extends the intermediate template for an interval target by adding GAM and autotuned tree, forest, neural network, and gradient boosting models. An ensemble model is also provided.	SAS Pipeline	Jan 28, 2021, 5:43:38 PM
Advanced Tuning + XGB	Advanced Tuning with Best, PCA, Autoencoder and a XGB node	Yen.Nguyen@sas.com	Jul 16, 2021, 3:36:45 PM
advanced.ratemaking.frequency.template.name	advanced.ratemaking.frequency.template.desc	SAS Pipeline	Aug 6, 2022, 11:59:02 AM
advanced.ratemaking.purepremium.template.name	advanced.ratemaking.purepremium.template.desc	SAS Pipeline	Aug 6, 2022, 11:59:05 AM

OK | Cancel

Build Pipelines

Use prebuilt templates or automatically generate the pipeline



New Pipeline

Name *

Pipeline 2

Description:

○ Select a pipeline template

Blank template ▾ Browse

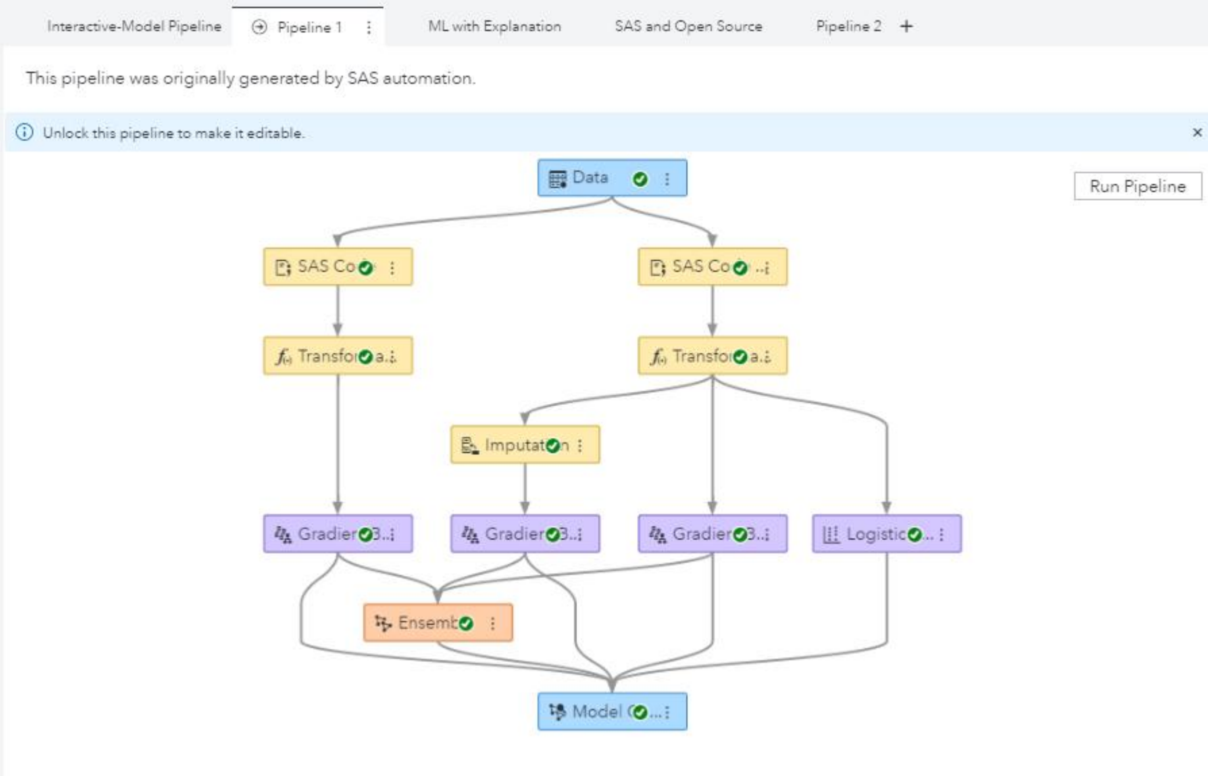
● Automatically generate the pipeline

☑ Set automation time limit

15 minutes

Advanced Settings

Save Cancel



This pipeline was originally generated by SAS automation.

Unlock this pipeline to make it editable.

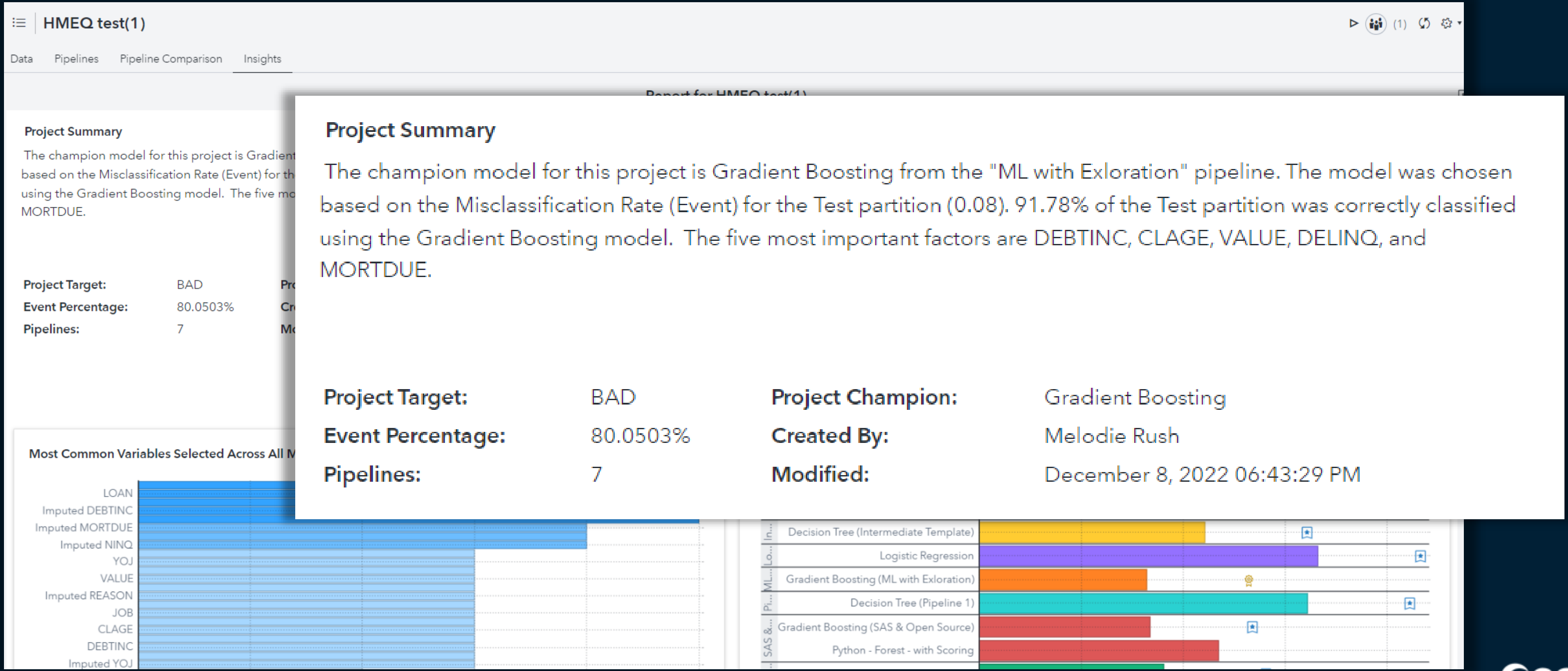
Automated Insights & Interpretability

Model Interpretability – What and Why?

A machine learning algorithm's interpretability refers to how ***easy*** *it is for* ***humans*** *to* ***understand the cause of a decision***

Automated Insights & Interpretability

Description in simple language



HMEQ test(1)

Project Summary

The champion model for this project is Gradient Boosting from the "ML with Exloration" pipeline. The model was chosen based on the Misclassification Rate (Event) for the Test partition (0.08). 91.78% of the Test partition was correctly classified using the Gradient Boosting model. The five most important factors are DEBTINC, CLAGE, VALUE, DELINQ, and MORTDUE.

Project Target:	BAD	**Project Champion:**	Gradient Boosting
Event Percentage:	80.0503%	**Created By:**	Melodie Rush
Pipelines:	7	**Modified:**	December 8, 2022 06:43:29 PM

Most Common Variables Selected Across All M

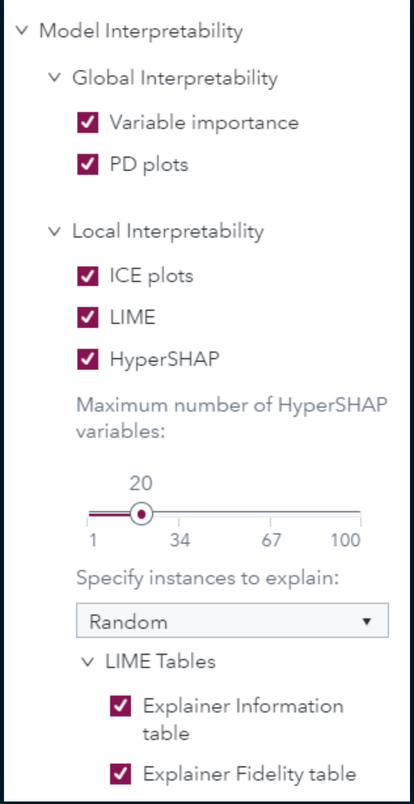
LOAN
Imputed DEBTINC
Imputed MORTDUE
Imputed NINQ
YOJ
VALUE
Imputed REASON
JOB
CLAGE
DEBTINC
Imputed YOJ
Decision Tree (Intermediate Template)
Logistic Regression
Gradient Boosting (ML with Exloration)
Decision Tree (Pipeline 1)
Gradient Boosting (SAS & Open Source)
Python - Forest - with Scoring
In...
Lo...
ML...
Pi...
SAS &...

Automated Insights & Interpretability

Model Interpretability Charts

To better understand model inputs/outputs

- Variable Importance Plots and Rankings
- Partial Dependence (PD) Plots
- LIME (Local Interpretable Model-agnostic Explanations)
- ICE (Individual Conditional Expectation) Plots
- Kernel SHAP Method (Shapley Values)



Model Interpretability

Global Interpretability

- ☑ Variable importance
- ☑ PD plots

Local Interpretability

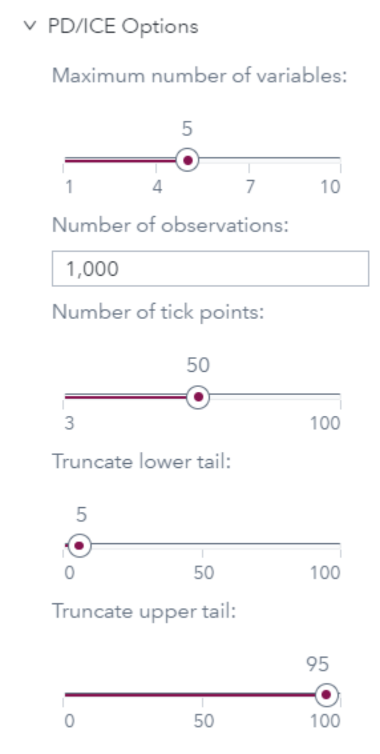
- ☑ ICE plots
- ☑ LIME
- ☑ HyperSHAP

Maximum number of HyperSHAP variables: 20

Specify instances to explain: Random ▾

LIME Tables

- ☑ Explainer Information table
- ☑ Explainer Fidelity table



PD/ICE Options

Maximum number of variables: 5

Number of observations: 1,000

Number of tick points: 50

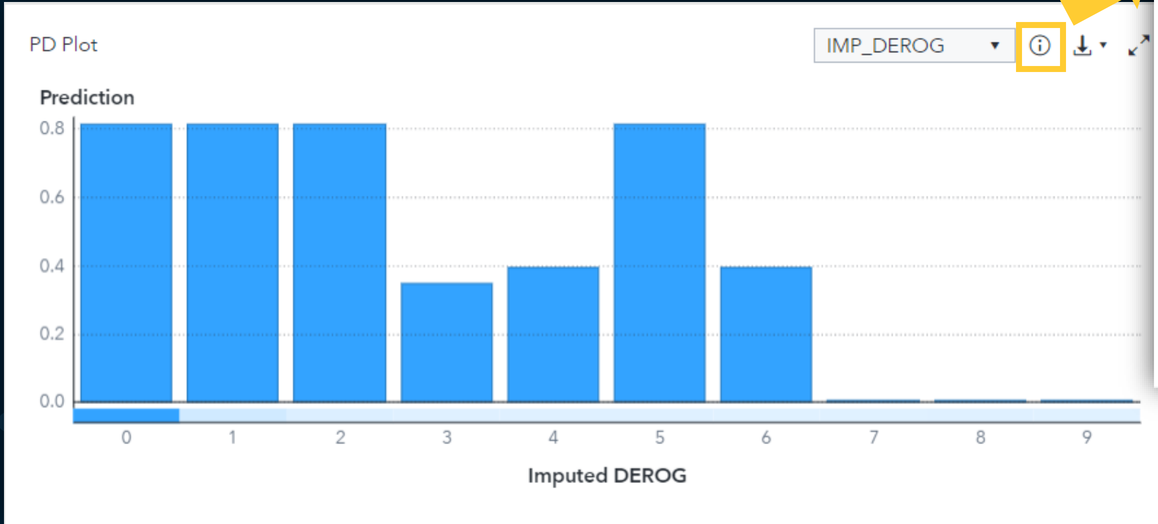
Truncate lower tail: 5

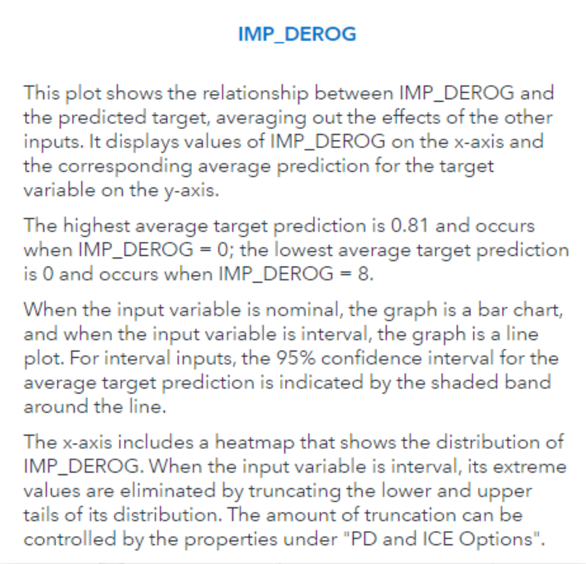
Truncate upper tail: 95

Automated Insights & Interpretability

Model Interpretability Charts

Each interpretability chart has insights included





IMP_DEROG

This plot shows the relationship between IMP_DEROG and the predicted target, averaging out the effects of the other inputs. It displays values of IMP_DEROG on the x-axis and the corresponding average prediction for the target variable on the y-axis.

The highest average target prediction is 0.81 and occurs when IMP_DEROG = 0; the lowest average target prediction is 0 and occurs when IMP_DEROG = 8.

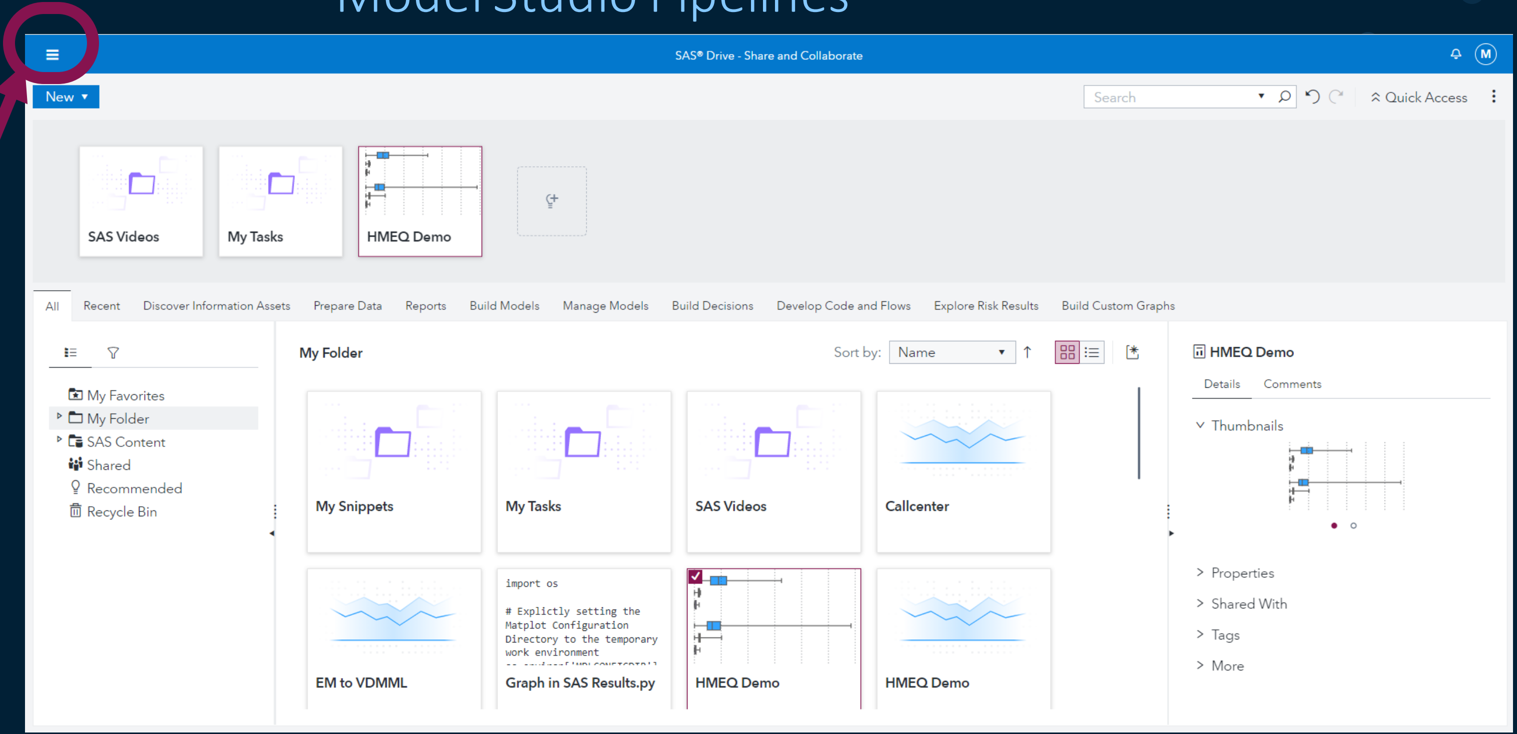
When the input variable is nominal, the graph is a bar chart, and when the input variable is interval, the graph is a line plot. For interval inputs, the 95% confidence interval for the average target prediction is indicated by the shaded band around the line.

The x-axis includes a heatmap that shows the distribution of IMP_DEROG. When the input variable is interval, its extreme values are eliminated by truncating the lower and upper tails of its distribution. The amount of truncation can be controlled by the properties under "PD and ICE Options".

SAS® Drive

Model Studio Pipelines

Click on applications menu and select Build Models



SAS® Drive - Share and Collaborate

New ▾

Search

Quick Access

SAS Videos

My Tasks

HMEQ Demo

All | Recent | Discover Information Assets | Prepare Data | Reports | Build Models | Manage Models | Build Decisions | Develop Code and Flows | Explore Risk Results | Build Custom Graphs

- My Favorites
- My Folder
- SAS Content
- Shared
- Recommended
- Recycle Bin

My Folder

Sort by: Name

My Snippets

My Tasks

SAS Videos

Callcenter

EM to VDMML

```
import os

# Explictly setting the
Matplot Configuration
Directory to the temporary
work environment
```

Graph in SAS Results.py

HMEQ Demo

HMEQ Demo

HMEQ Demo

Details | Comments

Thumbnails

- Properties
- Shared With
- Tags
- More

Visual Interface Demo

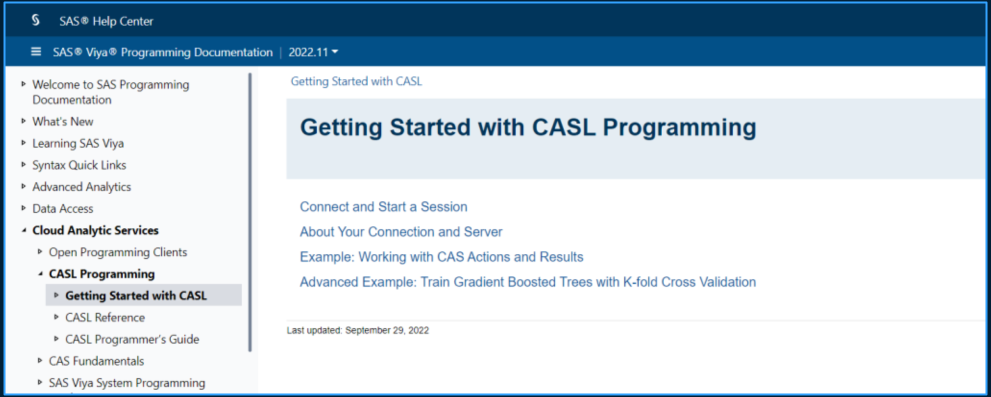
Model Studio - Pipelines

SAS Programming Interface

SAS Studio & CASL

Cloud Analytic Services Language (CASL)

- CASL is the language specification that enables you to access the CAS server.
- Designed to offer easy access to SAS Viya functionality
- Modeled after DATA step and IML
- Available via PROC CAS

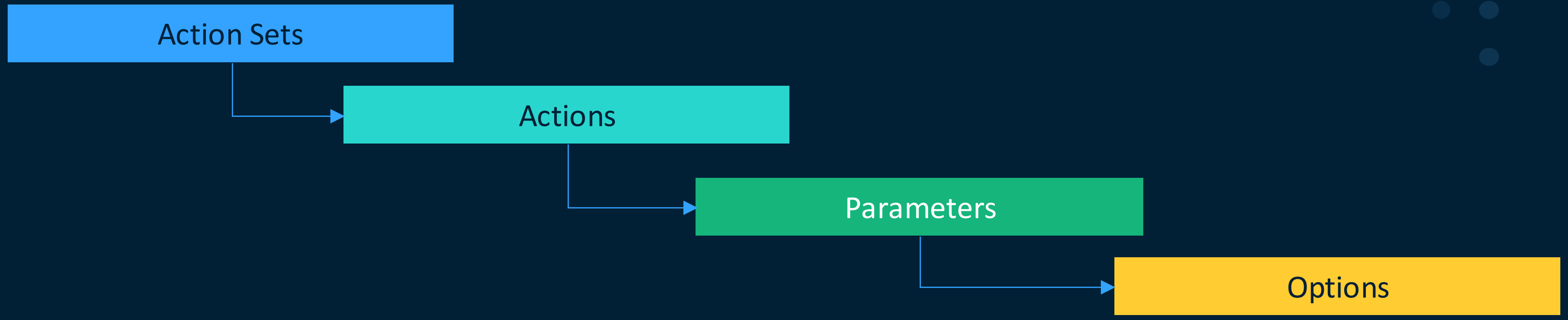


Getting Started with CASL Programming

Connect and Start a Session
About Your Connection and Server
Example: Working with CAS Actions and Results
Advanced Example: Train Gradient Boosted Trees with K-fold Cross Validation

Last updated: September 29, 2022

CAS Actions Hierarchies

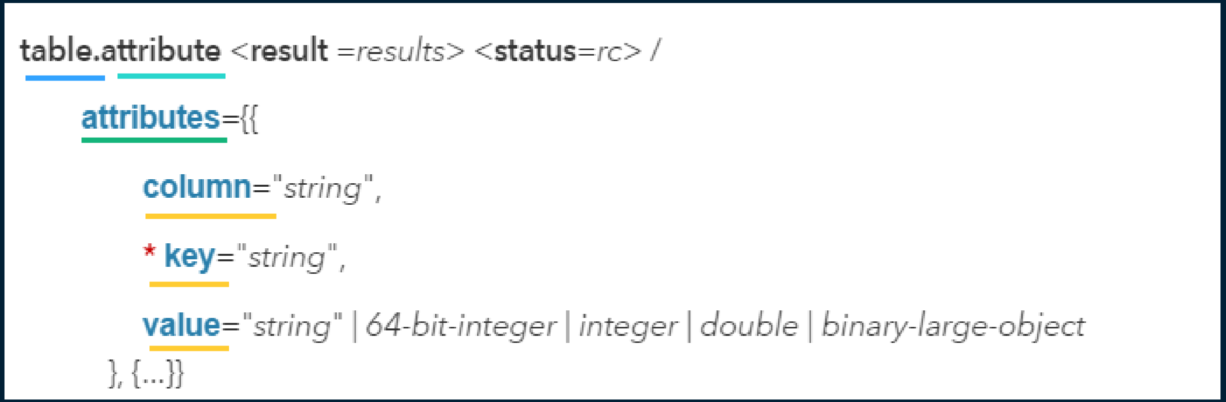


Action Sets

Actions

Parameters

Options



```
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

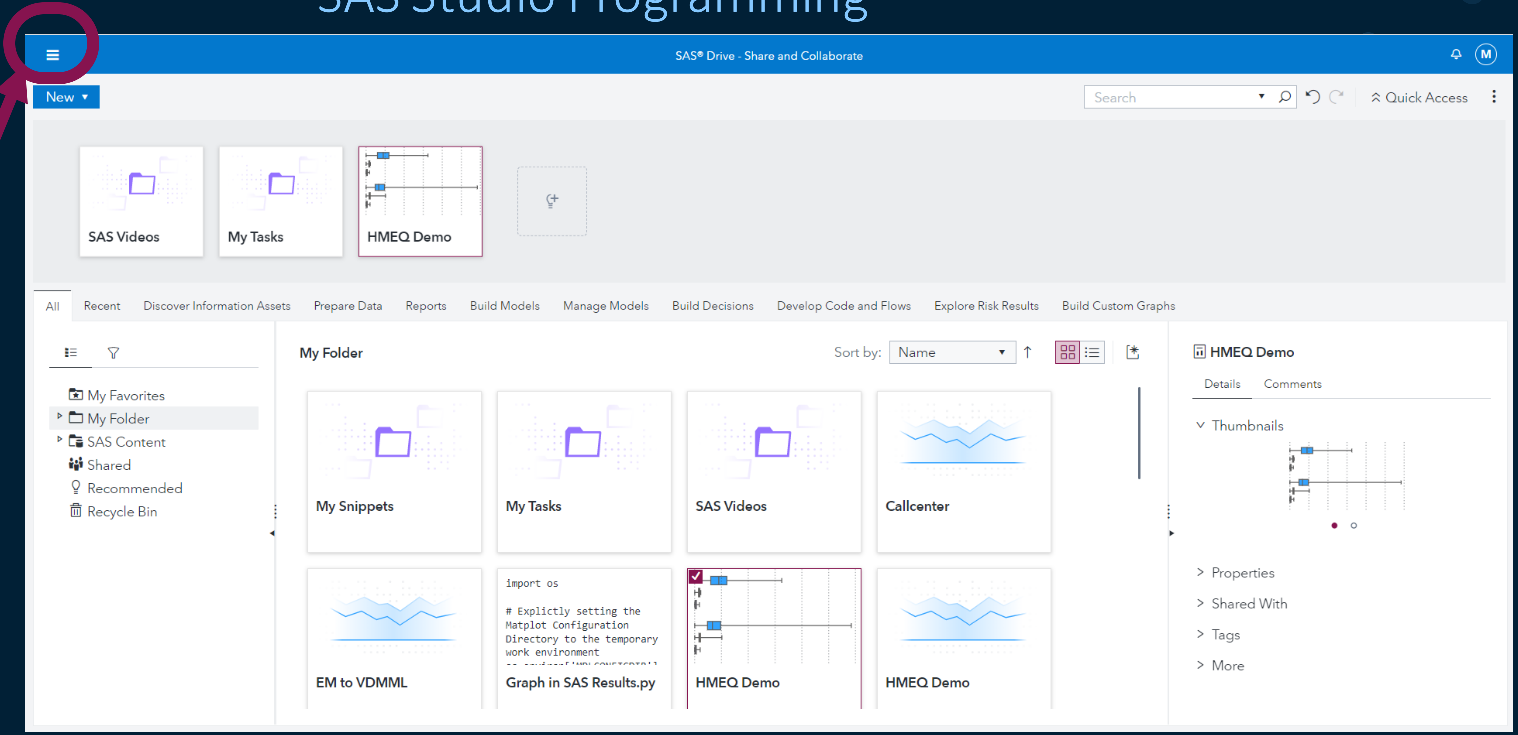
```
proc treesplit data=PUBLIC.HMEQ maxdepth=10;
         input LOAN MORTDUE VALUE YOJ DEROG NINQ CLNO DEBTINC / level=interval;
         input REASON JOB / level=nominal;
         target BAD / level=nominal;
         grow igr;
         prune none;
run;
```

```
proc cas;
         action decisionTree.dtreeTrain / table={name="HMEQ"}
         target="BAD" inputs={"LOAN", "MORTDUE", "VALUE", "YOJ",
             "DEROG", "NINQ", "CLNO", "DEBTINC", "REASON", "JOB"}
         nominals={"BAD", "REASON", "JOB"}
          maxLevel=11
         crit="GAINRATIO"
          varImp=TRUE
          missing="USEINSEARCH";
run;
```

SAS® Drive

SAS Studio Programming

Click on applications menu and select Develop Code and Flows



```
import os

# Explictly setting the
Matplot Configuration
Directory to the temporary
work environment
```

SAS Programming Interface

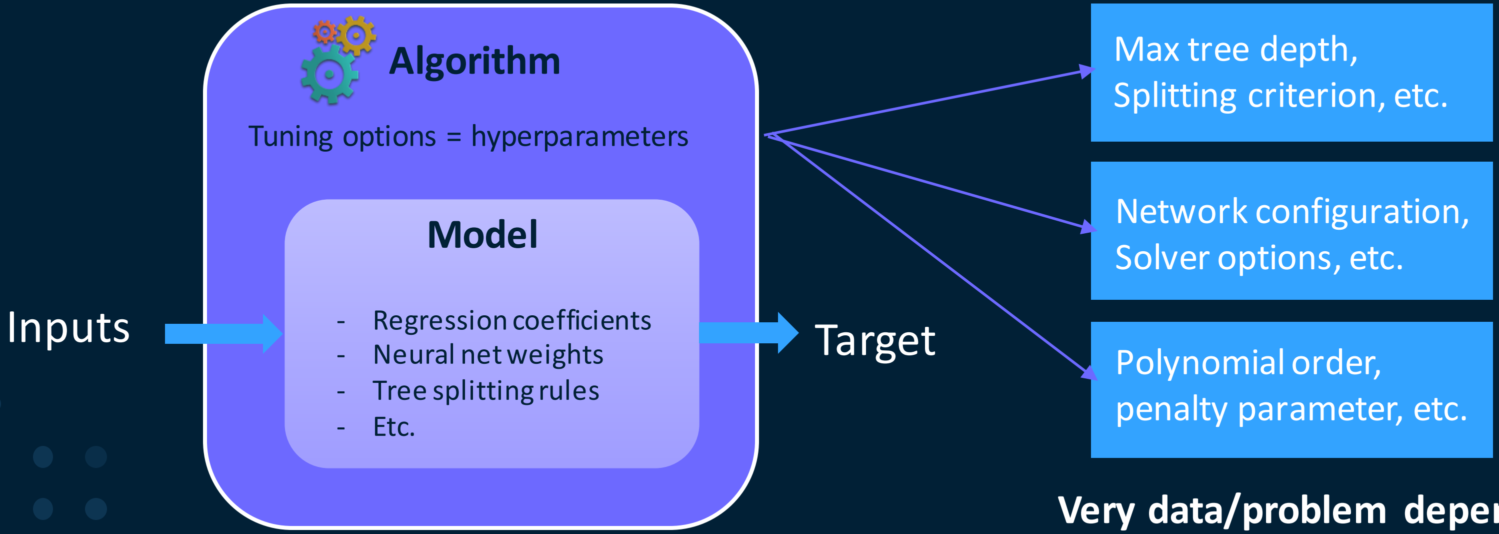
SAS Studio

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


Algorithm
Tuning options = hyperparameters
Model
- Regression coefficients
- Neural net weights
- Tree splitting rules
- Etc.
Inputs
Target
Max tree depth, Splitting criterion, etc.
Network configuration, Solver options, etc.
Polynomial order, penalty parameter, etc.

Very data/problem dependent

Autotuning

Methods

- SAS Visual Data Mining and Machine Learning offers:
  - Random search (highly parallelizable)
  - Latin Hypercube (highly parallelizable)
  - Genetic Algorithm - LH + proprietary SAS/OR algorithm (sequential in nature)
  - Bayesian – builds a kriging surrogate model, used in search process
  - Grid Search – uses all combinations

Autotuning Documentation

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

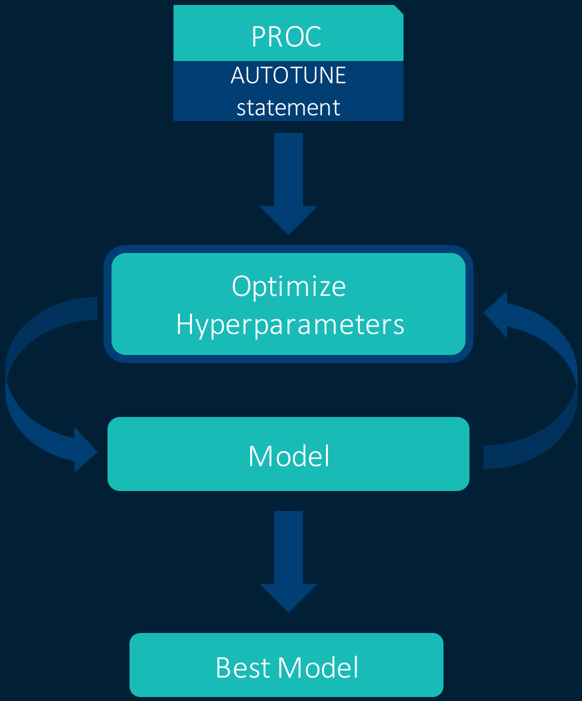
- Number of hidden layers
- Number of neurons in each hidden layer
- L1 regularization
- L2 regularization
- SGD options (annealing rate, learning rate)

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

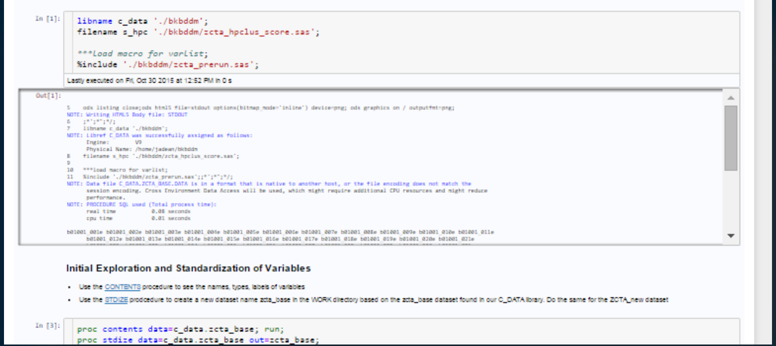
Open Source Interface

Jupyter Notebook

SAS® Scripting Wrapper for Analytics Transfer (SWAT)

Python and R

- Integration of SAS® Analytics in Python and R code
- R Studio and Jupyter Notebook support
- SWAT packages are available for Python and R free on GitHub or developer.sas.com.
- Download and install SWAT, connect to a CAS server, then write code to drive CAS actions.
- The SWAT package mimics much of the APIs of the native packages making it an easy addition for programmers familiar these languages.



In [1]:
```
libname c_data './bkbddm';
filename s_hpc './bkbddm/zcta_hpclus_score.sas';

***Load macro for varlist;
%include './bkbddm/zcta_prerun.sas';
```

Lastly executed on Fri, Oct 30 2015 at 12:52 PM in 0 s

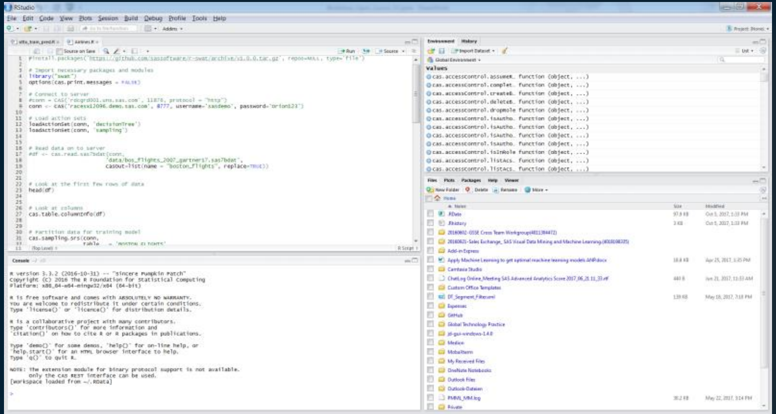
Out[1]:
```
5    ods listing close;ods html5 file=stdout options(bitmap_mode='inline') device=png; ods graphics on / outputfmt=png;
NOTE: Writing HTML5 Body file: STDOUT
6    ;*';*";*/;
7    libname c_data './bkbddm';
NOTE: Libref C_DATA was successfully assigned as follows:
      Engine:        V9
      Physical Name: /home/jadean/bkbddm
8    filename s_hpc './bkbddm/zcta_hpclus_score.sas';
9
10   ***load macro for varlist;
11   %include './bkbddm/zcta_prerun.sas';;*';*";*/;
NOTE: Data file C_DATA.ZCTA_BASE.DATA is in a format that is native to another host, or the file encoding does not match the
      session encoding. Cross Environment Data Access will be used, which might require additional CPU resources and might reduce
      performance.
NOTE: PROCEDURE SQL used (Total process time):
      real time           0.08 seconds
      cpu time            0.01 seconds

b01001_001e b01001_002e b01001_003e b01001_004e b01001_005e b01001_006e b01001_007e b01001_008e b01001_009e b01001_010e b01001_011e
      b01001_012e b01001_013e b01001_014e b01001_015e b01001_016e b01001_017e b01001_018e b01001_019e b01001_020e b01001_021e
```

Initial Exploration and Standardization of Variables

- Use the CONTENTS procedure to see the names, types, labels of variables
- Use the STDIZE prodcedure to create a new dataset name zcta_base in the WORK directory based on the zcta_base dataset found in our C_DATA library. Do the same for the ZCTA_new dataset

In [3]:
```
proc contents data=c_data.zcta_base; run;
proc stdize data=c_data.zcta_base out=zcta_base;
```



```
#install.packages('https://github.com/sassoftware/r-swat/archive/v1.0.0.tar.gz', repos=NULL, type='file')

# Import necessary packages and modules
library("swat")
options(cas.print.messages = FALSE)

# Connect to server
#conn = CAS('rdcgrd001.unx.sas.com', 11870, protocol = "http")
conn <- CAS('racesx12096.demo.sas.com', 8777, username='sasdemo', password='orion123')

# Load action sets
loadActionSet(conn, 'decisionTree')
loadActionSet(conn, 'sampling')


# Read data on to server
#df <- cas.read.sas7bdat(conn,
                        'data/bos_flights_2007_gartner17.sas7bdat',
                        casout=list(name = "boston_flights", replace=TRUE))


# Look at the first few rows of data
head(df)


# Look at columns
cas.table.columnInfo(df)


# Partition data for training model
cas.sampling.srs(conn,
                 table   = 'BOSTON_FLIGHTS'
```

```
R version 3.3.2 (2016-10-31) -- "Sincere Pumpkin Patch"
Copyright (C) 2016 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

NOTE: The extension module for binary protocol support is not available.
      Only the CAS REST interface can be used.
[Workspace loaded from ~/.RData]

>
```

Values
```
cas.accessControl.assumeR… function (object, ...)
cas.accessControl.complet… function (object, ...)
cas.accessControl.createB… function (object, ...)
cas.accessControl.deleteB… function (object, ...)
cas.accessControl.dropRole function (object, ...)
cas.accessControl.isAutho… function (object, ...)
cas.accessControl.isAutho… function (object, ...)
cas.accessControl.isAutho… function (object, ...)
cas.accessControl.isAutho… function (object, ...)
cas.accessControl.isInRole function (object, ...)
cas.accessControl.listAcs… function (object, ...)
cas.accessControl.listAcs… function (object, ...)
```

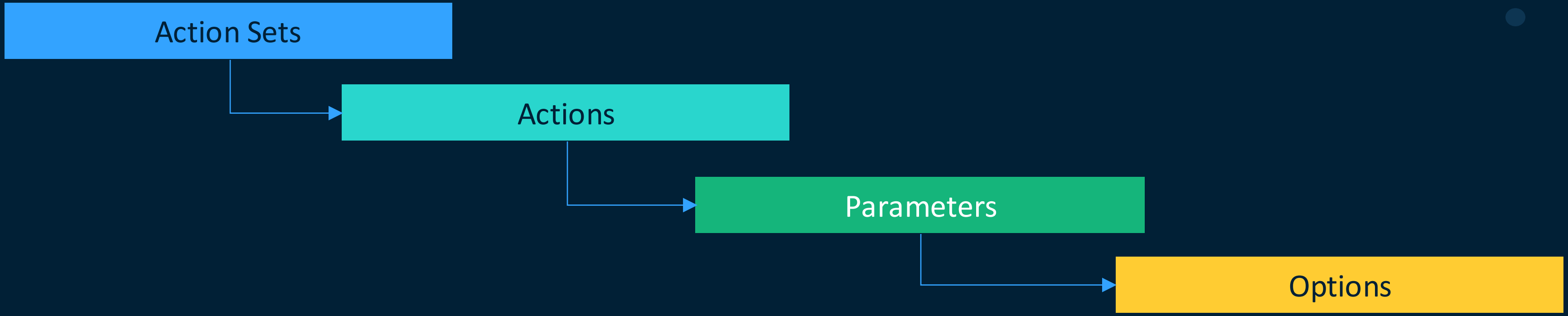
Name	Size	Modified
.RData	97.8 KB	Oct 5, 2017, 1:33 PM
.Rhistory	3 KB	Oct 5, 2017, 1:33 PM
20160802- GSSE Cross Team Workgroup(4811384472)		
20160621- Sales Exchange_ SAS Visual Data Mining and Machine Learning (4018198735)		
Add-in Express		
Apply Machine Learning to get optimal machine learning models ANP.docx	18.8 KB	Apr 25, 2017, 1:35 PM
Camtasia Studio		
ChatLog Online_Meeting SAS Advanced Analytics Score 2017_06_21 11_33.rtf	440 B	Jun 21, 2017, 11:33 AM
Custom Office Templates		
DF_Segment_Filter.xml	139 KB	May 18, 2017, 7:18 PM
Expenses		
GitHub		
Global Technology Practice		
jd-gui-windows-1.4.0		
Medion		
MobaXterm		
My Received Files		
OneNote Notebooks		
Outlook Files		
Outlook-Dateien		
PMML_MM.log	30.2 KB	May 22, 2017, 3:14 PM
Private		

Open to SAS®

Copyright © SAS Institute Inc. All rights reserved.

Programming Interfaces

CAS Actions Hierarchies



Action Sets

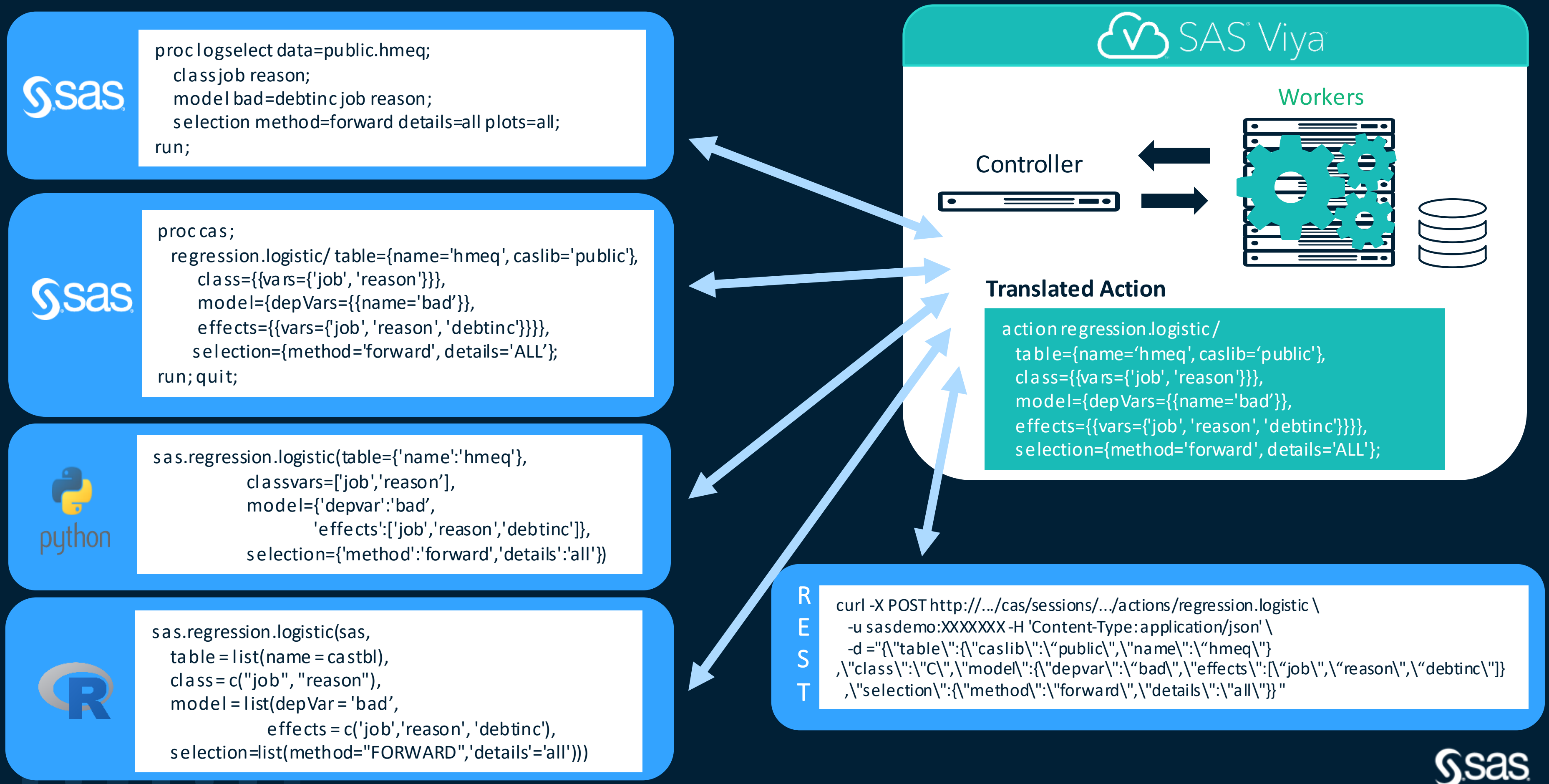
Actions

Parameters

Options

```
sas.datapreprocess.impute(
    table = dict(),
    inputs = value_list,
    methodContinuous = “median”,
    methodNominal=“mode”,
    casOut = dict()
    replace=TRUE)
)
```

How SAS Viya Single Code Source Works



```
proc logselect data=public.hmeq;
   class job reason;
   model bad=debtinc job reason;
   selection method=forward details=all plots=all;
run;
```

```
proc cas;
  regression.logistic/ table={name='hmeq', caslib='public'},
      class={{vars={'job', 'reason'}}},
      model={depVars={{name='bad’}},
      effects={{vars={'job', 'reason', 'debtinc'}}}},
     selection={method='forward', details='ALL’};
run; quit;
```

```
sas.regression.logistic(table={'name':'hmeq'},
              classvars=['job','reason’],
              model={'depvar':'bad’,
                        'effects':['job','reason','debtinc']},
              selection={'method':'forward','details':'all'})
```

```
sas.regression.logistic(sas,
   table = list(name = castbl),
   class = c("job", "reason"),
   model = list(depVar = 'bad’,
                  effects = c('job','reason', 'debtinc'),
   selection=list(method="FORWARD",'details'='all')))
```

SAS Viya

Translated Action

```
action regression.logistic/
  table={name=‘hmeq', caslib=‘public'},
  class={{vars={'job', 'reason'}}},
  model={depVars={{name='bad’}},
  effects={{vars={'job', 'reason', 'debtinc'}}}},
  selection={method='forward', details='ALL'};
```

REST

```
curl -X POST http://.../cas/sessions/.../actions/regression.logistic \
  -u sasdemo:XXXXXXX -H 'Content-Type: application/json' \
  -d ="{\"table\":{\"caslib\":\“public\",\"name\":\“hmeq\"}
,\"class\":\"C\",\"model\":{\"depvar\":\“bad\",\"effects\":[\“job\",\“reason\",\“debtinc\"]}
 ,\"selection\":{\"method\":\"forward\",\"details\":\"all\"}}"
```

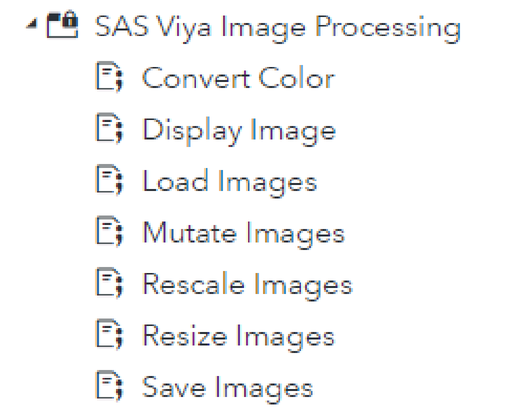
Open Source Interface Demo

Jupyter Notebooks

Other Exciting Features in SAS VDMML

Additional Analytical Algorithms and Options

- Tensor Factorization
- Neural Network Autoencoders
- Clustering mixed variables
- Deep Learning and Reinforcement Learning algorithms - Deep forward neural networks (DNNs), convolutional neural networks (CNNs) and recurrent neural networks (RNNs)
- Bayesian Network
- Market Basket Analysis
- Image Processing (CAS Actions)
  - 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



- SAS Viya Image Processing
  - Convert Color
  - Display Image
  - Load Images
  - Mutate Images
  - Rescale Images
  - Resize Images
  - Save Images

Resources

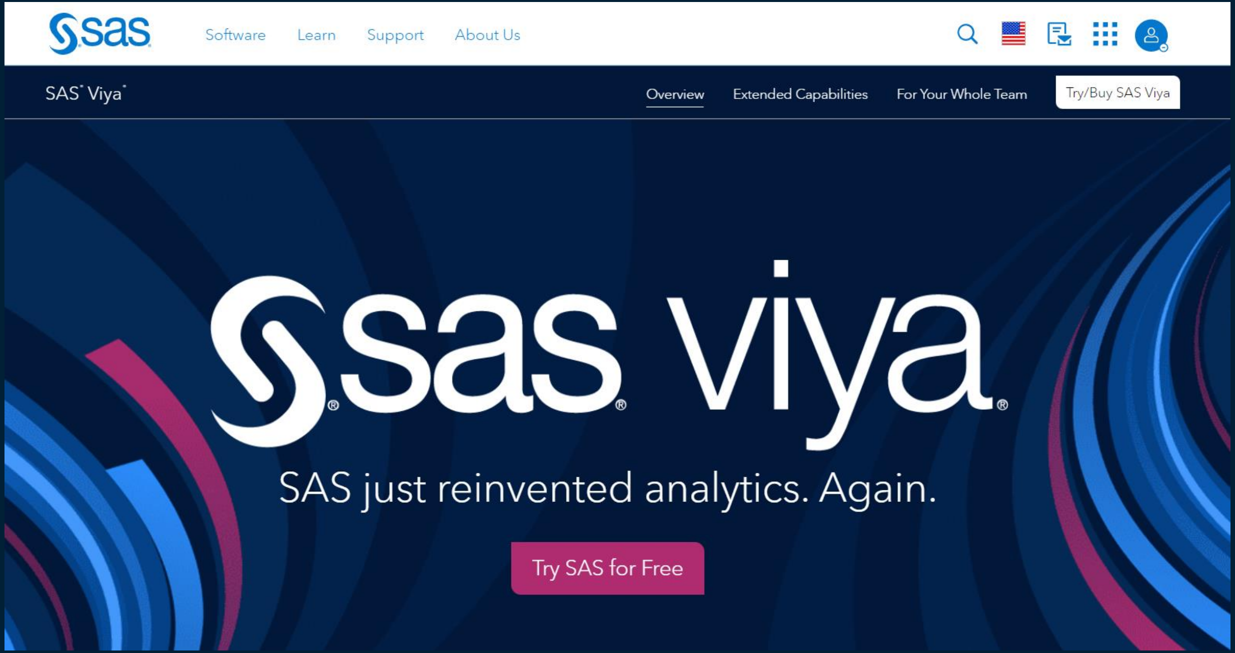
Where to learn more

SAS Viya

Resources

Overview, Training, Samples and Tips

- SAS Viya Overview
- SAS Viya Training
- A Beginner's Guide to Programming in the SAS® Cloud Analytics Services Environment



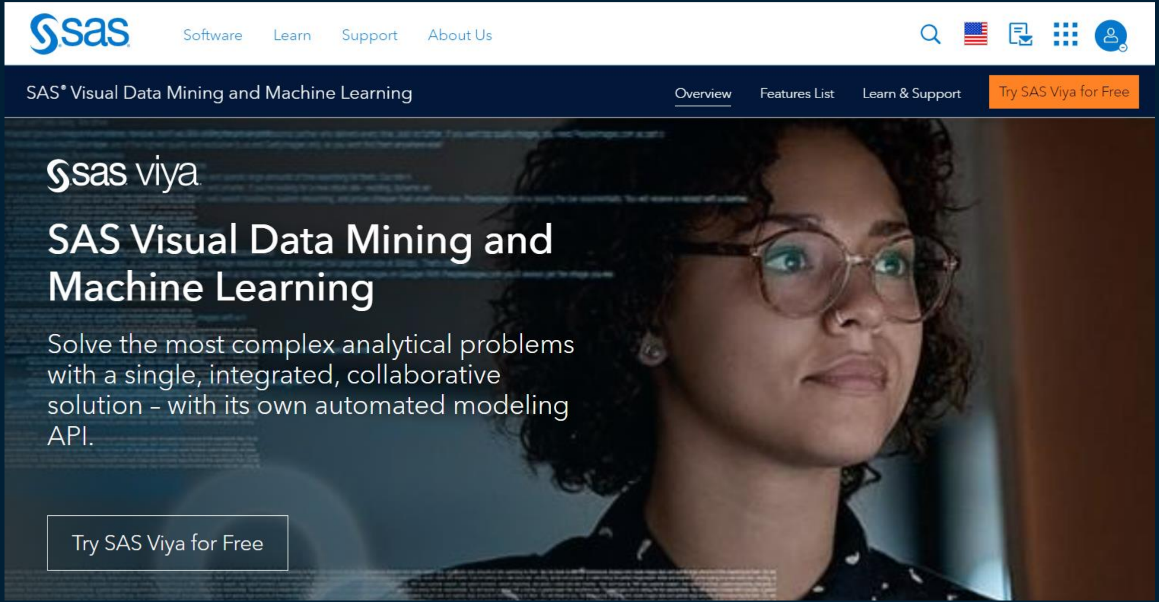
SAS® Visual Data Mining and Machine Learning

Where to Learn More

[Documentation]()

[Getting Started Learning Resources]()

[Video Tutorials]()



SAS Visual Data Mining and Machine Learning

Solve the most complex analytical problems with a single, integrated, collaborative solution – with its own automated modeling API.

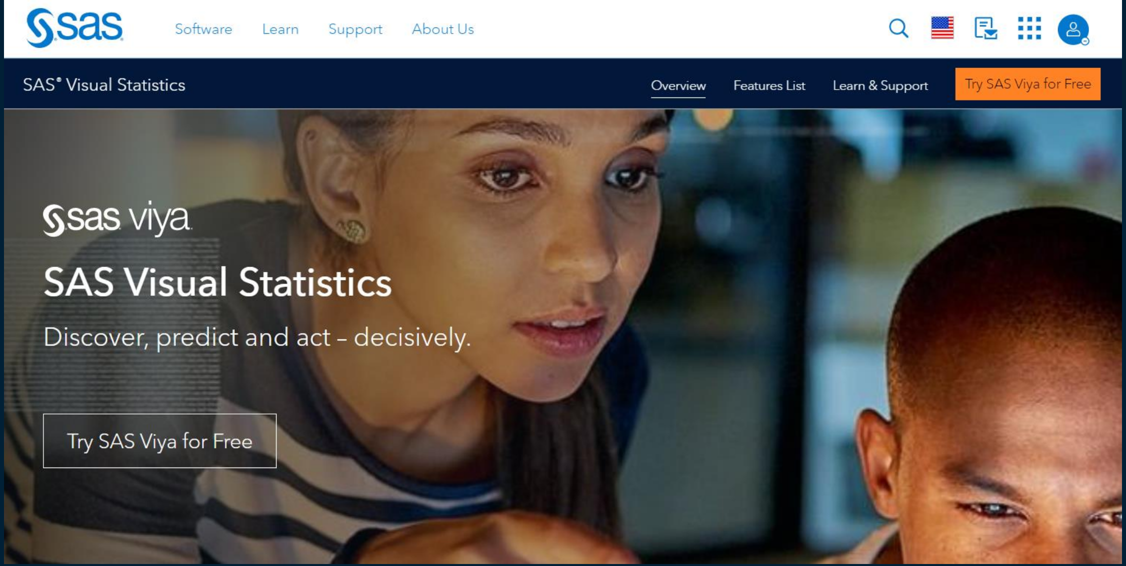
SAS® Visual Statistics

Where to Learn More

[Documentation]

[Getting Started Learning Resources]

[Video Tutorials]



Resources

Autotuning

- Local search optimization for hyperparameter tuning
- Optimization for machine learning and monster trucks
- The OPTLSO procedure
- SAS communities discussion – check out some of the current problems with auto-tuning, long run-times and SAS Studio locking you out etc.

DLPy Resources

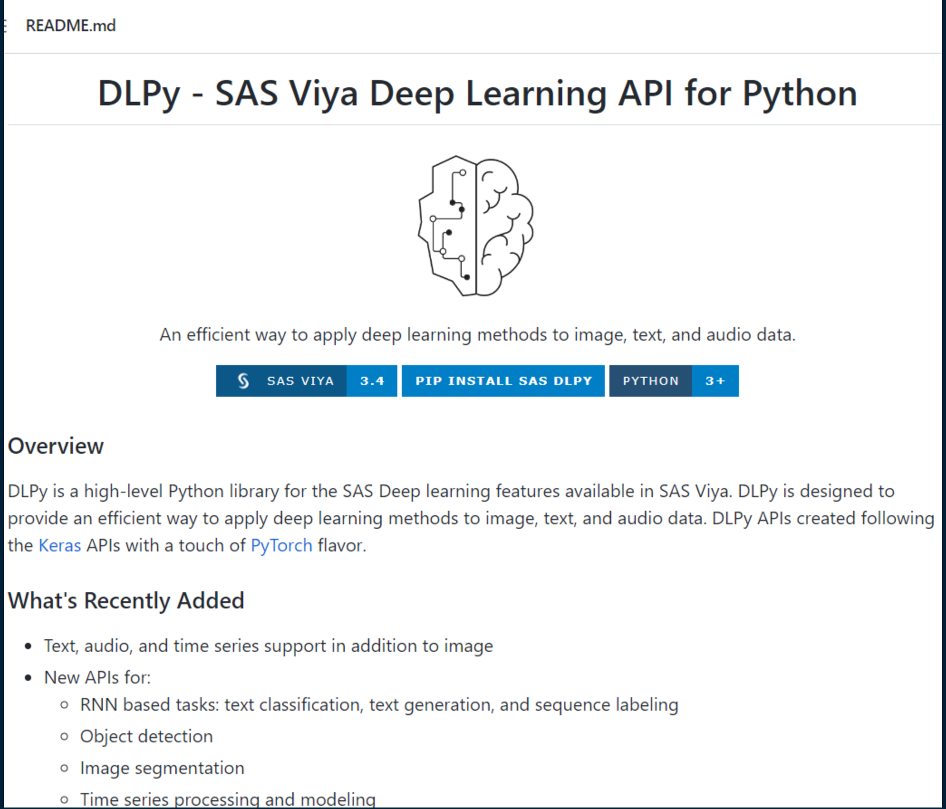
[DLPy Documentation](#)

[DLPy Github](#)

[DLPy Blog](#)

Introduction to DLPy and examples (YouTube):

- Introduction Deep Learning with Python (DLPy) and SAS Viya
- Image classification using CNNs
- Object detection using TinyYOLOv2
- Import and export deep learning models with ONNX
- Text classification and text generation using RNNs
- Time series forecasting using RNNs



README.md

DLPy - SAS Viya Deep Learning API for Python

An efficient way to apply deep learning methods to image, text, and audio data.

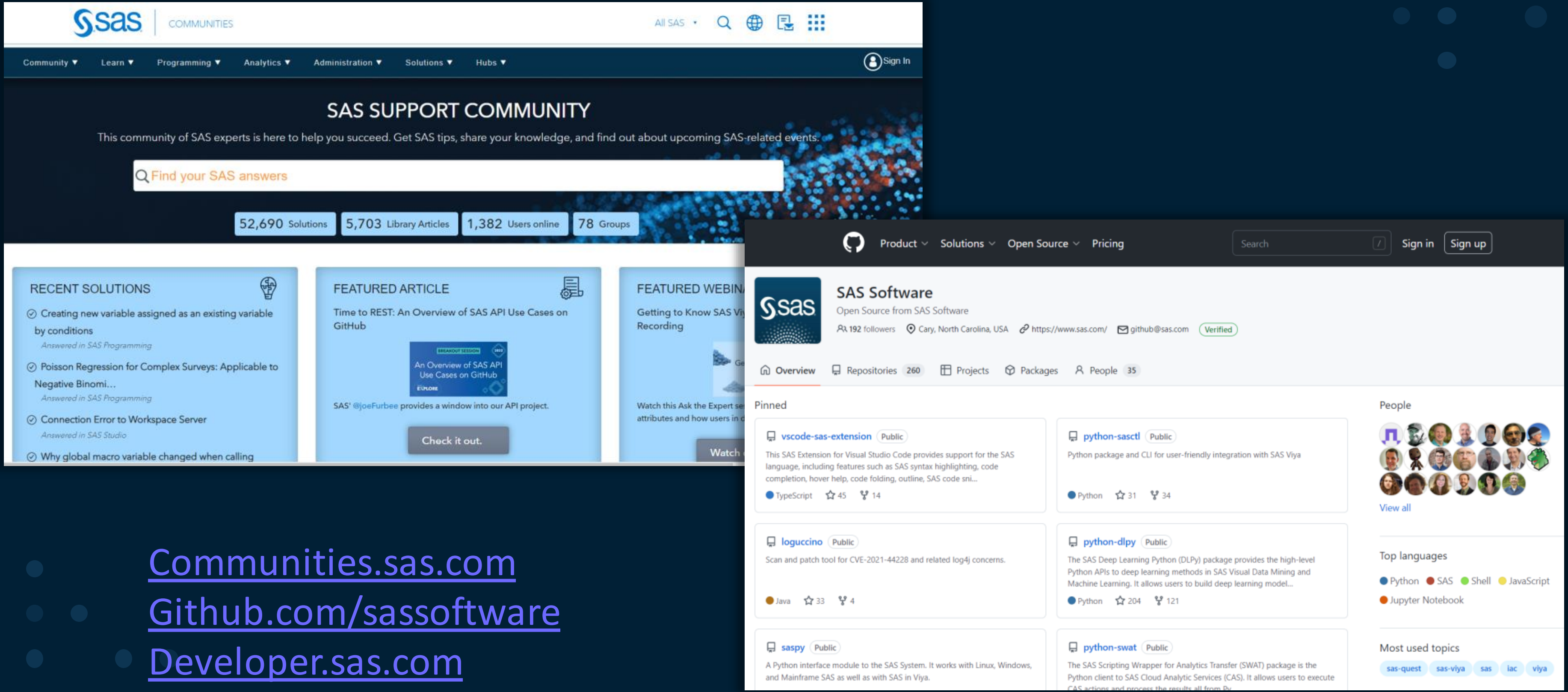
Overview

DLPy is a high-level Python library for the SAS Deep learning features available in SAS Viya. DLPy is designed to provide an efficient way to apply deep learning methods to image, text, and audio data. DLPy APIs created following the Keras APIs with a touch of PyTorch flavor.

What's Recently Added

- Text, audio, and time series support in addition to image
- New APIs for:
  - RNN based tasks: text classification, text generation, and sequence labeling
  - Object detection
  - Image segmentation
  - Time series processing and modeling

Communities



SAS SUPPORT COMMUNITY

This community of SAS experts is here to help you succeed. Get SAS tips, share your knowledge, and find out about upcoming SAS-related events.

Find your SAS answers

52,690 Solutions | 5,703 Library Articles | 1,382 Users online | 78 Groups

RECENT SOLUTIONS

- Creating new variable assigned as an existing variable by conditions
  Answered in SAS Programming
- Poisson Regression for Complex Surveys: Applicable to Negative Binomi...
  Answered in SAS Programming
- Connection Error to Workspace Server
  Answered in SAS Studio
- Why global macro variable changed when calling

FEATURED ARTICLE

Time to REST: An Overview of SAS API Use Cases on GitHub

SAS' @joeFurbee provides a window into our API project.

FEATURED WEBIN

Getting to Know SAS Vi
Recording

Watch this Ask the Expert se
attributes and how users in c

SAS Software

Open Source from SAS Software

192 followers · Cary, North Carolina, USA · https://www.sas.com/ · github@sas.com · Verified

Pinned

vscode-sas-extension Public
This SAS Extension for Visual Studio Code provides support for the SAS language, including features such as SAS syntax highlighting, code completion, hover help, code folding, outline, SAS code sni...
TypeScript ☆ 45 Forks 14

python-sasctl Public
Python package and CLI for user-friendly integration with SAS Viya
Python ☆ 31 Forks 34

loguccino Public
Scan and patch tool for CVE-2021-44228 and related log4j concerns.
Java ☆ 33 Forks 4

python-dlpy Public
The SAS Deep Learning Python (DLPy) package provides the high-level Python APIs to deep learning methods in SAS Visual Data Mining and Machine Learning. It allows users to build deep learning model...
Python ☆ 204 Forks 121

saspy Public
A Python interface module to the SAS System. It works with Linux, Windows, and Mainframe SAS as well as with SAS in Viya.

python-swat Public
The SAS Scripting Wrapper for Analytics Transfer (SWAT) package is the Python client to SAS Cloud Analytic Services (CAS). It allows users to execute CAS actions and process the results all from Py

People

View all

Top languages

● Python ● SAS ● Shell ● JavaScript ● Jupyter Notebook

Most used topics

sas-quest · sas-viya · sas · iac · viya

Communities.sas.com

Github.com/sassoftware

Developer.sas.com

Questions?
Thank you for your time and attention!

Connect with me:
LinkedIn: https://www.linkedin.com/in/melodierush

sas.com