Webinar:
Open Source Follow-Up and QA

26/8-2020 – in Teams

NORDIC SAS USER GROUP

8/26-2020 – in Teams

Webinar:

Open Source Follow-Up and Q&A

Agenda

- Introduction
- Follow-up and summary of presentation
- Q&A

NORDIC SAS USER GROUP

Whom am I

Host:

Frans Holm

Presenter

Daniel Ringqvist

- Responsible for FANS in Denmark
- Working +15 years in SAS

- Responsible for FANS in Sweden
- + 25 years experience of SAS

fans NORDIC SAS USER GROUP

Open Source Follow-Up 26aug2020

Summary slides

Daniel.ringqvist@sas.com

NORDIC SAS USER GROUP

Summary - Open Source in Viya

- Why Open Source code, what are we trying to solve?
  - Data Management
  - Analytics
  - Results (plots, lists, …)
- We learned
  - what we CAN do (and some what we can not)
  - How we set things up on server, for this to work
- Great documentation in the end of the slide deck

Summary - Open Source in Viya

- What CAN we do
  - Code nodes in Pipelines for R and Py
    - Data prep, analytics and results
  - SWAT in Jupyter Notebooks for R and Py
  - Model Manager
  - REST APIs
  - Proc FCMP to build functions running Py
  - Base SAS Java Object
  - Calling R from SAS/IML (SAS9 and possibly Viya)

fans NORDIC SAS USER GROUP

sas

Model Studio Pipelines

Open Source Code Node

Open Source Code node

CAN

- Support execution of Python/R code
  - Downloads data sample from Cloud Analytic Services (CAS)
- Display results from Python/R code execution
- Produce assessment statistics of Python/R models
- Enable comparison of Python/R models within Model Studio pipeline

Open Source Code node

CAN NOT

Be part of an Ensemble

Support Register, Publish or Download score code or score API

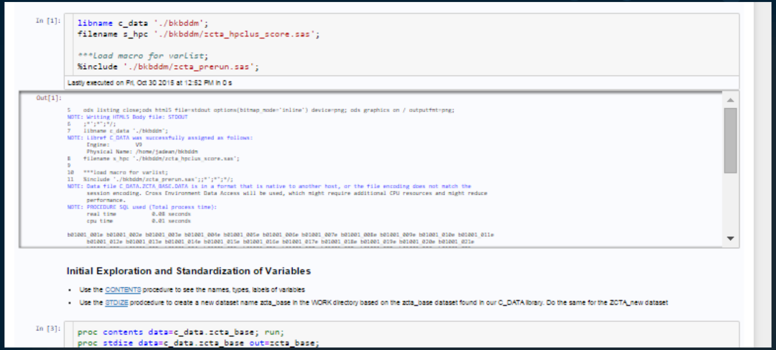
SWAT

For R and Python

SAS® Scripting Wrapper for Analytics Transfer

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';
```

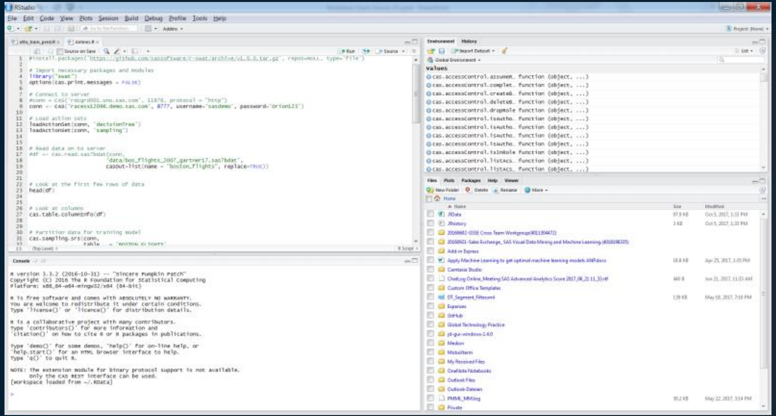
Lastly executed on Fri, Oct 30 2015 at 12:52 PM in 0 s

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;
```

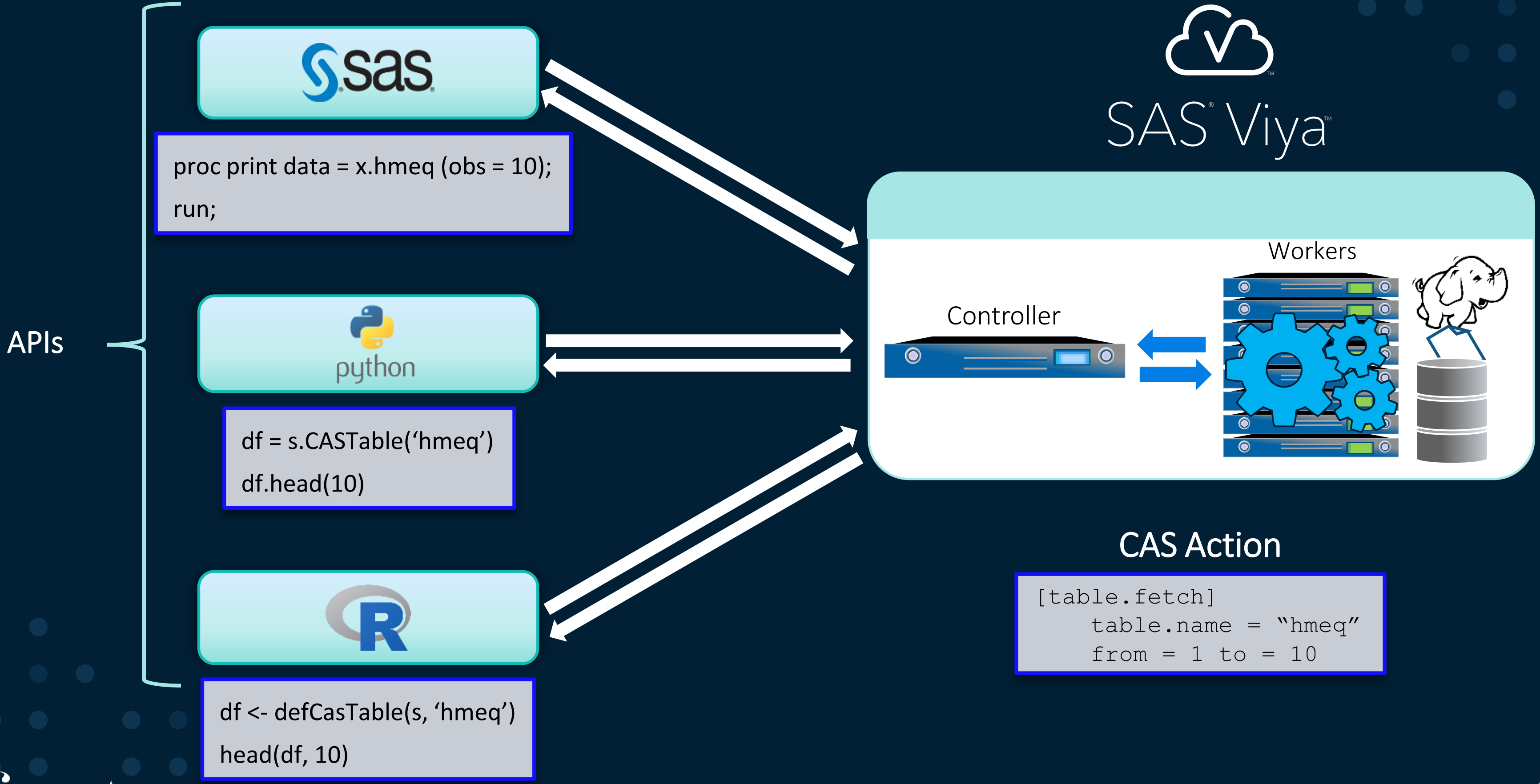

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Addins
Source on Save
Run
Source
Environment History
Import Dataset
Global Environment
Files Plots Packages Help Viewer
New Folder Delete Rename More
Home
Console
Project: (None)

```
#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', 11878, 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	97.9 KB	Oct 5, 2017, 1:33 PM
.Rhistory	3 KB	Oct 5, 2017, 1:33 PM
20160602-GSSE Cross Team Workgroup(401304472)		
20160921-Sales Exchange_SAS Visual Data Mining and Machine Learning (403038335)		
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, 11:13 AM
Custom Office Templates		
DT_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		



```
proc print data = x.hmeq (obs = 10);
run;
```

```
df = s.CASTable(‘hmeq’)
df.head(10)
```

```
df <- defCasTable(s, ‘hmeq’)
head(df, 10)
```

CAS Action

```
[table.fetch]
    table.name = “hmeq”
    from = 1 to = 10
```

SWAT

CAN

- Support execution of Python or R code
  - Connects to SAS CAS to run SAS Actions

- Can be used with IDE's such as Jupyter Notebooks and R Studio

- Python/R runs where IDE's are configured either locally or on compute server

- Mix SAS programming with Open Source (Python or R)

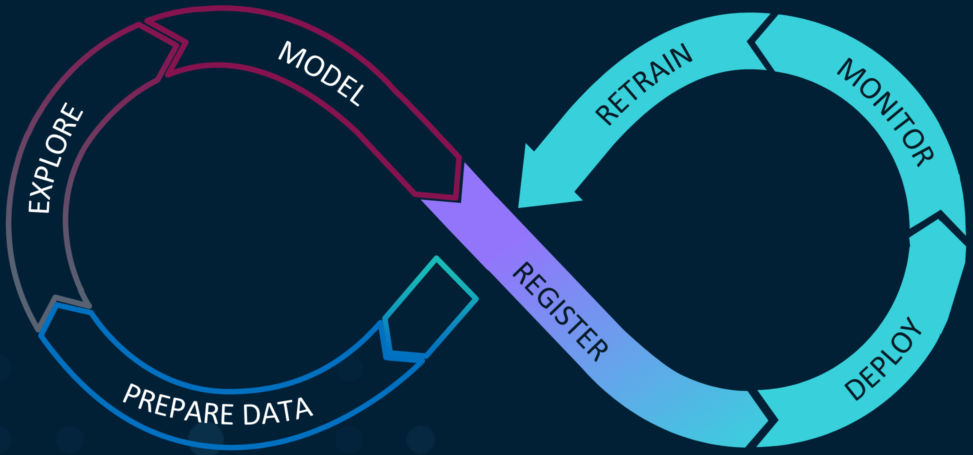
SAS Model Manager

And SAS Open Model Manager

fans
NORDIC SAS USER GROUP

sas

SAS Model Manager



- Register
- Validate
- Deploy
- Monitor

SAS Model Manager

CAN

- Supports the registration, validation, deployment and monitoring of Python and R models
- Available using both the point and click in the visual interface and through programming using packages sasctl and pzmm

Other Integration

REST APIs, PROC FCMP, SAS JAVA Object

REST APIs

REST APIs

What is a REST API?

An API is the messenger that takes a request, tells a system what you want to do and then returns the response back to you.

- A **RESTful API** is an application program interface (**API**) that uses HTTP requests to GET, PUT, POST and DELETE data. An **API** for a website is code that allows two software programs to communicate with each other.
- *“**REST** stands for **RE**presentational **S**tate **T**ransfer”*
- *“**A**PI means **A**pplication **P**rogramming **I**nterface”*

SAS Viya: Working with APIs – Ask the Expert

REST APIs

Two entry points into SAS Viya

APIs for application developers and admins

- designed for enterprise application developers
- intend to build on the work of model builders and data scientists, to deliver apps based on SAS Viya technology

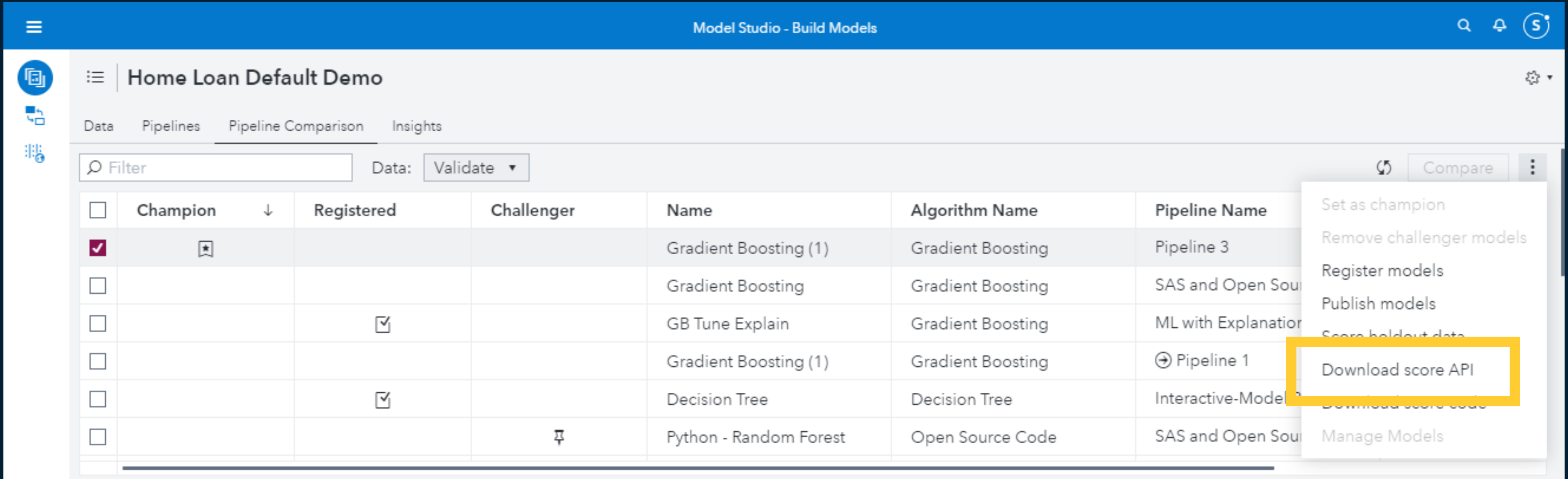
APIs for analysts and data scientist

- Designed for data scientist, programmers and administrators who need to interact with CAS directly
- Used to executing CAS actions, managing CAS sessions, monitoring the system and inspecting the CAS grid

developer.SAS.com

REST APIs

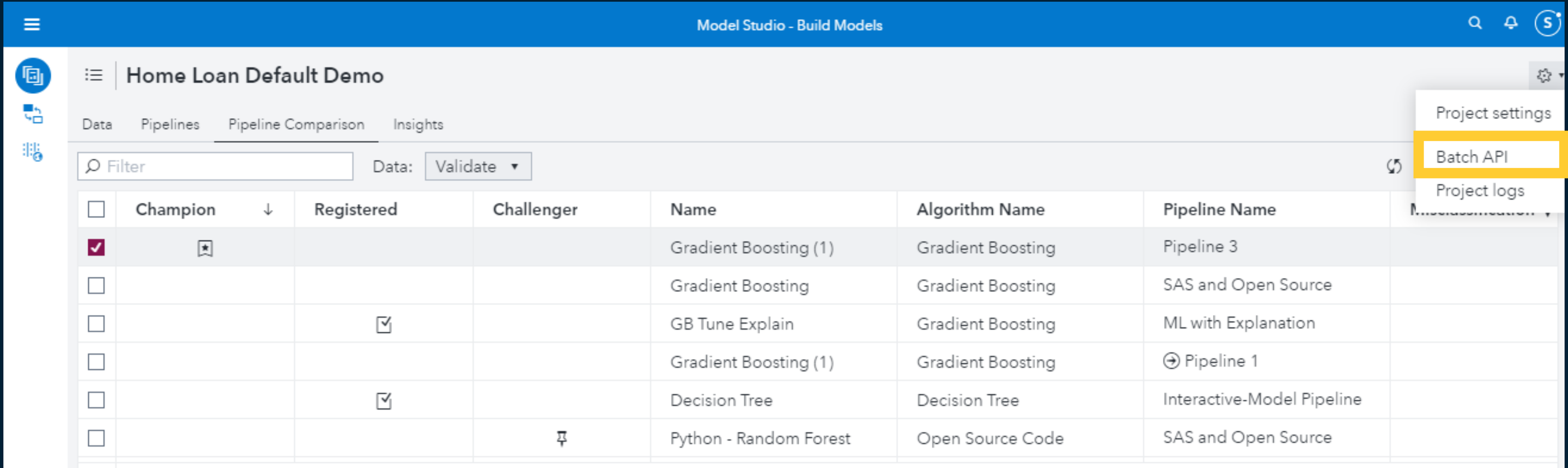
Scoring API



□	Champion ↓	Registered	Challenger	Name	Algorithm Name	Pipeline Name
☑	★			Gradient Boosting (1)	Gradient Boosting	Pipeline 3
□				Gradient Boosting	Gradient Boosting	SAS and Open Sou
□		☑		GB Tune Explain	Gradient Boosting	ML with Explanatio
□				Gradient Boosting (1)	Gradient Boosting	➔ Pipeline 1
□		☑		Decision Tree	Decision Tree	Interactive-Model
□			📌	Python - Random Forest	Open Source Code	SAS and Open Sou

REST APIs

APIs for analysts and data scientist



Home Loan Default Demo

Data: Validate

□	Champion ↓	Registered	Challenger	Name	Algorithm Name	Pipeline Name
☑	★			Gradient Boosting (1)	Gradient Boosting	Pipeline 3
□				Gradient Boosting	Gradient Boosting	SAS and Open Source
□		☑		GB Tune Explain	Gradient Boosting	ML with Explanation
□				Gradient Boosting (1)	Gradient Boosting	➔ Pipeline 1
□		☑		Decision Tree	Decision Tree	Interactive-Model Pipeline
□			📌	Python - Random Forest	Open Source Code	SAS and Open Source

PROC FCMP

PROC FCMP

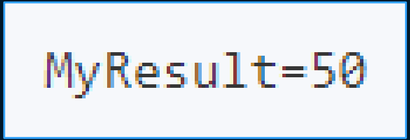
Using Python Functions in 5 Steps

Python Function Workflow

1. Declare a Python object & a dictionary object
2. Insert Python source code into SAS
3. Publish Python source code
4. Call the Python source code
5. Return results from the dictionary

```
proc fcmp;
declare object py(python);
submit into py;
def PyProduct(var1, var2):
   "Output: MyKey"
   newvar = var1 * var2
   return newvar,
endsubmit;
rc = py.publish();
rc = py.call("PyProduct", 5, 10);
MyResult =py.results["MyKey"];
put MyResult=;
```

Results



```
MyResult=50
```

Base SAS Java Object

Base SAS Java Object

Executes a Python or R file

```
/* Execute Python script */
data _null_;
        length rtn_val 8;
        declare javaobj
              j("com.sas.analytics.datamining.servertier.SASPythonExec",
                 "&dm_nodedir&dm_dsep.dm_srcfile.py");
         j.callVoidMethod("setOutputFile",
             "&dm_nodedir&dm_dsep&lang._output.txt");
         j.callIntMethod("executeProcess", rtn_val);
         j.delete();
         call symput('javaobj_rtnval', rtn_val);
run;
```

Calling R from SAS/IML

Comparison of matrix operations in IML and R

```
proc iml;
x = 1:3;                                    /* vector of sequence 1,2,3 */
m = {1 2 3, 4 5 6, 7 8 9};                  /* 3 x 3 matrix */
q = m * t(x);                          /* matrix multiplication */
print q;


submit / R;
  rx <- matrix( 1:3, nrow=1)                # vector of sequence 1,2,3
  rm <- matrix( 1:9, nrow=3, byrow=TRUE)    # 3 x 3 matrix
  rq <- rm %*% t(rx)                        # matrix multiplication
  print(rq)
endsubmit;
```

Q & A

fans
NORDIC SAS USER GROUP

sas

FANS

Webinars

Program 2020

sas.com/fans -> Events -> Webinars

- 18/9 Webinar – Migration to Viya
- 1/10 Webinar – Visual Analytics for Viya
- 11/11 Webinar – Enterprise Guide with Viya

fans NORDIC SAS USER GROUP

Thank you!

Frans.Holm@sas.com

Daniel.Ringqvist@sas.com