SVILCTENS

Quick guide to using Python preprocessing in SIMCA®-online

This document describes how to practically add a script to SIMCA® and use it in SIMCA®-online.

General information

Since version 16 it is possible for SIMCA[®] to use and embed Python scripts for preprocessing of data. This lets you morph the dataset for both modelling and predictions (including SIMCA[®]-Q and SIMCA[®]-online) row by row.

Note: Applying a Python script might slow down performance of the SIMCA®-Q and SIMCA®-online systems.

Creating a Python preprocessing script

This section describes how to create a Python preprocessing script. The actual Python code is left for you to implement.

1. Add the Development tab to SIMCA® (if it is not there already): In **File | Options**, in the Customize ribbon section, select to add the Development tab and click **OK**



SVILCIEVS

2. Create a new Spectral filter with Python by clicking **Create new script** in the Scripts group on the Developer tab, and then select **Spectral filter**



3. Give the script a name, and optionally also a prefix (that all new variables in the new dataset will be prefixed with).

Create filter		×
Filter name:		
MyPlugin		
Variable prefix (optional):		
PyPI		
	Restart SIMCA to load t	the new plugins.
	ОК	Cancel

A template is created and automatically opened. This is where you add your Python code



The example function (transform) just takes the square of all values, but this is where you need to enter your own transformation/preprocessing function. The matrix returned must always have the same number of rows as the input data collection but other than that anything is possible.

SVILCIEVS

Applying a preprocessing script

This section describes how a spectral filter and Python preprocessing script can be used in SIMCA[®]. Note that SIMCA[®] needs to be restarted to load any new Python add-ins from the add-in folder, and that all preprocessing scripts that you want to use need to be in the add-in folder of SIMCA[®].

1. On the Data tab, click **Preprocessing | Preprocessing...**

•	S ず 🔜 🐏 🦈 Bøkers Veast Elbrus - coded batch names.usp - SIMCA																
	File	Home	Data Ar	alyze Predict	t View	Tool	ls De	veloper	Add-Ins								
				×	ţ.	÷	ĸ	-	×	*		Ţ		Σ			
	Fror	n From Paste database data	Import to model	View Delete	Merge	Split Ti	ranspose	Generate variables	Trim-Winz	Local centering *	Pre	processing	Spectra	Dataset summary			
		Get external da	ita	Existing dataset	5	1	Modify da	taset		- 1	Sp	pectral filter	s		L		
	F [S Bakers Yeast	Elbrus - cod	ed batch names [M7]						E,	Preproce	ssing				
	n i	Number	Model	Type			A		N	R2X(cun	Ŧ	Wavelet of	compressi	on spectral	Datr	e Ti	de
	ň	Batch model 1									Ti	me series fi	lters		2011-08-0	18	
		Batch evolu	ition models	(BEM)							Ŧ	Wavelet of	:ompressi	on			
		Batch level	models (BLN	1)							Ŧ	Wavelet of	denoising	/decimation			
		4	M4	PCA-X			1	5	20	0,9	05			0,012	2011-08-0	18 P	1
		5	M5	PCA-X				3	20	0,	84			0,746	2011-08-0	18 P	2
		6	M6	PCA-X			1	3	20	0,8	16			0,712	2011-08-0	18 P	3
		L. 7	M7	PLS				3	20	0,	57		0,915	0,643	2011-08-0	8	

2. Select the Source dataset to apply the filtering to and then go through the wizard and select the Python script (or any of the built-in filters) that you want to use and click **Next** to continue to the Summary page.

Note that several filters can be applied in a chain

# Preprocessing	— D	×
Select data 📏 Variables 📏 Observations 📏 Filters 📏	Summary	
Select filters to apply		
Smoothing ()		
Savitzky-Golay EWMA WDS Moving AsLS novel smoothing smoothing		
Baseline correction ③		
Row-center Offset Linear AsLS correction		
Normalization ()		
Image: SNV Image: Arrow Peak Area		
Other		
MSC Derivatives MyPlugin Jon_special		
1. MyPlugin.MyPlugin ×		
< Back Next >	Can	ncel

3. Give the new (filtered) dataset a name and click Finish

Preprocessing			×
Select data 🗲 Variables 🗲 Observations 🗲 Filters 🗲	Summa	ry	
Dataset name: BEM48C data_Plusin			
MyPlugin.MyPlugin			^
Source datasets: BEM+BC data			
Included variables: Ethanol - pH			
MyPlugin.MyPlugin from filterabc import FilterABC			
#Default editor can be changed in File Options Python Editor Command #For example notepad++ to get better syntax highlight			
#see FilterABC for more optional functions			
class MyPlugin(FilterABC) :			
def_init_(self): #The name of the transform.			~
<back next=""> Fir</back>	nish	Can	icel

SVISCIEVS

Viewing an existing embedded Python preprocessing script

This section describes how to view the embedded Python script for a specific dataset in SIMCA®.

To view the existing Python preprocessing script that is already embedded in a dataset:

1. Open the dataset Properties:



2. View the Preprocessing tab:

Properties								×		
Variables	Observations	General	Phases	Batches	Preprocessing	Missing values	Trimming			
MyPlugin.MyPlugin Source datasets: BEM+BC data Included observations: 1 - 1660										
Include	Included variables: Ethanol - pH									
MyPlu	gin.MyPlugin									
from fi	from filterabc import FilterABC									
#Default editor can be changed in File Options Python Editor Command #For example notepad++ to get better syntax highlight										
#see F	#see FilterABC for more optional functions									
class N	lyPlugin(Filter/	ABC) :					~	1		
			Oł	<	Cancel	<u>A</u> pply	Help			

SVILCIEVS

Using a SIMCA project with an embedded Python preprocessing script in SIMCA®-online

This section describes using a SIMCA® project with an embedded Python script in SIMCA®-online.

Note that since Python scripts can potentially contain dangerous code, any project with existing Python scripts needs to be approved before it can be used in the products.

1. Upload SIMCA[®] project opens a warning-message:



To see the Python script content, click **View scripts**. After reviewing the script, provided you are content, click **Upload anyway** to move forward

- 2. Step through the project configuration as with any other project.
- 3. Display your new/filtered variables in the charts