“What’s new” - Coding Examples in SAS Viya – Da geht noch viel mehr!

Gerhard Svolba

Analytic Solutions Architect

SAS DACH

Twitter: gsvolba

https://github.com/gerhard1050

https://www.linkedin.com/in/gerhardsvolba/

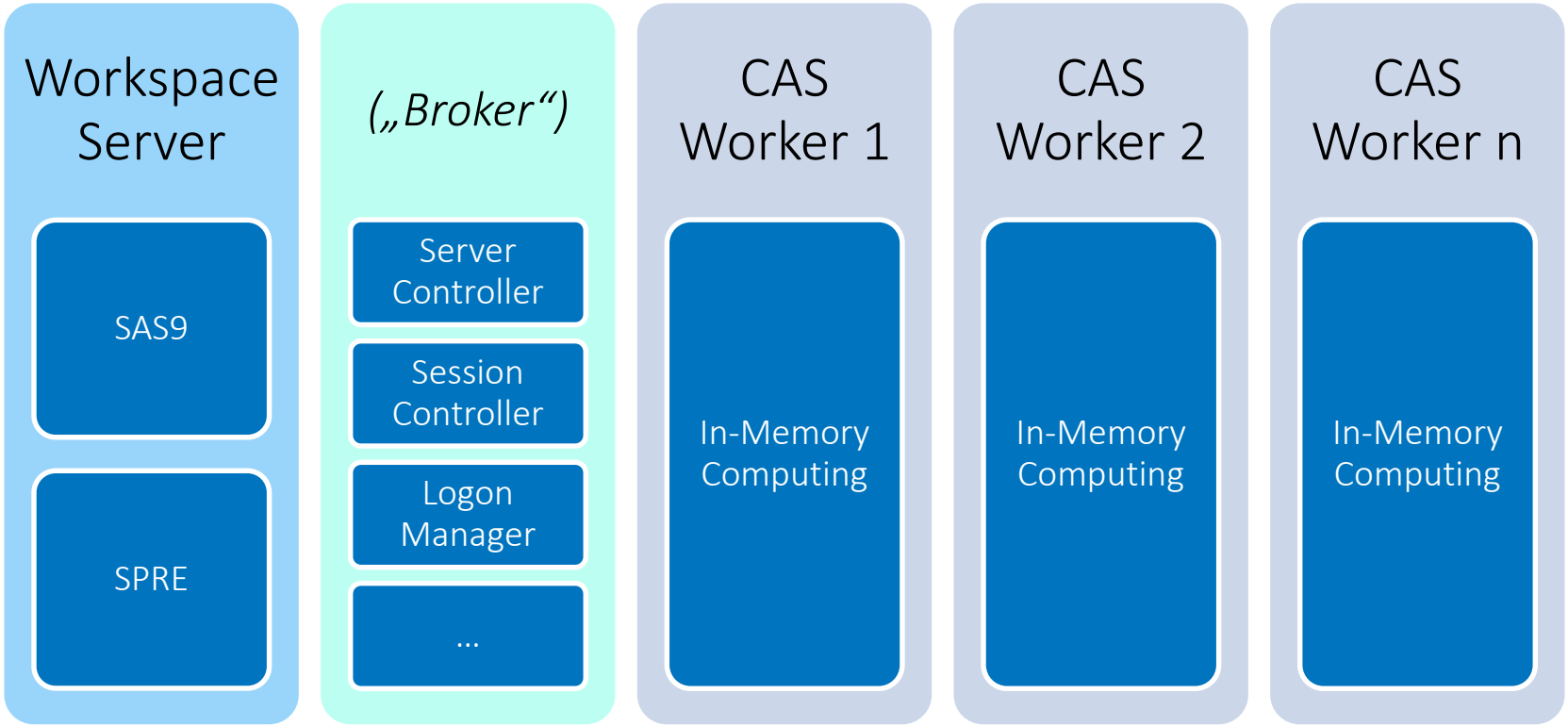
THE POWER TO KNOW.

- Tips and Techniques for Running CAS Actions on Youtube by Robert Cohen
  - https://www.youtube.com/watch?v=ZjT1YwpmI1I

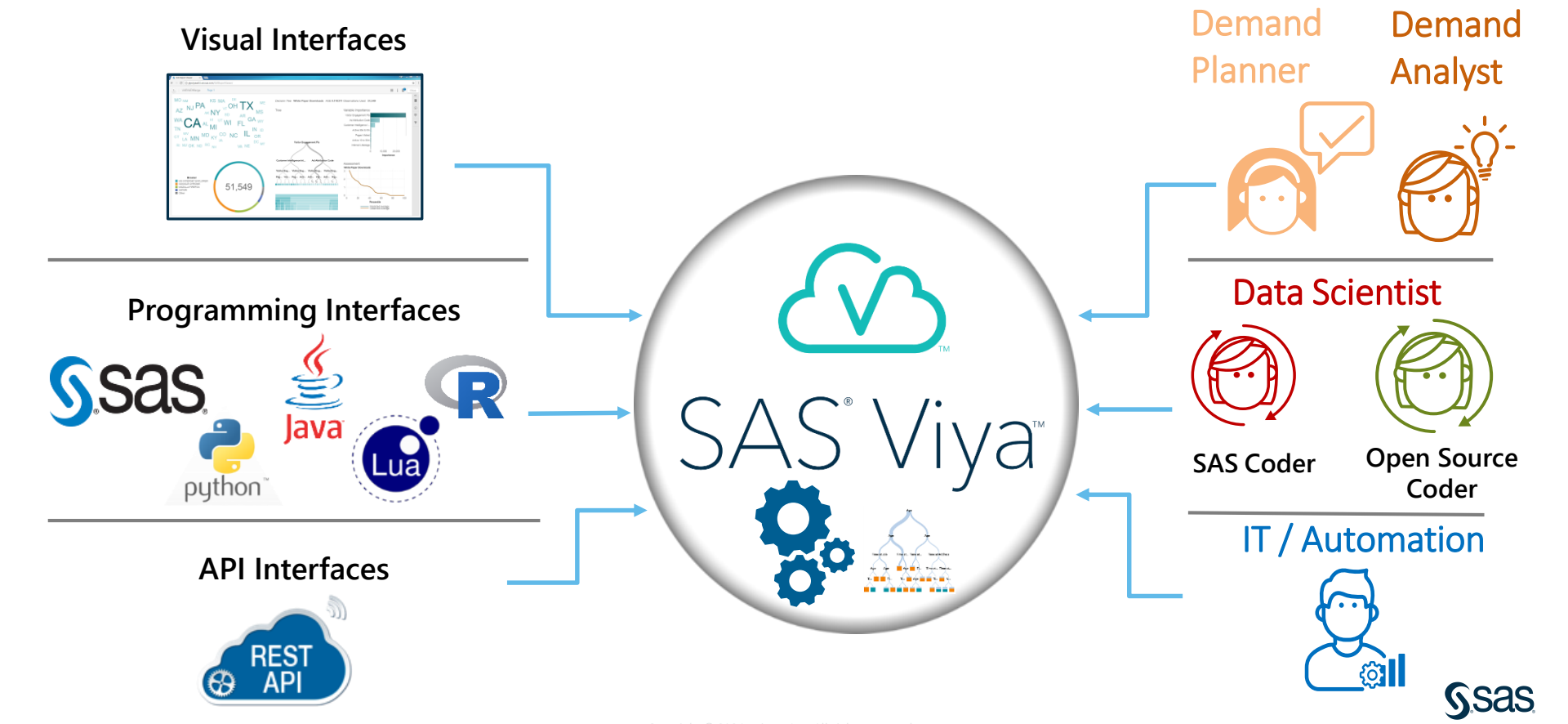
This presentation shows you

- In SAS Viya you can still use SAS Procs with the familar SAS syntax
- You can dig deeper than the SAS Procedure syntax and use CAS actions.
- CAS actions provide much more flexibility in the analysis
  - E.g. you can use PERL regular expressions
- You can use the CAS actions from the SAS program editor, Jupyter Notebook, R-Studio, …

Viya Architektur (schematisch, stark vereinfacht)



Openness of the SAS Platform



Multiple interfaces, single code base

Clients ask CAS to run “actions” on data

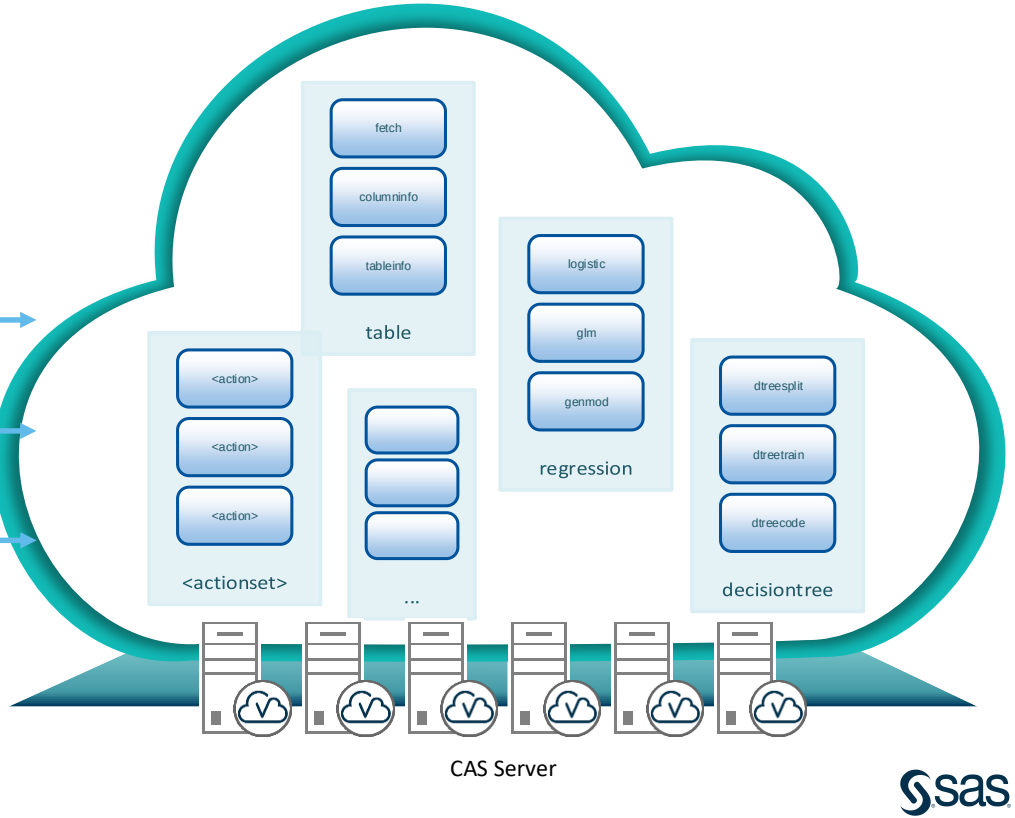
Visual Interfaces

Programming Interfaces

sas python Java Lua R

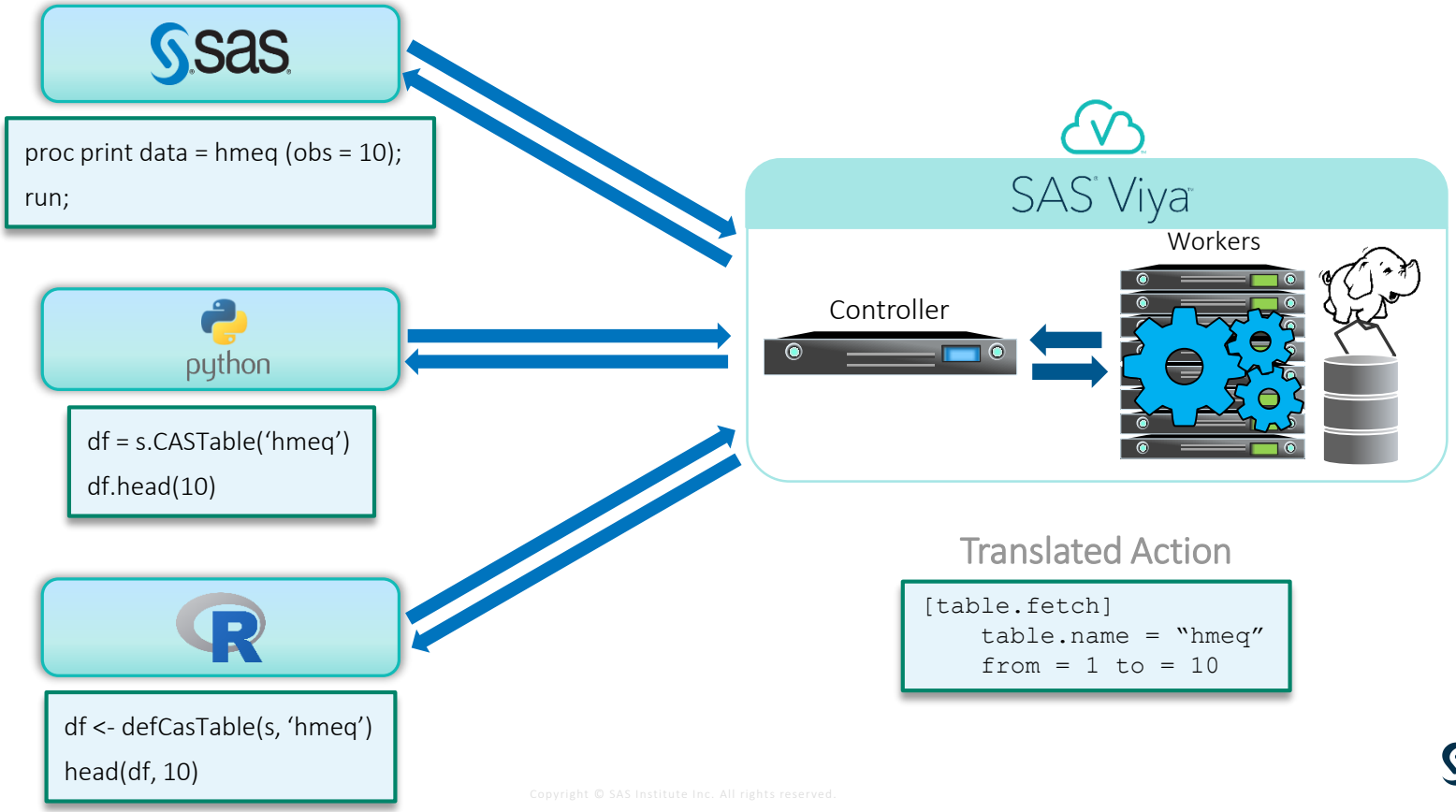
API Interfaces

REST API



SAS® Viya™: New Interface APIs

Different Languages – Same Power



Example Layout of a SAS Viya Machine Learning Procedure

Model Parameters

Partition Data

Perform Hyperparameter Tuning

Model Definition (Target, Inputs)

Generate Model Output Tables
(Fit, Variable Importance, …)

Save Model as an ASTORE

```
proc gradboost data=public.fraud_vsd
    earlystop(tolerance=0 stagnation=5)
    numBin=20 binmethod=BUCKET
    maxdepth=6
    maxbranch=2
    minleafsize=5
    assignmissing=USEINSEARCH minuseinsearch=1
    seed=12345
    printtarget
;
partition fraction (validate=0.7);
autotune useparameters=CUSTOM tuningparameters=(
    lasso(LB=0 UB=10 INIT=0)
    learningrate(LB=0.01 UB=1 INIT=0.1)
    ntrees(LB=20 UB=150 INIT=100)
    ridge(LB=0 UB=10 INIT=0)
    samplingrate(LB=0.1 UB=1 INIT=0.5)
    vars_to_try(LB=1 UB=19 INIT=19)
    )
    searchmethod=GA objective=KS maxtime=3600
    maxevals=50 maxiters=5 popsize=10
    targetevent='1'
;
target Fraud_Flag / level=nominal;
input age otherincome netincome / level=interval;
input gender marital_status education / level=nominal;
ods output
    VariableImportance    = &dm_lib..VarImp
    Fitstatistics         = &dm_data_outfit
    PredProbName = &dm_lib..PredProbName
    PredIntoName = &dm_lib..PredIntoName
    TunerResults = &dm_lib..tuneresults
    BestConfiguration = &dm_lib..tunebest(drop=name)
;
id 'accnbr'n 'age_emp_edu_peer_group'n 'ApplicationId'n '
savestate rstore=casdata.fraud_gradboost1;
run;
```

SAS Actions are generated automatically by the procedure

And can be reviewed with the `proc cas; history{first=-100}; run;` statement

```
proc cas;
  action builtins.loadActionSet / actionSet='decisionTree'; /* (SUCCESS) */
  action builtins.loadActionSet / actionSet='Sampling'; /* (SUCCESS) */
  action sampling.srs / table={name='FRAUD_VSD', caslib='public'}, sampPct=70, partInd=true, output={casOut={name='_data_',
   caslib='CASUSER(sasdemo01)', replace=true}, copyVars='ALL', partIndName='_Fraction_PartInd_'}; /* (SUCCESS) */
  action builtins.loadActionSet / actionSet='autotune'; /* (SUCCESS) */
  action autotune.tuneGradientBoostTree / tunerOptions={maxEvals=50, maxIters=5, maxTime=3600, popSize=10,
   userDefinedPartition=true, searchMethod='GA', objective='KS', targetEvent='1'}, useParameters='custom',
   tuningParameters={{namePath='nTree', lowerBound=20, upperBound=150, initValue=100}, {namePath='m', lowerBound=1,
   upperBound=19, initValue=19}, {namePath='learningRate', lowerBound=0.01, upperBound=1, initValue=0.1},
   {namePath='subSampleRate', lowerBound=0.1, upperBound=1, initValue=0.5}, {namePath='lasso', lowerBound=0, upperBound=10,
   initValue=0}, {namePath='ridge', lowerBound=0, upperBound=10, initValue=0}}, trainOptions={inputs={'age', 'OtherIncome',
   'NetIncome', 'Gender', 'Marital_Status', 'Education'}, table={name='_data_', casLib='casuser', where='_Fraction_PartInd_=0
   and Fraud_Flag NE .'}, casout={name='_model_', replace='TRUE', casLib='casuser'}, target='Fraud_Flag', nominals={'Gender',
   'Marital_Status', 'Education', 'Fraud_Flag'}, nbins=20, maxlevel=7, maxbranch=2, leafsize=5, missing='USEINSEARCH',
   minuseinsearch=1, ntree=100, seed=12345, binorder=true, varimp=true, mergebin=true, encodeName=true,
   saveState={name='FRAUD_GRADBOOST1', caslib='casdata', replace=true}, copyvars={'accnbr', 'age_emp_edu_peer_group',
   'ApplicationId', 'CCity', 'Company', 'CZipcode', 'Email', 'employer_date_peer_group', 'GivenName', 'NationalID', 'PCity',
   'PZipcode', 'Surname', 'Telephone'}, validTable={name='_data_', casLib='casuser', where='_Fraction_PartInd_=1 and Fraud_Flag
   NE .'}, earlyStop={stagnation=5}}, scoreOptions={table={name='_data_', casLib='casuser', where='_Fraction_PartInd_=1 and
   Fraud_Flag NE .'}, model={name='_model_', casLib='casuser'}, copyvars={'accnbr', 'age_emp_edu_peer_group', 'ApplicationId',
   'CCity', 'Company', 'CZipcode', 'Email', 'employer_date_peer_group', 'GivenName', 'NationalID', 'PCity', 'PZipcode',
   'Surname', 'Telephone'}, encodeName=true}; /* (WARNING) */
  action decisionTree.gbtreeScore / table={name='_data_', caslib='CASUSER(sasdemo01)', where='_Fraction_PartInd_=0 and
   Fraud_Flag NE .'}, modelTable={name='_model_', caslib='CASUSER(sasdemo01)'}, copyVars={'accnbr', 'age_emp_edu_peer_group',
   'ApplicationId', 'CCity', 'Company', 'CZipcode', 'Email', 'employer_date_peer_group', 'GivenName', 'NationalID', 'PCity',
   'PZipcode', 'Surname', 'Telephone'}, encodeName=true; /* (SUCCESS) */
  action decisionTree.gbtreeScore / table={name='_data_', caslib='CASUSER(sasdemo01)', where='_Fraction_PartInd_=1 and
   Fraud_Flag NE .'}, modelTable={name='_model_', caslib='CASUSER(sasdemo01)'}, copyVars={'accnbr', 'age_emp_edu_peer_group',
   'ApplicationId', 'CCity', 'Company', 'CZipcode', 'Email', 'employer_date_peer_group', 'GivenName', 'NationalID', 'PCity',
   'PZipcode', 'Surname', 'Telephone'}, encodeName=true; /* (SUCCESS) */
;
```

SAS Online Doc

[CAS Actions by Name]

Perl-Regular Expressions

- SGF: Paper 1754-2018, The Baker Street Irregulars Investigate: Discoveries Using Perl Regular Expressions and SAS®, Peter Eberhardt https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/1754-2018.pdf
- https://support.sas.com/resources/papers/proceedings/proceedings/sugi26/p188-26.pdf
- A regular expression is string of ‘normal’ (letters, numbers) characters coupled with some special meta-characters that, when applied to another text string, provides a concise and flexible means to "match" (specify and recognize) strings within the text, such as particular characters, words, or patterns of characters.
- In SAS: can be used for text and string search but also for a flexible definition of your model equation.

Some PERL examples for CAS Actions

- effects = { '/Amt$/','/Speed$/','/Temp$/','/Time$/','ambHumidity' }
  - /Temp$/ → Variable must contain "Temp"
- { vars = '!/score|evaluator/' },
  - All variables EXCEPT score, evaluator
- { vars = {'bakeTime','bakeTime'},interaction='CROSS' }
  - Quadratic term of bakeTime
- { vars = {'/Time/','/Time/'},interaction='BAR',maxinteract=2}
  - 2-way interactions and quadratic terms of all variables that contain „TIME"

CAS Servers allows to use non-sorted data in BY-Group Processing

```
proc regselect data=casdata.cars;
 by type;
 model mpg_highway =
              horsepower
              enginesize
              cylinders;
 ods select
     parameterestimates;
run;
```

Type	EngineSize	Cylinders	Horsepower
SUV	3.8	6	215
Wagon	4.5	8	315
Sedan	2.7	6	170
Wagon	3.5	6	280
Wagon	2.3	4	155
Sports	3.2	6	290
Sedan	3	6	210
Sedan	1.8	4	124
Sedan	3.3	6	230
Sedan	1.8	4	126
Sedan	2.5	4	165
SUV	2.7	6	173
Sedan	3.5	6	260

This presentation shows you

- In SAS Viya you can still use SAS Procs with the familar SAS syntax
- You can dig deeper than the SAS Procedure syntax and use CAS actions.
- CAS actions provide much more flexibility in the analysis
  - E.g. you can use PERL regular expressions
- You can use the CAS actions from the SAS program editor, Jupyter Notebook, R-Studio, …

Additional Links

- SAS Online Documentation
  - https://go.documentation.sas.com/?cdcId=pgmsascdc&cdcVersion=9.4_3.4&docsetId=pgmsashome&docsetTarget=home.htm&locale=en
- Tips and Techniques for Running CAS Actions on Youtube by Robert Cohen
  - https://www.youtube.com/watch?v=ZjT1YwpmI1I
- SAS Viya 14-Day Test – Free Software
  - https://www.sas.com/de_de/trials.html
- SAS University Edition (SAS9)
  - https://www.sas.com/en_us/software/university-edition.html
- Paper SAS2184-2018 Parallel Programming with the DATA Step: Next Steps David Bultman and Jason Secosky, SAS Institute Inc., Cary, NC
  - https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/2184-2018.pdf

SAS Bookstore → https://support.sas.com/en/books.html

Actual SAS 9.4 Version –> SAS9.4 M6