Send in the clones

- The first thing that this programmer learned about software is that details ***KILL.***
- 1 comma will ruin 1 million lines of code, one system feature will stop a project dead.
- It takes an extensive range of proven, reliable tools to deal with all the slings and arrows your data suppliers will throw at you.

Send in the clones

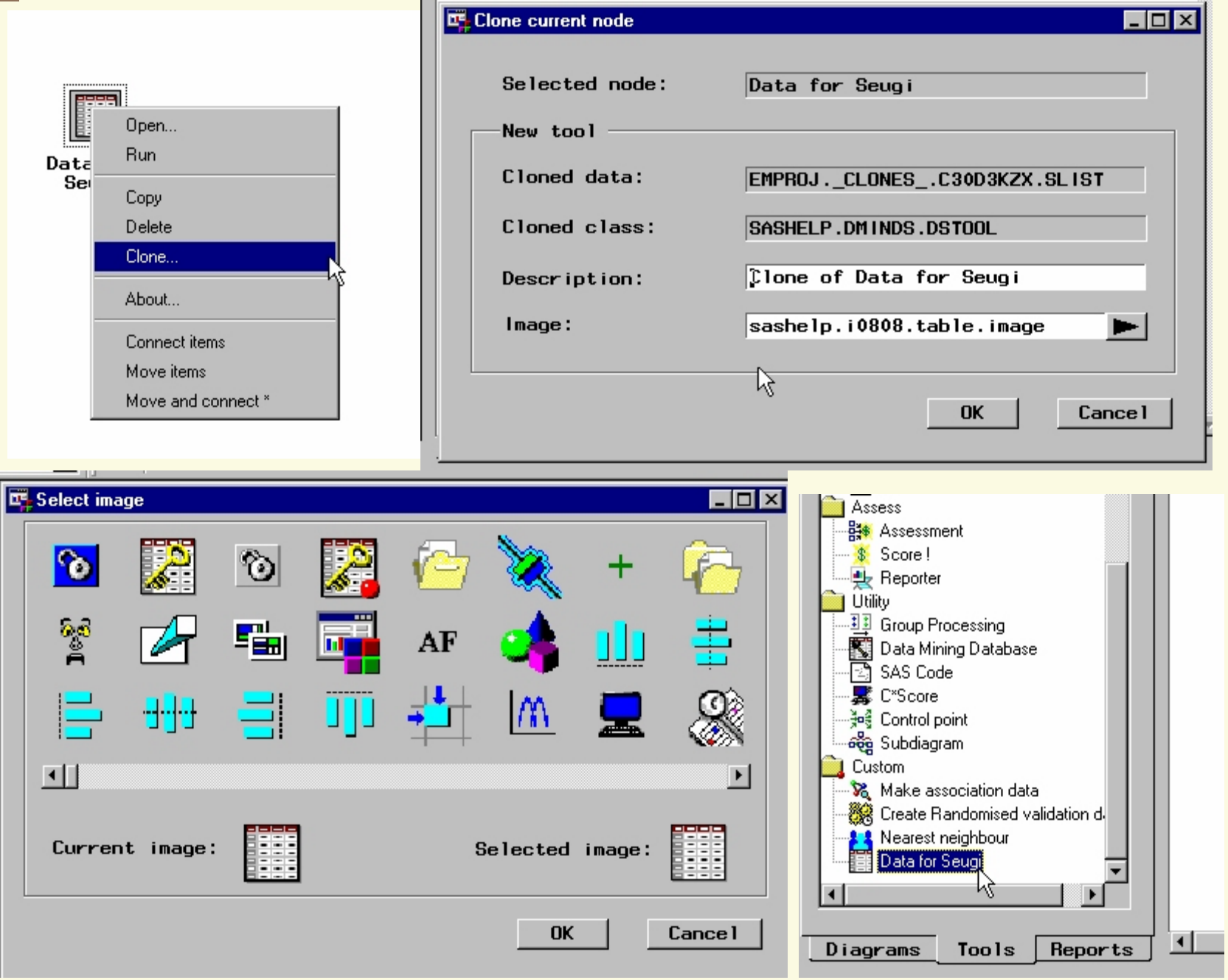
- ✓ Major strength of the SAS/Enterprise Miner is that it is built within the SAS system.
- ✓ So it comes with >20 years worth of user requested tools to get you out of any tight corner.
- ✓ It also gives you a rich set of PROCS to extend the data exploration and analysis capabilities of the miner.

How to exploit the environment.

- ✓ The SAS code node.
- ✓ The user defined model.
- ✓ The sub-diagram
- ✓ Cloning.

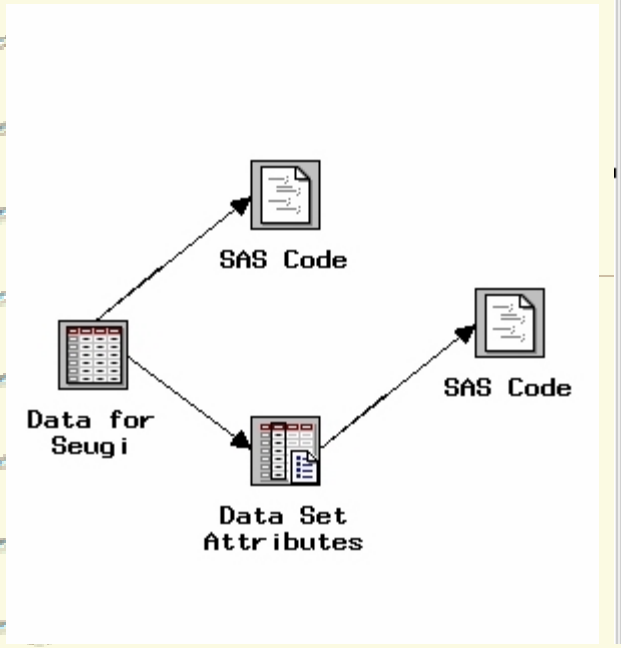
Cloning

- Cloning - easier than falling off a log. Right click on any node and there it is.
- Very useful tip -
- Clone the data source node for a project.
- Saves a lot of time.
- Easier to move data from project to project.


Open...
Run
Copy
Delete
Clone...
About...
Connect items
Move items
Move and connect *
Clone current node
Selected node:
Data for Seugi
New tool
Cloned data:
EMPROJ._CLONES_.C30D3KZX.SLIST
Cloned class:
SASHELP.DMINDS.DSTOOL
Description:
Clone of Data for Seugi
Image:
sashelp.i0808.table.image
OK
Cancel
Select image
Current image:
Selected image:
OK
Cancel
Assess
Assessment
Score !
Reporter
Utility
Group Processing
Data Mining Database
SAS Code
C*Score
Control point
Subdiagram
Custom
Make association data
Create Randomised validation d
Nearest neighbour
Data for Seugi
Diagrams
Tools
Reports

The code node.

- ✓ Exploiting the SAS code node.
- ✓ The SAS code node comes with a rich, but not complete set of macros describing the data sets and variables it can see.
- ✓ Use the data set attributes node in conjunction with the SAS code node to control the contents of these Macros to get what you want.



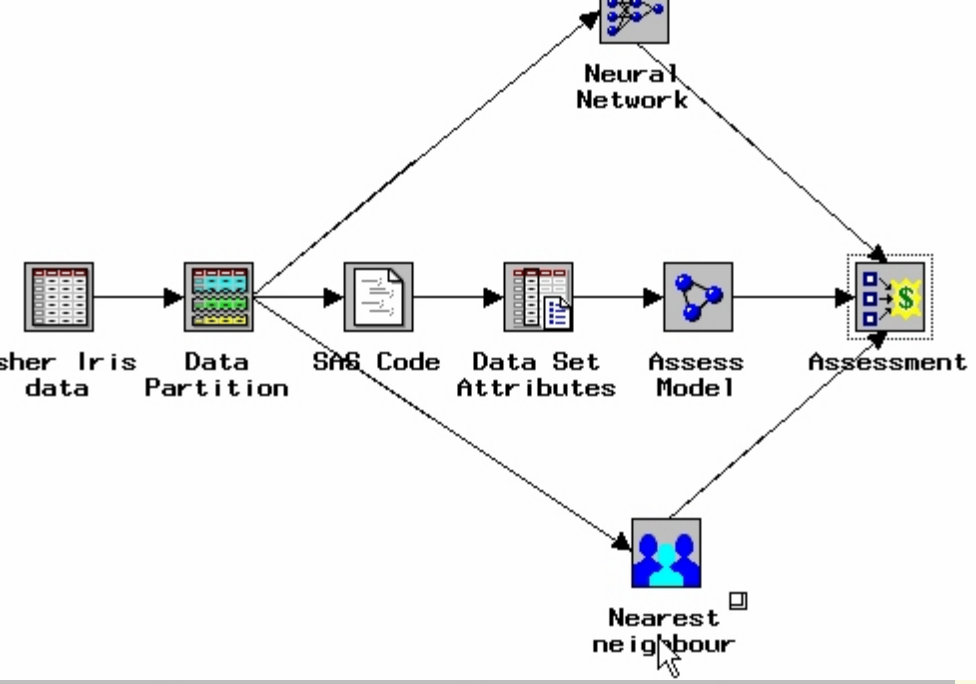
Macro Variable	Description	Current value	Data Set La
&_PLIB	Project library	EMPROJ	
&_DLIB	Data library	EMDATA	
&_TRAIN	Import: Train data set	EMDATA.VIEW_033	IBC_US.CH
&_VALID	Import: Validation data set		
&_TEST	Import: Test data set		
&_SCORE	Import: Score data set		
&_MAC_1	Import: Raw data set	EMDATA.VIEW_033	IBC_US.CH
&_VARS	All variables (not rejected)	HOUSEHOLD_ID SCCLIP SCCSLIM ...	
&_REJECTS	Rejected variables	AX10 AX100 AX101 AX102 AX103 AX...	
&_INPUTS	All inputs	SCCLIP SCCSLIM SCS SCSUNI SFDL...	
&_INTRVL	Interval inputs	SCCLIP SCCSLIM SCS SCSUNI SFDL...	
&_CLASS	Class inputs		
&_NOMINAL	Nominal inputs		
&_ORDINAL	Ordinal inputs		
&_BINARY	Binary inputs		
&_FORMATS	Input formats	MISS_INT9. MISS_INT9. MISS_INT9....	
&_TARGETS	All targets	V122	
&_TGMEAS	Target measurement	NOMINAL	

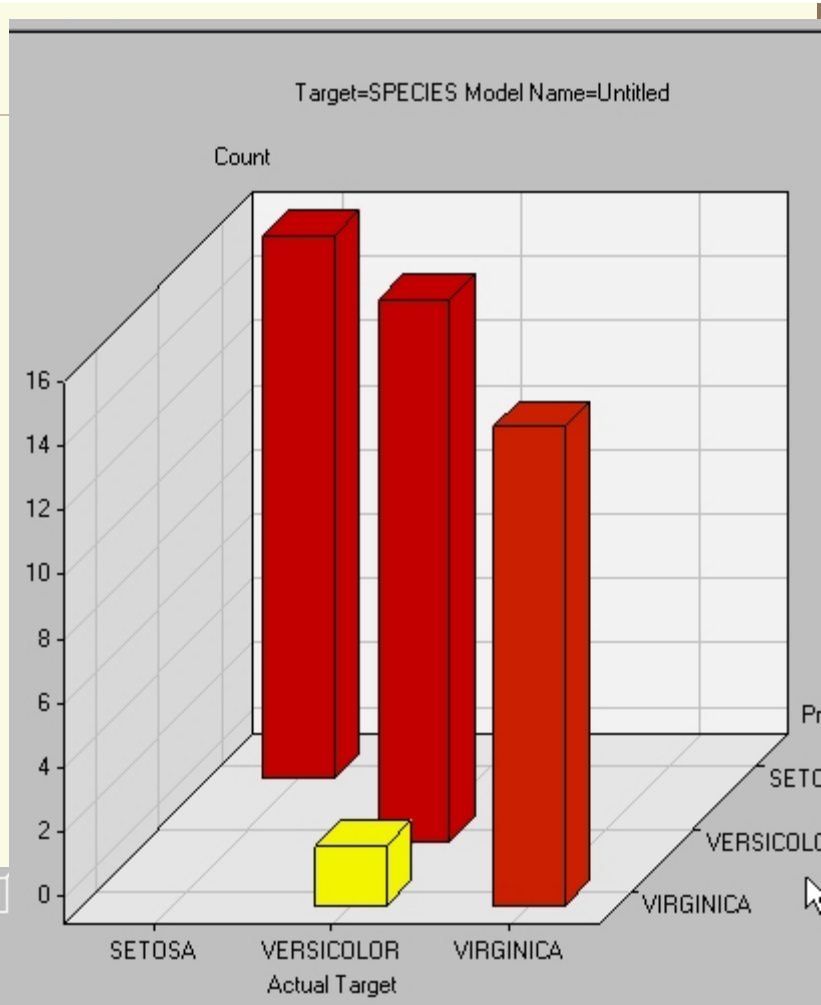
Name	Keep	Model Role	New Model Role
MAGQ3_7	Yes	rejected	input
MAGQ3_6	Yes	rejected	input
MAGQ3_5	Yes	rejected	input
MAGQ3_4	Yes	rejected	input
MAGQ3_3	Yes	rejected	input
MAGQ3_2	Yes	rejected	input
MAGQ3_1	Yes	rejected	input
HVWQ3_8	Yes	rejected	input
HVWQ3_7	Yes	rejected	input
HVWQ3_6	Yes	rejected	input
HVWQ3_5	Yes	rejected	input
HVWQ3_4	Yes	rejected	input
HVWQ3_3	Yes	rejected	input
HVWQ3_2	Yes	rejected	input
HVWQ3_1	Yes	rejected	input
HVWQ2_8	Yes	rejected	input
HVWQ2_7	Yes	rejected	input
HVWQ2_6	Yes	rejected	input

Macro Variable	Description	Current value
&_PLIB	Project library	EMPROJ
&_DLIB	Data library	EMDATA
&_TRAIN	Import: Train data...	EMDATA.EXXLO6JW
&_VALID	Import: Validation ...	
&_TEST	Import: Test data ...	
&_SCORE	Import: Score dat...	
&_MAC_1	Import: Raw data ...	EMDATA.EXXLO6JW
&_VARS	All variables (not r...	HOUSEHOLD_ID SCCLIP SCCSLIM SCS S
&_REJECTS	Rejected variables	AX10 AX100 AX101 AX102 AX103 AX103D
&_INPUTS	All inputs	SCCLIP SCCSLIM SCS SCSUNI SFDLIP SF
&_INTRVL	Interval inputs	SCCLIP SCCSLIM SCS SCSUNI SFDLIP SF
&_CLASS	Class inputs	HMRQ2_L HMRQ2_M HMRQ2_N HVWQ2
&_NOMINAL	Nominal inputs	
&_ORDINAL	Ordinal inputs	
&_BINARY	Binary inputs	HMRQ2_L HMRQ2_M HMRQ2_N HVWQ2
&_FORMATS	Input formats	MISS_INT9. MISS_INT9. MISS_INT9. MISS
&_TARGETS	All targets	V122
&_TGMEAS	Target measurem...	NOMINAL

Subdiagrams

- ✓ Fairly complicated routines can built in sub-diagrams.
- ✓ Once cloned, the routine can be retrieved with a single drag drop operation.
- ✓ Bad news, you usually have to run the code node first in order to get succeeding nodes to pick up the right arguments.
- ✓ Nice if SAS gave users more control here.





Data | Variables | Macros | Program

Run - Basic computations (client or server)

Enabled

```
proc discrim data=&_train testdata=&_valid
             method=npar
             out=&_tra testout=&_val k=1;
     class &_targets;
     var &_inputs;
run;
```

Data | Variables | Macros | Program

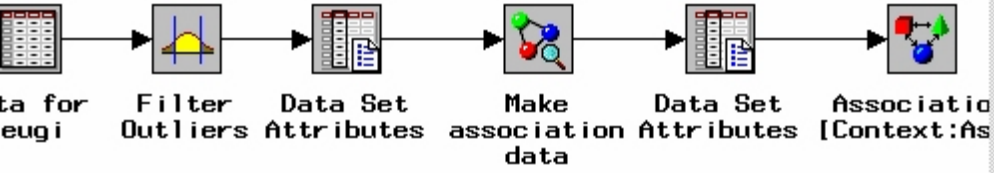
Pass imported data sets to successors

Exports:

Macro Variable	Description	Current value	Data Set Label
&_TRA	Export: Train data...	EMDATA.XTR_GQT6	TRAIN data from SAS ...
&_VAL	Export: Validate d...	EMDATA.XVL_H5CV	VALIDATION data fro...

Data manipulation - Transposing.

- One of the most useful Procedures in SAS is PROC TRANSPOSE.
- One step to change data from the format needed for modelling to the format needed for Association analysis.
- Saves ugly SQL coding and is usually much faster.



Name	Keep	Model Role	New Model Ro
HOUSEHOLD_ID	Yes	id	id
NAME	Yes	input	rejected
LABEL	Yes	input	target
COL1	Yes	rejected	rejected

Data | Variables | Macros | Program | Exportable labels

Run - Basic computations (client or server)

Enabled

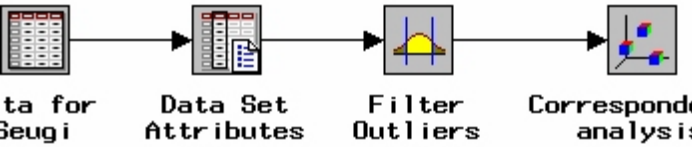
```
proc transpose data=&_mac_1 out=&_tra(where=(col1 ne '0'));
     by &_id notsorted;
     var &_binary;
run;
```

Name	Keep	Model Role	New Model Role	Measurement	New Measure
HOUSEHOLD_ID	Yes	id	id	interval	nominal
I170	Yes	input	input	nominal	binary
I169	Yes	input	input	nominal	binary
I168	Yes	input	input	nominal	binary
I167	Yes	input	input	nominal	binary
I166	Yes	input	input	nominal	binary
I165	Yes	input	input	nominal	binary
I164	Yes	input	input	nominal	binary
I163	Yes	input	input	nominal	binary
I162	Yes	input	input	nominal	binary
I161	Yes	input	input	nominal	binary
I160	Yes	input	input	nominal	binary
I159	Yes	input	input	nominal	binary
I158	Yes	input	input	nominal	binary
I157	Yes	input	input	nominal	binary
I156	Yes	input	input	nominal	binary
I155	Yes	input	input	nominal	binary
I154	Yes	input	input	nominal	binary
I153	Yes	input	input	nominal	binary
I152	Yes	input	input	nominal	binary
I151	Yes	input	input	nominal	binary
I150	Yes	input	input	nominal	binary
I149	Yes	input	input	nominal	binary

household id	NAME OF FORMER VARIABLE	LABEL
2000000	I167	Lipton Tea
2000000	I166	Wishbone
2000000	I155	Breyers
2000000	I153	Lever 2000
2000011	I163	Imperial
2000011	I158	Gorton's
2000011	I140	Sauve (TTL TTL)
2000013	I168	Lipton Soup
2000013	I164	Shedd's
2000013	I163	Imperial
2000013	I161	ICBINB
2000013	I155	Breyers
2000013	I148	Mentadent (TTL T
2000013	I140	Sauve (TTL TTL)
2000017	I167	Lipton Tea
2000017	I166	Wishbone
2000017	I164	Shedd's
2000017	I163	Imperial
2000017	I161	ICBINB
2000017	I160	Ragu
2000017	I158	Gorton's
2000017	I155	Breyers
2000017	I153	Lever 2000
2000017	I149	Degree
2000017	I147	Surf
2000017	I144	Snuggle
2000017	I141	Thermasilk

Correspondence analysis

- ✓ Very powerful visual technique.
- ✓ Works on summarised data, so can deal with large data sets.
- ✓ Annoying if client server project because very difficult to exploit JAVA, ACTIVEX graphics.



pondence analysis]

Output | Notes

18:07 Monday, April 16, 200

The CORRESP Procedure

rtia and Chi-Square Decomposition

Chi-uare	Percent	Cumulative Percent	2 4 6 8 10
65.7	6.61	6.61	**********
00.0	5.07	11.68	*********
15.1	4.19	15.87	********
39.3	4.01	19.88	********
84.8	3.65	23.53	********
63.5	3.61	27.14	********
07.4	3.48	30.61	********
64.7	3.38	33.99	********
82.9	3.19	37.18	********
42.5	3.10	40.27	********
26.6	3.06	43.33	********
14.2	3.03	46.36	********
89.4	2.97	49.34	*******

Data | Variables | Macros | Program | Exports

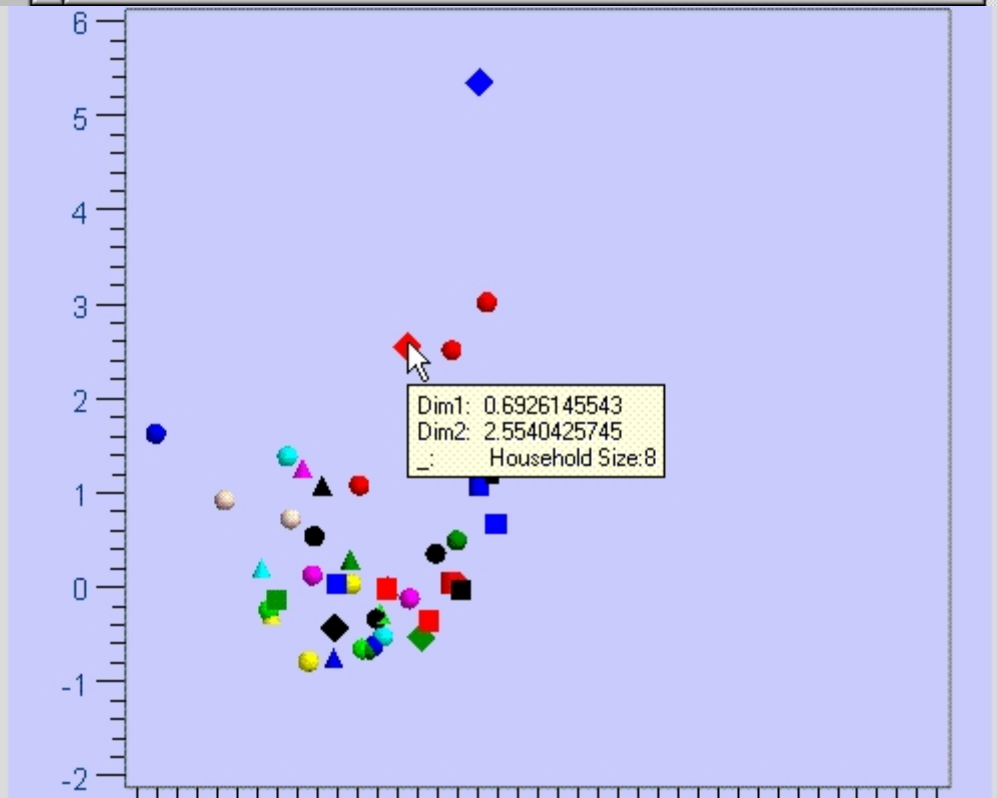
Run - Basic computations (client or server)

☑ Enabled

```
/* Default title */
%LET TITLE1=MULTIPLE CORRESPONDENCE ANALYSIS;

proc corresp data=&_mac_1(obs=1000) outc=&_res mca source;
     tables &_class;
run;

/* Tidy up coordinates */
data &_RES;
     set &_RES;
     where _type_ ne 'INERTIA';
     _=trim(_var_)||':'||trim(_name_);
run;
```



Re-coding class variables

- Bit by bit SAS is adding class statements to most PROCS.
- For those that remain, PROC GLMMOD will CLASS variables as dummy numeric for use in PROCs with no class statement.

Extra discrimination procedures

- ✓ PROC DISCRIM is rich in discrimination procedures including nearest neighbour.
- ✓ Use the User model Node to assess models and compare.
- ✓ Have to intervene to retype the prediction probabilities and identify them to the node. Would be nice if this could be done in the code node.

Customised reports

- ✓ Decision trees are rich in information and hard to summarise.
- ✓ Code node in conjunction with the data set produced by the new tree viewer and ODS can produce reports that go straight into Word.

Clones to explore the metasample

- ✓ Miner has a limited set of built in algorithms.
- ✓ Because computational speed with very large data sets is a serious limitation.
- ✓ The metasample is small, however, SAS could do a lot more with this.
- ✓ It would be nice if the name of the metasample data set were also a macro.

Explore structure and outliers

- ✓ Robust, resistant regression modules can be used.
- ✓ Nodes to explore outliers in the meta sample

Exploring the distribution

- ✓ The distribution explorer is nice but ...
- ✓ Looks at absolute frequencies and percentages of the total sample.
- ✓ Analyst needs marginal percentages and relative concentration of target values.
- ✓ Can do this utilising an AF application.
- ✓ Bit clumsy, nice if SAS gave you more control on when and how these execute.

Wish list to SAS

- More control about when and where the code node is run (on client on server).
- Saving more results from the node so it doesn't always have to be re-run.
- More macros - the metasample, the decision tree data set.

Wish list to SAS

- A macro for the original data. The macro in the node contains a view, so operations requiring direct access like point= used in re-sampling won't work.
- The ability to control the execution of AF entries.

Conclusions ...

- ✓ The Enterprise miner is far more powerful than most users realise.
- ✓ With some enhancements, that power could become very easy to access.
- ✓ SAS should consider setting a library of code nodes on their web site as accessible as all those useful Macros they supply.