IRootLab Tutorials

Classification with optimization of the number of “Principal Components” (PCs) aka PCA factors

Julio Trevisan – juliotrevisan@gmail.com

1st/December/2012

  
This document is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License](http://creativecommons.org/licenses/by-nc-sa/3.0/).

Loading the dataset 1

Preparing the dataset 1

Setting up 2

Optimization of number of PCs 6

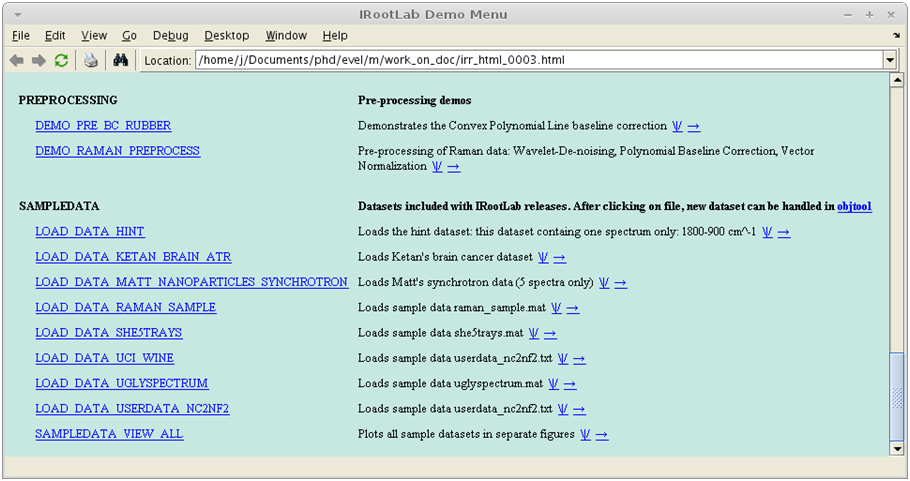
Using the optimal number of factors 9

References 14

# Loading the dataset

This tutorial uses Ketan’s Brain data[1], which is shipped with IRootLab.

1. At MATLAB command line, enter browse\_demos
2. Click on “LOAD\_DATA\_KETAN\_BRAIN\_ATR”
3. Click on “objtool” to launch objtool



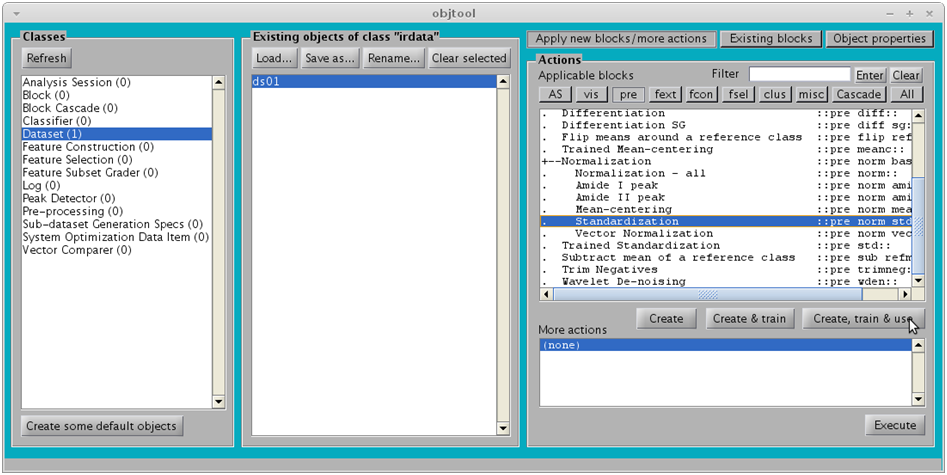
**2**

**3**

# Preparing the dataset

1. Click on ds01
2. Click on Apply new blocks/more actions
3. Click on pre
4. Click on Standardization
5. Click on Create, train & use

**Note** – Standardization[2] is mean-centering followed by scaling of each variable so that their standard deviations become 1. Mean-centering is an essential step, whereas scaling of the variables improves numerical stability.



**4**

**6**

**7**

**8**

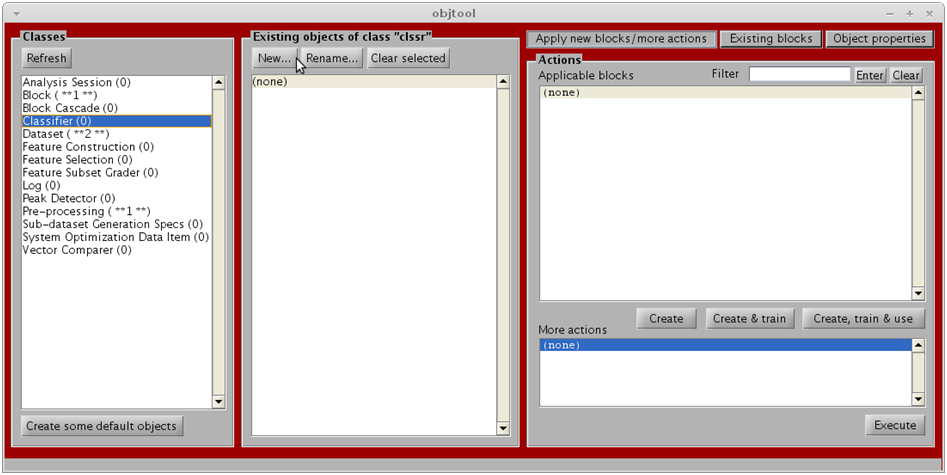
**5**

# Setting up some objects

A few objects need to be created first:

* A classifier block
* A PCA block
* A Sub-dataset Generation Specs (SGS) object

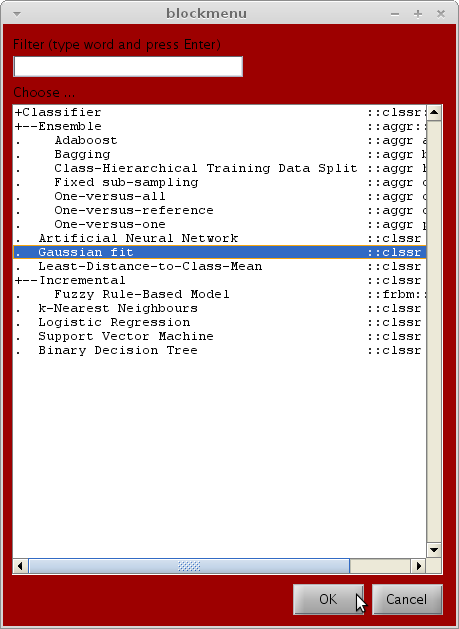
1. Click on Classifier
2. Click on New…



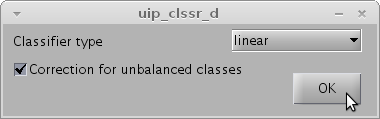
**9**

**10**

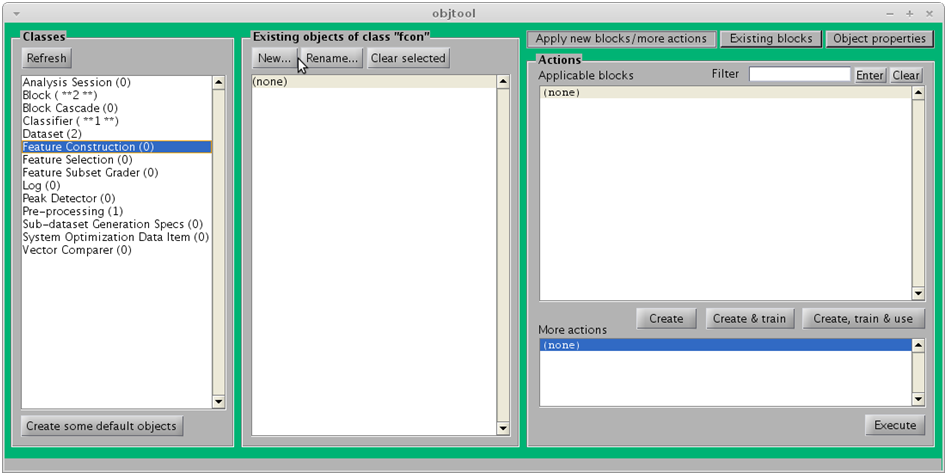
1. Click on Gaussian fit
2. Click on OK



1. Click on OK



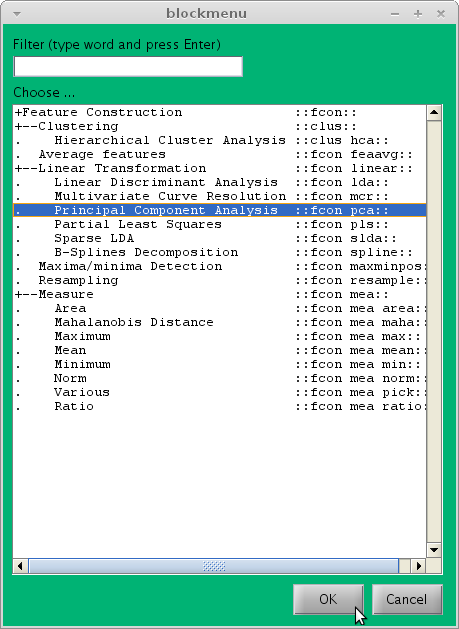
1. Click on Feature Construction
2. Click on New…



**14**

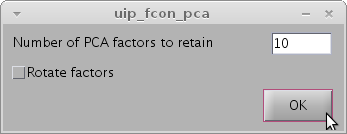
**15**

1. Click on Principal Component analysis
2. Click on OK

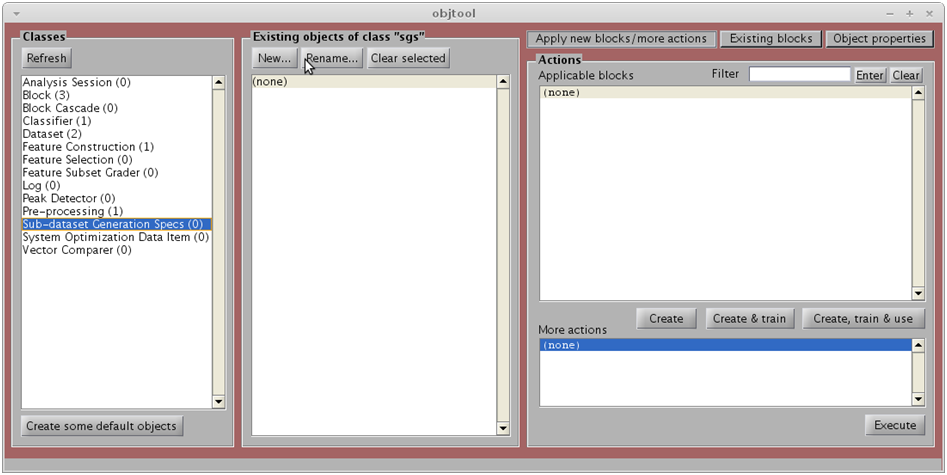


1. Click on OK

**Note** –The number of PCA factors to retainis irrelevant at this moment, as the point is to optimize this number (will be done below).



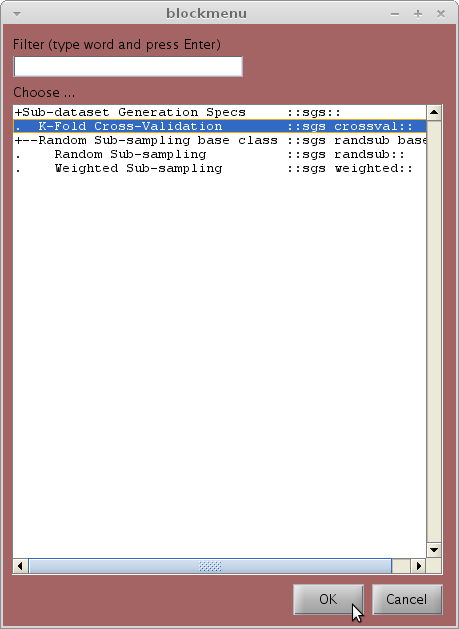
1. Click on Sub-dataset generation specs
2. Click on New…



**19**

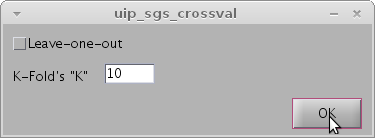
**20**

1. Click on K-Fold Cross-Validation
2. Click on OK



1. Click on OK

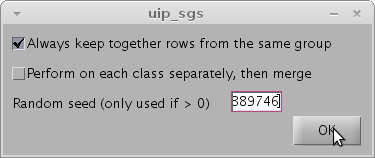
**Note -** 10-fold cross-validation is ok in the great majority of cases, except if the dataset is really small[2].



1. For the Random seed parameter, enter a random number containing a couple of digits

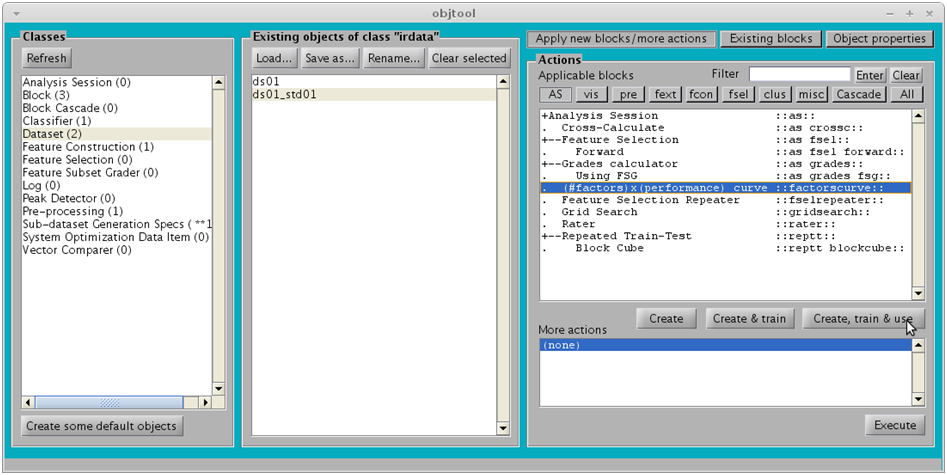
A Random seed > 0 makes the results to be exactly the same if the whole process is repeated.

In our case, we will be re-using this SGS as a parameter to a Rater later on, and we want the Rater results to be consistent with the optimization of number of PCs that will be done next.



# Optimization of number of PCs

1. Click on Dataset
2. Click on ds01\_std01
3. Click on AS
4. Click on (#factors)x(performance) curve
5. Click on OK



**25**

**26**

**27**

**28**

**29**

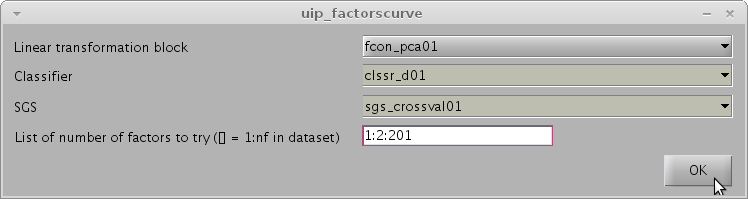
1. Specify the parameters as in the figure below
2. Click on OK

**Note** - The List of number of factors to try parameter has to be specified as a MATLAB vector.

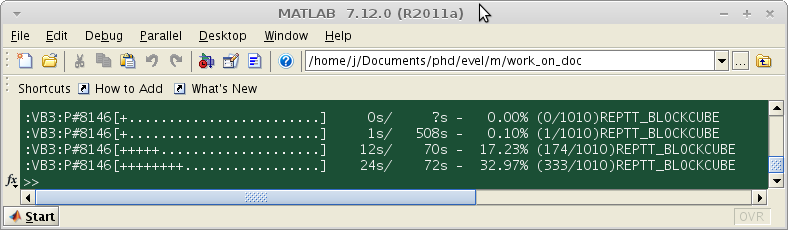
For those unfamiliar, **1:2:201** means from 1 to 201 at steps of two, *i.e.*, [1, 3, 5, 7, …, 199, 201]

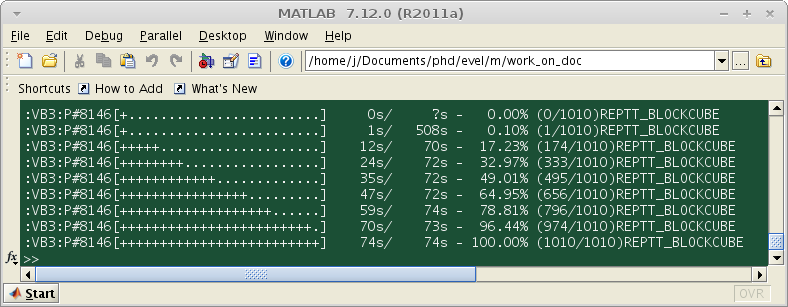
It was specified in steps of two to halve the calculation time. Even numbers of PCs will not be tested, but this will not make much of a difference in the generated curve

The maximum number of factors should be ≤ the number of features in the dataset (235 for this dataset).



You may want to monitor the calculation progress in MATLAB command window:

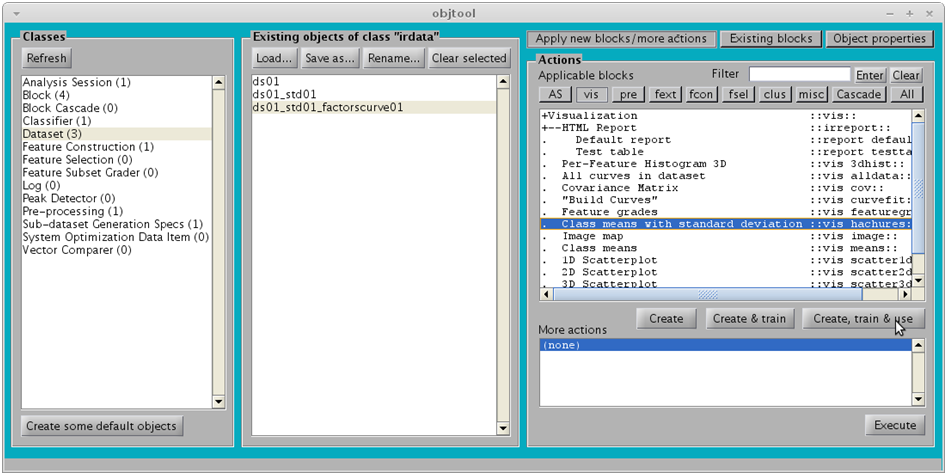




Now, visualizing the resulting curve

1. Click on ds01\_std01\_factorscurve01
2. Click on vis
3. Click on Class means with standard deviation
4. Click on Create, train & use

**Note** that (#factors)x(performance) curve output is a Dataset, unlike most Analysis Sessions (which output a Log).



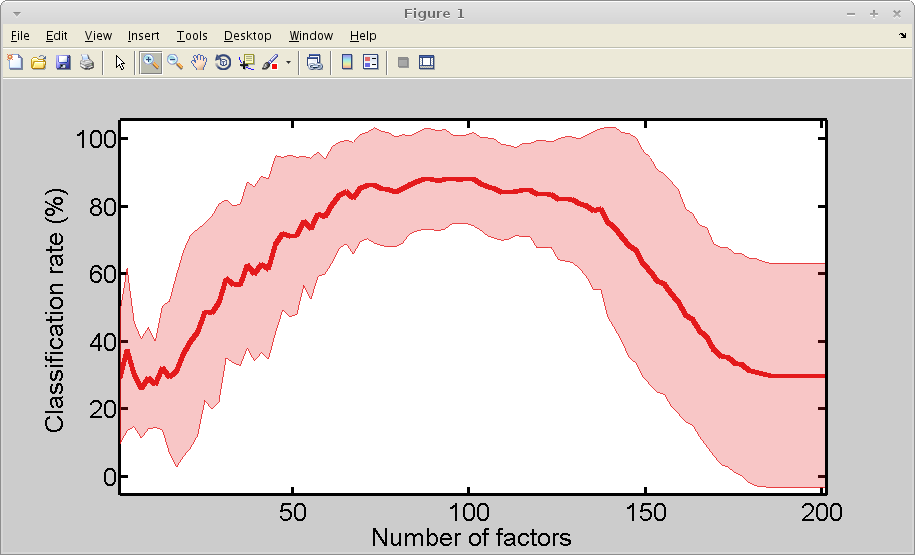
**32**

**33**

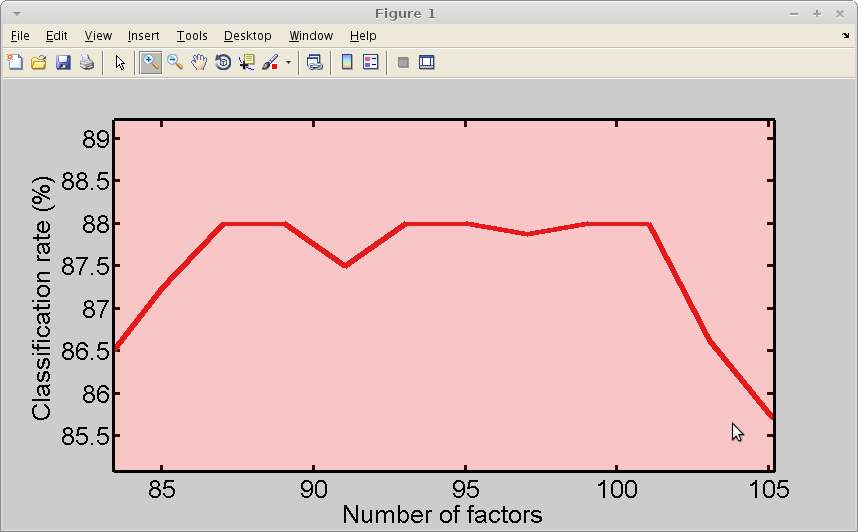
**34**

**35**

This should generate the following figure:



The figure below zooms into the previous figure. The optimal number of PCs is somewhere between 87 and 101. Let’s choose **95** (in the middle).

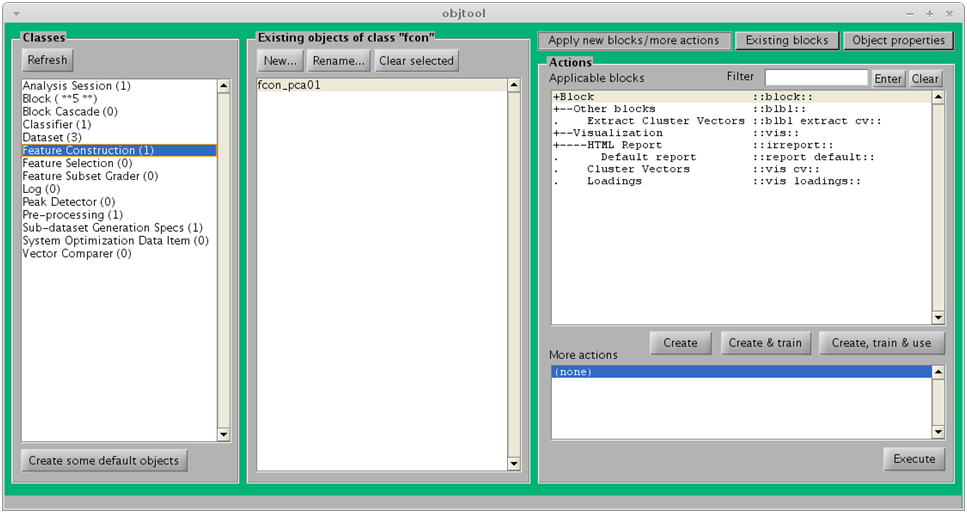


# Using the optimal number PCs

In this section, we will use a Rater to obtain further classification details (confusion matrix) using PCA with the optimal number of factors found previously.

This step will create a PCA block with **95** factors.

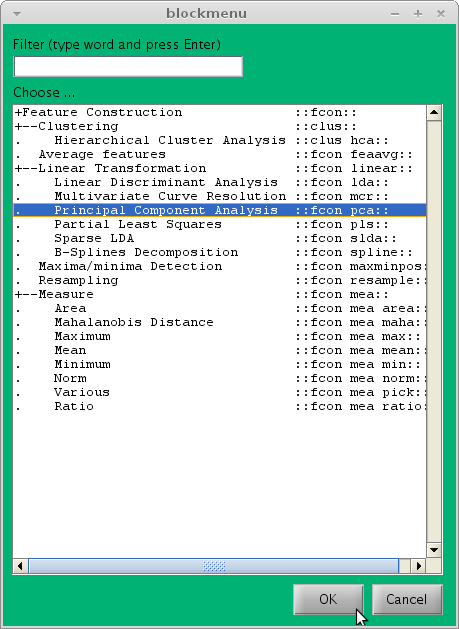
1. Click on Feature Construction
2. Click on New…



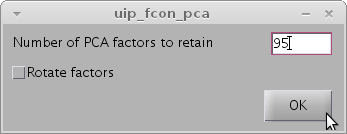
**36**

**37**

1. Click on Principal Component Analysis
2. Click on OK

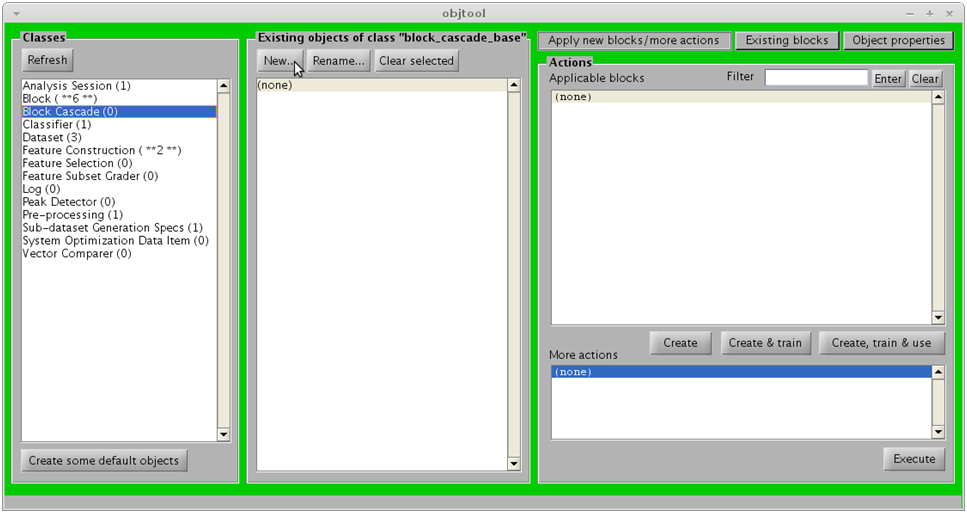


1. Enter the number **95** as below
2. Click on OK



The next step will create a classifier composed as a cascade sequence of 2 blocks.

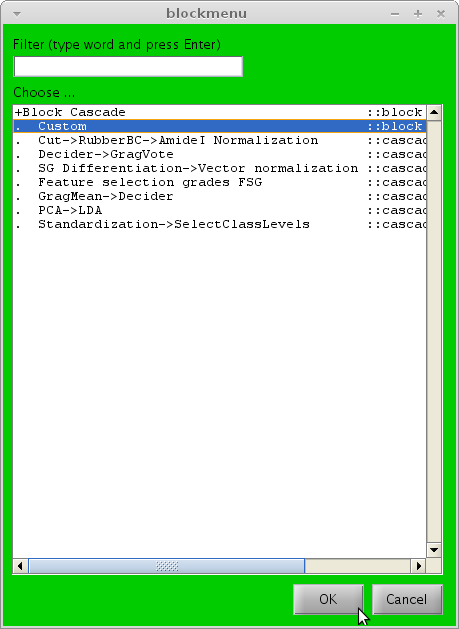
1. Click on Block cascade
2. Click on New…



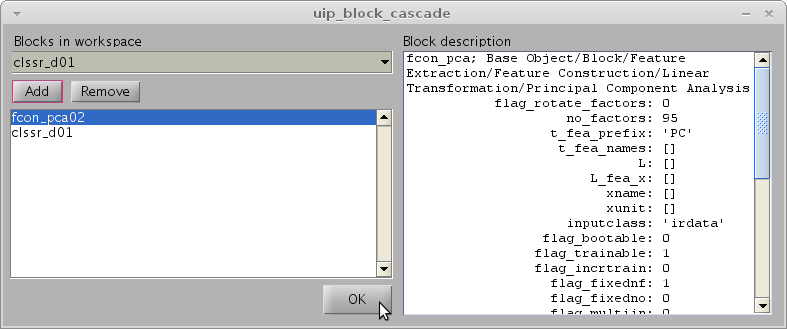
**42**

**43**

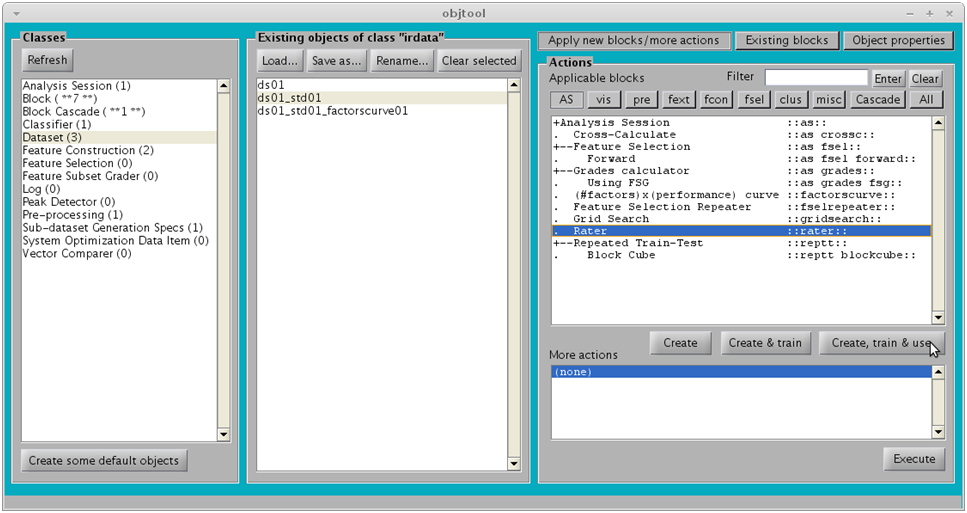
1. Click on Custom
2. Click on OK



1. Add the two blocks as below. Make sure that the blocks are added in the right sequence.
2. Click on OK



1. Click on Dataset
2. Click on ds01\_std01
3. Click on AS
4. Click on Rater
5. Click on Create, train & use



**48**

**49**

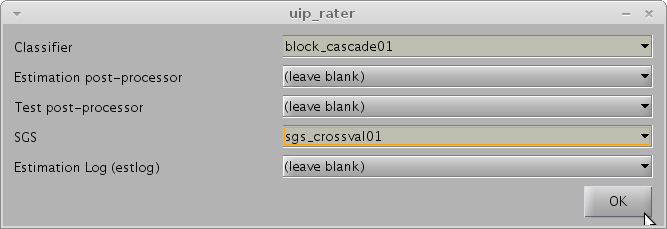
**50**

**51**

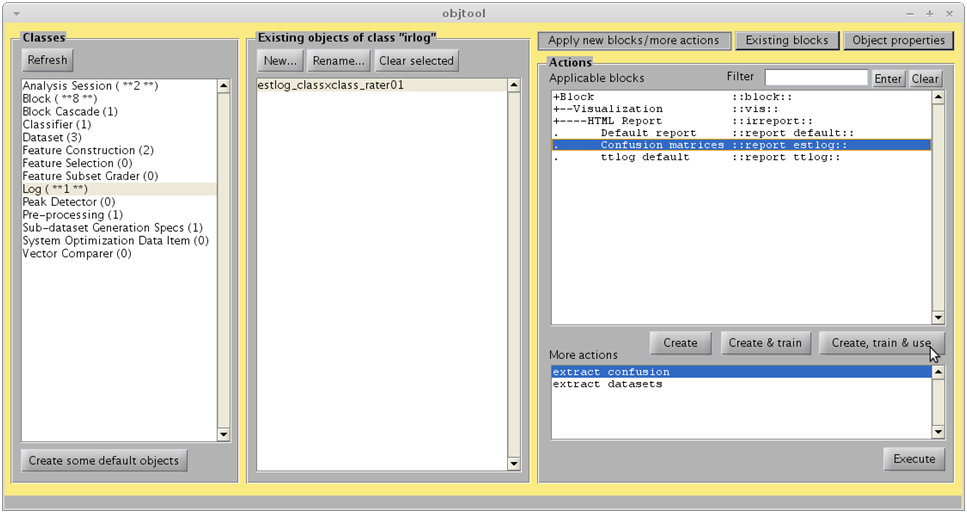
**52**

1. Specify the Classifier and SGS as below
2. Click on OK (after a few seconds, a new Log will be created)

This is where the random seed specified at step **24** starts to make sense. The data partitions used for the rater will be exactly the same used for the (#factors)x(performance) session run above; this is dictated by the sgs\_crossval01 object, which was passed as a parameter at step **30**, and is used here again.



1. Click on Log
2. Click on estlog\_classxclass\_rater01
3. Click on Confusion matrices
4. Click on Create, train & use



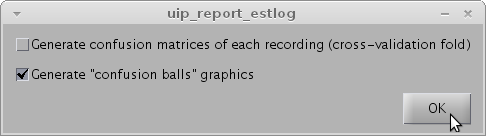
**55**

**56**

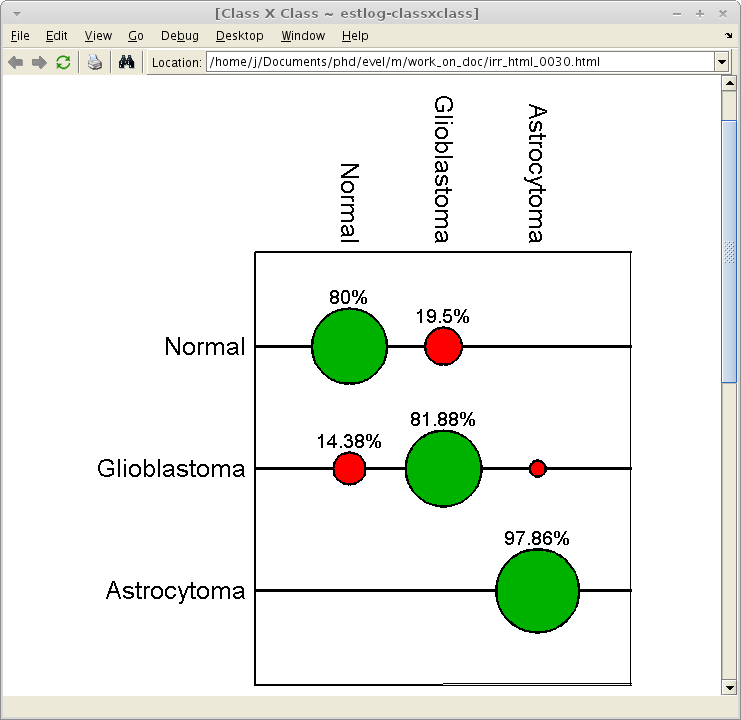
**57**

**58**

1. Click on OK



The following report should open:



# References

[1] K. Gajjar, L. Heppenstall, W. Pang, K. M. Ashton, J. Trevisan, I. I. Patel, V. Llabjani, H. F. Stringfellow, P. L. Martin-Hirsch, T. Dawson, and F. L. Martin, “Diagnostic segregation of human brain tumours using Fourier-transform infrared and/or Raman spectroscopy coupled with discriminant analysis,” *Analytical Methods*, vol. 44, no. 0, pp. 2–41, 2012.

[2] T. Hastie, J. H. Friedman, and R. Tibshirani, *The Elements of Statistical Learning*, 2nd ed. New York: Springer, 2007.