Multilevel PLS is based on the following matrix representation:

*X* - (*n x p*) matrix where *n* – number of samples, *p* – number of variables

*Xm* – (*n x p*) matrix containing average values for each variable repeated *n* times

*Xc* – (*n x p*) centered matrix

*Xb* – (*n x p*) matrix containing the values averaged over a subject (between-individual matrix)

*Xw* - (*n x p*) matrix containing the information on within-individual variation (within-individual matrix)

To perform Multilevel PLS first the matrix is represented in an aforementioned way and conventional PLS is applied either on matrix *Xb* or *Xw* depending on the question of interest.

There are two functions:

sk\_multilevel\_pls\_Xw – perform Multilevel PLS on matrix *Xw*

sk\_multilevel\_pls\_Xb – perform Multilevel PLS on matrix *Xb*

**So far these functions only work for discrimination of two groups!**

Arguments:

X – (*n x p*) data matrix

Y – (*n x 1*) label matrix (-1 for one group, 1 for another group)

subj – (*n x 1*) subject matrix (the same number for equal subjects)

ncomp – number of components in a model

scaling – perform univariate scaling (true/false)

nfold – number of subgroups in cross-validation (better to put 7)

nrepeats – number of repetitions of cross-validations (better to put 100)

nperm – number of permutation tests to be performed (0 – no test)

Output is a MATLAB structure of the form:

original\_data\_X – original (*n x p*) data matrix

original\_data\_Y – original (*n x 1*) label matrix

original\_data\_subj – original (*n x 1*) subject matrix

Xm - *Xm* matrix

Xc - *Xc* matrix

Xb - *Xb* matrix

Xw - *Xw* matrix

Ym – (*n x 1*) matrix filled with mean values of *Y* matrix

Yc – (*n x 1*) centered *Y* matrix

Xwstd/Xbstd – standard deviation of *Xw* or *Xb* matrix

Xwsc/Xbsc - *Xw* or *Xb* matrix scaled

Ycstd – standard deviation of *Yc* matrix

Ycsc - *Yc* matrix scaled

Px\_train – loadings corresponding to Xwsc/Xbsc (training set)

Tx\_train – scores corresponding to Xwsc/Xbsc (training set)

Py\_train – loadings corresponding to Ycsc (training set)

Ty\_train – scores corresponding to Ycsc (training set)

beta\_train – coefficients of multilinear regression (the first coefficient corresponds to intercept)

pred\_train – predicted *Y* values according to the training set

error\_train – training error of the form where *I(x)* is an indicator function

sq\_error\_train – training error of the form

sq\_error\_mod\_train – training error of the form

auc\_train – area under the curve (training set)

R2 – coefficient of determination of the form

cvsets – cross-validation subsets

cv\_error\_train – training error corresponding to the training subsets in cross-validation runs

cv\_sq\_error\_train - training error corresponding to the training subsets in cross-validation runs

cv\_sq\_error\_mod\_train - training error corresponding to the training subsets in cross-validation runs

cv\_error\_test – overall cross-validation error

cv\_sq\_error\_test – overall cross-validation error

cv\_sq\_error\_mod\_test – overall cross-validation error

cv\_auc – area under the curve in cross-validation

Q2 – coefficient of determination of the form

perm\_cv\_error – set of permutation errors of the form

perm\_Q2 – set of coefficients of determination of the form