ropls: PCA, PLS(-DA) and OPLS(-DA) for multivariate analysis and feature selection of omics data

Etienne A. Thévenot

January 7, 2016

1 The ropls package

The *ropls* R package implements the PCA, PLS(-DA) and OPLS(-DA) approaches with the original, NIPALS-based, versions of the algorithms [1][2]. It includes the R2 and Q2 quality metrics [3][4], the permutation diagnostics [5], the computation of the VIP values [6], the score and orthogonal distances to detect outliers [7], as well as many graphics (scores, loadings, predictions, diagnostics, outliers, etc).

The functionalities from *ropls* can also be accessed via a graphical user interface in the *Multivariate* module from the *Workflow4Metabolomics.org* online resource for computational metabolomics (http://workflow4metabolomics.org/) which provides a user-friendly, web-based environment for data pre-processing, statistical analysis, and annotation [8].

2 Context

2.1 Orthogonal Partial Least-Squares

Partial least-squares, which is a latent variable regression method based on covariance between the predictors and the response, has been shown to efficiently handle datasets with multi-collinear predictors, as in the case of spectrometry measurements [1]. More recently, Trygg and Wold introduced the orthogonal projection to latent structures (OPLS) algorithm to model separately the variations of the predictors correlated and orthogonal to the response [2].

OPLS has a similar predictive capacity compared to PLS and improves the interpretation of the predictive components and of the systematic variation [9]. In particular, OPLS modeling of single responses only requires one predictive component.

Diagnostics such as the Q2Y metrics and permutation testing are of high importance to avoid overfitting and assess the statistical significance of the model. The variable importance in projection (VIP), which reflects both the loading weights for each component and the variability of the response explained by this component [9][10] is often used for feature selection [2][9].

2.2 OPLS software

OPLS is available in the SIMCA-P commercial software (Umetrics, Umeå, Sweden; [3]). In addition, the kernel-based version of OPLS [11] is available in the open-source R statistical environment [12], and a single implementation of the linear algorithm in R has been described recently [13].

3 The sacurine metabolomics dataset

3.1 Study objective

The objective was to study the influence of age, body mass index and gender on metabolite concentrations in urine, by analysing 183 samples from a cohort of adults with liquid chromatography coupled to high-resolution mass spectrometry ([14]).

3.2 Pre-processing and annotation

Urine samples were analyzed by using an LTQ Orbitrap in the negative ionization mode. A total of 109 metabolites were identified or annotated at the MSI level 1 or 2. After retention time alignment with XCMS, peaks were integrated with Quan Browser. Signal drift and batch effect were corrected, and each urine profile was normalized to the osmolality of the sample. Finally, the data were log10 transformed.

3.3 Covariates

The volunteers' *age*, *body mass index*, and *gender* were recorded.

4 Hands-on

4.1 Loading

We first load the *ropls* package:

```
> library(ropls)
```

We then load the *sacurine* dataset which contains:

1. The *dataMatrix* matrix of numeric type containing the intensity profiles (log10 transformed)
2. The *sampleMetadata* data frame containg sample metadata
3. The *variableMetadata* data frame containg variable metadata

```
> data(sacurine)
> names(sacurine)
```

```
[1] "dataMatrix"       "sampleMetadata"   "variableMetadata"
```

We attach *sacurine* to the search path and display a summary of the content of the *dataMatrix*, *sampleMetadata* and *variableMetadata* with the *strF* **F**unction of the *ropls* package (see also *str*)

```
> attach(sacurine)
> strF(dataMatrix)
```

```
       dim  class    mode typeof   size NAs  min mean median max
 183 x 109 matrix numeric double 0.2 Mb   0 -0.3  4.2    4.3   6
    (2-methoxyethoxy)propanoic acid isomer (gamma)Glu-Leu/Ile ... Valerylglycine isomer 2
1                              3.019766011        3.888479324 ...              3.889078716
2                               3.81433889        4.277148905 ...              4.181765852
...                                    ...                ... ...                      ...
182                            3.748127215        4.523763202 ...              4.634338821
183                            4.208859398        4.675880567 ...               4.47194762
     Xanthosine
1   4.075879575
2   4.195761901
...         ...
182 4.487781609
183 4.222953354
```

```
> strF(sampleMetadata)
```

```
     age     bmi gender
 numeric numeric factor
 nRow nCol size NAs
  183    3 0 Mb   0
       age   bmi gender
HU_011   29 19.75      M
HU_014   59 22.64      F
...     ...   ...    ...
HU_208   27 18.61      F
HU_209 17.5 21.48      F
```

```
> strF(variableMetadata)
```

```
 msiLevel      hmdb chemicalClass
  numeric character     character
 nRow nCol size NAs
  109    3 0 Mb   0
                                      msiLevel      hmdb chemicalClass
(2-methoxyethoxy)propanoic acid isomer        2                 Organi
(gamma)Glu-Leu/Ile                            2                 AA-pep
...                                         ...       ...           ...
Valerylglycine isomer 2                       2           AA-pep:AcyGly
Xanthosine                                    1 HMDB00299        Nucleo
```

4.2 Principal Component Analysis (PCA)

We perform a PCA on the *dataMatrix* matrix (samples as rows, variables as columns):

```
> sacurine.pca <- opls(dataMatrix)
```

```
   R2X(cum) pre ort
h8    0.501   8   0
```

A summary of the model (8 components were selected) is printed. In addition the default `summary` 4-plot figure displays:

1. An inertia `overview` barplot (or *scree* plot): The graphic here suggests that 3 components may be sufficient to capture most of the inertia (see below),

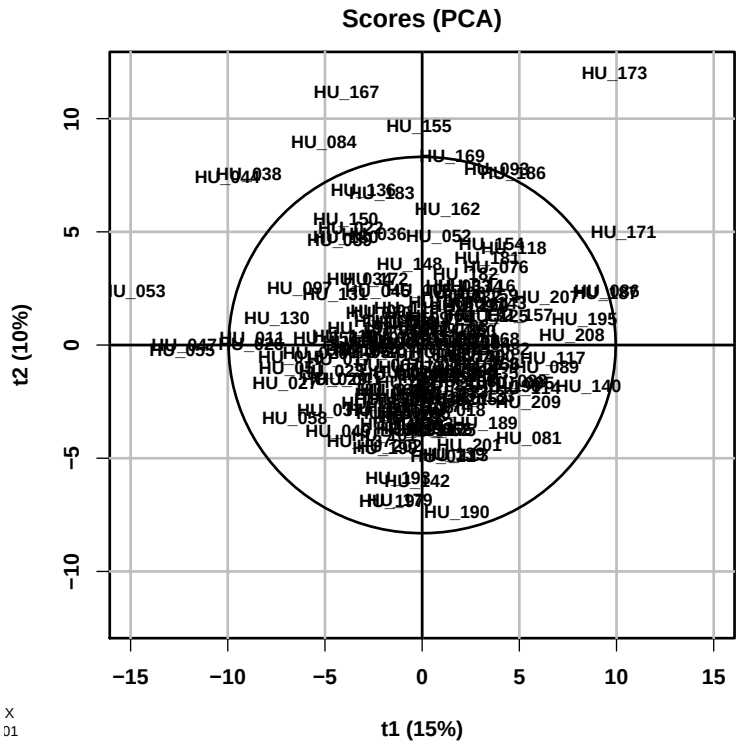

Figure 1: PCA `score` plot. The displayed components can be specified with `parCompVi` (plotting **par**ameter specifying the **Comp**onents: **V**ector of 2 **i**ntegers)

2. `outlier` diagnostics (distances within and orthogonal to the projection plane; [7]): The name of the samples with a high value for at least one of the distances are indicated,
3. The `x-score` plot,
4. The `x-loading` plot.

To plot the scores only (Figure 1):

```
> plot(sacurine.pca, typeVc = "x-score")
```

Note:

1. Since *dataMatrix* does not contain missing value, the singular value decomposition was used by default; NIPALS can be selected with the `algoC` argument specifying the **algo**rithm (**C**haracter),
2. The `predI = NA` default number of **pred**ictive components (**I**nteger) means that only components with a variance superior to the mean variance of all components are kept (note that this rule requires all components to be computed and can be quite time-consuming for large datasets with the NIPALS algorithm; in such cases, one may specify a limited number of components with the `predI` parameter).

Let us see if we notice any partition according to gender, by labeling/coloring the samples according to the gender and drawing the Mahalanobis ellipses for the male and female subgroups (Figure 2).

```
> genderFc <- sampleMetadata[, "gender"]
> plot(sacurine.pca, typeVc = "x-score",
+ parAsColFcVn = genderFc, parEllipsesL = TRUE)
```

Note that the plotting **par**ameter to be used **As Col**ors (**F**actor of **c**haracter type or **V**ector of **n**umeric type) has a length equal to the number of rows of the dataMatrix matrix (ie of samples) and that this qualitative or quantitative variable is converted into colors (by using an internal palette or color scale, respectively). We could have visualized the age of the individuals by specifying `parAsColFcVn = sampleMetadata[,`

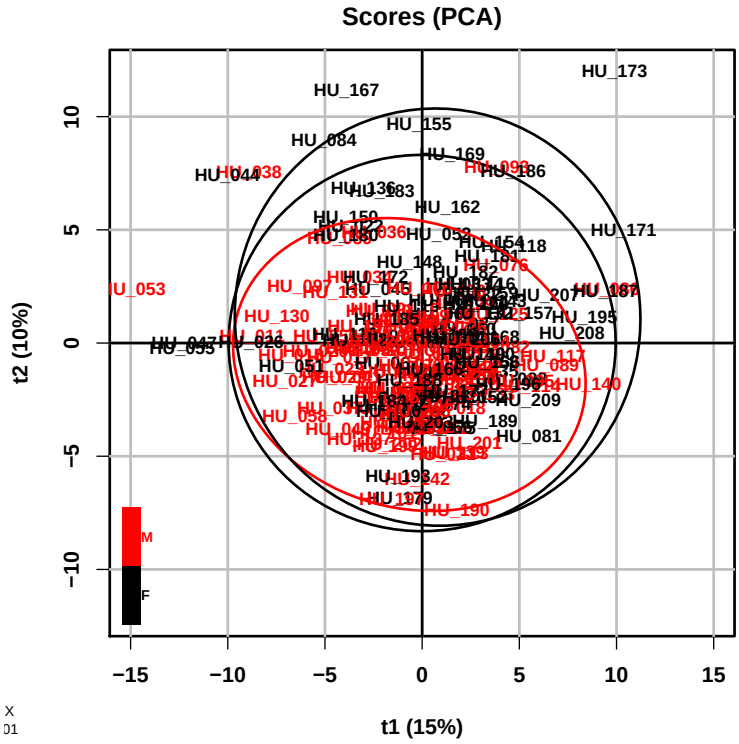

Figure 2: PCA `score` plot colored according to *gender*.

`"age"]`.

4.3 Partial least-squares: PLS and PLS-DA

For PLS (and OPLS), the Y response(s) must be provided. Y can be either a numeric vector (respectively matrix) for single (respectively multiple) (O)PLS regression, or a character factor for (O)PLS-DA classification as in the following example (The score plot is displayed on Figure 3).

```
> sacurine.plsda <- opls(dataMatrix, genderFc)
```

```
   R2X(cum) R2Y(cum) Q2(cum) RMSEE pre ort pR2Y pQ2
h3    0.275     0.73   0.584 0.262   3   0  0.1 0.1
```

Note:

1. When set to NA (as in the default), the number of components `predI` is determined automatically as follows ([3]): A new component *h* is added to the model if :
   (a) $R2Y_h \geq 1\%$, i.e., the percentage of Y dispersion (i.e., sum of squares) explained by component *h* is more than 1%, and
   (b) $Q2Y_h = 1 - PRESS_h / RSS_{h-1} \geq 0$ (or 5% when the number of samples is less than 100), i.e., the predicted residual sum of squares ($PRESS_h$) of the model including the new component *h* estimated by 7-fold cross-validation is less than the residual sum of squares (RSS_{h-1}) of the model with the previous components only (with RSS_0 being the sum of squared Y values).
2. The predictive performance of the full model is assessed by the cumulative Q2Y metric: $Q2Y = 1 - \prod_{h=1}^{r} (1 - Q2Y_h)$. We have $Q2Y \in [0, 1]$, and the higher the Q2Y, the better the performance. Models trained on datasets with a larger number of features compared with the number of samples can be prone to overfitting: in that case, high Q2Y values are obtained by chance only. To estimate

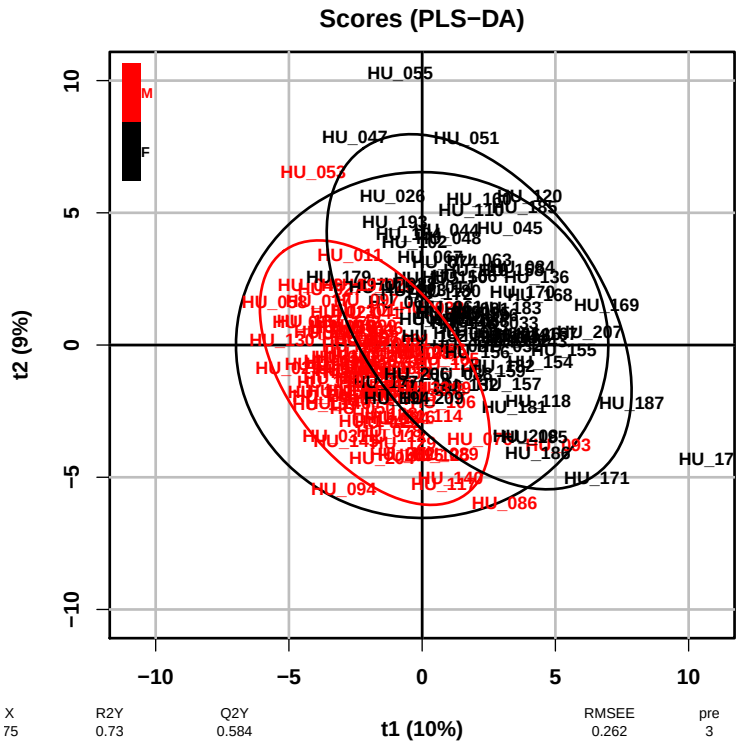

Figure 3: PLS-DA model of the *gender* response (`score` plot). The number of components and the cumulative R2X, R2Y and Q2Y are indicated below the plot.

the significance of Q2Y (and R2Y) for single response models, permutation testing can be used [5]: models are built after random permutation of the Y values, and $Q2Y_{perm}$ are computed. The *p*-value is equal to the proportion of $Q2Y_{perm}$ above $Q2Y$ (the default number of permutations is 10 as a compromise between quality control and computation speed; it can be increased with the `permI` parameter, e.g. to 1,000, to assess if the model is significant at the 0.05 level),

3. The NIPALS algorithm is used for PLS (and OPLS); *dataMatrix* matrices with (a moderate amount of) missing values can thus be analysed.

We see that our model with 3 predictive (`pre`) components has significant and quite high R2Y and Q2Y values.

4.4 Orthogonal partial least squares: OPLS and OPLS-DA

To perform OPLS(-DA), we set `orthoI` (number of components which are **ortho**gonal; **I**nteger) to either a specific number of orthogonal components, or to NA. Let us build an OPLS-DA model of the *gender* response (Figure 4). Note that for OPLS modeling of a single response, the number of predictive component is 1.

```
> sacurine.oplsda <- opls(dataMatrix, genderFc,
+ predI = 1, orthoI = NA)
```

```
    R2X(cum) R2Y(cum) Q2(cum) RMSEE pre ort pR2Y pQ2
sum    0.329    0.774   0.607  0.24   1   3  0.1 0.1
```

Let us assess the predictive performance of our model. We first train the model on a subset of the samples (here we use the odd subset value which splits the data set into two halves with similar proportions of samples for each class; alternatively, we could have used a specific subset of indices for training):

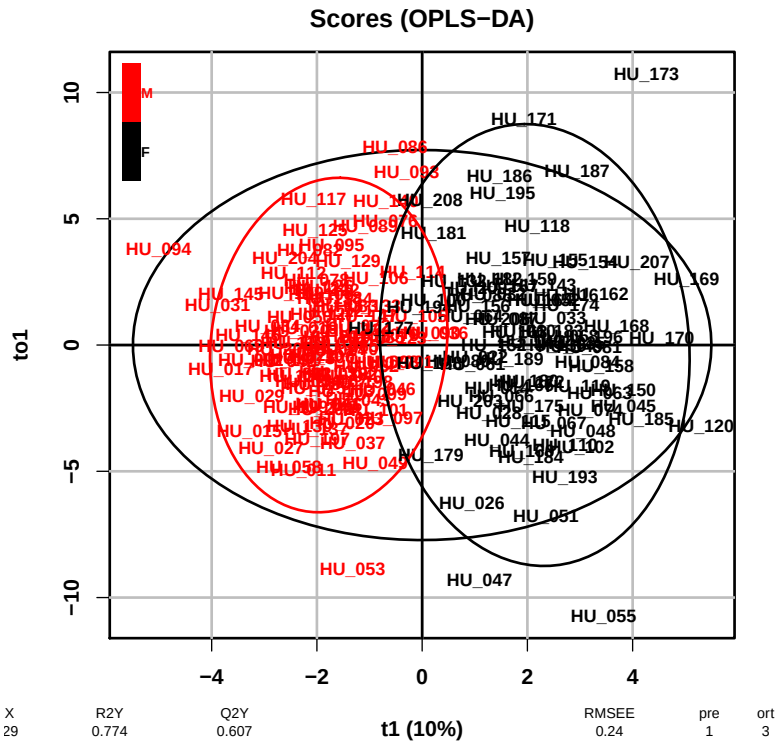

Figure 4: OPLS-DA model of the *gender* response (`score` plot). The predictive component is displayed as abscissa and the orthogonal component as ordinate.

```
> sacurine.oplsda <- opls(dataMatrix, genderFc, predI = 1, orthoI = NA, subset = "odd")
```

```
    R2X(cum) R2Y(cum) Q2(cum) RMSEE RMSEP pre ort
sum     0.18    0.767   0.563 0.245 0.342   1   1
```

We compute the performances on the training subset:

```
> trainVi <- sacurine.oplsda[["subset"]]
> table(genderFc[trainVi], fitted(sacurine.oplsda))
```

```
    M  F
  M 49  1
  F  3 39
```

We then compute the performances on the test subset:

```
> table(genderFc[-trainVi],
+       predict(sacurine.oplsda, dataMatrix[-trainVi, ]))
```

```
    M  F
  M 46  4
  F  4 37
```

As expected, the predictions on the test subset are (slightly) lower. The classifier however still achieves 91% of correct predictions.

Before closing this example session, we detach *sacurine* from the search path:

```
> detach(sacurine)
```

4.5 Session information

```
> sessionInfo()
```

- R version 3.2.2 (2015-08-14), x86_64-w64-mingw32
- Locale: LC_COLLATE=French_France.1252, LC_CTYPE=French_France.1252, LC_MONETARY=French_France.1252, LC_NUMERIC=C, LC_TIME=French_France.1252
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: ropls 1.3.6
- Loaded via a namespace (and not attached): BiocStyle 1.9.0, tools 3.2.2

5 Pre-processing and annotation of mass spectrometry data

To illustrate how *dataMatrix*, *sampleMetadata* and *variableMetadata* can be obtained from raw mass spectra file, we use the LC-MS data from the *faahKO* package [15]. We will pre-process the raw files with the *xcms* package [16] and annotate isotopes and adducts with the *CAMERA* package [17], as described in the corresponding vignettes (all these packages are from *bioconductor*).

Let us start by getting the paths to the 12 raw files (6 KO and 6 WT mice) in the ".cdf" open format. The files are grouped in two sub-directories ("KO" and "WT") since *xcms* can use sample class information when grouping the peaks and correcting retention times.

```
> library(faahKO)
> cdfpath <- system.file("cdf", package = "faahKO")
> cdffiles <- list.files(cdfpath, recursive = TRUE, full.names = TRUE)
> basename(cdffiles)
```

Next, *xcms* is used to pre-process the individual raw files, as described in the vignette.

```
> library(xcms)
> xset <- xcmsSet(cdffiles)
> xset
> xset <- group(xset)
> xset2 <- retcor(xset, family = "symmetric", plottype = "mdevden")
> xset2 <- group(xset2, bw = 10)
> xset3 <- fillPeaks(xset2)
```

Finally, the *annotateDiffreport* from *CAMERA* annotates isotopes and adducts and builds a peak table containing the peak intensities and the variable metadata.

```
> library(CAMERA)
> diffreport <- annotateDiffreport(xset3, quick=TRUE)
> diffreport[1:4, ]
```

We then build the dataMatrix, sampleMetadata and variableMetadata matrix and dataframes as follows:

```
> sampleVc <- grep("^ko|^wt", colnames(diffreport), value = TRUE)
> dataMatrix <- t(as.matrix(diffreport[, sampleVc]))
> dimnames(dataMatrix) <- list(sampleVc, diffreport[, "name"])
> sampleMetadata <- data.frame(row.names = sampleVc,
+ genotypeFc = substr(sampleVc, 1, 2))
> variableMetadata <- diffreport[, !(colnames(diffreport) %in% c("name", sampleVc))]
> rownames(variableMetadata) <- diffreport[, "name"]
```

The data can now be analysed with the *ropls* package as described in the previous section (i.e. by performing a PCA and a PLS-DA):

```
> library(ropls)
> opls(dataMatrix)
> opls(dataMatrix, sampleMetadata[, "genotypeFc"])
```

6 Other datasets

In addition to the *sacurine* dataset presented above, the package contains the following datasets to illustrate the functionalities of PCA, PLS and OPLS (see the examples in the documentation of the opls function):

aminoacids Amino-Acids Dataset. Quantitative structure property relationship (QSPR; [1]).

cellulose NIR-Viscosity example data set to illustrate multivariate calibration using PLS, spectral filtering and OPLS (Multivariate calibration using spectral data. Simca tutorial. Umetrics, Sweden).

cornell Octane of various blends of gasoline: Twelve mixture component proportions of the blend are analysed [4].

foods Food consumption patterns accross European countries (FOODS). The relative consumption of 20 food items was compiled for 16 countries. The values range between 0 and 100 percent and a high value corresponds to a high consumption. The dataset contains 3 missing data [3].

linnerud Three physiological and three exercise variables are measured on twenty middle-aged men in a fitness club [4].

lowarp A multi response optimization data set (LOWARP) [3].

mark Marks obtained by french students in mathematics, physics, french and english. Toy example to illustrate the potentialities of PCA [18].

References

[1] S. Wold, M. Sjöström, and L. Eriksson. Pls-regression: a basic tool of chemometrics. *Chemometrics and Intelligent Laboratory Systems*, 58:109–130, 2001. URL: http://dx.doi.org/10.1016/S0169-7439(01)00155-1.

[2] J. Trygg and S. Wold. Orthogonal projection to latent structures (o-pls). *Journal of Chemometrics*, 16:119–128, 2002. URL: http://dx.doi.org/10.1002/cem.695.

[3] I. Eriksson, E. Johansson, N. Kettaneh-Wold, and S. Wold. *Multi- and megavariate data analysis. Principles and applications*. Umetrics Academy, 2001.

[4] M. Tenenhaus. *La regression PLS : theorie et pratique*. Editions Technip, 1998.

[5] Ewa Szymanska, Edoardo Saccenti, AgeK. Smilde, and JohanA. Westerhuis. Double-check: validation of diagnostic statistics for pls-da models in metabolomics studies. *Metabolomics*, 8(1):3–16, 2012. URL: http://dx.doi.org/10.1007/s11306-011-0330-3, doi:10.1007/s11306-011-0330-3.

[6] B.-H. Mevik and R. Wehrens. The pls package: Principal component and partial least squares regression in r. *Journal of Statistical Software*, 18, 2007.

[7] M. Hubert, PJ. Rousseeuw, and K. Vanden Branden. Robpca: a new approach to robust principal component analysis. *Technometrics*, 47:64–79, 2005.

[8] Franck Giacomoni, Gildas Le Corguillé, Misharl Monsoor, Marion Landi, Pierre Pericard, Mélanie Pétéra, Christophe Duperier, Marie Tremblay-Franco, Jean-François Martin, Daniel Jacob, Sophie Goulitquer, Etienne A. Thévenot, and Christophe Caron. Workflow4metabolomics: a collaborative research infrastructure for computational metabolomics. *Bioinformatics*, 31(9):1493–1495, 2015. URL: http://bioinformatics.oxfordjournals.org/content/31/9/1493.abstract, arXiv:http://bioinformatics.oxfordjournals.org/content/31/9/1493.full.pdf+html.

[9] Rui Climaco Pinto, Johan Trygg, and Johan Gottfries. Advantages of orthogonal inspection in chemometrics. *Journal of Chemometrics*, 26(6):231–235, 2012. URL: http://dx.doi.org/10.1002/cem.2441, doi:10.1002/cem.2441.

[10] Tahir Mehmood, Kristian Hovde Liland, Lars Snipen, and Solve Saebo. A review of variable selection methods in partial least squares regression. *Chemometrics and Intelligent Laboratory Systems*, 118(0):62–69, 2012. URL: http://www.sciencedirect.com/science/article/pii/S0169743912001542, doi:http://dx.doi.org/10.1016/j.chemolab.2012.07.010.

[11] Max Bylesjo, Mattias Rantalainen, Jeremy Nicholson, Elaine Holmes, and Johan Trygg. K-opls package: Kernel-based orthogonal projections to latent structures for prediction and interpretation in feature space. *BMC Bioinformatics*, 9(1):106, 2008. URL: http://www.biomedcentral.com/1471-2105/9/106, doi:10.1186/1471-2105-9-106.

[12] R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. ISBN 3-900051-07-0. URL: http://www.R-project.org.

[13] R. Gaude, F. Chignola, D. Spiliotopoulos, A. Spitaleri, M. Ghitti, JM. Garcia-Manteiga, S. Mari, and G. Musco. muma, an r package for metabolomics univariate and multivariate statistical analysis. *Current Metabolomics*, 1:180–189, 2013.

[14] Etienne A. Thévenot, Aurélie Roux, Xu Ying, Eric Ezan, and Christophe Junot. Analysis of the human adult urinary metabolome variations with age, body mass index and gender by implementing a comprehensive workflow for univariate and opls statistical analyses. *Journal of Proteome Research*, 14(8):3322–3335, 2015. doi:10.1021/acs.jproteome.5b00354.

[15] Alan Saghatelian, Sunia A. Trauger, Elizabeth J. Want, Edward G. Hawkins, Gary Siuzdak, and Benjamin F. Cravatt. Assignment of endogenous substrates to enzymes by global metabolite profiling. *Biochemistry*, 43(45):14332–14339, November 2004. URL: http://dx.doi.org/10.1021/bi0480335.

[16] C. A. Smith, E. J. Want, G. O'Maille, R. Abagyan, and G. Siuzdak. Xcms: Processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Analytical Chemistry*, 78(3):779–787, 2006. URL: http://dx.doi.org/10.1021/ac051437y, doi:10.1021/ac051437y.

[17] Carsten Kuhl, Ralf Tautenhahn, Christoph Bottcher, Tony R. Larson, and Steffen Neumann. Camera: An integrated strategy for compound spectra extraction and annotation of liquid chromatography/mass spectrometry data sets. *Analytical Chemistry*, 84(1):283–289, 2012. URL: http://pubs.acs.org/doi/

abs/10.1021/ac202450g, arXiv:http://pubs.acs.org/doi/pdf/10.1021/ac202450g, doi:10.1021/ac202450g.

[18] A. Baccini. Statistique descriptive multidimensionnelle (pour les nuls), 2010.