MultiCoLA

v1.4

Angélique Gobet & Alban Ramette, September 2012

MultiCoLA Manual

Angélique Gobet, Alban Ramette, September 2012

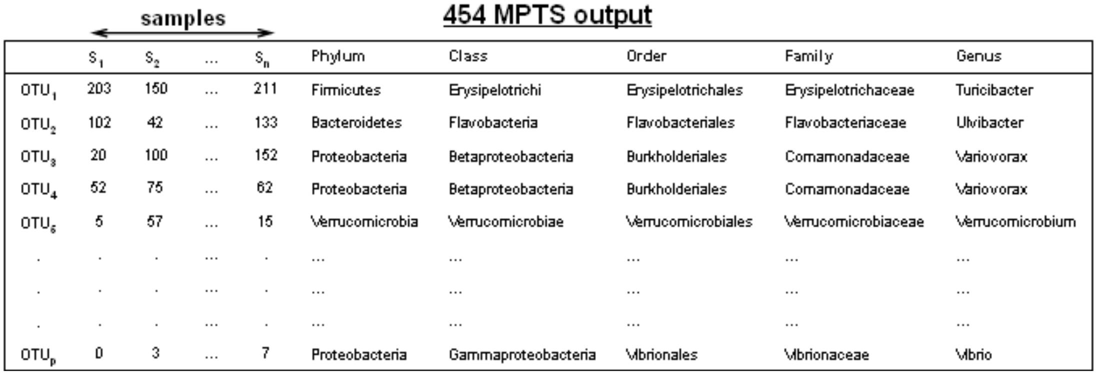
Version 1.4

Table of contents

1. Prepare the input file

Abundance table with the according taxonomy (e.g. output from the application of 454 massively parallel pyro-tag sequencing (MPTS)): Sample by [OTUs and taxonomy] (abundance matrix) to save as a .txt file, e.g. “input.txt”. In case there is no taxonomic annotation available, the input file can also be an abundance table (e.g. sample by OTUs).
You will find as an example “input.txt” in the .zip file which consists of a simplified 454 MPTS dataset with OTUs abundances and the according taxonomy.

Example:



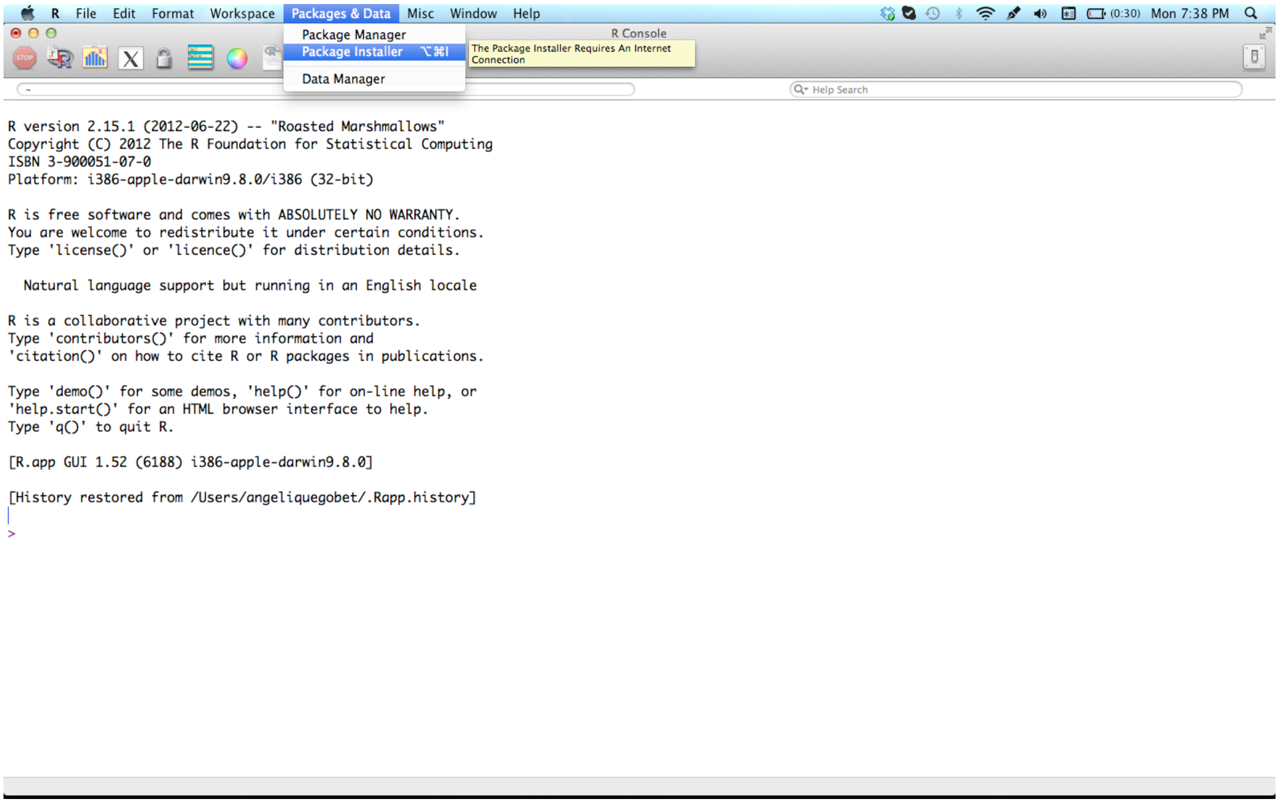
samples

454 MPTS output

	S_1	S_2	...	S_n	Phylum	Class	Order	Family	Genus
OTU_1	203	150	...	211	Firmicutes	Erysipelotrichi	Erysipelotrichales	Erysipelotrichaceae	Turicibacter
OTU_2	102	42	...	133	Bacteroidetes	Flavobacteria	Flavobacteriales	Flavobacteriaceae	Ulvibacter
OTU_3	20	100	...	152	Proteobacteria	Betaproteobacteria	Burkholderiales	Comamonadaceae	Variovorax
OTU_4	52	75	...	62	Proteobacteria	Betaproteobacteria	Burkholderiales	Comamonadaceae	Variovorax
OTU_5	5	57	...	15	Verrucomicrobia	Verrucomicrobiae	Verrucomicrobiales	Verrucomicrobiaceae	Verrucomicrobium
.	.	.	...	.	...	...	...	...	...
.	.	.	...	.	...	...	...	...	...
.	.	.	...	.	...	...	...	...	...
OTU_p	0	3	...	7	Proteobacteria	Gammaproteobacteria	Vibrionales	Vibrionaceae	Vibrio

2. Start the R interface (freely available at: http://www.r-project.org/)[1] and install *vegan* and *MASS* packages[1]

- Go to “Packages/Install packages”:

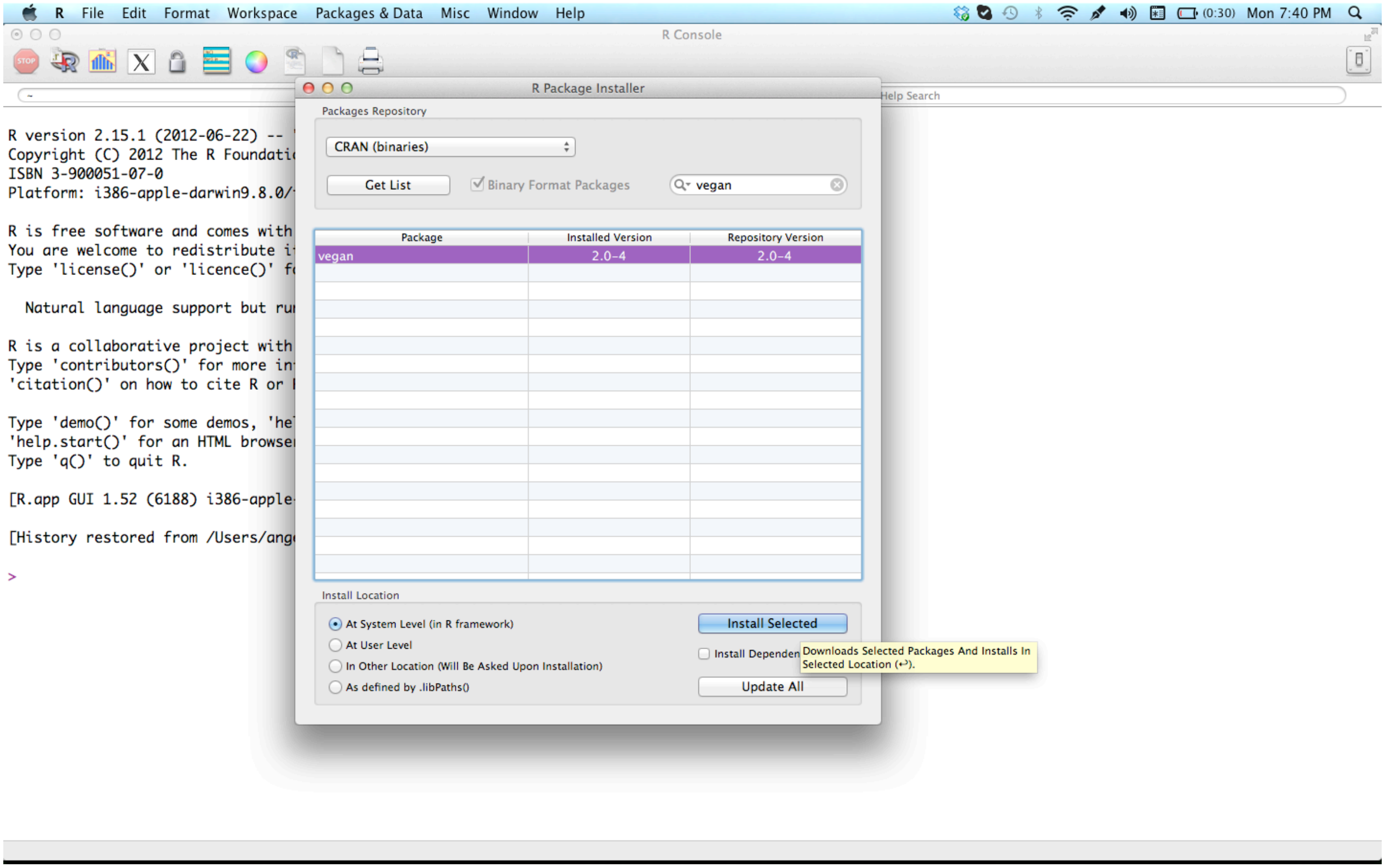


The Package Installer Requires An Internet Connection

```
R version 2.15.1 (2012-06-22) -- "Roasted Marshmallows"
Copyright (C) 2012 The R Foundation for Statistical Computing
ISBN 3-900051-07-0
Platform: i386-apple-darwin9.8.0/i386 (32-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[R.app GUI 1.52 (6188) i386-apple-darwin9.8.0]

[History restored from /Users/angeliquegobet/.Rapp.history]

>
```

[1] The editor Tinn-R may be a convenient way to read all the scripts of MultiCoLa – and eventually modify them. It can be downloaded here: http://sciviews.org/Tinn-R/.

- Select a CRAN mirror closer to the place where you work and select the package you would like to install (e.g. *vegan*):



```
R version 2.15.1 (2012-06-22) -- '
Copyright (C) 2012 The R Foundati
ISBN 3-900051-07-0
Platform: i386-apple-darwin9.8.0/

R is free software and comes with
You are welcome to redistribute i
Type 'license()' or 'licence()' f

  Natural language support but ru

R is a collaborative project with
Type 'contributors()' for more in
'citation()' on how to cite R or 

Type 'demo()' for some demos, 'he
'help.start()' for an HTML browse
Type 'q()' to quit R.

[R.app GUI 1.52 (6188) i386-apple-

[History restored from /Users/ang

>
```

Packages Repository

CRAN (binaries)

Get List ☑ Binary Format Packages vegan

Package	Installed Version	Repository Version
vegan	2.0-4	2.0-4

Install Location

● At System Level (in R framework)
○ At User Level
○ In Other Location (Will Be Asked Upon Installation)
○ As defined by .libPaths()

Install Selected

□ Install Dependen

Downloads Selected Packages And Installs In Selected Location (↩).

Update All

3. Load the data into the R workspace and run the scripts: community structure

First, set the directory where you want to work, i.e. where your input file and the series of scripts should be, and where you will find the several outputs from these scripts. The directory should be created beforehand and the name should not contain spaces to be readable by the software R (e.g. use underscore to separate words). The path to the working directory (*e.g.* "454_MPTS") may be indicated as followed:

```
setwd("/Users/angeliquegobet/R/454_MPTS")
```

The data can then be stored into the object M in the workspace[2,3]:

```
M<-read.table("input.txt",header=TRUE,row.names=1)
```

The scripts can now be used on the sample by OTUs (or taxonomy) matrix M according to the different steps:

3.1. To obtain a matrix for each taxonomic level (when the taxonomic annotation is available only):

```
source("taxa.pooler.1.4.r")
```

Some explanations about the function are then appearing. To execute the script, the output can be stored in the R workspace under a name of your choice, for instance:

```
all_taxa_pooled<-taxa.pooler(M)
```

Some questions will then appear:

- To give the total number of samples in the data set

```
Number of samples? (e.g. 16)...
```

- To give the total number of taxonomic levels

```
Number of taxonomic levels? (e.g. phylum+class+order+family+genus=5)...
```

- To choose the output as tables with presence/absence of OTUs (y) or abundance of OTUs (n)

```
Presence/absence tables as output? (y/n)
```

- To export the tables as text files (can be useful for further use under Microsoft Excel, for instance)

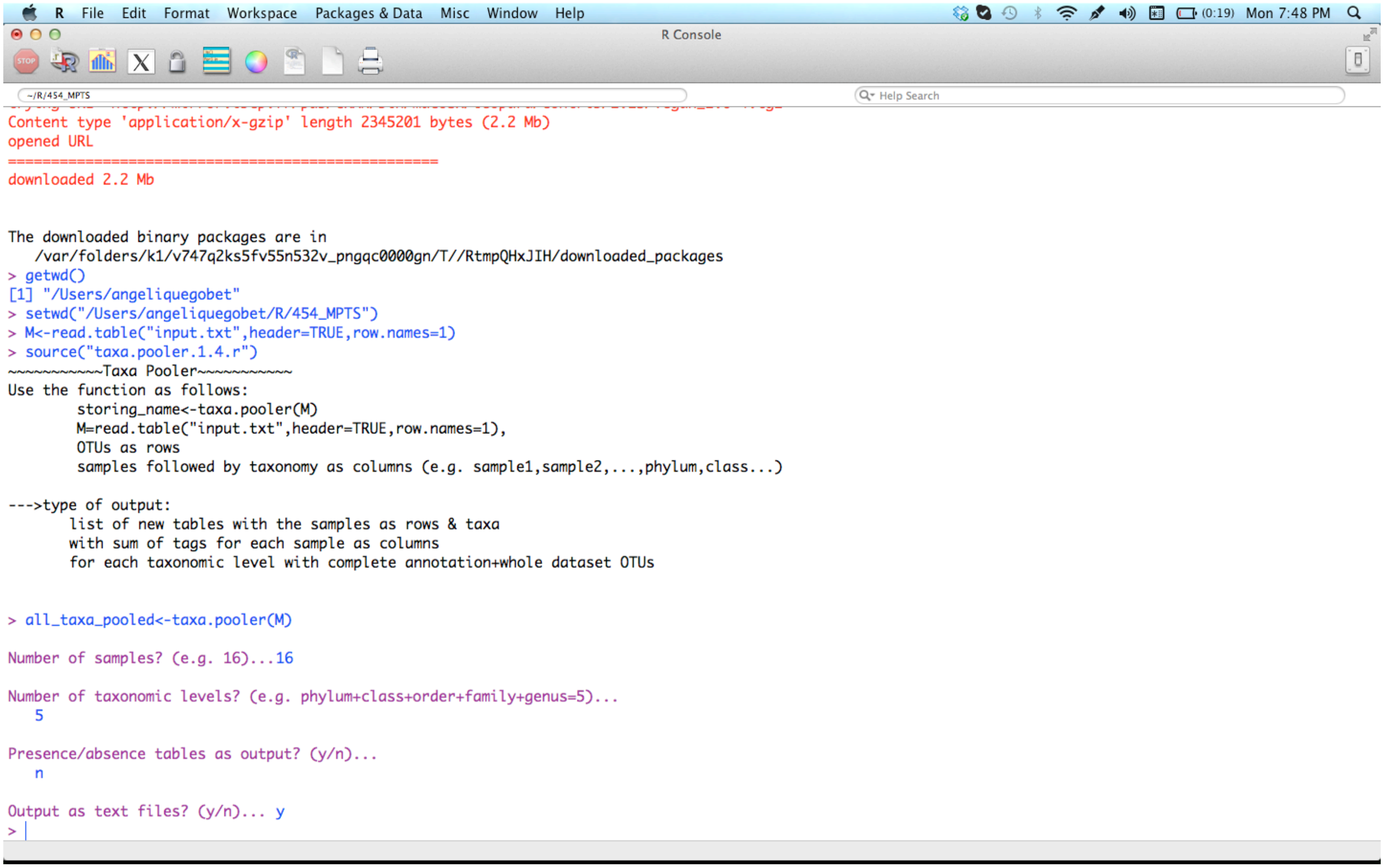
```
Output as text files? (y/n)... y
```

The output (all_taxa_pooled) is a list of matrices for each taxonomic level and two other matrices describing the occurrence of each OTU: one for only OTUs with a complete annotation and another one with all the OTUs. The output matrices can be either abundance matrices or presence/absence matrices.

[2] When OTUs are not fully annotated, complete by writing NA (e.g. some OTUs have an annotation from the Phylum to the Genus level while others have an annotation from the Phylum to the Family level then write NA in the missing Genus level)

[3] In the case of having a OTU table without annotation, one does not need to use the taxa.pooler function but the table should be transformed as a list to be used with the successive functions (e.g. COtables), for instance:

```
M<-list(M)
```



```
Content type 'application/x-gzip' length 2345201 bytes (2.2 Mb)
opened URL
==================================================
downloaded 2.2 Mb


The downloaded binary packages are in
   /var/folders/k1/v747q2ks5fv55n532v_pngqc0000gn/T//RtmpQHxJIH/downloaded_packages
> getwd()
[1] "/Users/angeliquegobet"
> setwd("/Users/angeliquegobet/R/454_MPTS")
> M<-read.table("input.txt",header=TRUE,row.names=1)
> source("taxa.pooler.1.4.r")
~~~~~~~~~~~Taxa Pooler~~~~~~~~~~~
Use the function as follows:
        storing_name<-taxa.pooler(M)
        M=read.table("input.txt",header=TRUE,row.names=1),
        OTUs as rows
        samples followed by taxonomy as columns (e.g. sample1,sample2,...,phylum,class...)

--->type of output:
       list of new tables with the samples as rows & taxa
       with sum of tags for each sample as columns
       for each taxonomic level with complete annotation+whole dataset OTUs


> all_taxa_pooled<-taxa.pooler(M)

Number of samples? (e.g. 16)...16

Number of taxonomic levels? (e.g. phylum+class+order+family+genus=5)...
   5

Presence/absence tables as output? (y/n)...
   n

Output as text files? (y/n)... y
> 
```

3.2. Application of successive cutoffs on each original matrix

```
source("COtables.1.4.r")
```

The truncated datasets can be stored as follows:

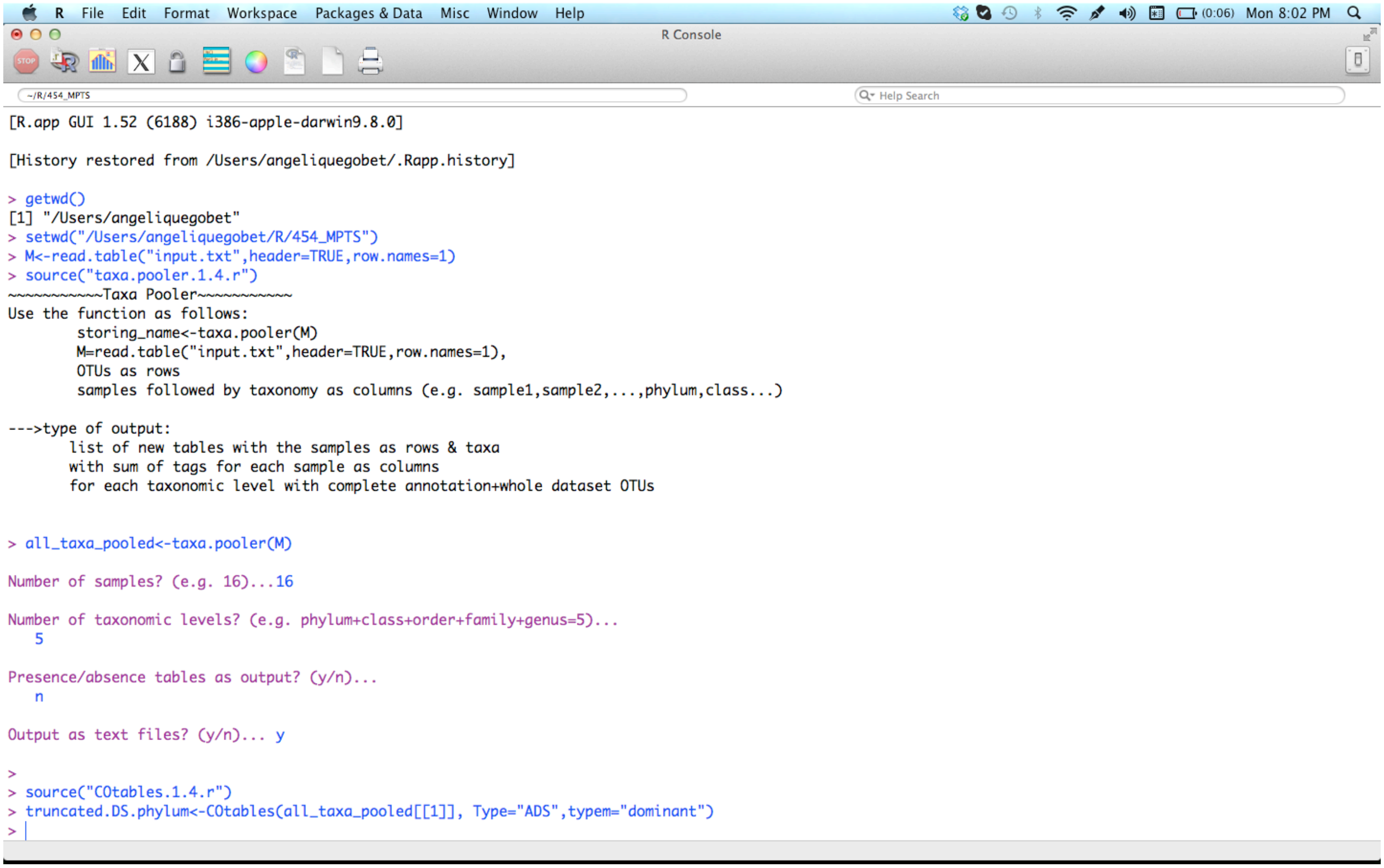
```
truncated.DS.i<-COtables(all_taxa_pooled[[i]], Type="ADS",typem="dominant")
```

With:

- The input: “all_taxa_pooled[[i]]”, representing one of the matrix obtained from the taxa.pooler(), with i from 1 (phylum level here) to the total number of taxonomic levels (here, 7), for example:

```
truncated.DS.phylum<-COtables(all_taxa_pooled[[1]], Type="ADS",typem="dominant")
truncated.DS.class<-COtables(all_taxa_pooled[[2]], Type="ADS",typem="dominant")
                                    .
                                    .
                                    .
truncated.DS.OTUwholeDS<-COtables(all_taxa_pooled[[7]], Type="ADS",typem="dominant
```

- Type = Type of cutoff: all dataset-,”ADS”, or sample-,”SAM”, based;
- typem = choice of the fraction of the matrix to work on: “dominant” types or “rare” types.



```
[R.app GUI 1.52 (6188) i386-apple-darwin9.8.0]

[History restored from /Users/angeliquegobet/.Rapp.history]

> getwd()
[1] "/Users/angeliquegobet"
> setwd("/Users/angeliquegobet/R/454_MPTS")
> M<-read.table("input.txt",header=TRUE,row.names=1)
> source("taxa.pooler.1.4.r")
~~~~~~~~~~~Taxa Pooler~~~~~~~~~~~
Use the function as follows:
        storing_name<-taxa.pooler(M)
        M=read.table("input.txt",header=TRUE,row.names=1),
        OTUs as rows
        samples followed by taxonomy as columns (e.g. sample1,sample2,...,phylum,class...)

--->type of output:
       list of new tables with the samples as rows & taxa
       with sum of tags for each sample as columns
       for each taxonomic level with complete annotation+whole dataset OTUs


> all_taxa_pooled<-taxa.pooler(M)

Number of samples? (e.g. 16)...16

Number of taxonomic levels? (e.g. phylum+class+order+family+genus=5)...
   5

Presence/absence tables as output? (y/n)...
   n

Output as text files? (y/n)... y

>
> source("COtables.1.4.r")
> truncated.DS.phylum<-COtables(all_taxa_pooled[[1]], Type="ADS",typem="dominant")
> |
```

3.3. Calculation of Spearman (or Pearson, Kendall) correlations and Procrustes correlations between the original dataset and the truncated ones

In this script, the truncated datasets are automatically calculated.

```
source("cutoff.impact.1.4.r")
```

Some explanations about the function are then appearing. Store the output in the R workspace under a name of your choice, for instance:

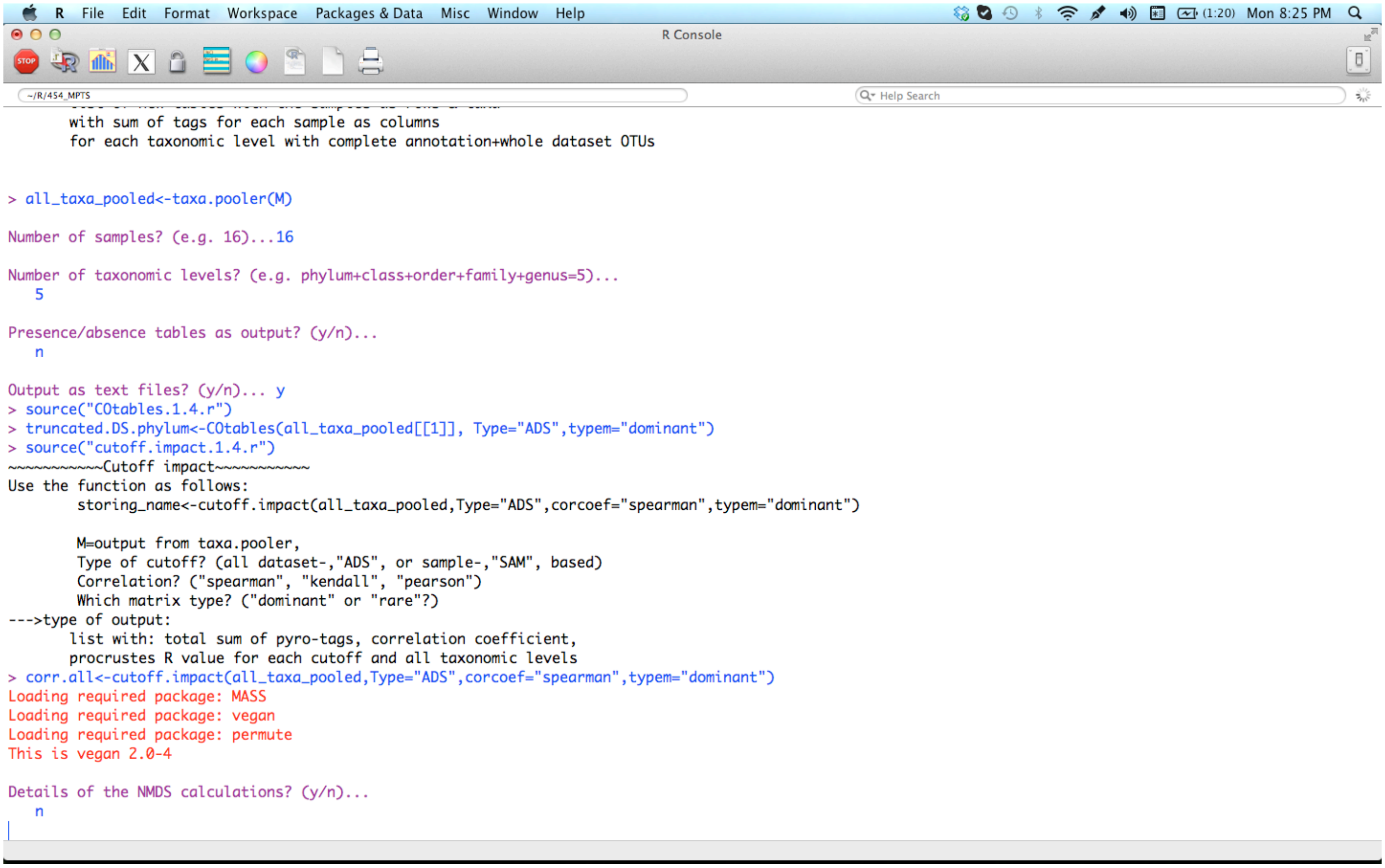
```
corr.all<-cutoff.impact(all_taxa_pooled,Type="ADS",corcoef="spearman",typem="dominant")
```

With:

- The input, “all_taxa_pooled” here, should be a list (e.g. the output from the taxa.pooler);
- Type = Type of cutoff: all dataset-, ”ADS”, or sample-, ”SAM”, based;
- corcoef = the chosen non-parametric correlation coefficient: “spearman” (“pearson” for a linear coefficient);
- typem = choice of the fraction of the matrix to work on: “dominant” types or “rare” types.

Also, if one does not need to see the details of the NMDS calculations, some computing time might be saved by answering no (“n”) to the following question:

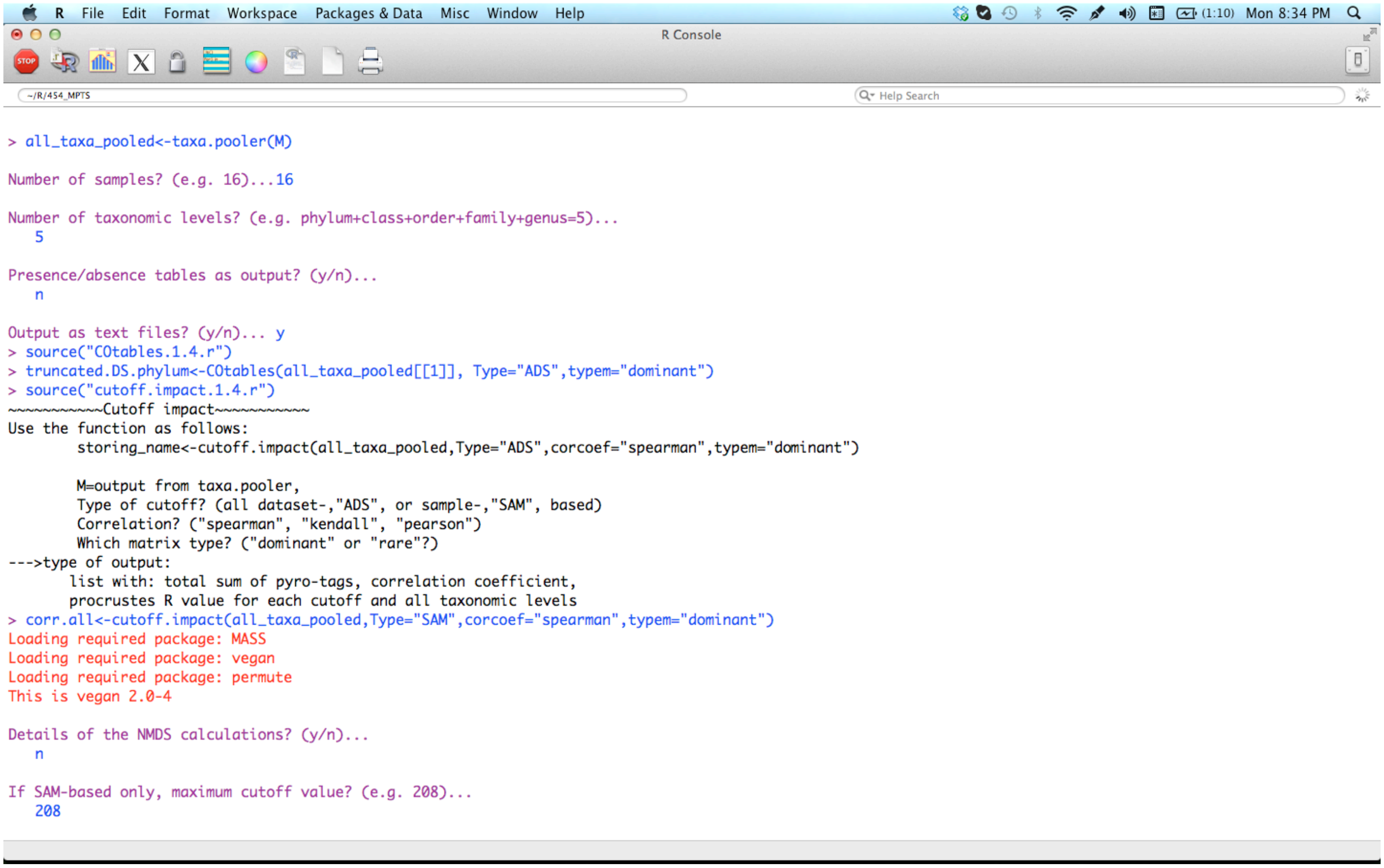
```
Details of the NMDS calculations? (y/n)...
```


R File Edit Format Workspace Packages & Data Misc Window Help
(1:20) Mon 8:25 PM
R Console
~/R/454_MPTS
Help Search

```
        with sum of tags for each sample as columns
        for each taxonomic level with complete annotation+whole dataset OTUs


> all_taxa_pooled<-taxa.pooler(M)

Number of samples? (e.g. 16)...16

Number of taxonomic levels? (e.g. phylum+class+order+family+genus=5)...
   5

Presence/absence tables as output? (y/n)...
   n

Output as text files? (y/n)... y
> source("COtables.1.4.r")
> truncated.DS.phylum<-COtables(all_taxa_pooled[[1]], Type="ADS",typem="dominant")
> source("cutoff.impact.1.4.r")
~~~~~~~~~~~Cutoff impact~~~~~~~~~~~
Use the function as follows:
        storing_name<-cutoff.impact(all_taxa_pooled,Type="ADS",corcoef="spearman",typem="dominant")

        M=output from taxa.pooler,
        Type of cutoff? (all dataset-,"ADS", or sample-,"SAM", based)
        Correlation? ("spearman", "kendall", "pearson")
        Which matrix type? ("dominant" or "rare"?)
--->type of output:
       list with: total sum of pyro-tags, correlation coefficient,
       procrustes R value for each cutoff and all taxonomic levels
> corr.all<-cutoff.impact(all_taxa_pooled,Type="ADS",corcoef="spearman",typem="dominant")
Loading required package: MASS
Loading required package: vegan
Loading required package: permute
This is vegan 2.0-4

Details of the NMDS calculations? (y/n)...
   n
|
```

If sample-based cutoff chosen, the following question will appear:

```
If SAM-based only, maximum cutoff value? (e.g. 208)...
```


R File Edit Format Workspace Packages & Data Misc Window Help
(1:10) Mon 8:34 PM
R Console
~/R/454_MPTS
Help Search

```
> all_taxa_pooled<-taxa.pooler(M)

Number of samples? (e.g. 16)...16

Number of taxonomic levels? (e.g. phylum+class+order+family+genus=5)...
   5

Presence/absence tables as output? (y/n)...
   n

Output as text files? (y/n)... y
> source("COtables.1.4.r")
> truncated.DS.phylum<-COtables(all_taxa_pooled[[1]], Type="ADS",typem="dominant")
> source("cutoff.impact.1.4.r")
~~~~~~~~~~~Cutoff impact~~~~~~~~~~~
Use the function as follows:
        storing_name<-cutoff.impact(all_taxa_pooled,Type="ADS",corcoef="spearman",typem="dominant")

        M=output from taxa.pooler,
        Type of cutoff? (all dataset-,"ADS", or sample-,"SAM", based)
        Correlation? ("spearman", "kendall", "pearson")
        Which matrix type? ("dominant" or "rare"?)
--->type of output:
       list with: total sum of pyro-tags, correlation coefficient,
       procrustes R value for each cutoff and all taxonomic levels
> corr.all<-cutoff.impact(all_taxa_pooled,Type="SAM",corcoef="spearman",typem="dominant")
Loading required package: MASS
Loading required package: vegan
Loading required package: permute
This is vegan 2.0-4

Details of the NMDS calculations? (y/n)...
   n

If SAM-based only, maximum cutoff value? (e.g. 208)...
   208
```

The output is a list of tables with the different assigned cutoffs (all dataset- or sample-based) by the sum of each truncated table, the correlation value between the original table and the truncated table, and the Procrustes value between the non-metric multidimensional scaling (NMDS) from the original table and the truncated table for all taxonomic levels.

!!! This script requires some time and a certain computing power (10 min of calculations for the example matrix with 1,000 OTUs on an Intel Pentium 4)

In order to obtain similar figures as Fig. 3 in the article, another script is needed:

```
source("cutoff.impact.fig.1.4.r")

output.all<-cutoff.impact.fig(corr.all)
```

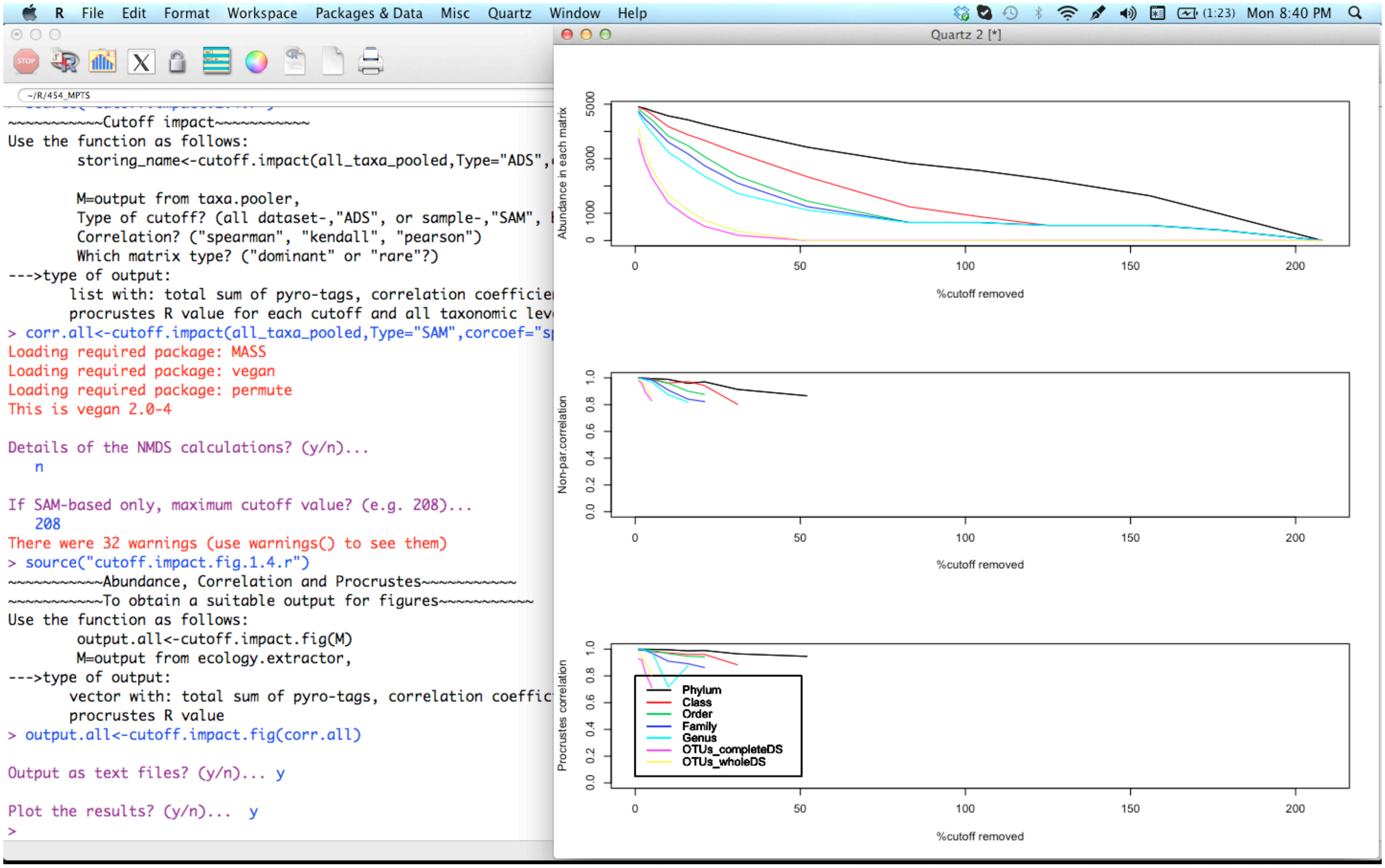
With the input, "corr.all" here, as a list (e.g. the output from the cutoff.impact) and you can choose to have the output as a text file:

```
Output as text files? (y/n)...
```

Then three files will appear in the directory:
- "abundance.txt"
- "non-par.correlation.txt"
- "procrustes.txt"

And they can be further used to produce figures with Microsoft Excel for example.
Or you can also choose if you want to directly plot the data:

```
Plot the results? (y/n)...
```



```
~~~~~~~~~~~Cutoff impact~~~~~~~~~~~
Use the function as follows:
        storing_name<-cutoff.impact(all_taxa_pooled,Type="ADS",

        M=output from taxa.pooler,
        Type of cutoff? (all dataset-,"ADS", or sample-,"SAM",
        Correlation? ("spearman", "kendall", "pearson")
        Which matrix type? ("dominant" or "rare"?)
--->type of output:
       list with: total sum of pyro-tags, correlation coefficie
       procrustes R value for each cutoff and all taxonomic lev
> corr.all<-cutoff.impact(all_taxa_pooled,Type="SAM",corcoef="s
Loading required package: MASS
Loading required package: vegan
Loading required package: permute
This is vegan 2.0-4

Details of the NMDS calculations? (y/n)...
   n

If SAM-based only, maximum cutoff value? (e.g. 208)...
   208
There were 32 warnings (use warnings() to see them)
> source("cutoff.impact.fig.1.4.r")
~~~~~~~~~~~Abundance, Correlation and Procrustes~~~~~~~~~~~
~~~~~~~~~~~To obtain a suitable output for figures~~~~~~~~~~~
Use the function as follows:
        output.all<-cutoff.impact.fig(M)
        M=output from ecology.extractor,
--->type of output:
       vector with: total sum of pyro-tags, correlation coeffic
       procrustes R value
> output.all<-cutoff.impact.fig(corr.all)

Output as text files? (y/n)... y

Plot the results? (y/n)...  y
>
```

4. Load and run the scripts: ecological patterns

4.1. Variation partitioning at several cutoff levels for all taxonomic levels

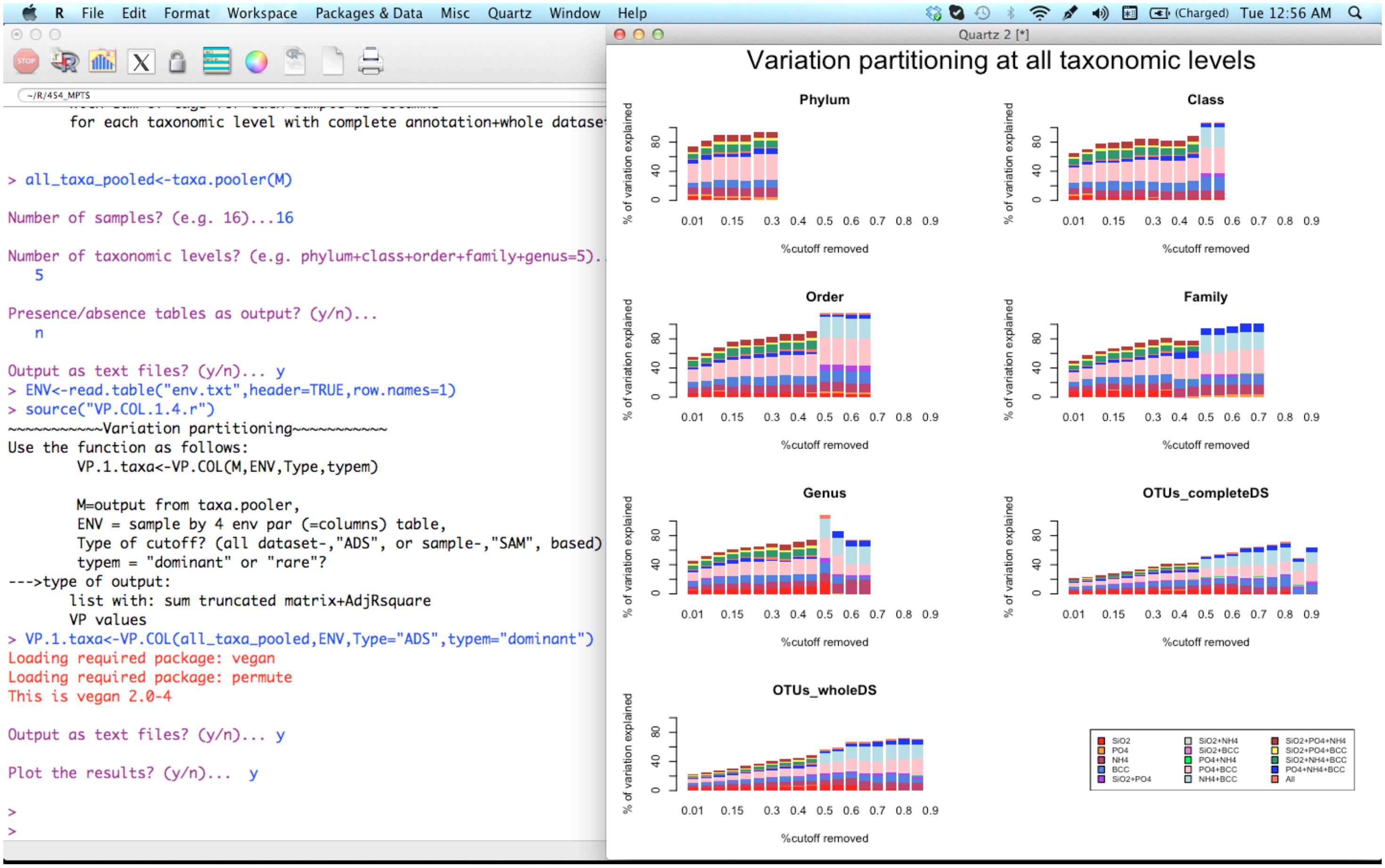
Load the environmental table with samples as rows and environmental parameters as columns (here the script is written for an environmental table with 4 columns) and the script:

```
ENV<-read.table("env.txt",header=TRUE,row.names=1)
source("VP.COL.1.4.r")
```

Some explanations about the function are then appearing. Store the output in the R workspace under a name of your choice, for instance:

```
VP.1.taxa<-VP.COL(all_taxa_pooled,ENV,Type="ADS",typem="dominant")
```

With:
- The input, "all_taxa_pooled" here, is the output from the taxa.pooler;
- ENV = the environmental table;
- Type = Type of cutoff: all dataset-,"ADS", or sample-,"SAM", based.



```
        for each taxonomic level with complete annotation+whole datase


> all_taxa_pooled<-taxa.pooler(M)

Number of samples? (e.g. 16)...16

Number of taxonomic levels? (e.g. phylum+class+order+family+genus=5)..
   5

Presence/absence tables as output? (y/n)...
   n

Output as text files? (y/n)... y
> ENV<-read.table("env.txt",header=TRUE,row.names=1)
> source("VP.COL.1.4.r")
~~~~~~~~~~~Variation partitioning~~~~~~~~~~~
Use the function as follows:
        VP.1.taxa<-VP.COL(M,ENV,Type,typem)

        M=output from taxa.pooler,
        ENV = sample by 4 env par (=columns) table,
        Type of cutoff? (all dataset-,"ADS", or sample-,"SAM", based)
        typem = "dominant" or "rare"?
--->type of output:
       list with: sum truncated matrix+AdjRsquare
       VP values
> VP.1.taxa<-VP.COL(all_taxa_pooled,ENV,Type="ADS",typem="dominant")
Loading required package: vegan
Loading required package: permute
This is vegan 2.0-4

Output as text files? (y/n)... y

Plot the results? (y/n)...  y

>
>
```

The output is a list of two tables, for each taxonomic level:
- one with the partition of the variation by the different assigned cutoffs (all dataset- or sample-based);
- one with the different assigned cutoffs by the sum of each truncated table, and the adjusted R square.

You can choose if you want the output as a text file:

```
Output as text files? (y/n)...
```

Then two files x the number of taxonomic level will appear in the directory:
- "taxonomiclevel.VarPart.txt"
- "taxonomiclevel.sum.adjRsq.txt"

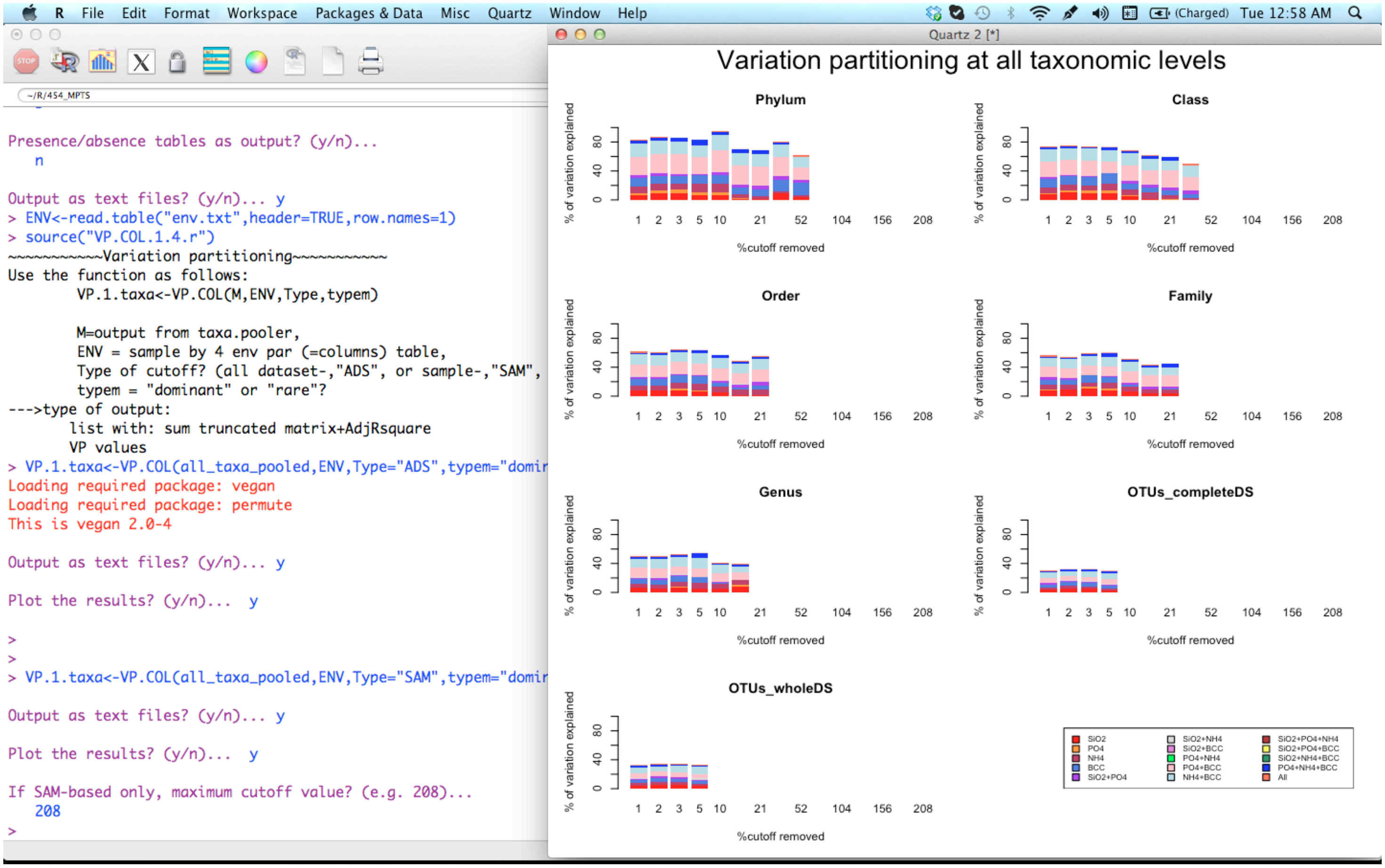
And then can be further used to produce figures with Microsoft Excel for example.

Or you can also choose if you want to plot the data:

```
Plot the results? (y/n)...
```

If sample-based cutoff chosen, the following question will appear:

```
If SAM-based only, maximum cutoff value? (e.g. 208)...
```



```
Presence/absence tables as output? (y/n)...
   n

Output as text files? (y/n)... y
> ENV<-read.table("env.txt",header=TRUE,row.names=1)
> source("VP.COL.1.4.r")
~~~~~~~~~~~Variation partitioning~~~~~~~~~~~
Use the function as follows:
        VP.1.taxa<-VP.COL(M,ENV,Type,typem)

        M=output from taxa.pooler,
        ENV = sample by 4 env par (=columns) table,
        Type of cutoff? (all dataset-,"ADS", or sample-,"SAM",
        typem = "dominant" or "rare"?
--->type of output:
       list with: sum truncated matrix+AdjRsquare
       VP values
> VP.1.taxa<-VP.COL(all_taxa_pooled,ENV,Type="ADS",typem="domir
Loading required package: vegan
Loading required package: permute
This is vegan 2.0-4

Output as text files? (y/n)... y

Plot the results? (y/n)...  y

>
>
> VP.1.taxa<-VP.COL(all_taxa_pooled,ENV,Type="SAM",typem="domir

Output as text files? (y/n)... y

Plot the results? (y/n)...  y

If SAM-based only, maximum cutoff value? (e.g. 208)...
   208
>
```

4.2. Calculation of correlation coefficients for the environmental parameters (for the first RDA axis)

Load the following script:

```
source("corrcoeff.ENV.1.4.r")
```

However, a whole “automatic” script could not be realized as the R software can present some scoping problems. Instead, you may copy and paste the following lines (here an example for the original table at the OTU level with the whole dataset; we work here on the 7th element of the VP.1.taxa output):

- for all dataset-based cutoffs:

```
#create a matrix to store corrcoeff output at all 21 cutoffs
corrcoeff.table.ADS<-matrix(NA,21,5)
row.names(corrcoeff.table.ADS)<-c(paste("CO_",c(0.01,seq(0.05,0.95,by=0.05),0.99),sep=""))
colnames(corrcoeff.table.ADS)<-c("Sum",paste("RDA1.",colnames(ENV),sep=""))

#store the original matrix
#7: whole dataset at the OTU level
#3: where the cutoff matrices are
OTU.ADS<-VP.1.taxa[[c(7,3)]]

#application of corrcoeff at all cutoffs
SPE<-OTU.ADS[[1]];corrcoeff.table.ADS[1,]<-corrcoeff(SPE,ENV);rm(SPE)
SPE<-OTU.ADS[[2]];corrcoeff.table.ADS[2,]<-corrcoeff(SPE,ENV);rm(SPE)
.
.
.
SPE<-OTU.ADS[[21]];corrcoeff.table.ADS[21,]<-corrcoeff(SPE,ENV);rm(SPE)

#application of corrcoeff on the original table with no cutoff
SPE<-all_taxa_pooled[[7]]
corrcoeff.table.ADS.orig<-corrcoeff(SPE,ENV)
row.names(corrcoeff.table.ADS.orig)<-c("CO_1")
corrcoeff.table.ADS<-rbind(corrcoeff.table.ADS,corrcoeff.table.ADS.orig)

#output as a text file
write.table(corrcoeff.table.ADS,"corrcoeff.table.ADS.txt",quote=FALSE)
```

- for sample-based cutoffs:

```
#create a matrix to store corrcoeff output at all 15 cutoffs
corrcoeff.table.SAM<-matrix(NA,15,5)
row.names(corrcoeff.table.SAM)<-
c(paste("CO_",c(1,2,3,5,10,15,20,30,55,80,105,130,155,180,208),sep=""))
colnames(corrcoeff.table.SAM)<-c("Sum",paste("RDA1.",colnames(ENV),sep=""))

#store the original matrix
#7: whole dataset at the OTU level
#3: where the cutoff matrices are
OTU.SAM<-VP.1.taxa[[c(7,3)]]

#application of corrcoeff at all cutoffs
SPE<-OTU.SAM[[1]];corrcoeff.table.SAM[1,]<-corrcoeff(SPE,ENV);rm(SPE)
SPE<-OTU.SAM[[2]];corrcoeff.table.SAM[2,]<-corrcoeff(SPE,ENV);rm(SPE)
.
.
.
SPE<-OTU.SAM[[15]];corrcoeff.table.SAM[15,]<-corrcoeff(SPE,ENV);rm(SPE)
```

```
#output as a text file
write.table(corrcoeff.table.SAM,"corrcoeff.table.SAM.txt",quote=FALSE)
```

4.3. Calculation of the significance of the whole variation partitioning model and the impact of the pure environmental parameters

Load the following script:

```
source("signif.1.4.r")
```

However, a whole "automatic" script could not be realized as the R software can present some scoping problems. Instead, you may copy and paste the following lines (here an example for the original table at the OTU level with the whole dataset; we work here on the 7th element of the VP.1.taxa output):

- for all dataset-based cutoffs:

```
#create a matrix to store signif output at all 21 cutoffs
signif.table.ADS<-matrix(NA,21,5)
row.names(signif.table.ADS)<-c(paste("CO_",c(0.01,seq(0.05,0.95,by=0.05),0.99),sep=""))
colnames(signif.table.ADS)<- c("whole.sig","ENV1.sig","ENV2.sig","ENV3.sig","ENV4.sig")

#store the original matrix
#7: whole dataset at the OTU level
#3: where the cutoff matrices are
OTU.ADS<-VP.1.taxa[[c(7,3)]]

#application of signif at all cutoffs
SPE<-OTU.ADS[[1]];signif.table.ADS[1,]<-signif(SPE,ENV);rm(SPE)
SPE<-OTU.ADS[[2]];signif.table.ADS[2,]<-signif(SPE,ENV);rm(SPE)
.
.
.
SPE<-OTU.ADS[[21]];signif.table.ADS[21,]<-signif(SPE,ENV);rm(SPE)

#application of signif on the original table with no cutoff
SPE<-all_taxa_pooled[[7]]
signif.table.ADS.orig<-signif(SPE,ENV)
row.names(signif.table.ADS.orig)<-c("CO_1")
signif.table.ADS<-rbind(signif.table.ADS, signif.table.ADS.orig)

#output as a text file
write.table(signif.table.ADS,"signif.table.ADS.txt",quote=FALSE)
```

- for sample-based cutoffs:

```
#create a matrix to store signif output at all 15 cutoffs
signif.table.SAM<-matrix(NA,15,5)
row.names(signif.table.SAM)<-
c(paste("CO_",c(1,2,3,5,10,15,20,30,55,80,105,130,155,180,208),sep=""))
colnames(signif.table.SAM)<- c("whole.sig","ENV1.sig","ENV2.sig","ENV3.sig","ENV4.sig")

#store the original matrix
#7: whole dataset at the OTU level
#3: where the cutoff matrices are
OTU.SAM<-VP.1.taxa[[c(7,3)]]

#application of signif at all cutoffs
SPE<-OTU.SAM[[1]];signif.table.SAM[1,]<-signif(SPE,ENV);rm(SPE)
SPE<-OTU.SAM[[2]];signif.table.SAM[2,]<-signif(SPE,ENV);rm(SPE)
```

```
.
.
.
SPE<-OTU.SAM[[15]];signif.table.SAM[15,]<-signif(SPE,ENV);rm(SPE)

#output as a text file
write.table(signif.table.SAM,"signif.table.SAM.txt",quote=FALSE)
```

5. Save your R workspace

```
save.image("MultiCoLA.RData")
```

All variables will then be saved and then available to work on them without running all the scripts again.

How to cite the script?
Gobet, A., Quince, C., and Ramette, A. 2010. **Multivariate Cutoff Level Analysis (MultiCoLA) of Large Community Datasets.** *Nucl. Acids Res.*

Comments and corrections are always welcome. Please address email correspondence to:
Angélique Gobet: angeliquegobet@gmail.com
or
Alban Ramette: aramette@mpi-bremen.de