**Workflow of the second Shiny app.**

**Name : DRomics\_multilevels**

**Examples :**

* Scenedesmus, triclosan (cf. triclosanSVcontigres.txt, triclosanSVcontigannot.txt, triclosanSVmetabres.txt, triclosanSVmetabannot.txt – faire un essai en ajoutant des lignes avec des id non annotés)
* Nicolas Creusot data ? PB because it was not aimed to do metabolomics
* PICT study when it will be published
* Another data with metabolomics time (Mechthild)
* IRSN data zebra fish at different times (sujet de M1 avril-mai 2022)

**Step 1**

**Import and merge of DRomics results and annotation data**

First input : the number of experimental levels e.g. different molecular levels (transcriptomics, metabolomis, …) , different experimental time points or different biological models (different species, different experimental settings), …

**The name of the column that gives the id in annotation file is asked for only for the first level : the name of the other column is stored and a message printed to say that the column used for the annotation at the other levels should always have this name.**

**We should clearly indicate that each file for annotation data must have only two columns**

Check that there is no redundancy (eliminate equivalent rows)

Step to repeat (+ button or numerical input with the number of results ):

Merge for each results data frame form the workflow, corresponding to a condition (PICT or not), a molecular level (metabolo or transcripto), a time, … -> identified by an **experimental level**

**We can consider that ten levels is a big max !**

Input :

* the name of the experimental level (string)
* the results data frame from the DRomics workflow (DRomics\_output ) : .txt
* the annotation data frame that may be longer (one item with more than one **annotation**) or shorter (items without any annotation) than DRomics\_output with only two columns (only one annotation) : .txt (check in defensive programming that there is no duplicated lines and remove duplicate lines with a warning in case)

Action :

* merge on id + add of a column named “**experimental\_level**” -> **extendedres\_ith**
* At the end -> rbind of all the extendedres\_ith -> extendedres

The app. must check that the the different data to rbind have the same columns and that there is at least one BMD ‘BMD.zSD of BMD.xfold. **Is it necessary to define a function in the package to do those steps ? To think… Possibility also to make enrichment test in case of annotated organisms ?**

If one BMD is missing, the corresponding button must be inactive in the following

Output of page 1 -> for the names of the levels, just put the given names (no paste needed)

Show the first rows of each explevel and not only the first rows

**Step 2**

**Title of the page : trend and sensitivity plots**

Numerical input : minimal number of items (default 3, from 1 to 10 ) to keep the item for one factor level

Do we keep the item in the other factor levels if it appears more than min\_nb\_of\_items in another factor level ? YES or give the choice. For the moment we calculate the max nb of items per pathway per experimental level

METTRE UNE FONCTION dans le package pour cela ? selectgroups (en cours)

Pour le moment j’ai mis dans le share une fonction selectgroups qui fait le job, séparément par niveau expérimental. L’amélioration qu’on pourrait faire à mon avis serait de pouvoir mettre une vecteur dans nitemsmin, pour pouvoir sélectionner sur un nb différent en multi-omiques, par exemple si on a beaucoup plus d’items qui sortent en transcripto qu’en métabolo.

Is it possible if one level is specified, to keep the inputs when cjanging the nb of levels ?

Change colorby in colorby for sensitivityplot, which must appear only if tehre is more than one explevel

selectgroups()

PB with quantiles – write a message to say that quantiles on low number of items are not reliable

Radio- button : Ordering of the annotations : **alphabetic order** (must be ordered within Shiny in decreasing alphabetic order before calling sensitivityplot and use ECDF\_plot = FALSE), or order of apparition (chosen by the user – NOT POSSIBLE FROM a.txt File) in the input, **ordered by total number of items in all the experimental levels (or if only one experimental level : ordered by number of items**) (must be ordered in increasing order of number of items within Shiny before calling sensitivityplot and use ECDF\_plot = FALSE), **and if only one experimental level : ordered by BMD summary value** (call to sensitivityplot with ECDF\_plot = TRUE)

Try to implement a widget to change interactively the order of groups

Action :

* Filtering of annotations to plot
* Trendplot and sensitivity

trendplot()

* (extendedres)
* (group = annotation)
* (facetby = label)
* add.color = TRUE

sensitivityplot()

* (extendedres)
* BMD type = (“zSD”, “xfold”)
* BMD summary = c(“first.quartile”, “median”, “median.and.IQR”)
* (group = annotation)
* colorby = label or nothing (if there is more than one expe level, colorby is automatically put to expelev or not – ask Elise if it could be interesting to group the levels in the sensitivity plot)
* if no colorby ECDF\_plot = TRUE
* BMD\_log\_transfo = FALSE

Each with possible exportation of the plot

* Step 2 remplacer BMDmax par maximum for the BMD summary et ce serait bien de remplacer "Selection of items to plot" par " Selection of annotation groups to plot" et pour le blabla associé on pourrait mettre : "to limit the number of annotation groups you can use the thresholds on the number of items representing the group and/or the BMDsummary value of the group."
* Attention en step 2 même si on a un seul niveau experimental "ordered by BMD summary value" ne fonctionne pas

**Step 3**

**Title : BMD plots**

Put again the input of page 2 with the last value proposed by the user as the default value

Optional focus on some annotations : mark the chosen annotations (by default none are marked) with a button to mark all

**Common arguments (in bold)** so only one question for both functions, except add.CI and colorby

bmdplot()

* **(Extendedres)**
* **BMD type = (“zSD”, “xfold”)**
* add.CI = FALSE
* **facetby (for columns) = annotation or label or none**
* **facetby2 (for rows) = label or annotation or none (conditioned by facetby)**
* colorby (select input in the columns of extendedres)
* **shapeby** (select input in the columns of extendedres) (not proposed if add.label = TRUE)
* **BMD\_log\_transfo = FALSE**
* **add.label = FALSE**

bmdplotwithgradient()

* **(extendedres)**
* **BMD type = (“zSD”, “xfold”)**
* **facetby (for columns) = annotation or label or none**
* **facetby2 (for rows) = label or annotation or none (conditioned by facetby)**
* **BMD\_log\_transfo = FALSE**
* **add.label = FALSE**
* **shapeby** (select input in the columns of extendedres) (not proposed if add.label = TRUE)

When there is only one experimental level, it should be nice if we could use only one of the facetby argument (not facetby2) to have the different plots in more than one row (see page “Metabolomics responses of S. vauolatus exposed to triclosan – BMD plot with confidence intervals” of the training course

**Step 4**

**Title Curves plot**

Put again the input of page 3 with the last value proposed by the user as the default value

Optional focus on some annotations : mark the chosen annotations (by default the ones marked on page 3) with a button to mark all

curvesplot()

* (extendedres)
* Min and max doses to define the range for DR curves
* facetby (for columns) = annotation or label or none (by default the one in page 3)
* facetby2 (for rows) = label or annotation or none (conditioned by facetby) (by default the one in page 3)
* colorby (select input in the columns of extendedres)
* dose\_log\_transfo = (by default the one in page 3 for BMD\_log\_transfo)

For the plot the option for faceting dose yet works (always one plot even if more than one annotations for one expe level), but the same comment as for Step 3 may stand. And it would also be perhaps nice to enable the facet by label if there is only one exp level and one chosen annotation

**Page 5**

The script – With examples of scripts corresponding to the workflow, see if some functions must be added in the package to simplify the code and/or give some useful functions in the package