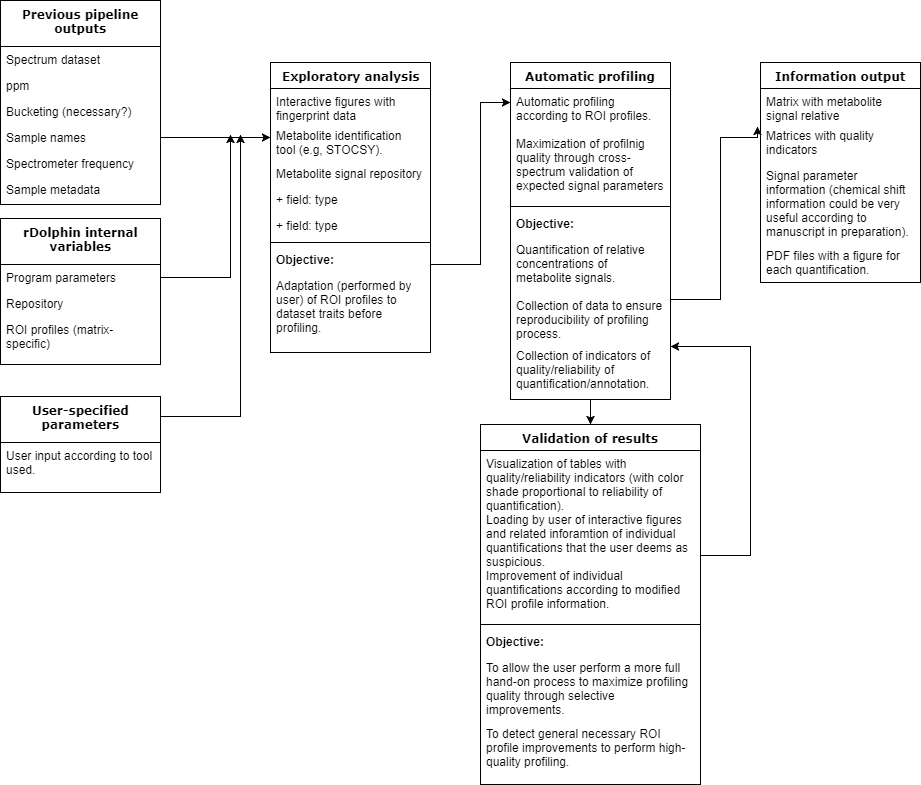
Integration rDolphin into Phenomenal

# Outline of document:

rDolphin is an R package which provides tools for the automatic profiling of 1H 1D NMR datasets. It also provides tools for the previous exploratory analysis of the dataset to profile so the user can adapt the profiling to the dataset properties. In addition, it provides options to validate the quality of the profiling performed and, if necessary, to update individual metabolite signal quantifications.

I have prepared a general diagram of the kind of inputs necessary to provide, the steps of the package workflow for atuomatic profiling and the kinds of output generated.



During the hackathon, I will prepare a live demo of the capabilities of each tool in the diagram to potentially help in the Phenomenal workflow.

On next sections, I have summed up the inputs and outputs necessary for each tool.

# Exploratory analysis

## Exploratory analysis figures:

rDolphin provides two kinds of exploratory visualization to ease as much as possible the evaluation of the main phenomena to monitor in the spectrum dataset analyzed.

* exemplars\_plot: if reduces the spectrum\_dataset to a set of around 10 spectra representative of the variability present in the dataset. It clusters the spectrum dataset into ten groups and selects the best exemplar of each spectrum. Then, it provides an interactive figure of these spectra.
* median\_plot: It calculates the median spectrum of each sample type according to metadata inputted. Then, it provides an interactive figure of these spectra.

In addition, fingerprint data is provided below these spectra. This fingeprint data helps the user evaluate possible signals difficult to profile but which might be the most informative ones in the phenomenon analyzed.

### Necessary input:

* imported\_data: list with at least:
* dataset - Provided from nmrProcFlow (each row a spectrum)
* ppm : provided from NMRProcFlow
* Experiments: Info coming from nmrProcFlow output (equivalent to ‘samples’).
* Metadata: provided by nmrML, ISA?

### Output:

Plotly interactive figure.

## Statistical heterospectroscopy:

Statistical method to develop metabolite identification through STOCSY or RANSY (note: I’m developing the [improvement](https://danielcanueto.shinyapps.io/dependence_measures/) of this technique; if desired, it could already be added to the workflow). Through the ‘identification\_tool’ function.

### Necessary input:

* dataset - Provided from nmrProcFLow
* ppm : provided from NMRProcFlow
* Driver\_peak\_edges: left and right edges (in ppm) of the driver peak to use for metabolite identification. Provided by the user.
* Method: STOCSY with Pearson correlation, STOCSY with Spearman correlation, RANSY. Provided by the user.

### Output:

Plotly interactive plot to analyze regions of the spectrum related to the driver peak analyzed.

## HMDB metabolite signal repository:

Data.frame provided by rDolphin with relevant information for each metabolite signal reported in the HMDB. rDolphin users can specify the biofluid analyzed so to ensure the metabolites annotated are typical in the biofluid analyzed and no wrong metabolite identification is performed.  
  
In the rDolphin Shiny GUI, the user can play with an interactive ‘DT’ table to filter signals by the spectrum region, the kind of multiplet and the number of times (and mean concentration) the signal has been reported in HMDB in the biofluid analyzed. Furthermore, in case of doubt, the user can click a button to go directly to the HMDB metabolite webpage to get additional information. Ideally, this interactivity should be maintained in the Phenomenal version so to maximize the information added.

## Other additional options which might be enumerated:

Interactive figure of how profiling looks like when performed in a model spectrum.

# Automatic profiling

## Automatic profiling of a dataset.

Through the ‘automatic\_profiling’ function.

### Necessary input:

* imported\_data: list with at least:
* dataset - matrix (each row a spectrum) provided from nmrProcFLow
* ppm : provided from NMRProcFlow
* buck\_step: constant buck step, can be calculated from ppm info from NMRProcFlow. If not constant, we have to talk about how to approach it.
* Experiments: Info coming from nmrProcFlow output
* Freq: Info coming from nmrML?
* Program\_parameters: Internal macro from rDolphin package with lots of parameters that can be tuned in order to adapt integration/deconvolution to ideally desired characteristics. Just for very advanced users (basically me).
* ROI\_data: data frame provided by the user, with the ROI profile information necessary to perform automatic profiling. It is imported from csv, but it might be by default given by Phenomenal Galaxy according to the matrix specified (and then modified by the user to adapt it to the spectrum dataset traits).
* Optimization: by default TRUE. It maximizes the quality of automatic profiling through the post-profiling evaluation of the internal consistency of the signal parameters’ information. Workflow of this maximization in manuscript preparation.
* Spectra\_to\_profile: by default NA, and all spectra are profiled. If a vector of indexes is inputted, only the spectra from imported\_data$Experiments with the indexes specified are profiled.

### Output:

* Final\_output: list with:
* Quantification: matrix with relative concentration estimated for each metabolite signal.
* Fitting\_error: matrix with fitting error (normalized RMSE) estimated for each fitting of metabolite signal.
* Signal\_area\_ratio: matrix with % of total spectrum region area explained by the metabolite signal.
* Half\_bandwidth: matrix with half bandwidth estimated for each metabolite signal.
* Chemical\_shift: matrix with chemical shift estimated for each metabolite signal.
* Intensity: matrix with intensity estimated for each metabolite signal.
* Reproducibility\_data: list with necessary data to reproduce quantifications so to to validate them and maybe to improve them (e.g., through loading figures and associated quality information, outputting figures for each quantification, calculating the difference between the expected chemical shift and the estimated chemical shift).

# Validation of results

## Profiling validation tables:

The final\_output list provided by the ‘automatic\_profiling’ function outputted relevant inforamtion about the quality of the performed profiling. This information can be evaluated through the ‘validation’ function. This function processes the ‘final\_output’ information to generate values indicative of possible inaccurate quantifications and annotations. The, it generates an interactive ‘DT’ table with a color shade that helps identify the most suspicious quantifications.

The current processing workflow has a limited effectiveness because of the computing time demands of typical CPUs. Depending of Phenomenal H2020 computing potential, the effectiveness might be maximized.

### Necessary input:

* final\_output: list with:
* Quantification: matrix with relative concentration estimated for each metabolite signal.
* Fitting\_error: matrix with fitting error (normalized RMSE) estimated for each fitting of metabolite signal.
* Signal\_area\_ratio: matrix with % of total spectrum region area explained by the metabolite signal.
* Half\_bandwidth: matrix with half bandwidth estimated for each metabolite signal.
* Chemical\_shift: matrix with chemical shift estimated for each metabolite signal.
* Intensity: matrix with intensity estimated for each metabolite signal.
* alarm\_matrix: by default NA. Depending on processing capabilities, I would explain what this variable consists of.
* validation\_type: which kind of information will be visualized by the user (1: fitting error, 2: signal total area ratio, 3: difference expected-obtained chemical shift, 4: difference expected-obtained half bandwidth, 5: difference expected-obtained intensity)

### Output:

A list with these components:

* alarm\_matrix=list with these five kinds of information:
* fitting\_error: matrix with fitting error (normalized RMSE) estimated for each fitting of metabolite signal.
* signal\_area\_ratio: matrix with % of total spectrum region area explained by the metabolite signal.
* Half\_bandwidth: Difference between expected and obtained half bandwidth.
* Chemical\_shift: Difference between expected and obtained chemical shift.
* Intensity: Difference between expected and obtained intensity.
* Shown\_matrix=matrix with concrete information to be visualized according to option selected on ‘validation\_type’.
* Brks and clrs: information necessary to provide a different color shade for each signal area quantification according to deviation from optimal values.

With this list, a ‘DT’ interactive table is created.

## Other additional options which might be enumerated:

Loading of information of individual quantifications.

Individual profiling: lineshape fitting of individual signals. Especially useful to improve individual quantifications after checking quantifications that might be improved.

# Information output

## Output of ‘final\_output’ collected information:

The ‘write\_info’ function generates a CSV file for each element of the ‘final\_output’ list with the information generated during profiling. In addition, it associates each metabolite signal with an HMDB code so information can be easily integrated to a complete metabolomic workflow.

### Necessary input:

* export\_path: file path where csv files will be outputted.
* final\_output: list with the information generated during profiling.
* ROI\_data: data frame with the HMDB code information necessary. It is imported from csv provided by user.

### Output:

CSV files.

## 

## Figures for each signal area quantification:

To validate the quality of the profiling performed. Associated information of fitting error and signal to total area ratio. Through ‘write\_plots’ function.

### Necessary input:

* export\_path: file path where to export output.
* final\_output: list with the information generated during profiling.
* reproducibility\_data: list with necessary data to reproduce quantifications performed during profiling.
* signals\_to\_plot: by default NA. If indexes are specified, only metabolite signals specified are profiled.

### Output:

‘Plots’ folder with a PDF for each metabolite signal with all quantifications performed.