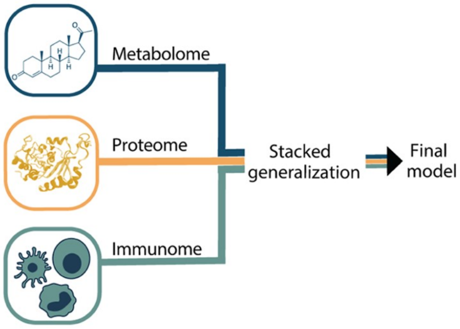
7 ExpoMo module

7.1 Application domain

Multi-omics research is a method of exploring the interactions between multiple substances in biological systems, including genomics, proteomics, metabolomics, etc., which together affect the phenotype and traits of organisms. With the rapid development of high-throughput sequencing and multiomics, we can develop more comprehensive predictive models and identify important features by integrating multi-omic data, which facilitates a deeper understanding of the complexity of biology. ***exmo*** package is designed to integrate the multi-omic data. It mainly aims to construct various stacked generalization models to predict the response or probability of outcome incidence, as well as providing the statistical explanation. Additionly, the package can provide visualization plots with high quality of the final calculation results to make it easier for users to understand.

7.2 Theory

The theory of the package ***exmo*** is based on "Integrated trajectories of the maternal metabolome, proteome, and immunome predict labor onset". And the code for machine learning and visualization of the results are based on "mlr3" package and "ggplot2" package.



Stelzer IA, Ghaemi MS, Han X, et al. Integrated trajectories of the maternal metabolome, proteome, and immunome predict labor onset. Sci Transl Med. 2021;13(592):eabd9898. doi:10.1126/scitranslmed.abd9898

7.3 Work pipeline

Initialize package

Make sure that the required packages is already installed.

```
# The following two packages should be installed in advance
# devtools::install_github("ExposomeX/exmo", force = TRUE)
# devtools::install_github("ExposomeX/extidy", force = TRUE)

# library(exmo)
# library(extidy)
library(tidyverse)

# devtools::install_github("ExposomeX/exposomex", force = TRUE)
library(exposomex)
```

The first step,you need to initialize the environment using "InitMO".

```
res = InitMO()
```

Here, we can see that the returned value "res" is an R6 object. It contains an unique program ID of res$PID (e.g., "100737GJMWJA"), which is random generated by the system. Users need use it in the following step for further data process.

Upload data

```
res1 = LoadMO(PID = res$PID,
              UseExample="example#1")
```

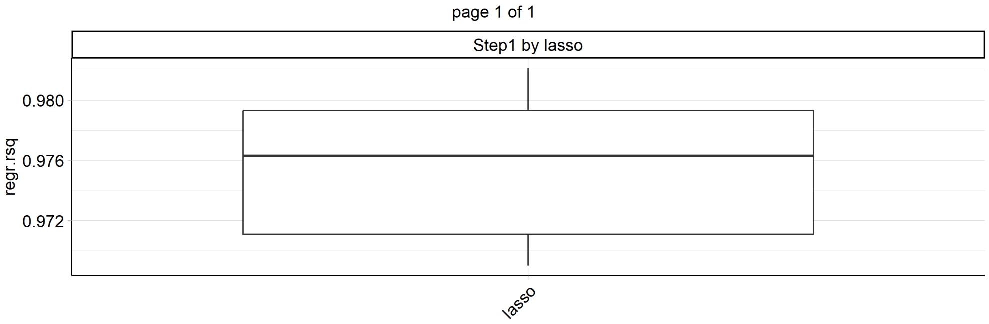
Tidy data

```
res2 = TransImput(PID = res$PID,
                  Group="T",
                  Vars="all.x",
                  Method="lod")

res3 = DelNearZeroVar(PID = res$PID)

res4 = DelMiss(PID = res$PID)

res5 = TransType(PID = res$PID,
                 Vars="all.x",
                 To="numeric")

res6 = TransClass(PID = res$PID,
                  Group="F",
                  Vars="C1",
                  LevelTo="4")

res7 = TransScale(PID = res$PID,
                  Group="T",
                  Vars="all.x",
                  Method="normal")

res8 = TransDistr(PID = res$PID,
                  Vars="C2",
                  Method="log10")

res10 = TransDummy(PID = res$PID,
                   Vars="default")
```

Modeling This step is the most critical part of the entire module. we will build multiomics model with function “MulOmicsCros”.You can select one or more arbitrary learning methods in parameter SG_Lrns. Here we choose lasso(least absolute shrinkage and selection operator) and rf(random forest) as examples. The calculation time depends on the characteristics of your data, the number of learning methods, and the tuning method. For parameter TuneMethod, the default option can provide faster calculations but less accurate results than other autotune methods. If you want to train a better model, choose other auto-tune method and increase the number of tuning times.

```
res11 = MulOmicsCros(PID = res$PID,
                     OutPath = "default",
                     OmicGroups = "immunome,metabolome,proteome",
                     VarsY = "Y1",
                     VarsC = "all.c",
                     TuneMethod = "default",
                     TuneNum = 5,
                     RsmpMethod = "cv",
                     Folds = 5,
                     Ratio = 0.67,
                     Repeats = 5,
                     VarsImpThr = 0.85,
                     SG_Lrns ="lasso")
```

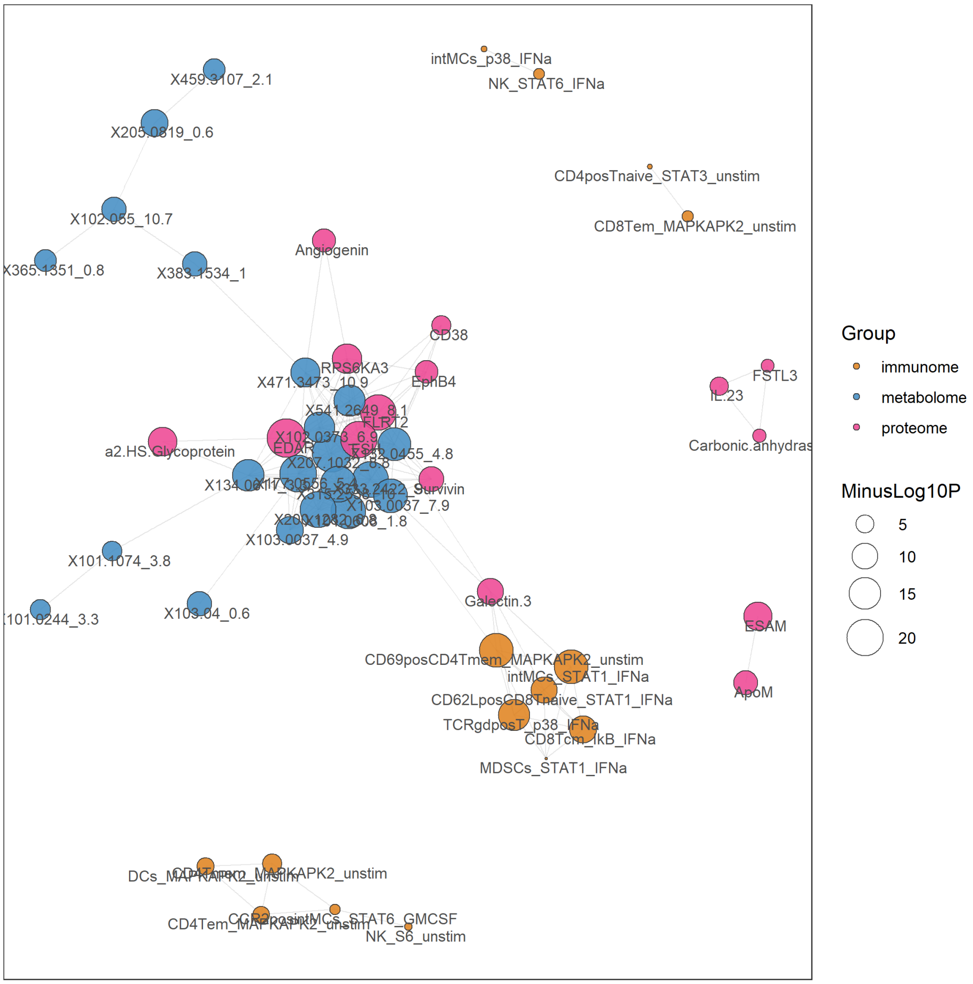
```
knitr::include_graphics("@Figures/Model assessment_boxplot.png")
```



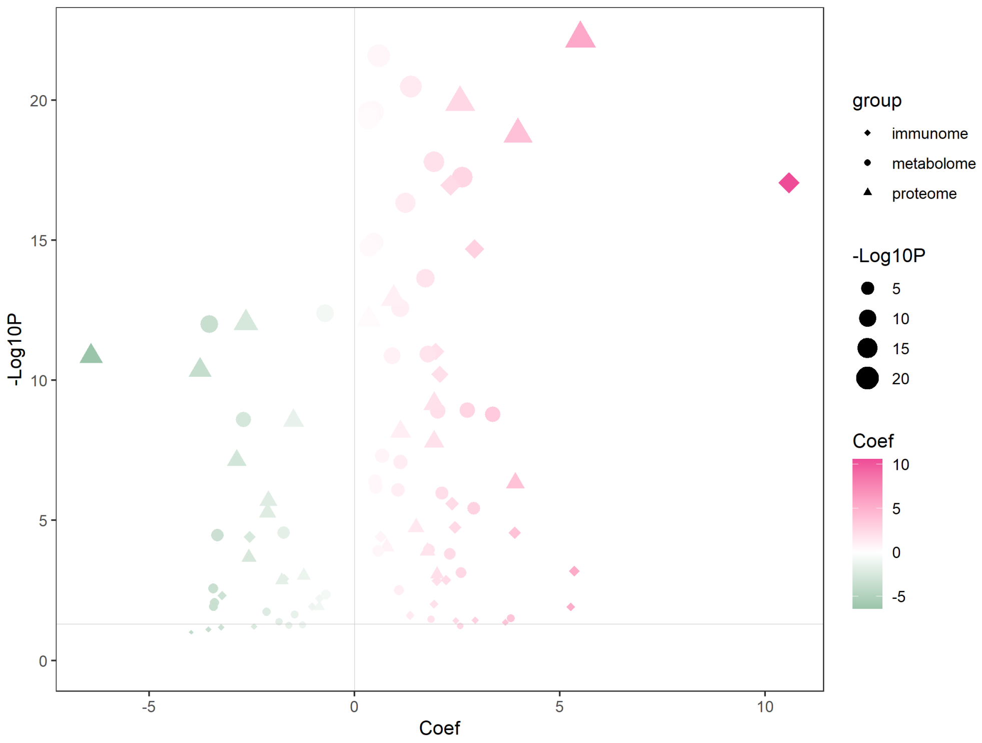
Visualize model This step will visualize multi-omic data with function "VizMulOmicsCros". You can get different styles of images by selecting different parameters.

```
res12 = VizMulOmicCros(PID = res$PID,
                       OutPath = "default",
                       VarsY = "Y1",
                       NodeNum = 100,
                       EdgeThr= 0.45,
                       Layout = "force-directed",
                       Brightness = "light",
                       Palette = 'nejm')
```

```
knitr::include_graphics("@Figures/NetworkPlot_lasso_force-directed_light_nejm.png")
```



```
knitr::include_graphics("@Figures/NodePlot_lasso_light_nejm.png")
```



```
FuncExit(PID = res$PID)
```

```
## [1] "Success to exit. Thanks for using ExposomeX platform!"
```