5 ExpoViz module

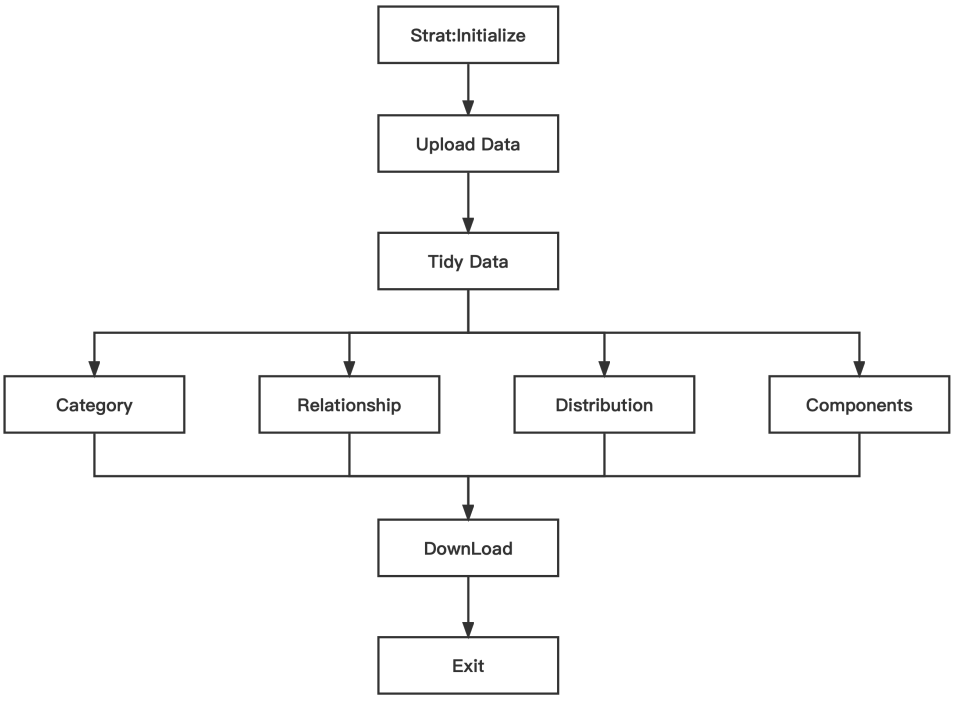
5.1 Application domain

The ***exviz*** package is designed for the data visualization of different statistical and biological analysis in a user friendly and easy way, including four typical classes of visualization. The visualization of the high dimension data is also very useful for the users in the field. The data visualization of different statistical analysis as well as the biological interaction can save much time for the data interpretation. Here, we mainly classify all the visualization task into four types, including category (distinguishing the characteristics of various groups), relationship (statistical relationship between various features), distribution (the distribution characters of the single or multiple factors), and components (the inclusion relationship between the part components and the whole). For each type, users only need to provide the original dataset by following the template and the target types, the ***exviz*** module can generate various potential visualization types.

5.2 Theory

The ***exviz*** package is based on "ggplot2" package and its extension packages used for four typical classes of data visualization.Accurate and beautiful visualization can make readers understand the paper presentation easily.The main functions of the visualization include: (1) To show the data truthfully,accurately and comprehensively; (2) To carry more information in a smaller space; (3) To reveal the essence,relationship and rule of data.

5.3 Work pipeline



Initialize package

Make sure that the packages needed is already installed.

```
# The following two packages should be installed in advance
```

```
# devtools::install_github("ExposomeX/exviz", force = TRUE)
# devtools::install_github("ExposomeX/extidy", force = TRUE)

#library(exviz)
# library(extidy)
library(tidyverse)

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

Besides, we also strongly recommended the users to install the package *extidy* for data processing.

At first, you need to initialize the calculation environment using a series of initialization functions, e.g., InitCros, InitMo, InitTidy, InitViz, InitBiolink, etc. Here, we use the package "exviz" for data visualization for example. The detailed information about the functions and returned value will be introcuced in the following chapters.

```
res = InitViz()
res
```

```
## <eSet>
##   Public:
##     AddCommand: function (x)
##     AddLog: function (x)
##     clone: function (deep = FALSE)
##     ExcecutionLog: Complete initializing the Exoverse module.2022.12.14 14. ...
##     Expo: list
##     ExpoDel: list
##     FileDirIn: NULL
##     FileDirOut: /home/ubuntu/@changxin/R_Exposome_1.0/output_144722MBWWKV
##     PID: 144722MBWWKV
##     RCommandLog: eSet <- InitVisual(PID = Any ID your like, FileDirOut =  ...
```

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

Secondly, you need to load data file for visualization. The PID Parameter, you can enter res$PID which is random generated by the system when you run the InitViz function. If you want to try this package at first, you can input the UseExample Parameter with "example#1" to use our example data. Or you can enter *UseExample = "default"* to use your own data, you can enter *DataPath =* and *VocaPath =* to choose you own input data file and vocabulary file directories. The demand of these Parameters can see in the help of *LoadViz* function.

You should note that the format of the input data file and vocabulary file is ruled. You can see the demand of the data format by visiting the following website http://www.exposomex.cn/#/expoviz.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

UseExample

chr. Method of uploading data. If "default", user should upload their own data files,or use "example#1" provided by this module.

ExdataPath

chr. Input data file directory, e.g. "D:/test/eg_expoviz_data.xlsx". It should be noted that the slash symbol is "/", not "\".

VodataPath

chr. Input vocabulary file directory, e.g. "D:/test/eg_expoviz_voca.xlsx". It should be noted that the slash symbol is "/", not "\".

```
res1 <- LoadViz(res$PID,
                UseExample = "example#1")

res1$Expo$Voca %>%
  dplyr::slice(1:20) %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

SerialNo	SerialNo_Raw	FullName	GroupName	Lod
Y1	Y1	Y_disc	Outcome	NA
Y2	Y2	Y_cont	Outcome	NA
C1	C1	Cov_1	Demography	NA
C2	C2	Cov_2	Demography	NA
C3	C3	Cov_3	Demography	NA
C4	C4	Cov_4	Demography	NA
C5	C5	Cov_5	Demography	NA
C6	C6	Cov_6	Demography	NA
X1	X1	TE_1	Chemical	0.5
X2	X2	TE_2	Chemical	0.5
X3	X3	TE_3	Chemical	0.5
X4	X4	TE_4	Chemical	0.5
X5	X5	TE_5	Chemical	0.5
X6	X6	TE_6	Chemical	0.5
X7	X7	TE_7	Chemical	0.5
X8	X8	TE_8	Chemical	0.5
X9	X9	CH1	Chemical	5.0
X10	X10	CH2	Chemical	5.0
X11	X11	CH3	Chemical	5.0
X12	X12	CH4	Chemical	5.0

```
res1$Expo$Data %>%
  dplyr::select(SampleID:C2) %>%
  dplyr::slice(1:20) %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

SampleID	SubjectID	Group	Y1	Y2	C1	C2
Tr1	S1	train	1	-101	26.86773	25.35056
Tr2	S2	train	0	-51	30.91822	23.94432
Tr3	S3	train	0	-37	25.82186	23.04579
Tr4	S4	train	1	-61	37.97640	21.21191
Tr5	S5	train	0	-28	31.64754	19.53762
Tr6	S6	train	0	-8	25.89766	20.77442
Tr7	S7	train	1	-63	32.43715	27.00009
Tr8	S8	train	0	-35	33.69162	22.13620
Tr9	S9	train	0	-14	32.87891	19.84672
Tr10	S10	train	1	-99	28.47306	29.60787
Tr11	S11	train	0	-60	37.55891	25.27530
Tr12	S12	train	0	-32	31.94922	23.28406
Tr13	S13	train	0	-73	26.89380	27.17545
Tr14	S14	train	0	-18	18.92650	26.65927
Tr15	S15	train	0	-48	35.62465	22.14227
Tr16	S16	train	0	-20	29.77533	30.61831
Tr17	S17	train	0	-9	29.91905	23.23492
Tr18	S18	train	1	-98	34.71918	19.72652
Tr19	S19	train	0	-70	34.10611	23.56680
Tr20	S20	train	0	-36	32.96951	24.62261

Here, we can see that the returned value "res1" is an R6 object. It contains two data frames which are your input data. Users need use it in the following step for further data visualization.

Tidy data

The third step is to tidy your input data into appropriate form. You should install our expotidy package before use it. If users have installed the *exposomex* package before, you can ignore this. By the way, this step is not necessary, users need to decide whether to do this step based on their own data. Noted that almost most of the functions below in the tidy module are from the package *extidy*, where the users can access for detailed information.

```
#Imputation for variables that have missing values
res2 = TransImput(PID = res$PID,
                  Group = F,
                  Vars = "all.x",
                  Method ="lod")

#Delete variables with low variance
res3 = DelNearZeroVar(PID = res$PID)

#Delete variables with missing values
res4 = DelMiss(PID = res$PID)

#Transform datatype
res5 = TransType(PID = res$PID,
                 Vars = "Y1",
                 To="factor")

#Scale
res6 = TransScale(PID = res$PID,
```

```
                  Group = F,
                  Vars = "all.x",
                  Method = "normal")

#Distribution
res7 = TransDistr(PID = res$PID,
                  Vars = "all.x",
                  Method = "ln")
```

By now, we have prepared our dataset ready for the following visualization.

Category Visualization

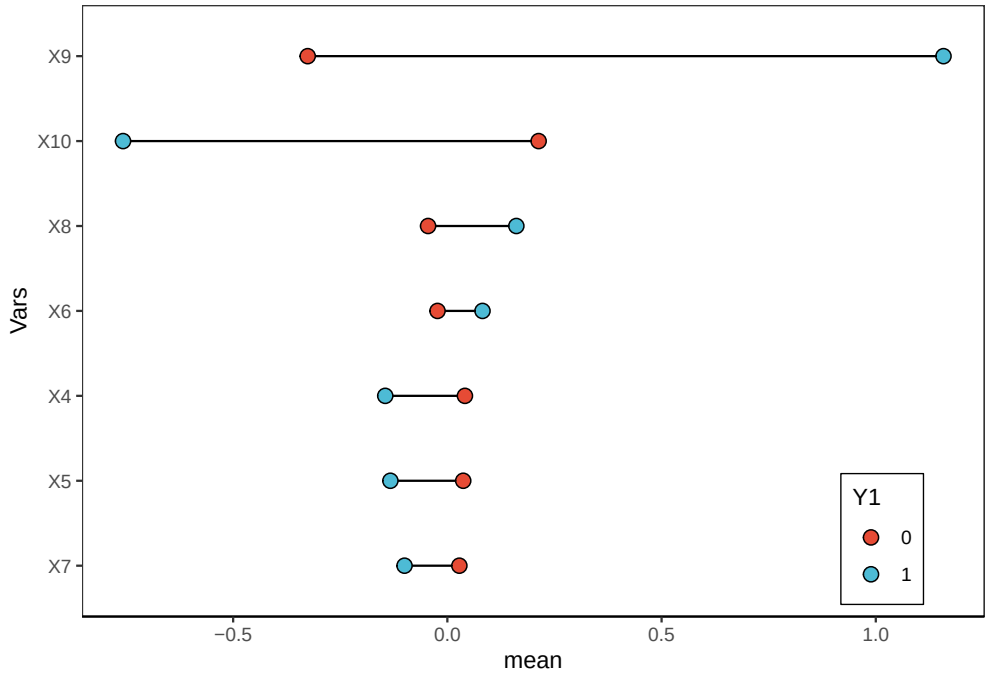
A category comparison chart generally contains two types of data: numerical and categorical, often used to compare the size of data. The input data must contain a categorical variable to use this function. We have a function to visualize data via dot plot.

The *Vars* Parameter is necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:
PID
chr. Program ID. It must be the same with the PID generated by InitViz.
OutPath
chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.
Group
lgl.Whether to separate dataset into train and test datasets for data imputation,including T or F.The default option is F.
Vars
chr. Specifying the variables.Available options include:
"all.x", all independent variables;
"all.c", all covariate variables;
"all.cx", combination of All x and All x;
or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").No more than 50 variables be entered is recommended (< 50 variables).
Parameter
chr. Specifying which parameter of the data to be the ordinate of the output plot. Available options include: "mean", "median", "min", "max", "mad" or "sd".Default is "mean".
Brightness
chr. Visualization brightness. Available options include "light" and "dark".
Palette
chr. Visualization palette. Available options include "default1", "default2" and "Journal". The "Journal" option provides several journal preference styles including cell, nature, science, lancet, nejm, and jama.

```
res8 = VizCateDot(PID = res$PID,
                  OutPath = "default",
                  Group = "F",
                  Vars = "X4,X5,X6,X7,X8,X9,X10",
                  Parameter = "mean",
                  Brightness = "light",
                  Palette = "default1")
```

```
res8$light_default1
```



Here, we can see that the returned value "res8" is a ggplot2 plot. The dot plot is used to display the relative position of two data points in the same time period, or compare the difference between the two categorical variables. The vertical coordinates are ordered by dividing the absolute value of the difference by the mean value.

Relationship Visualization

Data relational chart includes relatioanal, hierarchical and network relational charts, which respectively show the relationships between two or more variabels, the hierarchical relationships between data individuals and the visualization of relational data without hierarchical structures.

Network Visualization

The *VarsY* Parameter is necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

OutPath

chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.

VarsY

chr. Specifying the dependent variables(e.g."Y2").

VarsC

chr. Specifying the covariate variable.Available options include:

"all.c", all covariate variables;

or input a character string specifying the variables,separated by comma "," without space(e.g."C1,C2").

VarsX

chr. Specifying the independent variables. Available options include:

"all.x", all independent variables;

or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").

Family

chr. Specifying the data distribution in order to determine the visualization method. Available options include:"gaussian","poisson" and "logistic".Notice that the family are determined by data type of an outcome, or the plot can not be visualized.

Layout

chr. Visualization layout. Available options include "force-directed" and "degree-circle".

CutOff

num. Partial outcomes to visualize which is determined by correlation coefficient r. The range must between 0 and 1.

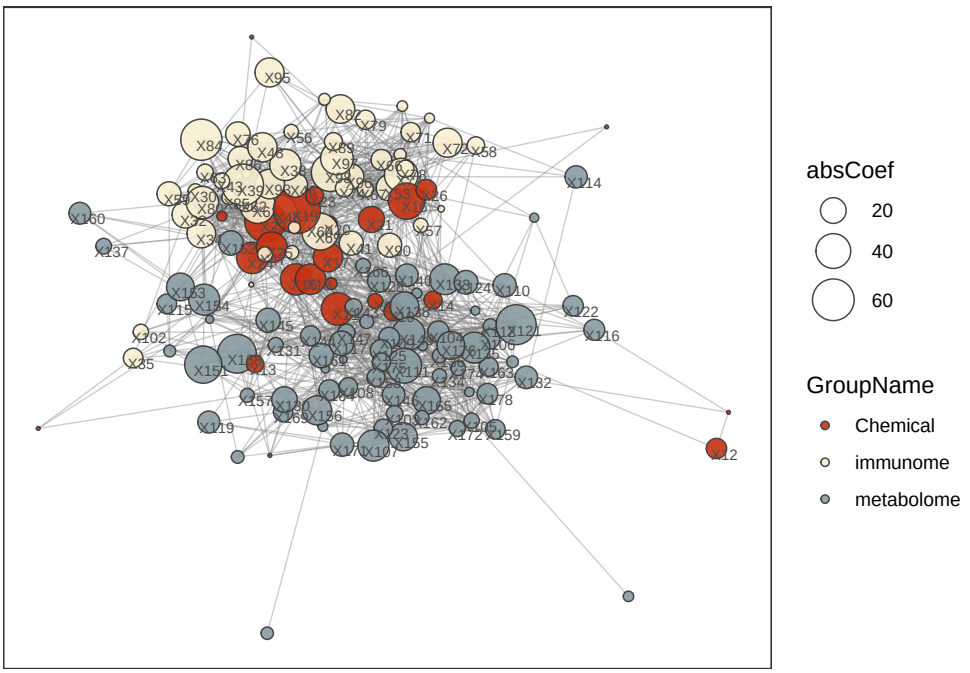
Brightness

chr. Visualization brightness. Available options include "light" and "dark".

Palette

chr. Visualization palette. Available options include "default1", "default2" and "Journal". The "Journal" option provides several journal preference styles including cell, nature, science, lancet, nejm, and jama.

```
res9 = VizRelatNetwork(PID = res$PID,
                       OutPath = "default",
                       VarsY = "Y2",
                       VarsC = "all.c",
                       VarsX = "all.x",
                       Family = "gaussian",
                       Layout = "force-directed",
                       CutOff = 0.8,
                       Brightness = "light",
                       Palette = "default1")
res9$light_default1
```



Here, we can see that the returned value "res9 is a ggplot2 plot. The network plot is used to visualize the relationship between the input variables.The point color is defined by the coefficient which is calculate by the

generalized linear model, the point size is defined by the p-value which is determined by the relationship between the input variables.

Edge Bundling Visualization

The *VarsY* Parameter is necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

OutPath

chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.

VarsY

chr. Specifying the dependent variables(e.g."Y2").

VarsC

chr. Specifying the covariate variable.Available options include:

"all.c", all covariate variables;

or input a character string specifying the variables,separated by comma "," without space(e.g."C1,C2").

VarsX

chr. Specifying the independent variables. Available options include:

"all.x", all independent variables;

or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").

Family

chr. Specifying the data distribution in order to determine the visualization method. Available options include:"gaussian","poisson" and "logistic".Notice that the family are determined by data type of an outcome, or the plot can not be visualized.

SizeFor

chr. Parameter to represent the size of the points in the output plot. Available options include "pvalue" and "beta".The default option is "pvalue".

CutOff

num. Partial outcomes to visualize which is determined by correlation coefficient r. The range must between 0 and 1.

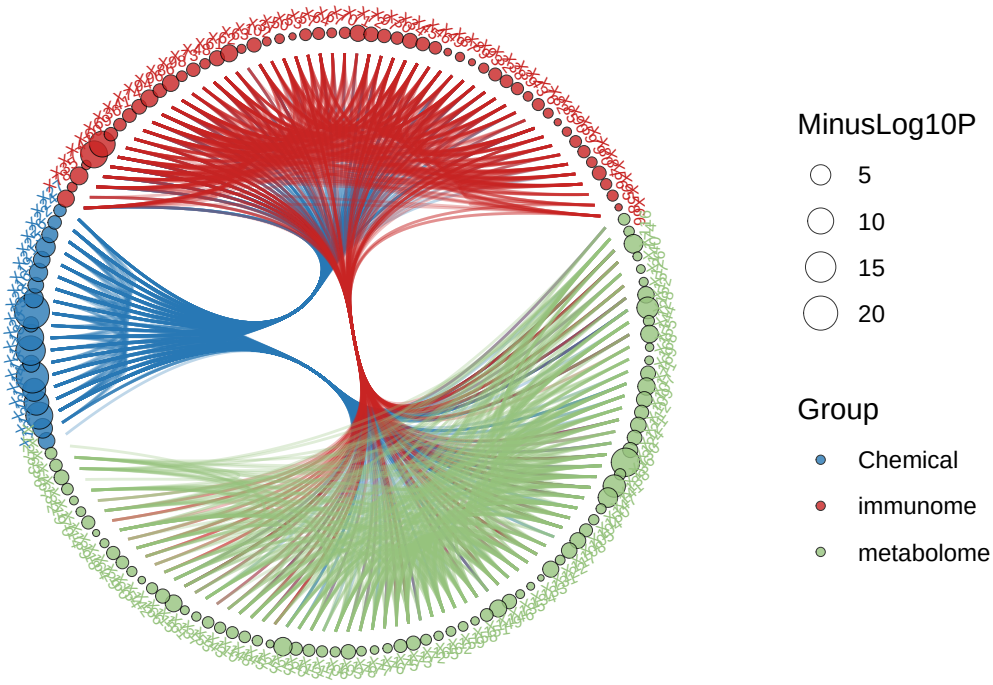
Brightness

chr. Visualization brightness. Available options include "light" and "dark".

Palette

chr. Visualization palette. Available options include "default1", "default2" and "Journal". The "Journal" option provides several journal preference styles including cell, nature, science, lancet, nejm, and jama.

```
res10 = VizRelatEdgeBundling(PID = res$PID,
                             OutPath = "default",
                             VarsY = "Y2",
                             VarsC = "all.c",
                             VarsX = "all.x",
                             Family = "gaussian",
                             SizeFor = "pvalue",
                             Brightness = "light",
                             Palette = "default1")
res10$light_default1
```



Here, we can see that the returned value "res10" is a ggplot2 plot. The edge bundling plot is used to bundle the edges closely in order to reduce complexity. The point color is defined by the group of the input independent variables, the point size is defined by the p-value which is determined by the relationship between the input variables.

Heatmap Visualization

The *VarsY* and *VarsX* Parameter are necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

OutPath

chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.

Group

lgl.Whether to separate dataset into train and test datasets for data imputation,including T or F.The default option is F.

VarsY

chr. Specifying the dependent variables(e.g."Y2").

VarsX

chr. Specifying the independent variables. Available options include:

"all.x", all independent variables;

or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").

Method

chr. Method to calculate the correlation. Default option is "spearman".

Brightness

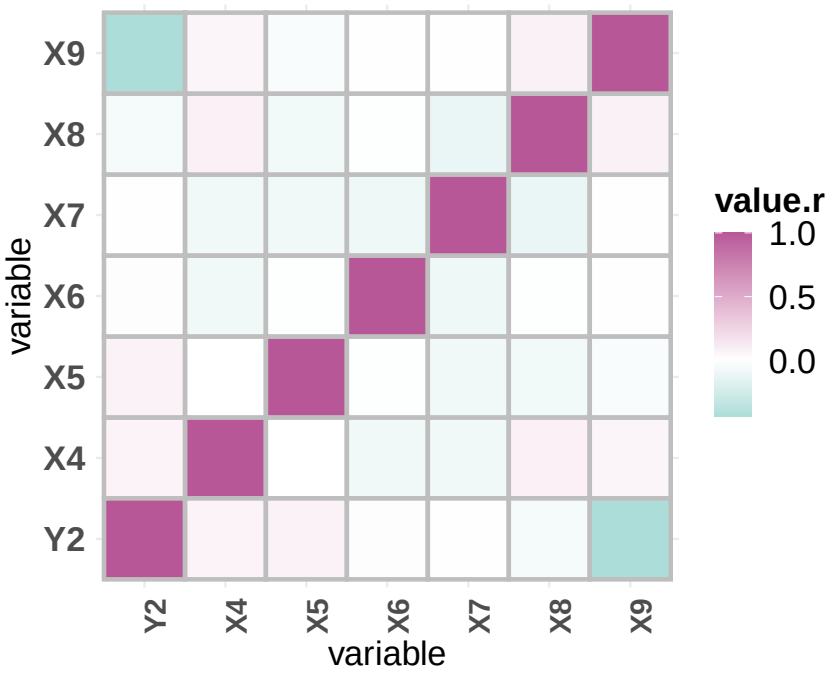
chr. Visualization brightness. Available options include "light" and "dark".

Palette

chr. Visualization palette. Available options include "default1", "default2" and "Journal". The "Journal" option provides several journal preference styles including cell, nature, science, lancet, nejm, and jama.

```
res11 = VizRelatHeatmap(PID = res$PID,
                        OutPath = "default",
                        Group = "F",
                        VarsY = "Y2",
                        VarsX = "X4,X5,X6,X7,X8,X9",
                        Method = "spearman",
                        Brightness = "light",
                        Palette = "default1")

res11$light_default1
```



Here, we can see that the returned value "res11" is a ggplot2 plot. The heatmap plot is used to display data in color changes as a matrix. The point color is defined by the coefficient r which is determined by the relationship between the input variables.

Matrix Visualization

The *VarsY* and *VarsX* Parameter are necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

OutPath

chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.

Group

lgl.Whether to separate dataset into train and test datasets for data imputation,including T or F.The default option is F.

VarsY

chr. Specifying the dependent variables(e.g."Y2").

VarsX

chr. Specifying the independent variables. Available options include:

"all.x", all independent variables;

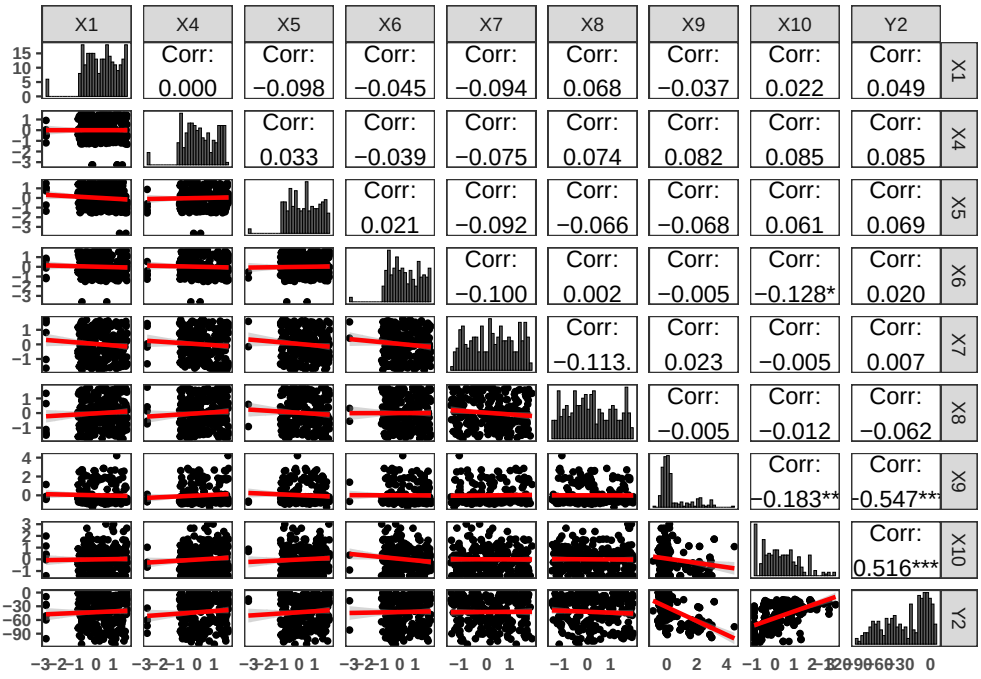
or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").

Method

chr. Method to calculate the correlation. Default option is "spearman".

```
res12 = VizRelatMatrix(PID = res$PID,
                       OutPath = "default",
                       Group = "F",
                       VarsY = "Y2",
                       VarsX = "X1,X4,X5,X6,X7,X8,X9,X10",
                       Method = "spearman")

res12$ALL
```



Here, we can see that the returned value "res12" is a ggplot2 plot. The matrix plot is used to make a matrix of plots with a given data set.

Distribution Visualization

Data distribution type chart mainly shows the values in the data set and their frequency or distribution rule. Generally, the horizontal axis represents the data type and the vertical axis represents the distribution.We have a function to visualize data via sierra plot.

The *Vars* Parameter is necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

OutPath

chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.

Group

lgl.Whether to separate dataset into train and test datasets for data imputation,including T or F.The default option is F.

Vars

chr. Specifying the variables.Available options include:

"all.x", all independent variables;

"all.c", all covariate variables;

"all.cx", combination of All x and All x;

or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").No more than 50 variables be entered is recommended (< 50 variables).

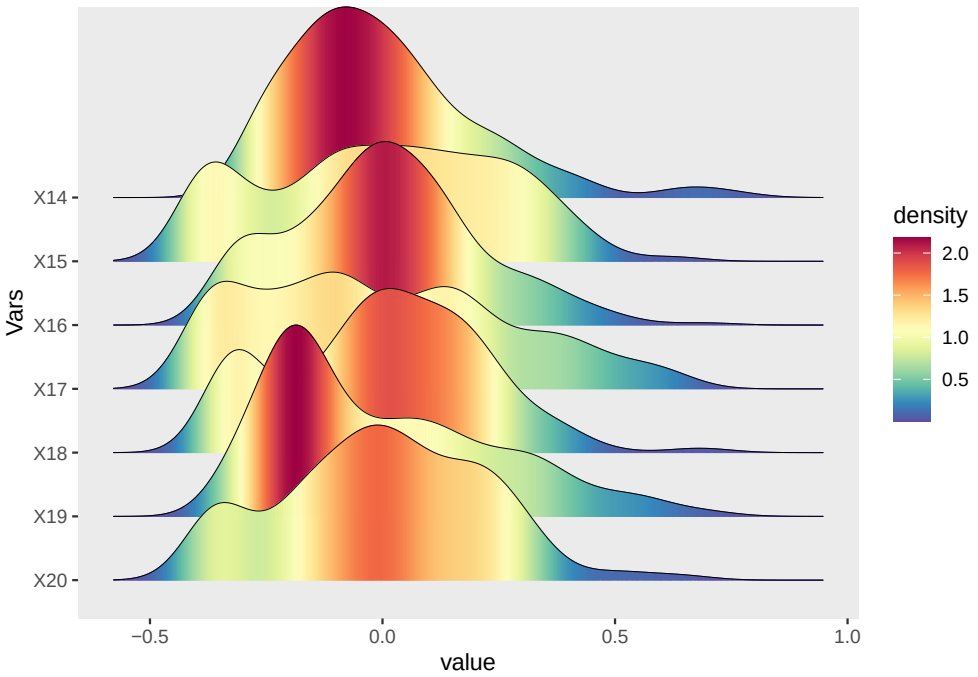
Brightness

chr. Visualization brightness. Available options include "light" and "dark".

Palette

chr. Visualization palette. Available options include "default1", "default2" and "Journal". The "Journal" option provides several journal preference styles including cell, nature, science, lancet, nejm, and jama.

```
res13 = VizDistrSierra(PID = res$PID,
                       OutPath = "default",
                       Group = "F",
                       Vars = "X14,X15,X16,X17,X18,X19,X20",
                       Brightness = "light",
                       Palette = "default1")
res13$light_default1
```



Here, we can see that the returned value "res13" is a ggplot2 plot. The sierra plot is used to visualize the kernel density estimation of data.

Components Visualization

Local integrity chart can show the proportion information of the local component and the whole. We have a function to visualize data via dendrogram plot.

The *Vars* Parameter is necessarily needed, other Parameters have default option. Or users can choose to enter other options.

The details for each parameter are listed below:

PID

chr. Program ID. It must be the same with the PID generated by InitViz.

OutPath

chr. Output file directory, e.g. "D:/test". It should be noted that the slash symbol is "/", not "\". If "default", the current working directory will be set.

Group

lgl.Whether to separate dataset into train and test datasets for data imputation,including T or F.The default option is F.

Vars

chr. Specifying the variables.Available options include:

"all.x", all independent variables;

"all.c", all covariate variables;

"all.cx", combination of All x and All x;

or input a character string specifying the variables,separated by comma "," without space(e.g."X4,X5,X6,X7,X8,X9,X10").No more than 50 variables be entered is recommended (< 50 variables).

Parameter

chr. Specifying which parameter of the data to be the ordinate of the output plot. Available options include: "mean", "median", "min", "max", "mad" or "sd".Default is "mean".

DistMethod

chr.The distance measure. This must be one of "euclidean", "maximum" or "manhattan".Default is "euclidean".

ClusterMethod

chr.The agglomeration method. This should be one of "ward.D", "ward.D2" or "single".Default is "ward.D".

ClusterNum

num. The number of groups for cutting the tree.Default is 4.

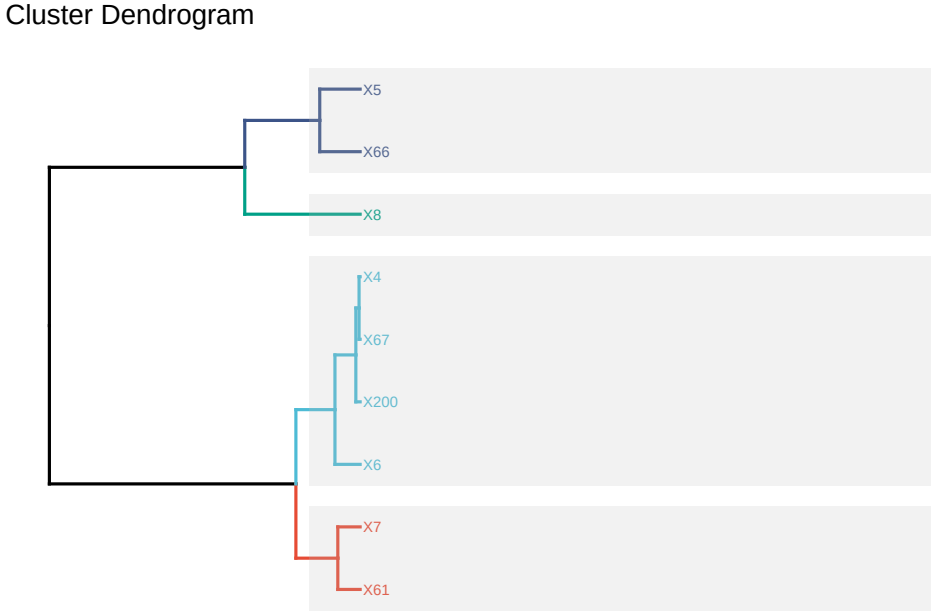
Brightness

chr. Visualization brightness. Available options include "light" and "dark".

Palette

chr. Visualization palette. Available options include "default1", "default2" and "Journal". The "Journal" option provides several journal preference styles including cell, nature, science, lancet, nejm, and jama.

```
res14 = VizCompoDendrogram(PID = res$PID,
                           OutPath = "default",
                           Group = "T",
                           Vars = "X4,X5,X6,X7,X8,X61,X66,X67,X200",
                           Parameter = "median",
                           DistMethod = "euclidean",
                           ClusterMethod = "ward.D2",
                           ClusterNum = "4",
                           Brightness = "light",
                           Palette = "default1")
res14$all_light_default1
```



Here, we can see that the returned value "res9" is a list containing three ggplot2 plots. Here we show one of them. The dendrogram plot is used to plot beautiful dendrograms.

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

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