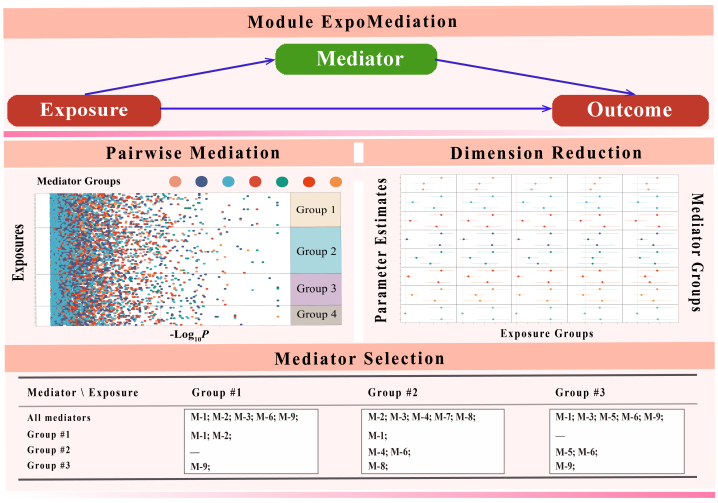
8 ExpoMedt module



8.1 Application domain

The **ExpoMedt** module is designed for estimating and screening potential mediation pathways of indigenous biomarkers (i.e., mediator) between external environmental exposure and health outcome in a user friendly and efficient way. Especially, when the number and category of exposures and mediators are of high dimension. Please see the website (http://www.exposomex.cn/#/expomediation) for more information.

8.2 Theory

The module **ExpoMedt** is realized by the package "*exmedt*". The core theory of the *exmedt* package contains two parts: pairwise mediation modelling and exposure/mediator dimension reduction. Pairwise mediation modelling allows the users to estimate the mediation effects of all potential combination of exposure-mediator pairs. For given m exposures and n mediators, an exhaustive rule will be executed and $m * n$ pair-wised modelling are fitted. In most cases or scenarios, the users may deal with datasets with numerous exposures and mediators. Using *exmedt* might be a good choice to efficiently establish batch mediaton modelling.

Dimension reduction for exposure/mediator is also an important module in *exmedt* package. By providing subgroup information for each exposure/mediator, users can conduct dimension reduction for user-specified variables or for all variables automatically (default). In *exmedt* package, various alternative methods are provided for dimension reduction. Users can choose one that may be suitable for their data or try all possible methods to obtain results for sensitive analyses.

8.3 Work pipeline

8.3.1 Initialize package

Before we start using *exmedt*, Make sure that the required packages is already installed.

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

# library(exmedt)
# library(extidy)
library(tidyverse)
```

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

At first, you need to initialize the calculation environment using the initialization function, e.g., *InitMedt*.

```
res = InitMedt()
res
```

```
## <eSet>
##   Public:
##     AddCommand: function (x)
##     AddLog: function (x)
##     clone: function (deep = FALSE)
##     DataStr: NULL
##     EpiDesign: NULL
##     ExcecutionLog: Complete initializing the mediation module.
##     2022-12-14  ...
##     Expo: list
##     ExpoDel: list
##     FileDirOut: /home/ubuntu/@changxin/R_Exposome_1.0/output_145226LTXFBP
##     PID: 145226LTXFBP
##     RCommandLog: eSet <- InitExpoData(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.

8.3.2 Upload data

In *exmedt*, we use the function *LoadMedt* to upload data file as well as vocabulary file for the following ExpoMediation analyses.

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

res1$Expo$Voca %>%
  dplyr::select(SerialNo:Lod) %>%
  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	SubgroupName	DataType	Lod
Y1	Y1	Outcome1	outcome	none	numeric	NA
Y2	Y2	Outcome2	outcome	none	numeric	NA
C1	C1	Race	demography	none	factor	NA
C2	C2	Age	demography	none	numeric	NA
C3	C3	BMI	demography	none	numeric	NA
X1	X1	NAP	chemical	polycyclic_aromatic_hydrocarbons	numeric	NA
X2	X2	ANY	chemical	polycyclic_aromatic_hydrocarbons	numeric	NA
X3	X3	ACE	chemical	polycyclic_aromatic_hydrocarbons	numeric	0.01
X4	X4	FLU	chemical	polycyclic_aromatic_hydrocarbons	numeric	NA
X5	X5	PHE	chemical	polycyclic_aromatic_hydrocarbons	numeric	NA
X6	X6	Pb	chemical	trace_metals	numeric	NA
X7	X7	Hg	chemical	trace_metals	numeric	NA
X8	X8	As	chemical	trace_metals	numeric	NA
X9	X9	Cd	chemical	trace_metals	numeric	NA
X10	X10	Zn	chemical	trace_metals	numeric	NA
M1	M1	medi_1	mediator	Pathway1	numeric	NA
M2	M2	medi_2	mediator	Pathway1	numeric	NA
M3	M3	medi_3	mediator	Pathway1	numeric	NA
M4	M4	medi_4	mediator	Pathway1	numeric	NA
M5	M5	medi_5	mediator	Pathway1	numeric	NA

```
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
1	1	train	36.08893	1	1	30.41476
2	2	train	37.77839	0	1	35.94014
3	3	train	37.12210	0	2	47.89022
4	4	train	38.31941	0	1	35.02661
5	5	train	35.68648	1	2	32.83606
6	6	train	37.50690	0	1	32.34126
7	7	train	39.01246	0	3	36.94808
8	8	train	38.20825	0	1	38.20296
9	9	train	36.19353	1	1	31.94458
10	10	train	38.56214	0	3	25.56944
11	11	train	37.46305	0	1	29.73365
12	12	train	38.78437	0	1	36.90168
13	13	train	36.67348	1	1	34.98749
14	14	train	39.96011	0	1	36.42101
15	15	train	38.81336	0	2	28.54855
16	16	train	39.70540	0	3	33.47661
17	17	train	40.86476	0	1	31.10082
18	18	train	38.72371	0	1	30.72760
19	19	train	39.09666	0	3	31.82517
20	20	train	37.41684	0	3	36.24989

8.3.3 Tidy 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
res3 = TransImput(PID = res$PID,
                  Group="F",
                  Vars="all.x",
                  Method="lod")

#Transform datatype
res4 = TransType(PID = res$PID,
                 Vars="C1",
                 To="factor")

#Scale
res5 = TransScale(PID = res$PID,
                  Group="F",
                  Vars="all.x",
                  Method="normal")
res6 = TransScale(PID = res$PID,
                 Group="F",
                 Vars="all.m",
                 Method="normal")

#Distribution
res7 = TransDistr(PID = res$PID,
                  Vars="all.x",
                  Method="ln")
res8 = TransDistr(PID = res$PID,
                 Vars="all.m",
                 Method="ln")

#Transform factor variables into dummy variables
res9 = TransDummy(PID = res$PID,
                   Vars="default")

#Divide exposures and mediators into their different subgroups
res10 <- XMlists(PID = res$PID)
res10$ExpoList
```

```
## $polycyclic_aromatic_hydrocarbons
## # A tibble: 161 x 5
##        X1     X2     X3       X4     X5
##     <dbl>  <dbl>  <dbl>    <dbl>  <dbl>
##  1 -0.191 -0.310 -0.300 -0.239   -0.364
##  2 -0.318 -0.290 -0.273 -0.210   -0.217
##  3 -0.267 -0.388 -0.359 -0.303   -0.356
##  4 -0.304 -0.251 -0.232 -0.268   -0.102
##  5 -0.112 -0.326 -0.326 -0.295   -0.418
##  6  0.593  0.817  0.548  0.752    0.287
##  7 -0.298 -0.471 -0.462 -0.372   -0.313
```

```
##  8 -0.352 -0.444 -0.436 -0.338   -0.321
##  9 -0.264 -0.450 -0.360 -0.0239  -0.174
## 10 -0.161 -0.461 -0.294 -0.00260 -0.370
## # ... with 151 more rows
##
## $trace_metals
## # A tibble: 161 x 5
##        X6      X7      X8      X9    X10
##     <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1 -0.707  0.0617 -0.698  -0.114  -0.198
##  2 -0.297 -0.406   0.198  -0.0726 -0.541
##  3 -0.629 -0.206  -0.701  -0.140  -0.510
##  4 -0.420  0.0298  0.341  -0.120  -0.679
##  5 -0.214 -0.433  -0.0128 -0.163  -0.716
##  6  6.29  -0.344  -0.829  -0.0181 -0.422
##  7 -0.111 -0.420  -0.659  -0.115  -0.192
##  8 -0.489 -0.0383  0.278   0.0397  0.785
##  9 -0.655 -0.228   1.17   -0.117  -0.618
## 10 -0.759 -0.530  -0.333  -0.0567 -0.469
## # ... with 151 more rows
```

```
res10$MediList
```

```
## $Pathway1
## # A tibble: 161 x 5
##        M1     M2     M3      M4     M5
##     <dbl>  <dbl>  <dbl>   <dbl>  <dbl>
##  1  3.36  -0.366 -0.201  0.434  -0.378
##  2  2.38   2.04   1.04   1.31    2.14
##  3 -0.460 -0.528 -0.372 -0.112  -0.265
##  4 -0.498 -0.485 -0.634 -0.651  -0.499
##  5  1.61   1.33   2.14   0.218  -0.343
##  6  2.58   2.16   1.41  -0.0330 -0.479
##  7 -0.498 -0.525 -0.389 -0.328  -0.352
##  8 -0.539 -0.405 -0.176 -0.234  -0.441
##  9 -0.535 -0.561 -0.352 -0.478  -0.356
## 10 -0.388 -0.284 -0.705 -0.155  -0.430
## # ... with 151 more rows
##
## $Pathway2
## # A tibble: 161 x 5
##         M6     M7      M8       M9     M10
##      <dbl>  <dbl>   <dbl>    <dbl>   <dbl>
##  1  0.0647 -0.963  1.04    0.549    0.857
##  2  1.03   -1.24   0.589  -0.120   -0.0478
##  3 -0.438  -1.14   0.194   1.06    -0.286
##  4 -0.584  -1.11  -0.334  -0.0634  -0.405
##  5  7.61   -1.03  -0.795  -0.551   -0.238
##  6 -0.0329 -1.25  -0.363  -0.403   -0.548
##  7 -0.536  -0.609 -0.129   0.0613  -0.167
##  8 -0.721  -1.20   0.260  -0.0634  -0.191
##  9 -0.443  -0.792 -0.0408 -0.00676 -0.500
## 10 -0.511  -0.878 -0.436  -0.596    0.452
## # ... with 151 more rows
```

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

8.3.4 Modelling

8.3.4.1 Pairwise mediation modelling

First, we proposed a pairwise mediation modelling via the function *Pairwise*, where each given exposure (named as the form of "X*,X*"; i.e., "X1,X2") and mediator (named as the form of "M*,M*"; i.e., "M1,M2") will be selected to constructe mediation model.

In *Pairwise*, several parameters need to be specified before running. The details for each parameter are listed below:

PID:

A character indicating Program ID. It must be the same with the PID generated by *InitMedt*.

VarsY:

A character indicating the outcome variable. The outcome variables should be in the form of "Y*"; i.e., "Y1".Either continuous, binary or count variable is available in function *Pairwise*.

VarsX:

A character indicating the exposure variables. The exposure variables should be in the form of "X*,X*"; i.e., "X1,X2". Users can also specify "default" to include all exposure variables into the function.

VarsM:

A character indicating the mediator variables. The mediator variables should be in the form of "M*,M*"; i.e., "M1,M2". Users can also specify "default" to include all mediator variables into the function.

VarsC:

A character indicating the confounder variables. The confounder variables should be in the form of "C*,C*"; i.e., "C1,C2". Users can also specify "default" to include all confounder variables into the function.

Family:

A character indicating the family of the mediation modellings. Available options include "linear" and "gaussian" for continuous outcome variables, and "binomial" as well as "poisson" options for binary and count outcome variables.

Iter:

A numeric value indicating the number of iteration times. Default is 500. To get a more stable result, a minimum iteration of 1,000 is recommended. To obtain more digits of P value of parameter estimation, a minimum iteration of 5,000 is suggested.

Noted that if the number of Iter, or the number of exposure-mediator pair, is large, the total execution time of the function might be time consumed. Users should be patient for waiting the outputs. The following code generate a set of 40 mediation models (10 exposures * 4 mediators) with 500 iterations for each model. The outcome variable "Y1" is a continuous variable and "linear" family was used.

In this example, we also used function *VizMedtPair* to visualize our pairwise mediation results. Noted that before using function *VizMedtPair*, make sure that the function *Pairwise* have been successfully executed. For details about this function, run "*?exmedt::VizMedtPair*" to see the help mannuals.

```
#1. Pairwise mediation modelling
res11 <- Pairwise(PID = res$PID,
                  VarsY = "Y1",
                  VarsX = "default",
                  VarsM = "M1,M2,M8,M9",
                  VarsC = "C1,C2,C3",
                  Family = "linear",
                  Iter = 500)
res11$MedtPairWise_Stats
```

```
## # A tibble: 40 x 17
##    Pairs       TE TE_LCL  TE_UCL TE_Pv~1       IE  IE_LCL  IE_UCL IE_Pv~2     DE
```

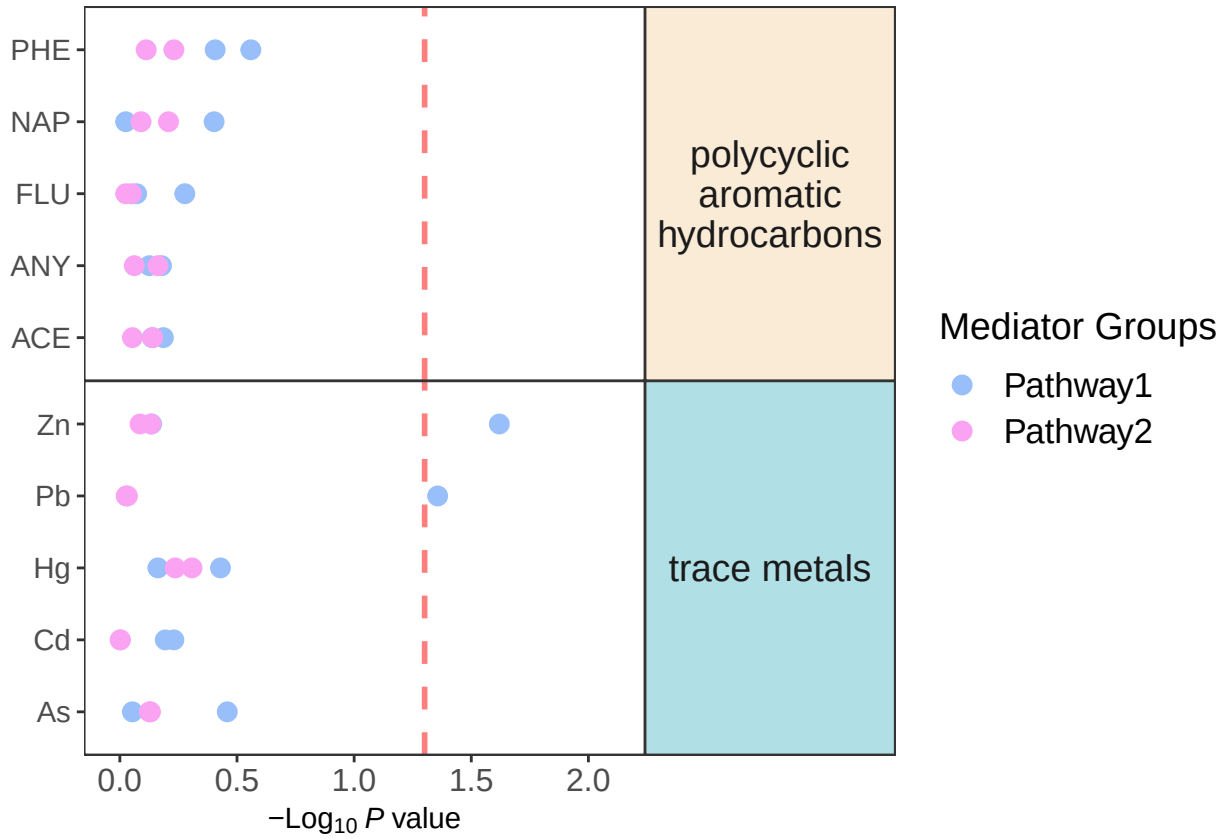
```
##    <chr>    <dbl>  <dbl>   <dbl>   <dbl>    <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1 X1 * M1 -0.347 -0.653 -0.0774   0.02   0.00106 -0.0714  0.0724   0.944 -0.348
##  2 X1 * M2 -0.364 -0.667 -0.0674   0.016 -0.0365  -0.124   0.0459   0.396 -0.328
##  3 X1 * M8 -0.349 -0.637 -0.0729   0.024 -0.00896 -0.0554  0.0267   0.62  -0.340
##  4 X1 * M9 -0.350 -0.646 -0.0575   0.016 -0.00372 -0.0353  0.0257   0.812 -0.346
##  5 X10 * ~ -0.382 -0.687 -0.0906   0.008 -0.0127  -0.0846  0.0587   0.732 -0.370
##  6 X10 * ~ -0.371 -0.653 -0.0781   0.016 -0.0925  -0.202  -0.0123   0.024 -0.278
##  7 X10 * ~ -0.382 -0.658 -0.0406   0.024 -0.00769 -0.0635  0.0290   0.736 -0.374
##  8 X10 * ~ -0.378 -0.660 -0.108    0.016  0.00406 -0.0289  0.0409   0.82  -0.382
##  9 X2 * M1 -0.388 -0.645 -0.0907   0.02   0.0154  -0.0576  0.111    0.664 -0.403
## 10 X2 * M2 -0.411 -0.726 -0.137    0.008 -0.0140  -0.108   0.0739   0.748 -0.397
## # ... with 30 more rows, 7 more variables: DE_LCL <dbl>, DE_UCL <dbl>,
## #   DE_Pvalue <dbl>, PctIE <dbl>, PctIE_LCL <dbl>, PctIE_UCL <dbl>,
## #   PctIE_Pvalue <dbl>, and abbreviated variable names 1: TE_Pvalue,
## #   2: IE_Pvalue
```

```
#2. Visualize pairwise mediation modelling result
res12 <- VizMedtPair(PID = res$PID,
                     Brightness = "light",
                     Pallette = "default1")

res12$plotdata%>%
  dplyr::slice(1:10) %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

Mediator	SubgrpM	NameM	Exposure	SubgrpX	NameX	IE_Pvalue
M1	Pathway1	medi_1	X1	polycyclic aromatic hydrocarbons	NAP	0.944
M1	Pathway1	medi_1	X2	polycyclic aromatic hydrocarbons	ANY	0.664
M1	Pathway1	medi_1	X3	polycyclic aromatic hydrocarbons	ACE	0.728
M1	Pathway1	medi_1	X4	polycyclic aromatic hydrocarbons	FLU	0.848
M1	Pathway1	medi_1	X5	polycyclic aromatic hydrocarbons	PHE	0.392
M1	Pathway1	medi_1	X6	trace metals	Pb	0.004
M1	Pathway1	medi_1	X7	trace metals	Hg	0.688
M1	Pathway1	medi_1	X8	trace metals	As	0.884
M1	Pathway1	medi_1	X9	trace metals	Cd	0.588
M1	Pathway1	medi_1	X10	trace metals	Zn	0.732

```
grid::grid.draw(res12$plot[[1]])
```



8.3.4.2 Exposure dimension reduction

In this part, we will implement an example of exposure dimension reduction. Exposure dimension reduction is realized by *RedX* function in *exmedt* function. In *RedX*, exposure variables in the datasets will be shrinkaged into different subgroups that users have defined in vocabulary file of SubGroupName column. For each shinkaged expoure variable, pairwise mediation with given mediators will be fitted at one step.

Same as *Pairwise*, several parameters are required to specify to run *RedX*. Among them, we recommended keeping the parameters of PID, VarsY, VarsC and Family in *RedX* as same as those in *Pairwise*, for better conprehension and explanation of the results. Noted that the parameter "VarsM" in *RedX* is only used for constructing mediation models, that is, the process of exposure dimension reduction in *RedX* is **independent** of "VarsM". Besides, one input parameter is unique in *RedX* function: Method.

Method:

A character indicating the method for exposure dimension reduction. Available options include "mean" and "gcdnet". We recommended choosing "gcdnet" for dimension reduction, inspite of its long time of execution.

```
#3. Exposures dimension reduction
res13 <- RedX(PID = res$PID,
              VarsY = "Y1",
              VarsC = "C1,C2,C3",
              VarsM = "M1,M2,M3",
              Method = "mean",
              Family = "linear",
              Iter = 500)
res13$MedtRedX_ERSall
```

```
## # A tibble: 161 x 1
##    ERS_all
##      <dbl>
##  1  -0.306
##  2  -0.242
##  3  -0.386
##  4  -0.201
##  5  -0.302
##  6   0.768
##  7  -0.341
##  8  -0.132
##  9  -0.172
## 10  -0.344
## # ... with 151 more rows
```

```
res13$MedtRedX_ERSlist
```

```
## # A tibble: 161 x 2
##    polycyclic_aromatic_hydrocarbons trace_metals
##                               <dbl>        <dbl>
##  1                           -0.281      -0.331
##  2                           -0.261      -0.224
##  3                           -0.335      -0.437
##  4                           -0.232      -0.170
##  5                           -0.296      -0.308
##  6                            0.600       0.936
##  7                           -0.383      -0.300
##  8                           -0.378       0.115
##  9                           -0.254      -0.0887
## 10                           -0.258      -0.430
## # ... with 151 more rows
```

```
res13$MedtRedX_Stats
```

```
## # A tibble: 9 x 17
##   Pairs        TE TE_LCL TE_UCL TE_Pv~1       IE  IE_LCL   IE_UCL IE_Pv~2     DE
##   <chr>     <dbl>  <dbl>  <dbl>   <dbl>    <dbl>   <dbl>    <dbl>   <dbl>  <dbl>
## 1 ERS_all~ -0.984 -1.57  -0.428   0     -2.77e-2 -0.187   1.16e-1   0.696 -0.956
## 2 ERS_all~ -0.975 -1.54  -0.398   0     -1.04e-1 -0.311   6.27e-2   0.26  -0.872
## 3 ERS_all~ -0.997 -1.58  -0.329   0.004 -1.27e-1 -0.312   7.99e-4   0.056 -0.870
## 4 polycyc~ -0.551 -0.923 -0.161   0.004  6.32e-4 -0.0909  9.83e-2   0.976 -0.552
## 5 polycyc~ -0.545 -0.927 -0.140   0.008 -4.66e-2 -0.185   7.06e-2   0.412 -0.498
## 6 polycyc~ -0.552 -0.914 -0.133   0.012 -8.31e-2 -0.190  -4.18e-3   0.032 -0.469
## 7 trace_m~ -0.749 -1.32  -0.216   0.004 -5.20e-2 -0.207   8.82e-2   0.44  -0.697
## 8 trace_m~ -0.762 -1.31  -0.193   0.012 -1.20e-1 -0.306   6.47e-2   0.16  -0.642
## 9 trace_m~ -0.752 -1.31  -0.206   0.008 -9.84e-2 -0.266   5.56e-3   0.072 -0.654
## # ... with 7 more variables: DE_LCL <dbl>, DE_UCL <dbl>, DE_Pvalue <dbl>,
## #   PctIE <dbl>, PctIE_LCL <dbl>, PctIE_UCL <dbl>, PctIE_Pvalue <dbl>, and
## #   abbreviated variable names 1: TE_Pvalue, 2: IE_Pvalue
```

8.3.4.3 Mediator dimension reduction

Similar to exposure dimension reduction, *exmedt* also provides dimension reducton for mediators via *RedM* function. In *RedM* function, two methods are available for dimension reduction including "mean" as well as "pdm1"(recommended but time consumed). Likewise, we recommended keeping parameters "VarsY",

"VarsC", "Family" as same as those in former steps.

However, it should be mentioned that when you choose using "pdm1" to execute mediator dimension reduction, the process of mediator dimension reduction is **NOT** independent of exposure variables. That is, the exposure variables specified in "VarsX" take effects in both dimension reduction and pairwise modedlling process in "pdm1" method settings. Therefore, we strongly recommended the users to specify **VarsX="default"** when **Method="pdm1"**.

```
#4. Mediators dimension reduction
res14 <- RedM(PID = res$PID,
              VarsY = "Y1",
              VarsX = "X1,X2,X3",
              VarsC = "C1,C2,C3",
              Method = "mean",
              Family = "linear",
              Iter = 500)
res14$MedtRedM_all
```

```
## # A tibble: 3 x 18
##   Expo  Mediator   TE    TE_LCL TE_UCL TE_Pv~1 IE    IE_LCL IE_UCL IE_Pv~2 DE
##   <chr> <chr>      <chr> <chr>  <chr>  <chr>   <chr> <chr>  <chr>  <chr>   <chr>
## 1 X1    All media~ -0.3~ -0.67~ -0.05~ 0.032   -0.0~ -0.21~ 0.007~ 0.076   -0.2~
## 2 X2    All media~ -0.3~ -0.72~ -0.10~ 0.016   -0.0~ -0.20~ 0.017~ 0.116   -0.3~
## 3 X3    All media~ -0.3~ -0.67~ -0.05~ 0.032   -0.0~ -0.20~ 0.017~ 0.116   -0.2~
## # ... with 7 more variables: DE_LCL <chr>, DE_UCL <chr>, DE_Pvalue <chr>,
## #   PctIE <chr>, PctIE_LCL <chr>, PctIE_UCL <chr>, PctIE_Pvalue <chr>, and
## #   abbreviated variable names 1: TE_Pvalue, 2: IE_Pvalue
```

```
res14$MedtRedM_list
```

```
## # A tibble: 6 x 18
##   Expo  Mediator TE      TE_LCL TE_UCL TE_Pv~1 IE    IE_LCL IE_UCL IE_Pv~2 DE
##   <chr> <chr>    <chr>   <chr>  <chr>  <chr>   <chr> <chr>  <chr>  <chr>   <chr>
## 1 X1    Pathway1 -0.352~ -0.67~ -0.05~ 0.028   -0.0~ -0.14~ 0.061~ 0.452   -0.3~
## 2 X2    Pathway1 -0.396~ -0.71~ -0.10~ 0.016   -0.0~ -0.12~ 0.085~ 0.776   -0.3~
## 3 X3    Pathway1 -0.351~ -0.67~ -0.05~ 0.028   -0.0~ -0.12~ 0.081~ 0.736   -0.3~
## 4 X1    Pathway2 -0.352~ -0.67~ -0.06~ 0.024   -0.0~ -0.16~ -0.00~ 0.024   -0.2~
## 5 X2    Pathway2 -0.395~ -0.71~ -0.10~ 0.016   -0.0~ -0.16~ -0.00~ 0.028   -0.3~
## 6 X3    Pathway2 -0.350~ -0.67~ -0.05~ 0.024   -0.0~ -0.16~ -0.00~ 0.016   -0.2~
## # ... with 7 more variables: DE_LCL <chr>, DE_UCL <chr>, DE_Pvalue <chr>,
## #   PctIE <chr>, PctIE_LCL <chr>, PctIE_UCL <chr>, PctIE_Pvalue <chr>, and
## #   abbreviated variable names 1: TE_Pvalue, 2: IE_Pvalue
```

8.3.4.4 Exposure and mediator dimension reduction

In this part, we use *RedXM* function to realize exposure and mediator dimension reduction. It's almost the same as what in *RedM*. The only difference between *RedM* and *RedXM* is that: in *RedXM*, the parameter VarsX is fixed as all shrinkaged exposure variables. Therefore, before using *RedXM*, make sure that you have executed *RedX* in your work directory.

In this example, we also used function *VizRedXM* to visualize our dimension reduction results. Noted that before using function *VizRedXM*, make sure that the function *RedXM* have been successfully executed. For details about this function, run "*?exmedt::VizRedXM*" to see the help mannuals.

```
#5. Exposure and mediator dimension reduction
res15 <- RedXM(PID = res$PID,
               VarsY = "Y1",
               VarsC = "C1,C2,C3",
```

```
                Method = "mean",
                Family = "linear",
                Iter = 500)
res15$MedtRedXM_all
```

```
## # A tibble: 3 x 18
##   Expo     Media~1 TE    TE_LCL TE_UCL TE_Pv~2 IE    IE_LCL IE_UCL IE_Pv~3 DE
##   <chr>    <chr>   <chr> <chr>  <chr>  <chr>   <chr> <chr>  <chr>  <chr>   <chr>
## 1 ERS_all  All me~ -1.0~ -1.59~ -0.49~ 0       -0.1~ -0.42~ -0.01~ 0.032   -0.8~
## 2 polycyc~ All me~ -0.5~ -0.96~ -0.22~ 0       -0.1~ -0.28~ -0.00~ 0.032   -0.4~
## 3 trace_m~ All me~ -0.8~ -1.40~ -0.26~ 0.008   -0.1~ -0.35~ 0.085~ 0.324   -0.6~
## # ... with 7 more variables: DE_LCL <chr>, DE_UCL <chr>, DE_Pvalue <chr>,
## #   PctIE <chr>, PctIE_LCL <chr>, PctIE_UCL <chr>, PctIE_Pvalue <chr>, and
## #   abbreviated variable names 1: Mediator, 2: TE_Pvalue, 3: IE_Pvalue
```

```
res15$MedtRedXM_list
```

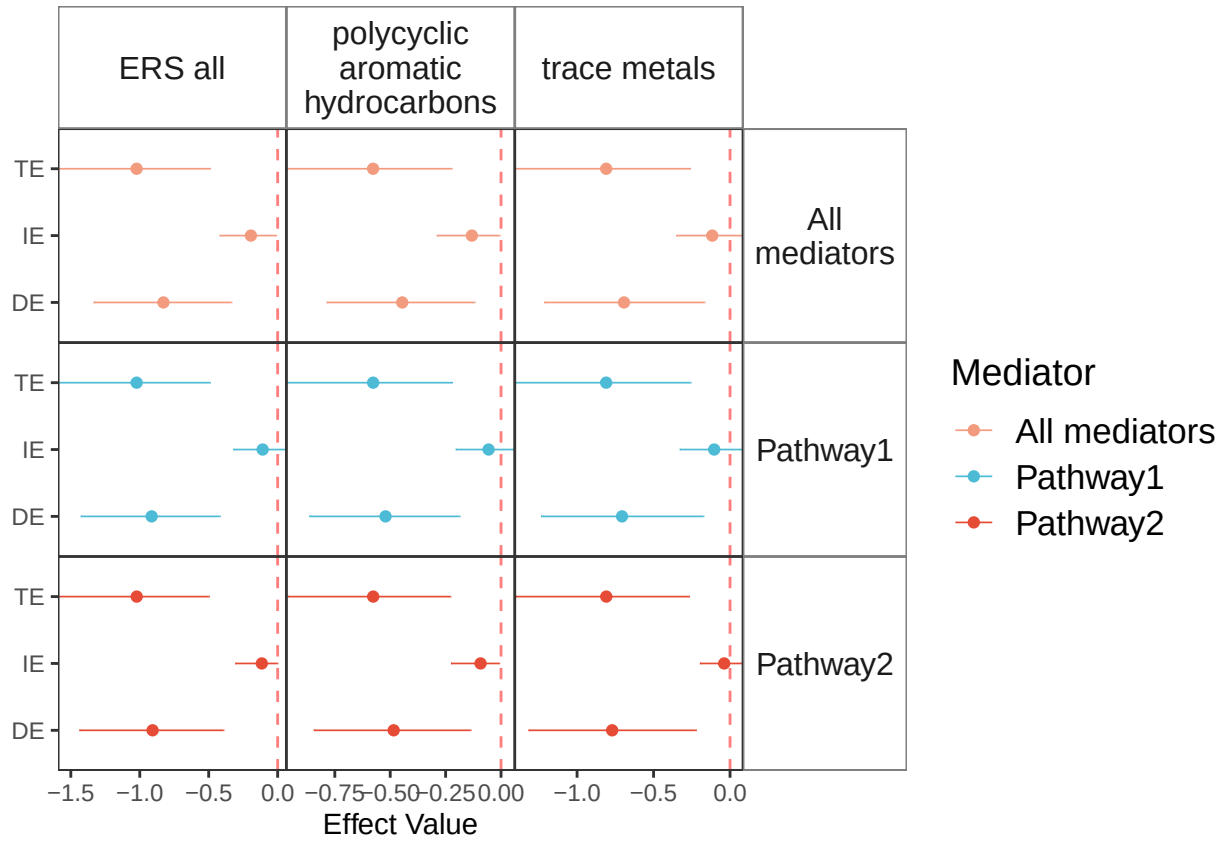
```
## # A tibble: 6 x 18
##   Expo     Media~1 TE    TE_LCL TE_UCL TE_Pv~2 IE    IE_LCL IE_UCL IE_Pv~3 DE
##   <chr>    <chr>   <chr> <chr>  <chr>  <chr>   <chr> <chr>  <chr>  <chr>   <chr>
## 1 ERS_all  Pathwa~ -1.0~ -1.59~ -0.49~ 0       -0.1~ -0.32~ 0.063~ 0.28    -0.9~
## 2 polycyc~ Pathwa~ -0.5~ -0.96~ -0.22~ 0       -0.0~ -0.20~ 0.061~ 0.416   -0.5~
## 3 trace_m~ Pathwa~ -0.8~ -1.40~ -0.25~ 0.008   -0.1~ -0.32~ 0.079~ 0.336   -0.7~
## 4 ERS_all  Pathwa~ -1.0~ -1.59~ -0.49~ 0       -0.1~ -0.30~ -0.00~ 0.048   -0.9~
## 5 polycyc~ Pathwa~ -0.5~ -0.96~ -0.22~ 0       -0.0~ -0.22~ -0.00~ 0.04    -0.4~
## 6 trace_m~ Pathwa~ -0.8~ -1.40~ -0.26~ 0.008   -0.0~ -0.19~ 0.083~ 0.584   -0.7~
## # ... with 7 more variables: DE_LCL <chr>, DE_UCL <chr>, DE_Pvalue <chr>,
## #   PctIE <chr>, PctIE_LCL <chr>, PctIE_UCL <chr>, PctIE_Pvalue <chr>, and
## #   abbreviated variable names 1: Mediator, 2: TE_Pvalue, 3: IE_Pvalue
```

```
#6. Visualize MedtRedXM mediation modelling result
res16 <- VizRedXM(PID = res$PID,
                  Brightness = "light",
                  Pallette = "nature")

res16$MedtRedXM_plotadta %>%
  dplyr::slice(1:10) %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

Expo	Mediator	Effect	Effect_Value	LowerCL	UpperCL
ERS all	Pathway1	IE	-0.1087993	-0.3213227	0.0634497
ERS all	Pathway1	DE	-0.9137781	-1.4278846	-0.4197218
ERS all	Pathway1	TE	-1.0225774	-1.5911547	-0.4917670
polycyclic aromatic hydrocarbons	Pathway1	IE	-0.0560722	-0.2044135	0.0618910
polycyclic aromatic hydrocarbons	Pathway1	DE	-0.5210358	-0.8650161	-0.1865901
polycyclic aromatic hydrocarbons	Pathway1	TE	-0.5771080	-0.9663433	-0.2211435
trace metals	Pathway1	IE	-0.1036513	-0.3290675	0.0795817
trace metals	Pathway1	DE	-0.7060029	-1.2359356	-0.1742808
trace metals	Pathway1	TE	-0.8096542	-1.4056128	-0.2578012
ERS all	Pathway2	IE	-0.1150173	-0.3068455	-0.0017912

```
res16$MedtRedXM_plot
```



8.3.4.5 Estimation the importace of mediators

ImptM function estimates the importance of mediators among given exposures. The estimation procedure is mainly realized by the hima function in HIMA package. *ImptM* also provides another evaluation method by the bama function in bama package. The users can search for these packages for further information.

For better comprehension and explanation of the result, we recommend that the parameter "VarsY" and "VarsC" unchanged. *ImptM* also supports shrinkaged exposure variables in VarsX, but *RedX* should be executed before including them into *ImptM*.

```
#7. Estimation the importance of mediators
res17  <- ImptM(PID = res$PID,
                VarsY = "Y1",
                VarsX = "X1,X2,X3",
                VarsC = "C1")

res17$MedtImptM_all %>%
  dplyr::select(exposure:pip,Bonferroni.p,BH.FDR) %>%
  dplyr::slice(1:10) %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

exposure	mediator	mediator_group	pip	Bonferroni.p	BH.FDR
X1	M1	All mediators	0.000	NA	NA
X1	M2	All mediators	0.002	1	0.9146337
X1	M3	All mediators	0.028	NA	NA
X1	M4	All mediators	0.002	1	0.9146337
X1	M5	All mediators	0.004	NA	NA
X1	M6	All mediators	0.010	1	0.9146337
X1	M7	All mediators	0.002	1	0.9146337
X1	M8	All mediators	0.006	NA	NA
X1	M9	All mediators	0.002	NA	NA
X1	M10	All mediators	0.134	NA	NA

```
res17$MedtImptM_list %>%
  dplyr::select(exposure:pip,Bonferroni.p,BH.FDR) %>%
  dplyr::slice(1:10) %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

exposure	mediator	mediator_group	pip	Bonferroni.p	BH.FDR
X1	M1	Pathway1	0.026	NA	NA
X1	M2	Pathway1	0.022	0.9815088	0.8977833
X1	M3	Pathway1	0.150	NA	NA
X1	M4	Pathway1	0.016	1.0000000	0.8977833
X1	M5	Pathway1	0.024	NA	NA
X2	M1	Pathway1	0.026	NA	NA
X2	M2	Pathway1	0.032	1.0000000	0.8044372
X2	M3	Pathway1	0.136	NA	NA
X2	M4	Pathway1	0.020	1.0000000	0.8044372
X2	M5	Pathway1	0.056	NA	NA

```
res17$MedtImptM_table1 %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

mediator_group	X1	X2	X3
All mediators	M2,M4,M6,M7	M2,M4,M6,M7	M2,M4,M6,M7
Pathway1	M2,M4	M2,M4	M2,M4
Pathway2	M6	M6	M6

```
res17$MedtImptM_table2 %>%
  knitr::kable(format = "latex",
               align = "l") %>%
  kableExtra::kable_styling(full_width = F,
                            latex_options = "striped",
                            position = "left",
                            font_size = 10)
```

mediator_group	X1	X2	X3
All mediators	M2,M4,M6,M7	M2,M4,M6,M7	M2,M4,M6,M7
Pathway1	M2,M4	M2,M4	M2,M4
Pathway2	M6	M6	M6

8.3.4.6 Exit exposome mediation analyses

When finishing your analyses task and saving all the results needed (**That is vital for the users.**), remenber run the exit function provided in that package, which will release the memory of R environment and is of helpful for both the users and the exposome platform.

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

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