main

January 4, 2022

1 Generate upsetplots per tissue for dataset comparison

```
[1]: library(ComplexHeatmap)
     library(dplyr)
```

```
Loading required package: grid

========================================
ComplexHeatmap version 2.10.0
Bioconductor page: http://bioconductor.org/packages/ComplexHeatmap/
Github page: https://github.com/jokergoo/ComplexHeatmap
Documentation: http://jokergoo.github.io/ComplexHeatmap-reference

If you use it in published research, please cite:
Gu, Z. Complex heatmaps reveal patterns and correlations in multidimensional
  genomic data. Bioinformatics 2016.

The new InteractiveComplexHeatmap package can directly export static
complex heatmaps into an interactive Shiny app with zero effort. Have a try!

This message can be suppressed by:
  suppressPackageStartupMessages(library(ComplexHeatmap))
========================================



Attaching package: ‘dplyr’


The following objects are masked from ‘package:stats’:

    filter, lag


The following objects are masked from ‘package:base’:

    intersect, setdiff, setequal, union
```

1.1 Functions

[2]:
```
subset_data <- function(tissue, direction, method){
    dt = data.table::fread("../../_m/clinical_overlap_ancestryDEGs.txt.gz") %>%
        filter(Tissue == tissue, Direction == direction, Method == method) %>%
        select(Dataset, Genes) #%>% tibble::column_to_rownames("Dataset")
    return(dt)
}
```

[3]:
```
get_annotation <- function(tissue, direction, method){
    ta = c(200, 60, 200, 250, 100, 40, 75, 125, 150, 40, 125, 150,
           100, 20, 75, 100, 50, 20, 40, 50, 50, 20, 40, 40)
    ra = c(500, 150, 500, 600, 250, 75, 250, 300, 300, 100, 300, 300,
           150, 30, 150, 150, 75, 20, 75, 75, 75, 20, 75, 75)
    annot = data.table::fread("../../_m/clinical_overlap_ancestryDEGs.txt.gz")␣
 ↪%>%
        select(-c(Genes, Dataset)) %>% distinct %>% mutate(TA=ta, RA=ra) %>%
        filter(Tissue == tissue, Direction == direction, Method == method) %>%
        select(TA, RA)
    return(annot)
}

get_data <- function(tissue, direction, method){
    datasets =␣
 ↪c('BS_Caudate_SZ','BS_DLPFC_SZ','BS_Hippocampus_SZ','CMC_DLPFC_SZ', 'PSY_SZ',
                'BS_Amyg_BD', 'BS_sACC_BD', 'PSY_BD', 'Parkinsons', 'PSY_ASD',
                'MAYO_AD',␣
 ↪'MSBB_MD10_AD','MSBB_MD22_AD','MSBB_MD36_AD','MSBB_MD44_AD','ROSMAP_AD')
    lt = list()
    for(dataset in datasets){
        #print(dataset)
        dt = subset_data(tissue, direction, method) %>% filter(Dataset ==␣
 ↪dataset)
        lt[[dataset]] <- dt$Genes
    }
    m = make_comb_mat(lt)
    return(m)
}

plot_upsetplot <- function(tissue, direction, method){
    annot <- get_annotation(tissue, direction, method)
    datasets =␣
 ↪c('BS_Caudate_SZ','BS_DLPFC_SZ','BS_Hippocampus_SZ','CMC_DLPFC_SZ', 'PSY_SZ',
                'BS_Amyg_BD', 'BS_sACC_BD', 'PSY_BD', 'Parkinsons', 'PSY_ASD',
             'MAYO_AD',␣
↪'MSBB_MD10_AD','MSBB_MD22_AD','MSBB_MD36_AD','MSBB_MD44_AD','ROSMAP_AD')
    m <- get_data(tissue, direction, method)
    cbb_palette <- ggpubr::get_palette(palette="npg", 16)
    label = gsub(" ", "_", paste(tolower(tissue), tolower(direction),␣
↪tolower(method), sep="_"))
    fn = paste0(label, ".pdf")
    ## Annotate
    right_annot = upset_right_annotation(m, ylim = c(0, annot$RA), gp =␣
↪gpar(fill = "black"),
                                     annotation_name_side = "top",␣
↪axis_param = list(side = "top"))
    top_annot = upset_top_annotation(m, height=unit(8, "cm"), ylim = c(0,␣
↪annot$TA),
                                 gp=gpar(fill=cbb_palette[comb_degree(m)]),␣
↪annotation_name_rot = 90)
    ## Save plot as PDF
    pdf(fn, width=18, height=6)
    ht = draw(UpSet(m, pt_size=unit(3, "mm"), lwd=3,␣
↪comb_col=cbb_palette[comb_degree(m)],
                    set_order = datasets, comb_order = order(-comb_size(m)),
                    row_names_gp = gpar(fontsize = 12, fontface='bold'),
                    right_annotation = right_annot, top_annotation = top_annot))
    od = column_order(ht); cs = comb_size(m)
    decorate_annotation("intersection_size", {
        grid.text(cs[od], x = seq_along(cs), y = unit(cs[od], "native") +␣
↪unit(6, "pt"),
        default.units = "native", just = "bottom", gp = gpar(fontsize = 9))
    })
    dev.off()
}
```

1.2 Main

[4]:
```
for(tissue in c("Caudate", "Dentate Gyrus", "DLPFC", "Hippocampus")){
    for(direction in c("All", "AA Bias", "EA Bias")){
        for(method in c("DEG", "TWAS")){
            plot_upsetplot(tissue, direction, method)
        }
    }
}
```

1.3 Reproducibility Information

[5]:
```
Sys.time()
proc.time()
options(width = 120)
sessioninfo::session_info()
```

```
[1] "2022-01-04 15:23:09 EST"
```

```
   user  system elapsed 
 57.133   7.464  69.383 
```

$platform **$version** 'R version 4.1.2 (2021-11-01)'

$os 'Arch Linux'

$system 'x86_64, linux-gnu'

$ui 'X11'

$language '(EN)'

$collate 'en_US.UTF-8'

$ctype 'en_US.UTF-8'

$tz 'America/New_York'

$date '2022-01-04'

$pandoc '2.14.1 @ /usr/bin/pandoc'

$packages A packages_info: 69 × 11

	package	ondiskversion	loadedversion	p
	<chr>	<chr>	<chr>	<
abind	abind	1.4.5	1.4-5	/
assertthat	assertthat	0.2.1	0.2.1	/
backports	backports	1.4.0	1.4.0	/
base64enc	base64enc	0.1.3	0.1-3	/
BiocGenerics	BiocGenerics	0.40.0	0.40.0	/
broom	broom	0.7.10	0.7.10	/
car	car	3.0.12	3.0-12	/
carData	carData	3.0.4	3.0-4	/
circlize	circlize	0.4.13	0.4.13	/
cli	cli	3.1.0	3.1.0	/
clue	clue	0.3.60	0.3-60	/
cluster	cluster	2.1.2	2.1.2	/
codetools	codetools	0.2.18	0.2-18	/
colorspace	colorspace	2.0.2	2.0-2	/
ComplexHeatmap	ComplexHeatmap	2.10.0	2.10.0	/
crayon	crayon	1.4.2	1.4.2	/
data.table	data.table	1.14.2	1.14.2	/
DBI	DBI	1.1.1	1.1.1	/
digest	digest	0.6.28	0.6.28	/
doParallel	doParallel	1.0.16	1.0.16	/
dplyr	dplyr	1.0.7	1.0.7	/
ellipsis	ellipsis	0.3.2	0.3.2	/
evaluate	evaluate	0.14	0.14	/
fansi	fansi	0.5.0	0.5.0	/
fastmap	fastmap	1.1.0	1.1.0	/
foreach	foreach	1.5.1	1.5.1	/
generics	generics	0.1.1	0.1.1	/
GetoptLong	GetoptLong	1.0.5	1.0.5	/
ggplot2	ggplot2	3.3.5	3.3.5	/
ggpubr	ggpubr	0.4.0	0.4.0	/
iterators	iterators	1.0.13	1.0.13	/
jsonlite	jsonlite	1.7.2	1.7.2	/
lifecycle	lifecycle	1.0.1	1.0.1	/
magrittr	magrittr	2.0.1	2.0.1	/
matrixStats	matrixStats	0.61.0	0.61.0	/
munsell	munsell	0.5.0	0.5.0	/
pbdZMQ	pbdZMQ	0.3.6	0.3-6	/
pillar	pillar	1.6.4	1.6.4	/
pkgconfig	pkgconfig	2.0.3	2.0.3	/
png	png	0.1.7	0.1-7	/
purrr	purrr	0.3.4	0.3.4	/
R.methodsS3	R.methodsS3	1.8.1	1.8.1	/
R.oo	R.oo	1.24.0	1.24.0	/
R.utils	R.utils	2.11.0	2.11.0	/
R6	R6	2.5.1	2.5.1	/
RColorBrewer	RColorBrewer	1.1.2	1.1-2	/
repr	repr	1.1.3	1.1.3	/
rjson	rjson	0.2.20	0.2.20	/
rlang	rlang	0.4.12	0.4.12	/
rstatix	rstatix	0.7.0	0.7.0	/

$hash $emoji 1. ' ' 2. ' ' 3. ' '

$emo_text 1. 'open hands: medium skin tone' 2. 'woman cook: light skin tone' 3. 'old woman: light skin tone'