Exploring the pan-cancer landscape of posttranscriptional regulation

Dr. Umesh Ghoshdastider

Contents

import libraries

```
library(ggplot2)
suppressPackageStartupMessages(library(circlize))
library(reshape2)
library(tidyr)
```

```
##
## Attaching package: 'tidyr'

## The following object is masked from 'package:reshape2':
##
##     smiths
```

```
library(dplyr)
```

```
##
## 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
```

```
library(ComplexHeatmap)
```

```
## Loading required package: grid

## ========================================
## ComplexHeatmap version 2.14.0
## 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 either one:
## - Gu, Z. Complex Heatmap Visualization. iMeta 2022.
## - 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))
## ========================================
```

```
library(viridis)
```

```
## Loading required package: viridisLite
```

```
library(forcats)
library(survival)
library(survivalAnalysis)
library(survminer)
```

```
## Loading required package: ggpubr
```

```
##
## Attaching package: 'survminer'
```

```
## The following object is masked from 'package:survival':
##
##     myeloma
```

```
library(enrichR)
```

```
## Welcome to enrichR
## Checking connection ...
```

```
## Enrichr ... Connection is Live!
## FlyEnrichr ... Connection is Live!
## WormEnrichr ... Connection is Live!
## YeastEnrichr ... Connection is Live!
## FishEnrichr ... Connection is Live!
## OxEnrichr ... Connection is Live!
```

```
library(stringr)
```

Fig 1G

```
df= read.csv('data/rna_prot_corr_quartile.csv', row.names = 1)
df = na.omit(df)

# use spearman correlation distance
dis_col = as.dist(1 - cor(df    , method = "spearman" ) )

# dont show gene names
pheatmap::pheatmap(df,
        show_rownames=FALSE,
        #filename = 'output/rna_prot_corr_heatmap_genes.pdf',
        cutree_rows=8,
        clustering_method = 'ward.D2',
        clustering_distance_cols = dis_col,
        width = 4,
        height = 10)
```
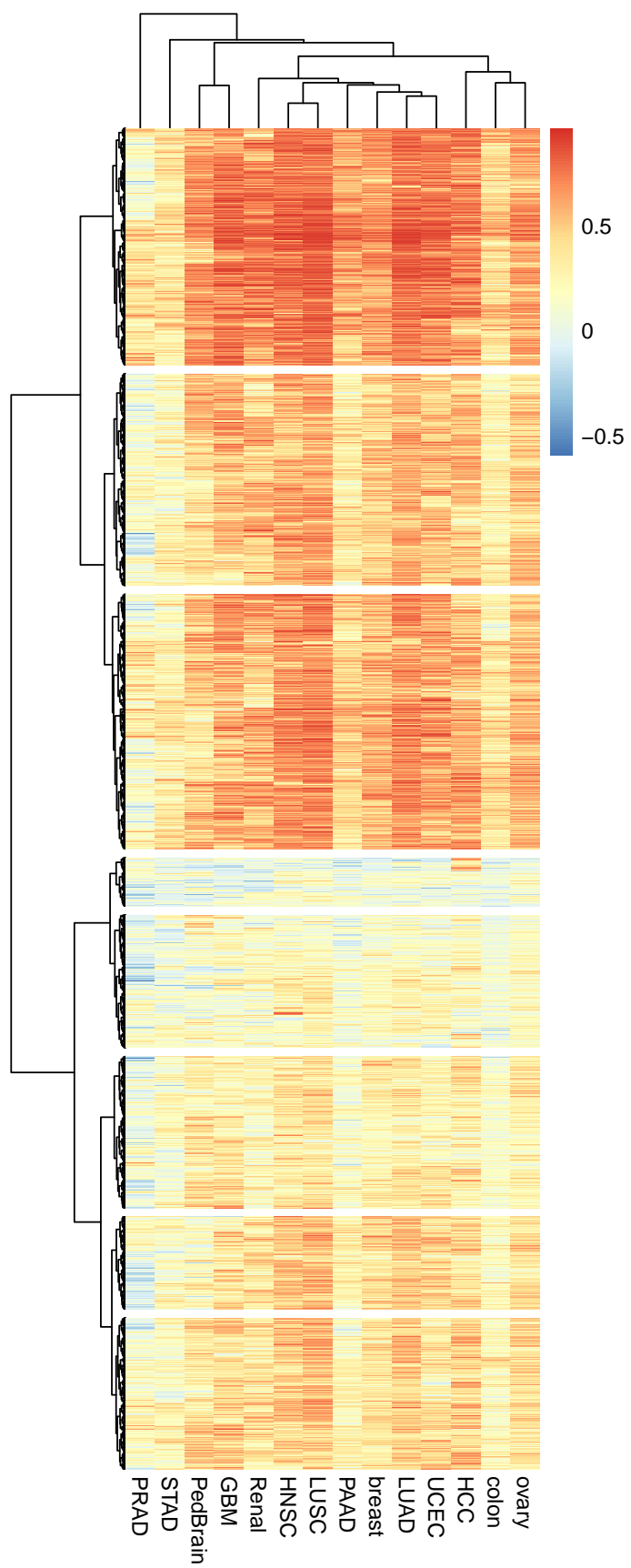
Fig 1F

```
pheatmap::pheatmap(cor(df),
         #filename = 'output/rna_prot_corr_heatmap_genes_corr.pdf',
         )
```
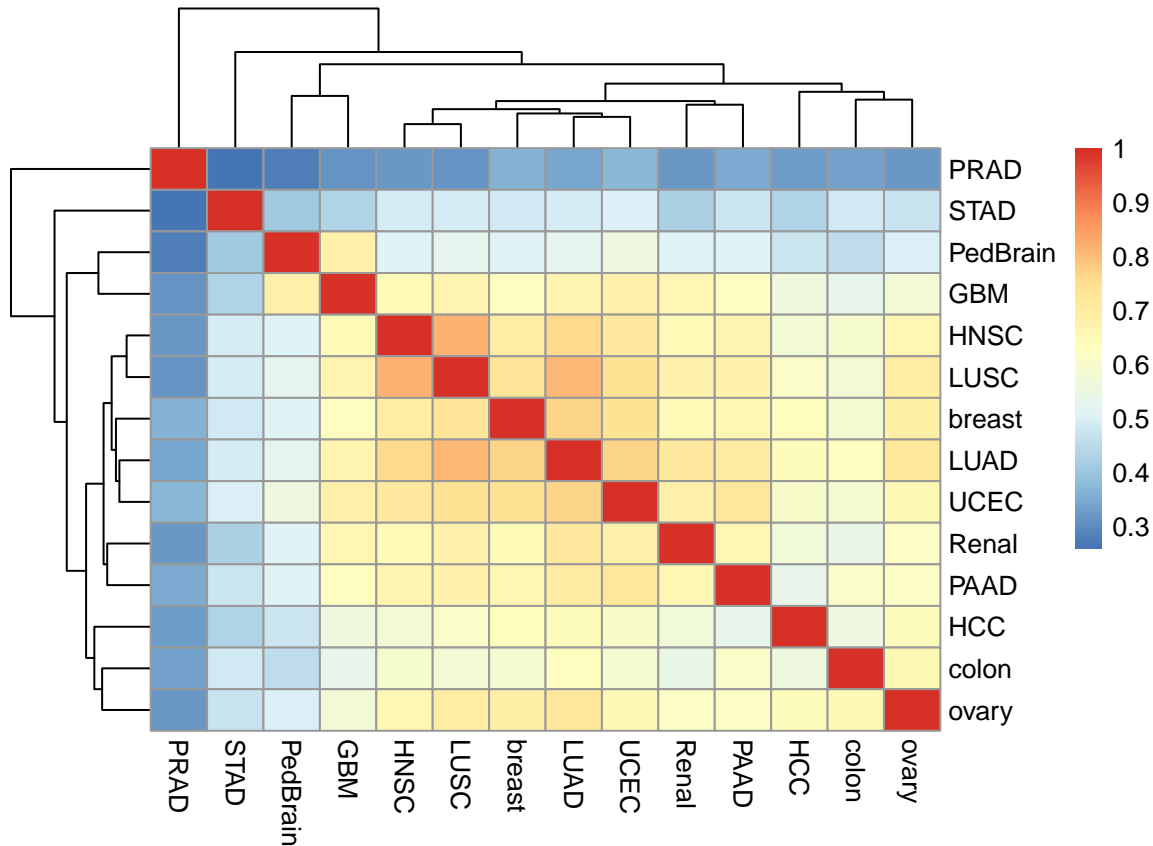

Fig 3E Oncoprint

```
# mutations annotations
c= read.csv('data/oncoprint_mutations_annotations.csv', row.names = 1)

# mutation data
m= read.csv('data/oncoprint_mutations.csv', row.names = 1)

rownames(c)=colnames(m)

col_fun = colorRamp2(c(0, 1), c("white", "dodgerblue4"))

#pdf('output/oncoprint_mutations.pdf', width = 10, height = 6)

ha = HeatmapAnnotation(
    empty = anno_empty(border = FALSE),
```
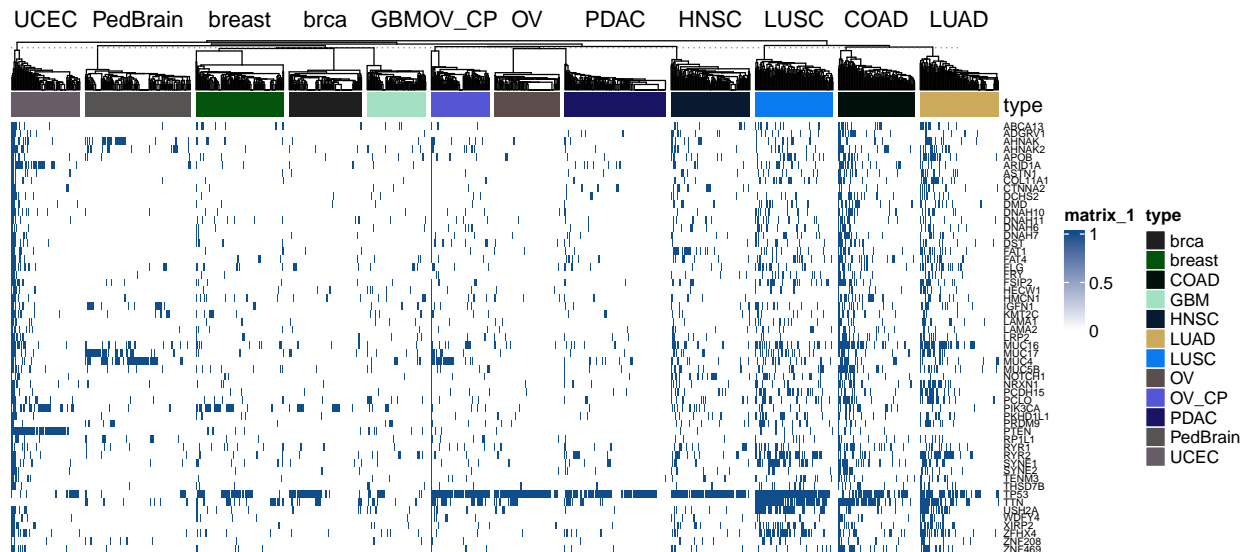
```
    foo = anno_block(gp = gpar(fill = 2:6), labels = unique(c$type))
)

ht_opt$heatmap_row_names_gp = gpar(fontsize = 6)

Heatmap(m,
        cluster_rows = FALSE,
        #cluster_columns = FALSE,
        show_column_names = FALSE,
        column_split = c$type,
        col=col_fun,
        top_annotation = HeatmapAnnotation(df = c)
        )
```

Warning: The input is a data frame-like object, convert it to a matrix.


```
#dev.off()
```

Fig 2A, Circos plot of driver genes correlations

```
# read corr data
median1=function(x) {median(x, na.rm = TRUE)}
sd1=function(x) {sd(x, na.rm = TRUE)}

# read correlations
d= read.csv('data/rna_prot_corr_quartile.csv', row.names = 1)
mf= read.csv('data/mutation_freq.csv')[c('Gene','Pancan.Frequency')]  %>% distinct()
ov=intersect(mf$Gene, rownames(d))
df=d[ov,]

# remove genes with missing data in 10 cancer types
```

```r
tmp=rowSums(is.na(df) )
tmp=tmp[tmp<10]

df=df[names(tmp),]

# get mutation freq
mf1=subset(mf, Gene %in% rownames(df))
rownames(mf1)=mf1$Gene
mf1$Gene=NULL

circos.clear()
circos.par(gap.degree = 5)

#pdf('output/mutation_corr_freq_circos.pdf')

col_fun = colorRamp2(seq(-1,1, length.out = 20), inferno(20))
#col_fun1 = colorRamp2(c(0, 0.4), c("white", "green"))
col_fun1 = colorRamp2(seq(0, 0.4, length.out = 10), viridis(10))

# heatmap of corr across cancer types
#co1[is.na(co1)] = 0

circos.heatmap(df,
               col = col_fun,
               rownames.side='outside',
               #cluster= FALSE,
               cell.border ='grey',
               dend.side = 'inside',
               rownames.cex = 0.4,
               #clustering.method = 'ward.D2',
               #split = 4,
               )

# add column names circos.text
circos.track(track.index = get.current.track.index(), panel.fun = function(x, y) {
        cn = colnames(df)
        n = length(cn)
        circos.text(rep(CELL_META$cell.xlim[2], n) + convert_x(1, "mm"),
            1:n - 0.5, cn,
            cex = 0.2, facing = "inside")

}, bg.border = NA)


## Note: 14 points are out of plotting region in sector 'group', track
## '3'.


# heatmap avg corr
circos.par("track.height" = 0.01)
#rowm = rowMeans(df[CELL_META$row_order,], na.rm =TRUE)
#med= apply(df[CELL_META$row_order,], 1, median1)
med= apply(df, 1, median1)
circos.heatmap(med, col = col_fun, cluster= FALSE) #
```

```r
# sd corr
circos.par("track.height" = 0.01)
sd= apply(df, 1, sd1)
circos.heatmap(sd, col = col_fun1, cluster= FALSE) # col = col_fun,

# barplot mutation freq
circos.par("track.height" = 0.3)
circos.track(ylim = c(0, 38), panel.fun = function(x, y) {
    circos.barplot( mf1[CELL_META$row_order,], 1:dim(mf1)[1]-0.5, col=3 )
})
circos.yaxis(side = "left", labels.cex= 0.25)


# draw colorbar
lgd1 = Legend(col_fun = col_fun, title = "corr")
lgd2 = Legend(col_fun = col_fun1, title = "sd")

lgd = packLegend(lgd1, lgd2)

draw(lgd, x = unit(2, "mm"), y = unit(2, "mm"), just = c("left", "bottom"))
```
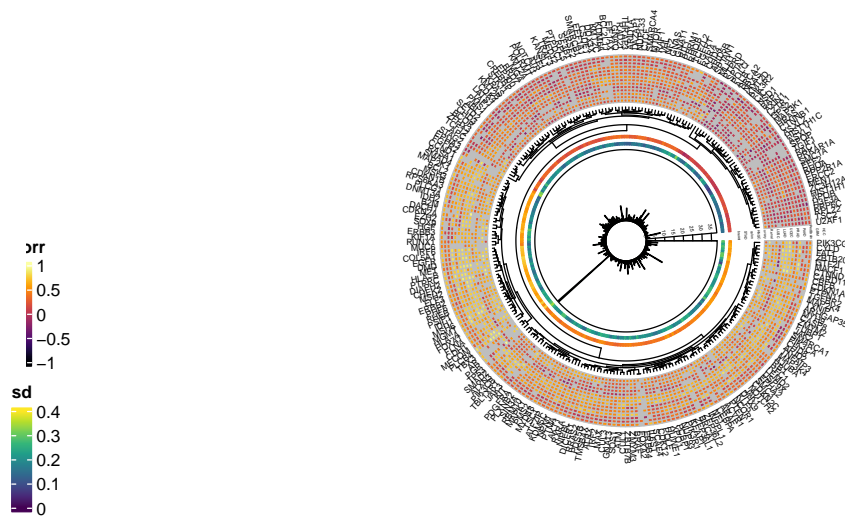


```r
#df$sd=sd
#df$median=med
#write.csv(df,'output/circos_input.csv')

#dev.off()
```

Fig 2A boxplot

```r
d= read.csv('data/rna_prot_corr_quartile.csv', row.names = 1)
mf= read.csv('data/mutation_freq.csv')[c('Gene','Pancan.Frequency')]  %>% distinct()
```

```
d1=melt(d)
```

```
## No id variables; using all as measure variables
```

```
d1$type='all'
```

```
d2=melt(d[mf$Gene,])
```

```
## No id variables; using all as measure variables
```
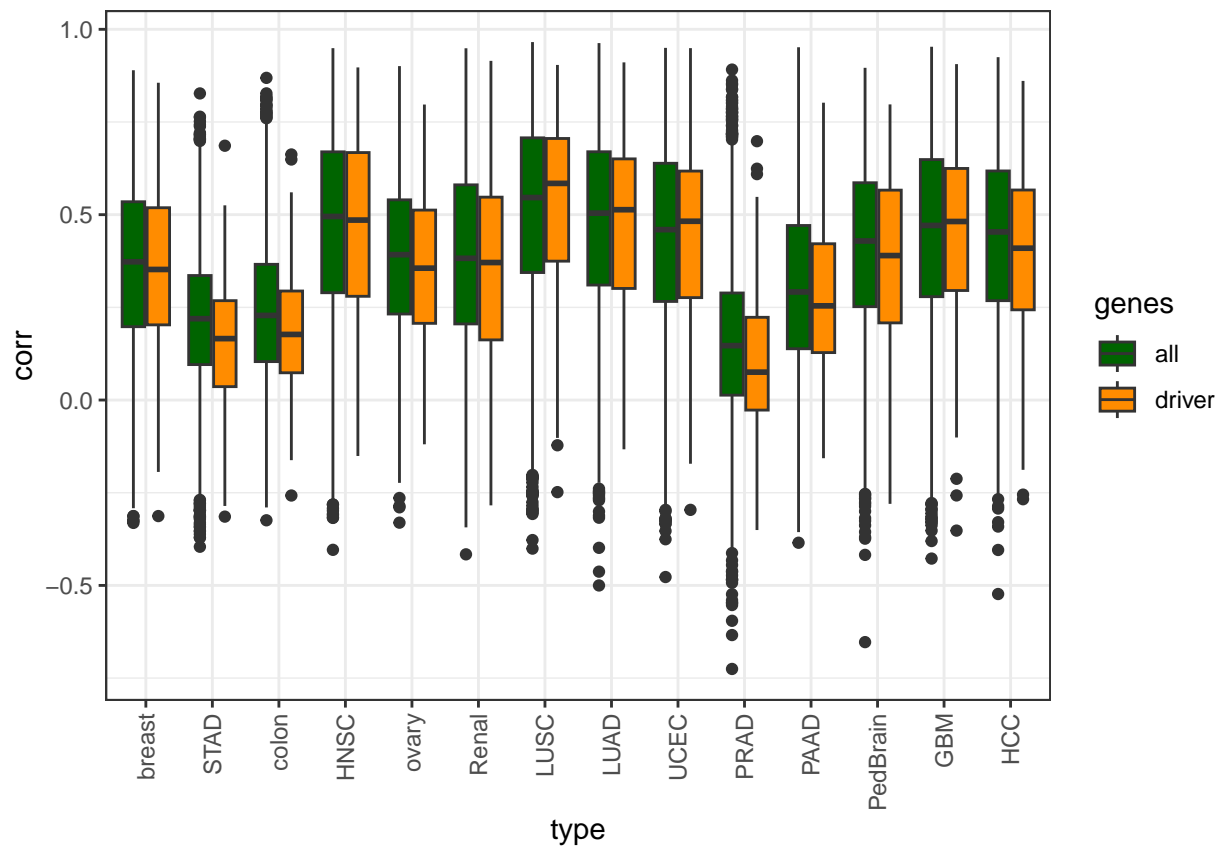
```
d2$type='driver'

d3=rbind(d1,d2)
colnames(d3)=c('type','corr','genes')

ggplot(d3, aes(type, corr, fill=genes)) +
  geom_boxplot() +
  theme_bw() +
  scale_fill_manual(values = c('darkgreen','darkorange')) +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

```
## Warning: Removed 68149 rows containing non-finite values ('stat_boxplot()').
```

```
#ggsave('output/rna_prot_corr_driver.pdf', width = 10, height = 8)
```

Fig 2B

```
df= read.csv('data/rna_prot_corr_quartile.csv', row.names = 1)

# read oncokb targets
ca=read.table('oncokb_targets')

ov=intersect(ca$V1, rownames(df))

df=df[ov,]

# remove genes with missing data in 10 cancer types
tmp=rowSums(is.na(df) )
tmp=tmp[tmp<11]

df=df[names(tmp),]

# horizontal
pheatmap::pheatmap(t(df),
        show_rownames=TRUE,
        color=inferno(20),
        #filename = 'output/rna_prot_corr_oncokb_targets.pdf',
        width = 20,
        height =4)
```
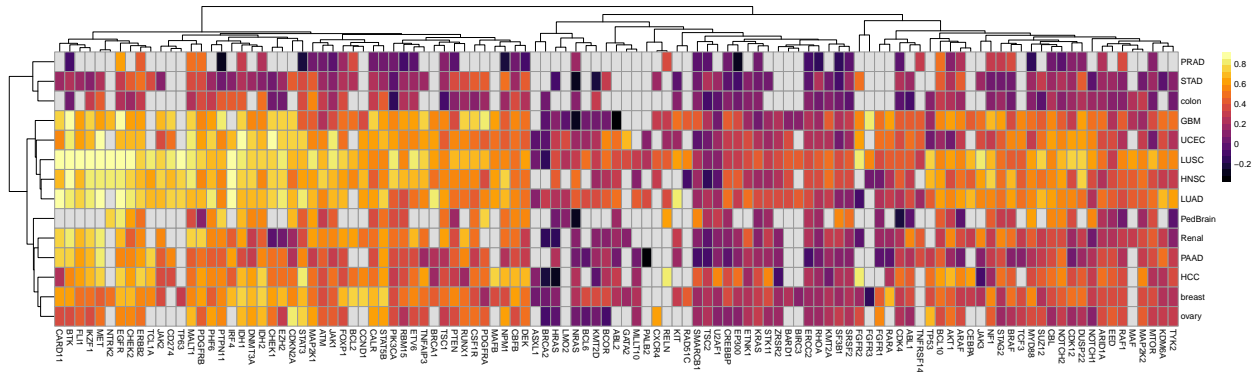

Fig 3A

```
df=read.csv('data/mutation_corr_wilcoxon.csv')
df$q=p.adjust(df$pv, method = 'BH')
d1=subset(df, q < 0.05)
#d1=subset(df, abs(diff)>0.01)

# select mutations in >1 cancer type
```

```
count = sort(table(d1$gene))
count=count[count>3]

d1= subset(d1, d1$gene %in% names(count))

d1$diff=d1$mm-d1$wm
d1$size= -log10(d1$q)

d1$shape=ifelse(d1$diff>0,'pos','neg')
#write.csv(d1, 'mutation_corr_wilcoxon_new1_fdr05.csv')

d1 %>%
  mutate(gene = fct_reorder(gene, desc(gene))) %>%
  ggplot( aes(ty, gene, size=size, color=diff)) +
  geom_point() +
  scale_colour_gradient2(low = 'blue', high='red', mid = 'white' ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
  geom_point(shape=21, color='black')
```
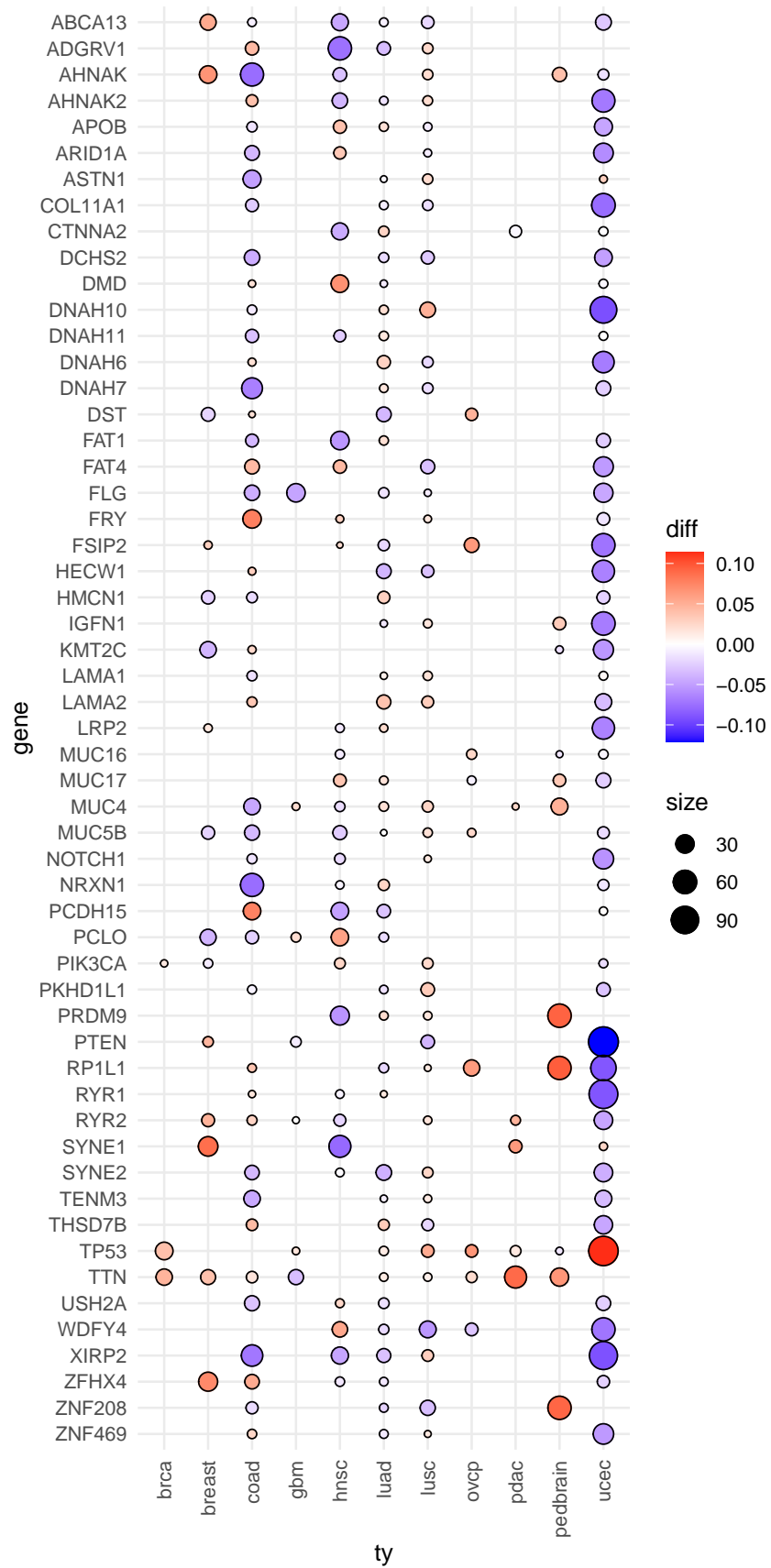
```
#ggsave('output/mutation_corr_wilcoxon.pdf', height = 10, width = 5)
```

Fig 3D

```
d=read.csv('data/gsea_merged_padj.csv')

ggplot(d, aes(ty,X, color=NES, size=-log10(FDR.q.val+0.001))) +
  geom_point() +
  scale_color_viridis() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```
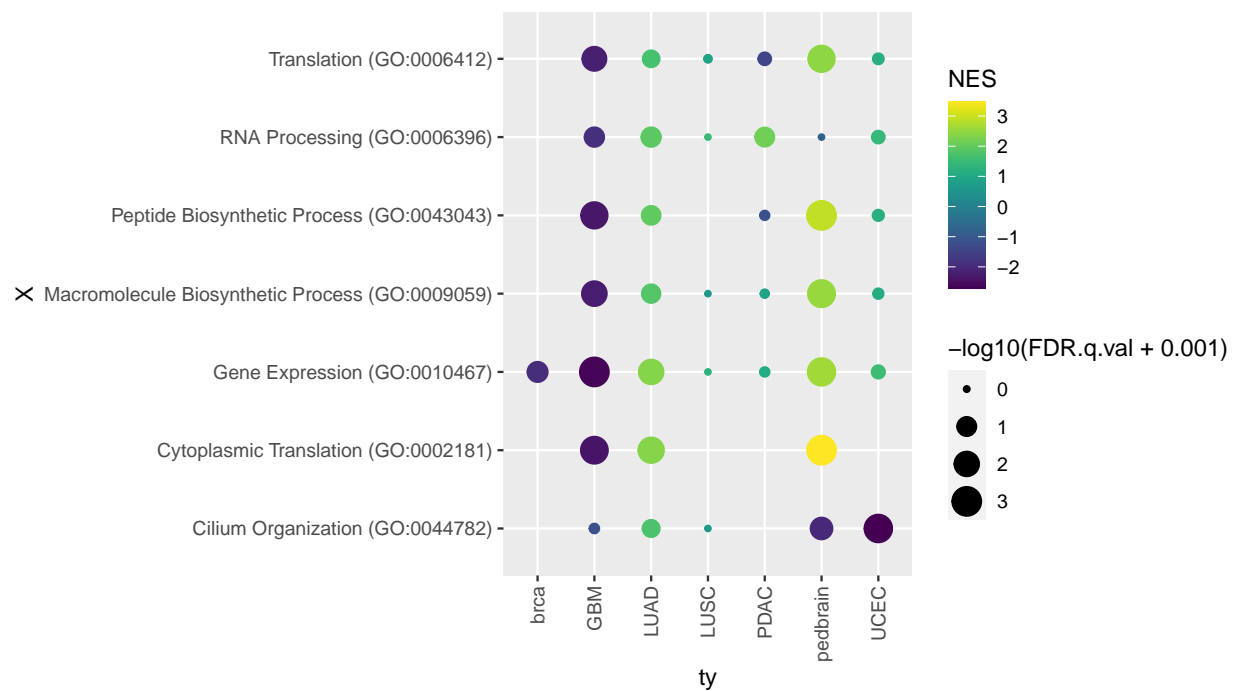

Fig 4F

```
library(EnhancedVolcano)
```

```
## Loading required package: ggrepel
```

```
a=read.csv('data/brca_protein_spearman_high_vs_low_tertile_de_padj.csv', row.names = 1)

EnhancedVolcano(a,
    lab = rownames(a),
    x = 'lfc',
    y = 'padj_ttest',
    pCutoff = 0.01,
```
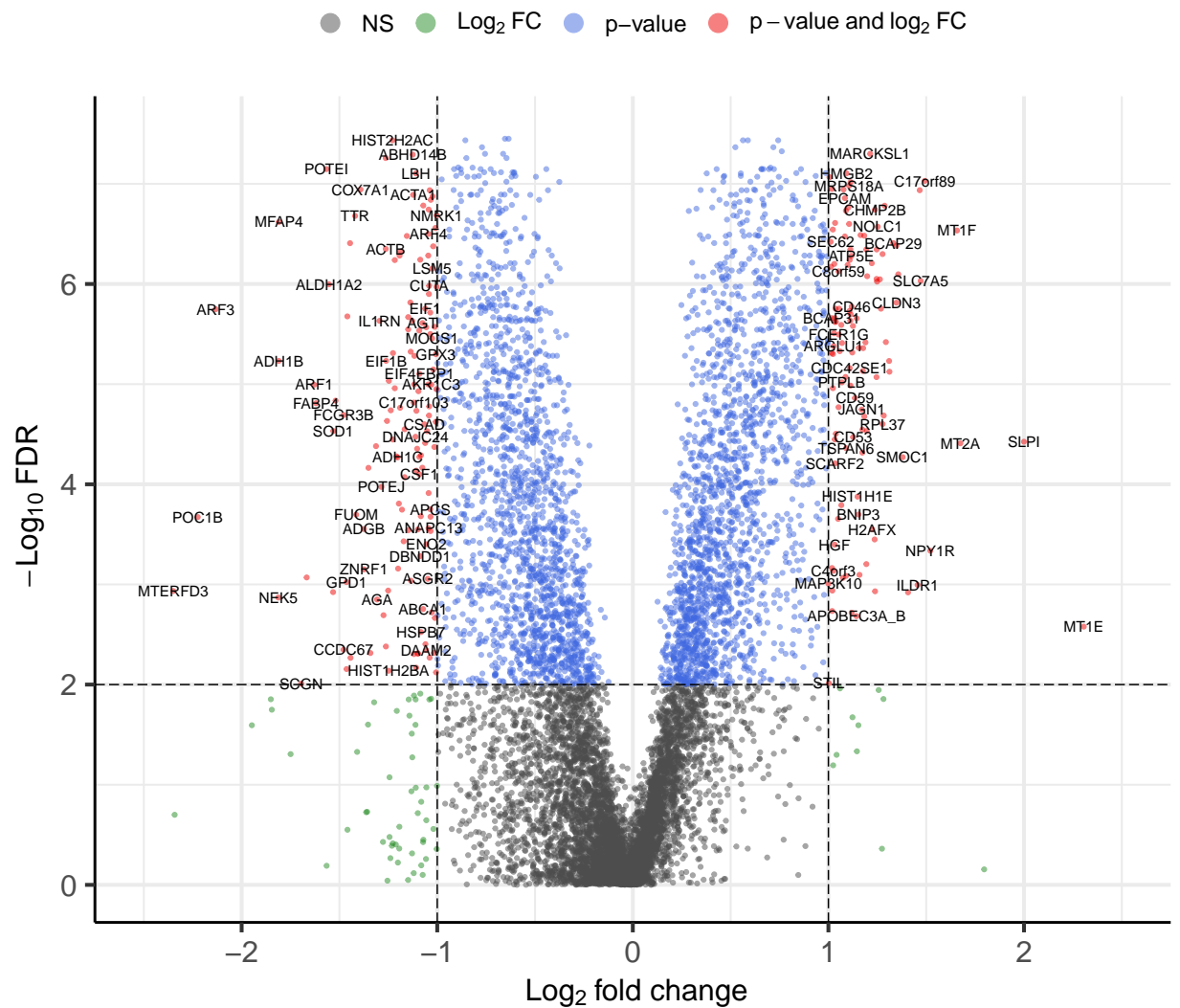
```
FCcutoff = 1,
pointSize = 1,
labSize = 3,
xlim = c(-2.5, 2.5),
ylim= c(0,7.5),
title = 'BRCA upper vs lower tertile',
ylab=bquote(~-Log[10]~ 'FDR'))
```

BRCA upper vs lower tertile

EnhancedVolcano

total = 9724 variables

```
#ggsave('output/brca_volcano.pdf', height = 10, width = 10)
```

Fig 4D

```r
b = read.csv('tcga_purity_tidyestimate_survival.csv')

b$spearman_10x = b$spearman*10

b %>% analyse_multivariate(vars(PFI.time, PFI),
                           covariates = vars(spearman_10x, age, purity, ty),
                           ) -> result
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `factor.name = map_chr(coefficient_labels, symbol_substring,
##   call_symbols)`.
## Caused by warning:
## ! `as_logical()` is deprecated as of rlang 0.4.0
## Please use `vctrs::vec_cast()` instead.
## This warning is displayed once every 8 hours.
```

```r
forest_plot(result, ggtheme = ggplot2::theme_bw(base_size = 10))
```

```
## Warning: `as_list()` is deprecated as of rlang 0.4.0
## Please use `vctrs::vec_cast()` instead.
## This warning is displayed once every 8 hours.
```

```
## Warning: `switch_type()` is soft-deprecated as of rlang 0.4.0.
## Please use `switch(typeof())` or `switch(my_typeof())` instead.
## This warning is displayed once every 8 hours.
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.36-1.90)' in 'mbcsToSbcs': dot substituted for <e2>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.36-1.90)' in 'mbcsToSbcs': dot substituted for <80>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.36-1.90)' in 'mbcsToSbcs': dot substituted for <93>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.99-1.02)' in 'mbcsToSbcs': dot substituted for <e2>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.99-1.02)' in 'mbcsToSbcs': dot substituted for <80>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.99-1.02)' in 'mbcsToSbcs': dot substituted for <93>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.43-1.47)' in 'mbcsToSbcs': dot substituted for <e2>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.43-1.47)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.43-1.47)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.28-0.53)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.28-0.53)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.28-0.53)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.36-1.90)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.36-1.90)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.36-1.90)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.99-1.02)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.99-1.02)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.99-1.02)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.43-1.47)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.43-1.47)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.43-1.47)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.28-0.53)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.28-0.53)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.28-0.53)' in 'mbcsToSbcs': dot substituted for <93>
```
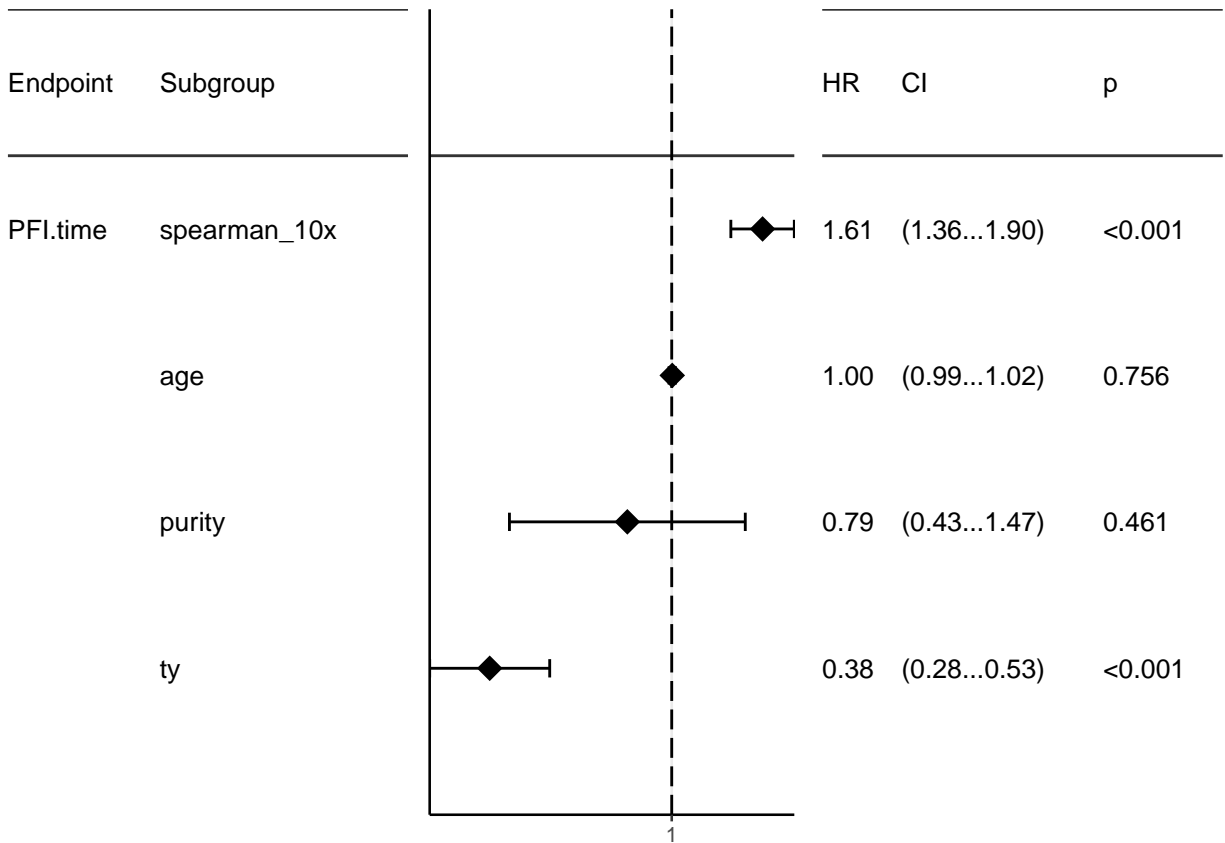
Endpoint	Subgroup		HR	CI	p
PFI.time	spearman_10x		1.61	(1.36...1.90)	<0.001
	age		1.00	(0.99...1.02)	0.756
	purity		0.79	(0.43...1.47)	0.461
	ty		0.38	(0.28...0.53)	<0.001

```
#ggsave('pdf/TCGA_all_pfi_new3_10x.pdf')
```

Fig 4E

```
b %>% analyse_multivariate(vars(OS.time, OS),
                     covariates = vars(spearman_10x, age, purity, ty),
                     ) -> result
forest_plot(result, ggtheme = ggplot2::theme_bw(base_size = 10))
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.14-1.61)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.14-1.61)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.14-1.61)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.02-1.05)' in 'mbcsToSbcs': dot substituted for <e2>
```

```
## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.02-1.05)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(1.02-1.05)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.40-1.60)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.40-1.60)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.40-1.60)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.34-0.68)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.34-0.68)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in strwidth(., family = ggtheme$text$family, units = "in"): conversion
## failure on '(0.34-0.68)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.14-1.61)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.14-1.61)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.14-1.61)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.02-1.05)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.02-1.05)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(1.02-1.05)' in 'mbcsToSbcs': dot substituted for <93>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.40-1.60)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.40-1.60)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.40-1.60)' in 'mbcsToSbcs': dot substituted for <93>
```

```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.34-0.68)' in 'mbcsToSbcs': dot substituted for <e2>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.34-0.68)' in 'mbcsToSbcs': dot substituted for <80>

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '(0.34-0.68)' in 'mbcsToSbcs': dot substituted for <93>
```
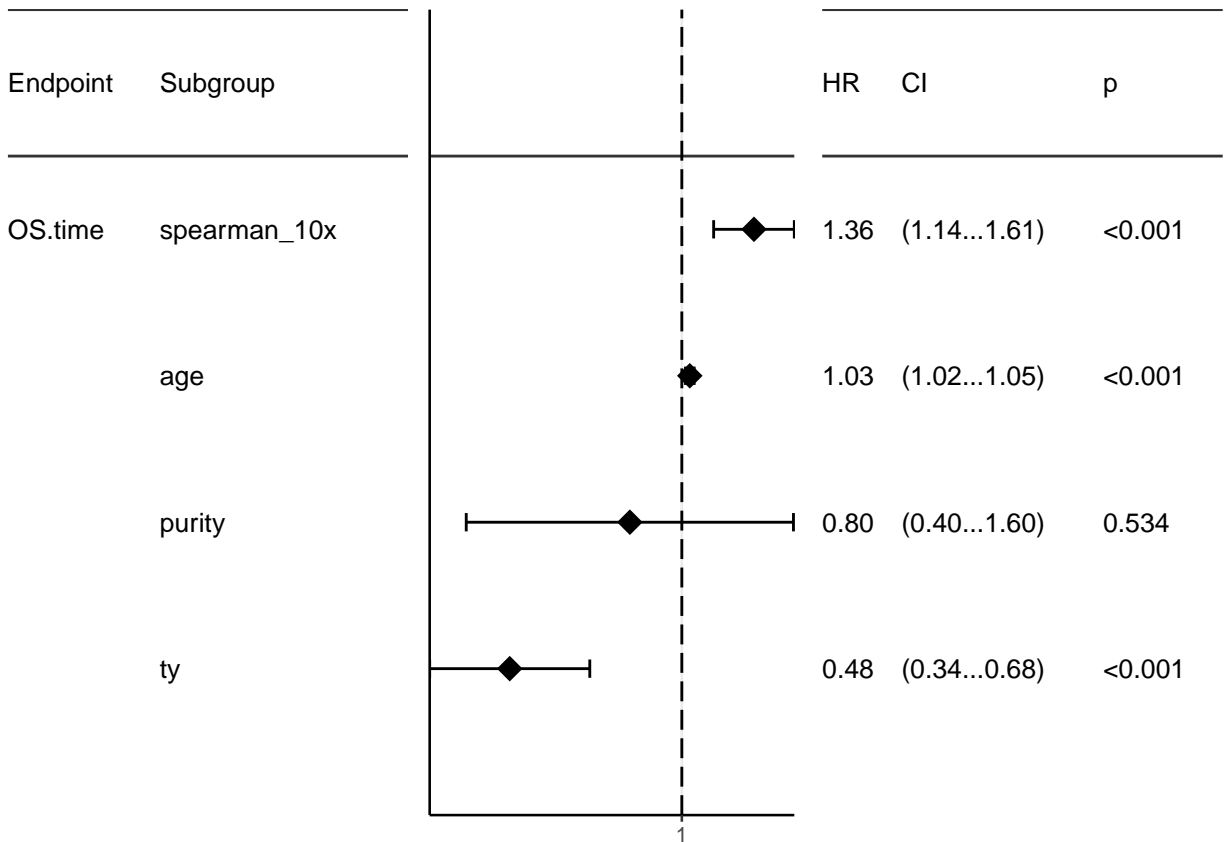
Endpoint	Subgroup		HR	CI	p
OS.time	spearman_10x		1.36	(1.14...1.61)	<0.001
	age		1.03	(1.02...1.05)	<0.001
	purity		0.80	(0.40...1.60)	0.534
	ty		0.48	(0.34...0.68)	<0.001

Fig 4B

```
low =filter(b, spearman < quantile(b$spearman,1/3,  na.rm=T)[[1]] )
high=filter(b, spearman > quantile(b$spearman,2/3,  na.rm=T)[[1]] )

low[,'group'] ='low'
high[,'group']='high'
d=rbind(low,high)

km=survfit( Surv(PFI.time, PFI) ~ group, d)
ggsurvplot(km, pval = T, title ='spearman', palette = c('red4','blue4'))
```

spearman

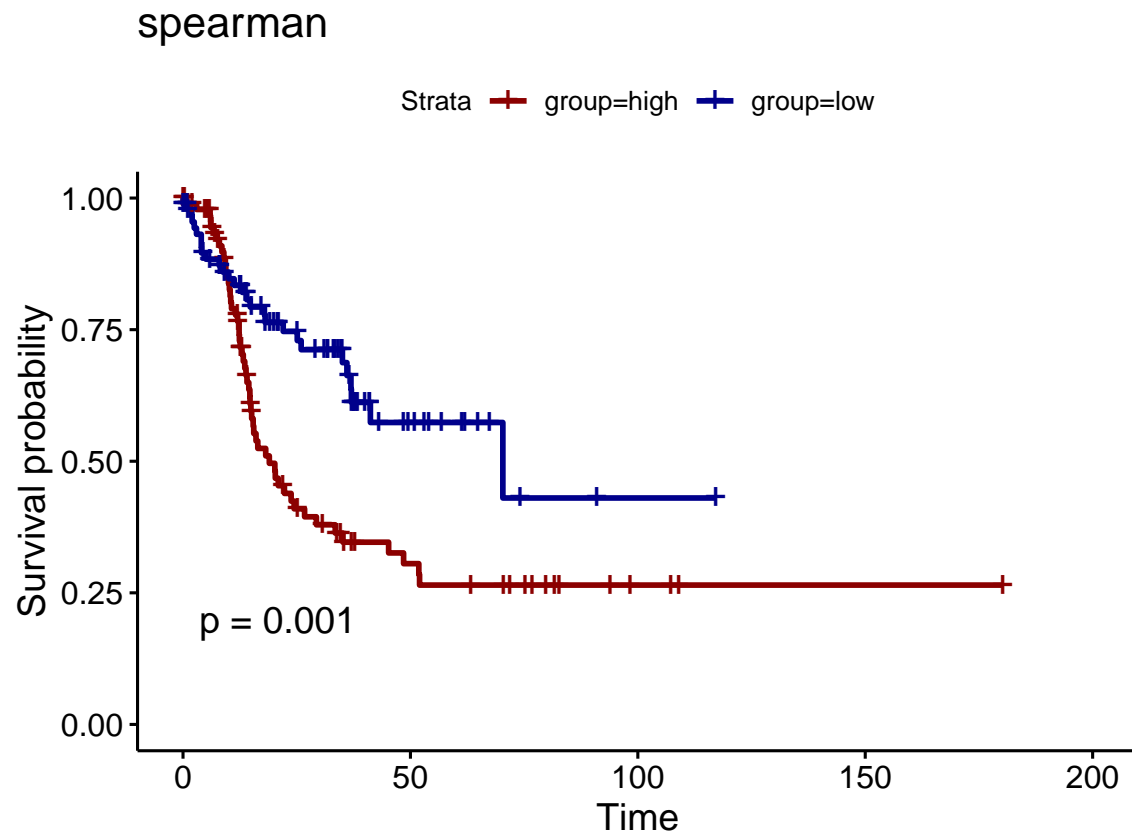**Fig 4A**

```
low =filter(b, spearman < quantile(b$spearman,1/3,  na.rm=T)[[1]] )
high=filter(b, spearman > quantile(b$spearman,2/3,  na.rm=T)[[1]] )

low[,'group'] ='low'
high[,'group']='high'
d=rbind(low,high)

km=survfit( Surv(OS.time, OS) ~ group, d)
ggsurvplot(km, pval = T, title ='spearman', palette = c('red4','blue4'))
```
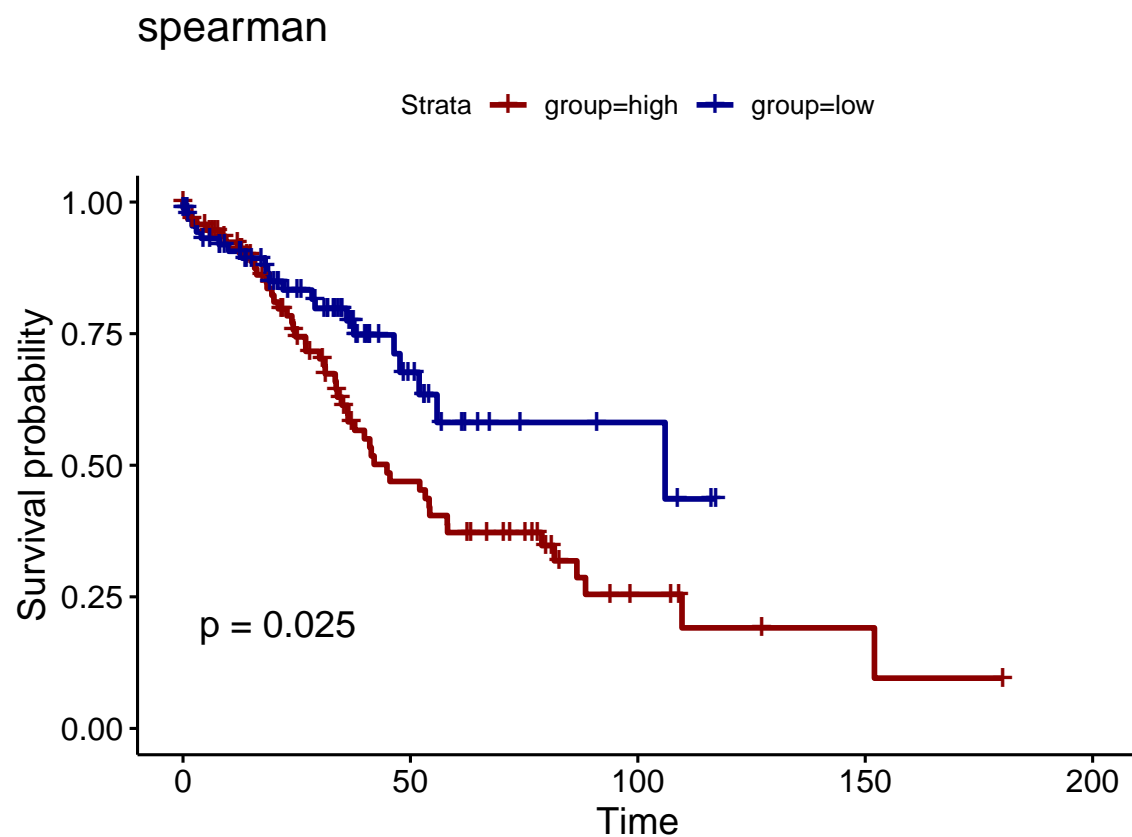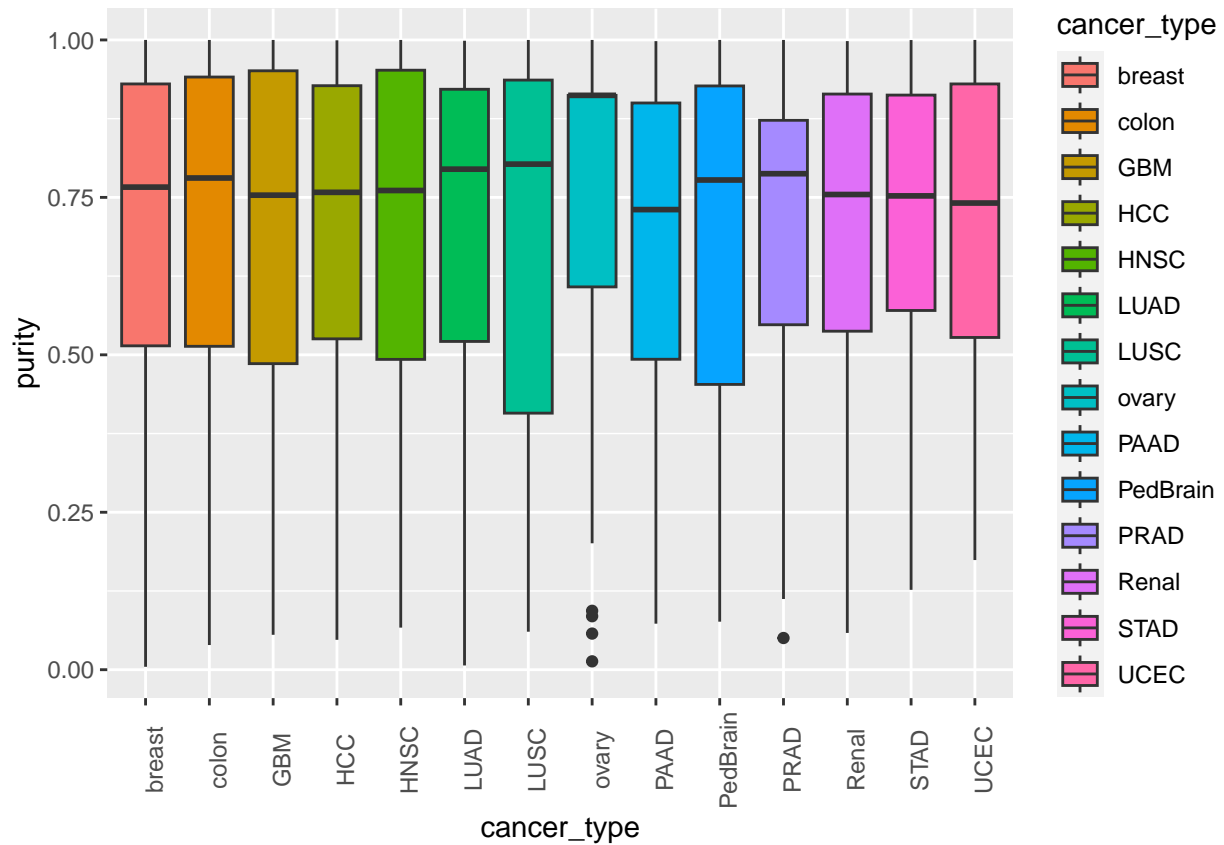
spearman

Fig S4

```
pur=read.csv('data/purity_tidyestimate_all_anno.csv')
ggplot(pur, aes(cancer_type,purity, fill=cancer_type)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 90))
```

```
## Warning: Removed 8 rows containing non-finite values ('stat_boxplot()').
```

```
#ggsave('output/tumor_purity_boxplot.pdf')
```
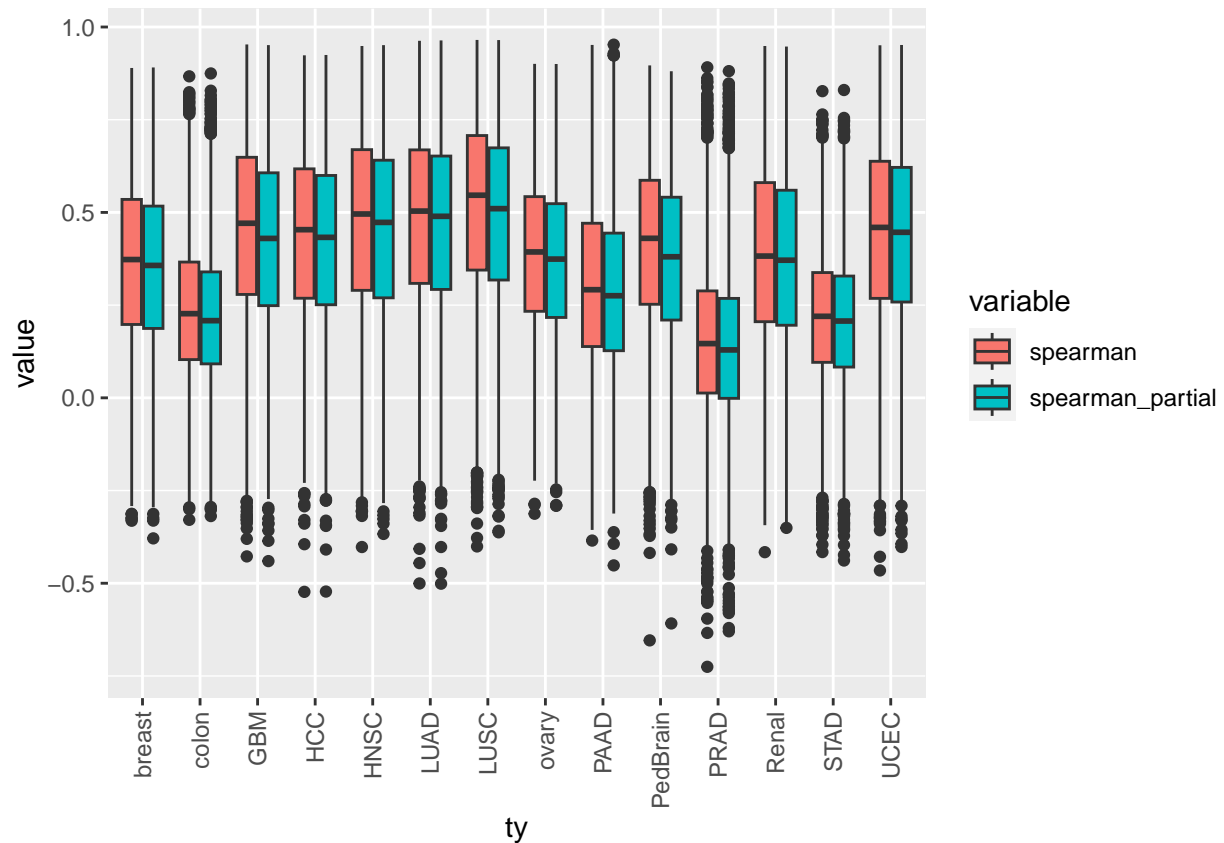
```
df=read.csv('data/spearman_partial_corr_cancer_types.csv')
```

```
d=reshape2::melt(df)
```

```
## Using X, ty as id variables
```

```
ggplot(d, aes(ty, value, fill=variable)) +
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))
```

```
## Warning: Removed 10 rows containing non-finite values ('stat_boxplot()').
```

Fig S4 C

```r
e=dcast(df, X ~ ty, value.var = 'spearman_partial' ) %>% tibble::column_to_rownames('X')
e=na.omit(e)

# not so good, use spearman correlation distance
dis_col = as.dist(1 - cor(e    , method = "spearman" ) )
#dis_row = as.dist(1 - cor(t(df), method = "spearman" ) )

# dont show gene names
pheatmap::pheatmap(e,
        show_rownames=FALSE,
        #filename = 'rna_prot_partial_corr_heatmap1.pdf',
        cutree_rows=8,
        clustering_method = 'ward.D2',
        clustering_distance_cols = dis_col,
        width = 4,
        height = 10)
```
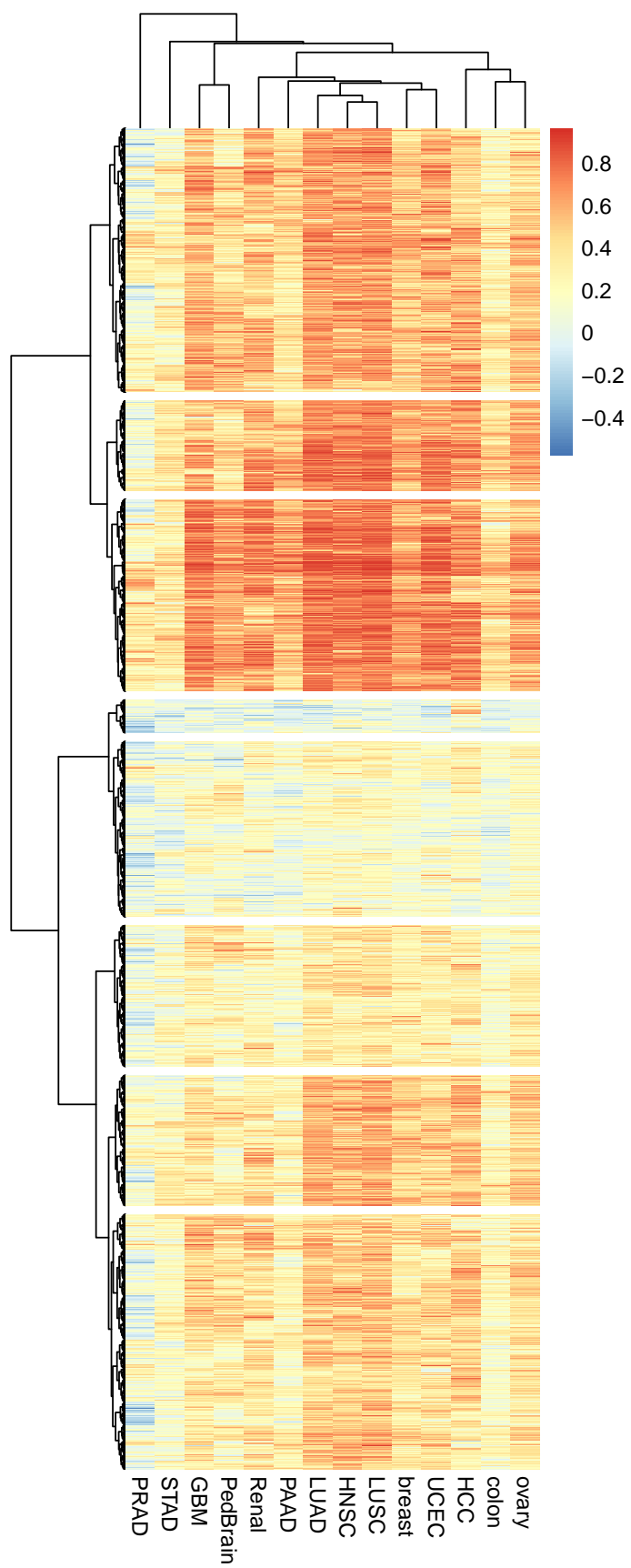
Fig S8

```
library(pathview)


##

## ###############################################################################
## Pathview is an open source software package distributed under GNU General
## Public License version 3 (GPLv3). Details of GPLv3 is available at
## http://www.gnu.org/licenses/gpl-3.0.html. Particullary, users are required to
## formally cite the original Pathview paper (not just mention it) in publications
## or products. For details, do citation("pathview") within R.
##
## The pathview downloads and uses KEGG data. Non-academic uses may require a KEGG
## license agreement (details at http://www.kegg.jp/kegg/legal.html).
## ###############################################################################

df=read.csv('data/brca_protein_spearman_high_vs_low_tertile_de_padj.csv', row.names = 1)
df=subset(df, padj_ttest<0.01)

pathview(gene.data =df[1], pathway.id = "03040", gene.idtype = 'SYMBOL')


## 'select()' returned 1:many mapping between keys and columns

## [1] "Note: 205 of 3794 unique input IDs unmapped."

## 'select()' returned 1:1 mapping between keys and columns

## Info: Working in directory /home/umesh/Downloads/proteomics_code

## Info: Writing image file hsa03040.pathview.png
```

Fig 1H, 1I, S2 pathway enrichment analysis

```
# # gene cluster annotations
# cl=read.csv('data/cluster_genes.csv')
#
# pdf('output/rna_prot_corr_cluster_GO.pdf', height=5,width=9 )
#
# # ,'GO_Cellular_Component_2021','GO_Molecular_Function_2021'
# for ( pathway in c('GO_Biological_Process_2021') ) {
#
#   for (c in c(1:max(cl$cluster))) {
#     er=enrichr(subset(cl, cluster==c)$gene,'GO_Biological_Process_2021')[[1]]
#
#     er1 = er %>% filter(Adjusted.P.value <0.25)
```
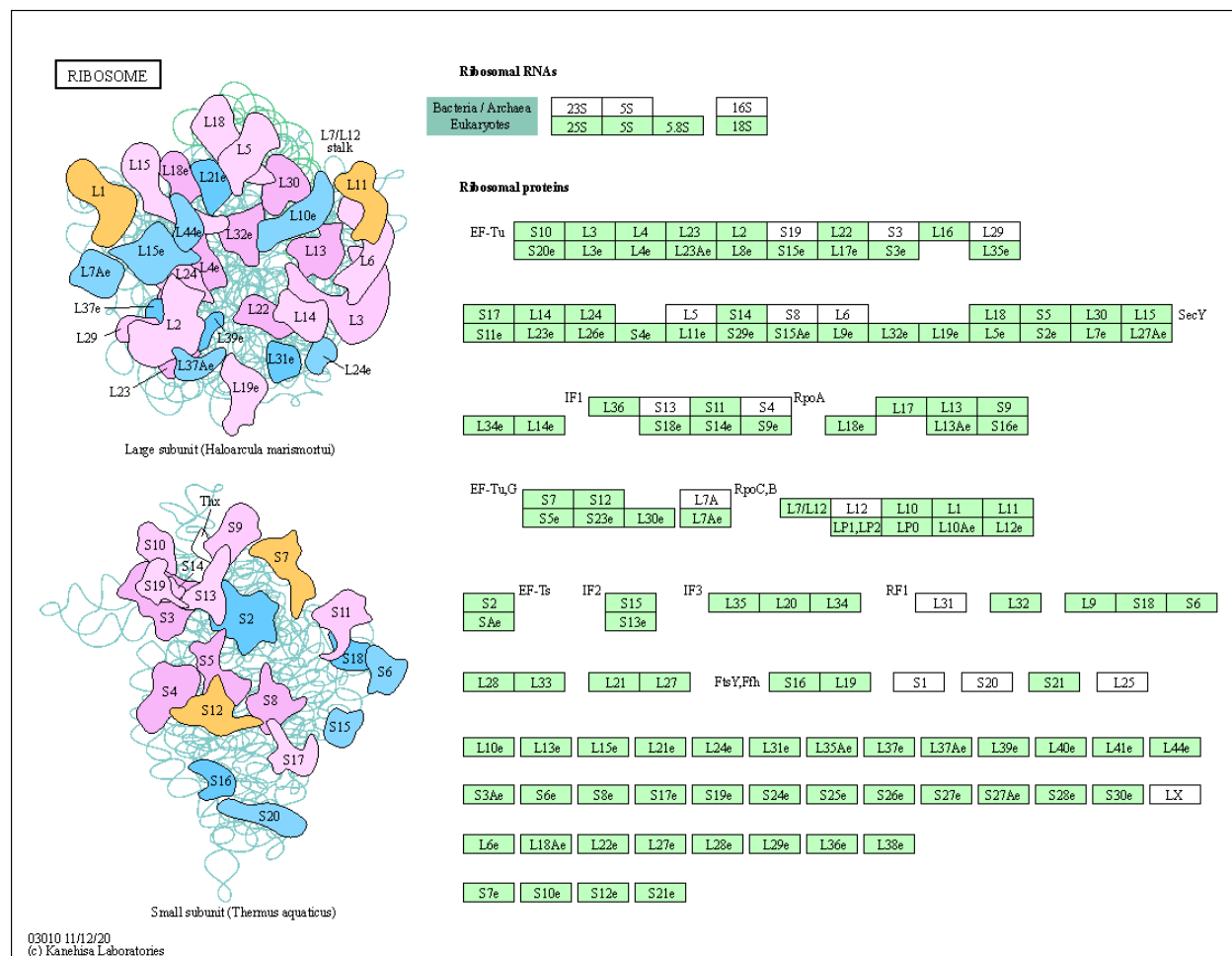
Figure 1: Fig S8

```
#
#     er1$fraction_overlap = sapply(er1$Overlap, function(x) eval(parse(text=x)))
#
#     er1$overlap = as.double(str_split_fixed(er1$Overlap,'/',2)[,1])
#
#     er1$padj=er1$Adjusted.P.value
#
#     p= ggplot(er1[1:20,], aes(fraction_overlap, Term, size= overlap, color=-log10(padj))) + geom_poin
#
#     print(p)
#   }
#
# }
#
# dev.off()
```

```
sessionInfo()
```

```
## R version 4.2.3 (2023-03-15)
## Platform: x86_64-conda-linux-gnu (64-bit)
## Running under: Ubuntu Lunar Lobster (development branch)
##
## Matrix products: default
## BLAS/LAPACK: /home/umesh/miniconda3/envs/r4.2/lib/libopenblasp-r0.3.21.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_GB.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=de_CH.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=de_CH.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=de_CH.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] grid      stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] pathview_1.38.0        EnhancedVolcano_1.16.0 ggrepel_0.9.3
##  [4] stringr_1.5.0          enrichR_3.2            survminer_0.4.9
##  [7] ggpubr_0.6.0           survivalAnalysis_0.3.0 survival_3.5-5
## [10] forcats_1.0.0          viridis_0.6.2          viridisLite_0.4.1
## [13] ComplexHeatmap_2.14.0  dplyr_1.1.1            tidyr_1.3.0
## [16] reshape2_1.4.4         circlize_0.4.15        ggplot2_3.4.1
##
## loaded via a namespace (and not attached):
##   [1] colorspace_2.1-0     ggsignif_0.6.4       rjson_0.2.21
##   [4] XVector_0.38.0       GlobalOptions_0.1.2  clue_0.3-64
##   [7] rstudioapi_0.14      farver_2.1.1         bit64_4.0.5
##  [10] AnnotationDbi_1.60.0 fansi_1.0.4          codetools_0.2-19
##  [13] splines_4.2.3        doParallel_1.0.17    cachem_1.0.7
##  [16] knitr_1.42           broom_1.0.4          km.ci_0.5-6
##  [19] cluster_2.1.4        png_0.1-8            pheatmap_1.0.12
```

```
##  [22] graph_1.76.0          compiler_4.2.3        httr_1.4.5
##  [25] backports_1.4.1       Matrix_1.5-3          fastmap_1.1.1
##  [28] cli_3.6.1             htmltools_0.5.5       tools_4.2.3
##  [31] gtable_0.3.3          glue_1.6.2            GenomeInfoDbData_1.2.9
##  [34] Rcpp_1.0.10           Biobase_2.58.0        carData_3.0-5
##  [37] vctrs_0.6.1           Biostrings_2.66.0     iterators_1.0.14
##  [40] xfun_0.38             lifecycle_1.0.3       WriteXLS_6.4.0
##  [43] rstatix_0.7.2         XML_3.99-0.14         org.Hs.eg.db_3.16.0
##  [46] zlibbioc_1.44.0       zoo_1.8-11            scales_1.2.1
##  [49] parallel_4.2.3        KEGGgraph_1.58.0      RColorBrewer_1.1-3
##  [52] yaml_2.3.7            curl_4.3.3            memoise_2.0.1
##  [55] gridExtra_2.3         KMsurv_0.1-5          stringi_1.7.12
##  [58] RSQLite_2.3.0         highr_0.10            S4Vectors_0.36.0
##  [61] foreach_1.5.2         BiocGenerics_0.44.0   shape_1.4.6
##  [64] GenomeInfoDb_1.34.8   rlang_1.1.0           pkgconfig_2.0.3
##  [67] matrixStats_0.63.0    bitops_1.0-7          evaluate_0.20
##  [70] lattice_0.20-45       purrr_1.0.1           labeling_0.4.2
##  [73] bit_4.0.5             cowplot_1.1.1         tidyselect_1.2.0
##  [76] plyr_1.8.8            magrittr_2.0.3        R6_2.5.1
##  [79] IRanges_2.32.0        generics_0.1.3        DBI_1.1.3
##  [82] pillar_1.9.0          withr_2.5.0           tidytidbits_0.3.2
##  [85] KEGGREST_1.38.0       abind_1.4-5           RCurl_1.98-1.12
##  [88] tibble_3.2.1          crayon_1.5.2          car_3.1-2
##  [91] survMisc_0.5.6        utf8_1.2.3            rmarkdown_2.21
##  [94] GetoptLong_1.0.5      data.table_1.14.8     blob_1.2.4
##  [97] Rgraphviz_2.42.0      digest_0.6.31         xtable_1.8-4
## [100] stats4_4.2.3          munsell_0.5.0
```