

R implementation

Sleiman Bassim, PhD

February 19, 2016

1 Loaded functions:

```
#source("/media/Data/Dropbox/humanR/01funcs.R")
rm(list=ls())
#setwd("/media/Data/Dropbox/humanR/PD/")
#setwd("~/Dropbox/humanR/PD/")
###load("PD.Rdata", .GlobalEnv)
#lsos(pat="")
```

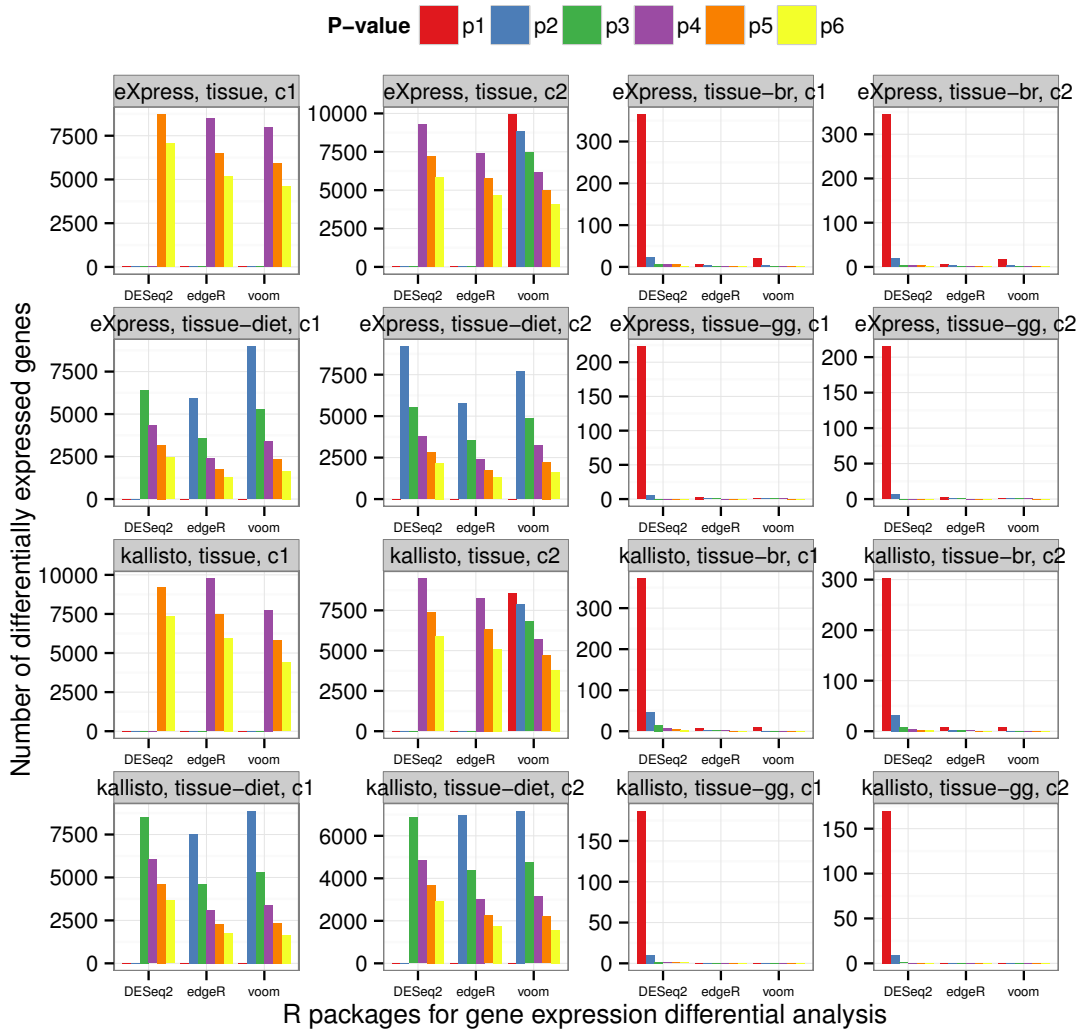
2 Load packages.

```
pkgs <- c('xlsx', 'caret', 'leaps', 'glmnet', 'lattice',
          'latticeExtra', 'dplyr', 'tidyr')
lapply(pkgs, require, character.only = TRUE)
```

3 1 Differentially expressed genes

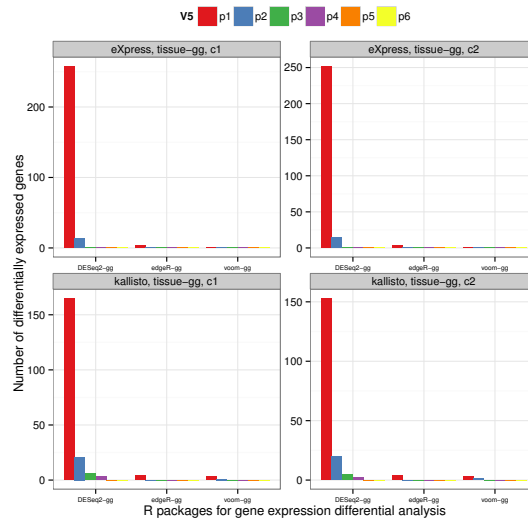
4 Differentially expressed genes are counted from mapping **both gills and ganglia** sequenced samples to
5 reference transcriptome built from all samples.

```
read.table("./data/summary.raw.all.txt") %>%
  ggplot(aes(
    x = V1,
    y = V8,
    fill = V6)) +
  theme_bw() +
  geom_bar(stat = "identity",
    position = "dodge") +
  facet_wrap(~ V4 + V5 + V7,
    ncol = 4,
    scales = "free") +
  scale_fill_brewer(type = "qual", palette = 6,
    name = "P-value") +
  labs(x = "R packages for gene expression differential analysis",
    y = "Number of differentially expressed genes") +
  theme(legend.position = "top",
    axis.text.x = element_text(vjust = .5,
      size = 6))
```



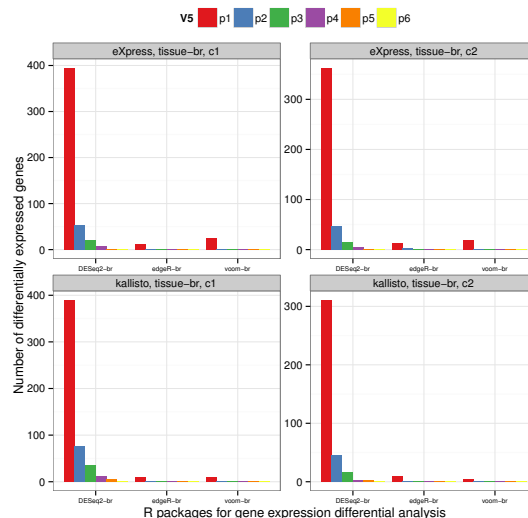
6
7 Differentially expressed genes are counted from mapping **gills** sequenced samples to reference tran-
8 scriptome built from all samples.

```
read.table("./data/summary.gg.txt") %>%
  ggplot(aes(
    x = V2,
    y = V8,
    fill = V5)) +
  geom_bar(stat = "identity",
    position = "dodge") +
  facet_wrap(~ V3 + V4 + V6,
    ncol = 2,
    scales = "free") +
  scale_fill_brewer(type = "qual", palette = 6) +
  theme_bw() +
  theme(legend.position = "top",
    axis.text.x = element_text(vjust = .5,
      size = 6)) +
  labs(x = "R packages for gene expression differential analysis",
    y = "Number of differentially expressed genes")
```



9
10 Differentially expressed genes are counted from mapping **ganglia** sequenced samples to reference trans-
11 criptome built from all samples.

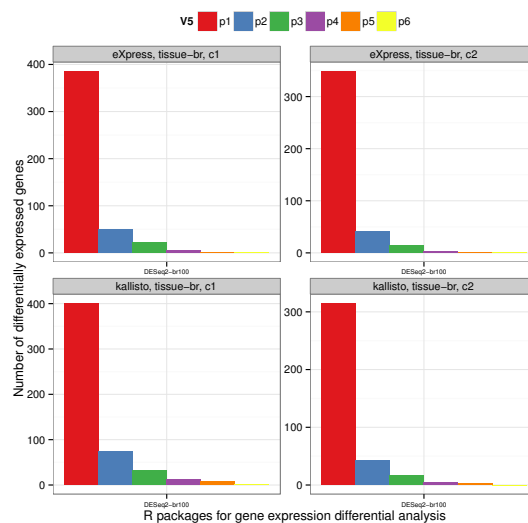
```
read.table("./data/summary.br.txt") %>%
  ggplot(aes(
    x = V2,
    y = V8,
    fill = V5)) +
  geom_bar(stat = "identity",
    position = "dodge") +
  facet_wrap(~ V3 + V4 + V6,
    ncol = 2,
    scales = "free") +
  scale_fill_brewer(type = "qual", palette = 6) +
  theme_bw() +
  theme(legend.position = "top",
    axis.text.x = element_text(vjust = .5,
      size = 6)) +
  labs(x = "R packages for gene expression differential analysis",
    y = "Number of differentially expressed genes")
```



12
13 **1.1 Increasing DEG by changing the trimming rates of raw reads**
14 Getting gene expression by mapping the original raw reads **without trimming** to the **gills** de novo trans-
15 criptome.

† De novo assembly was carried out with trimmed reads though

```
read.table("./data/summary.br_35078.txt") %>%
  ggplot(aes(
    x = V2,
    y = V8,
    fill = V5)) +
  geom_bar(stat = "identity",
    position = "dodge") +
  facet_wrap(~ V3 + V4 + V6,
    ncol = 2,
    scales = "free") +
  scale_fill_brewer(type = "qual", palette = 6) +
  theme_bw() +
  theme(legend.position = "top",
    axis.text.x = element_text(vjust = .5,
      size = 6)) +
  labs(x = "R packages for gene expression differential analysis",
    y = "Number of differentially expressed genes")
```



2 A linear representation of gene expression

Only selected genes can be represented as follow.

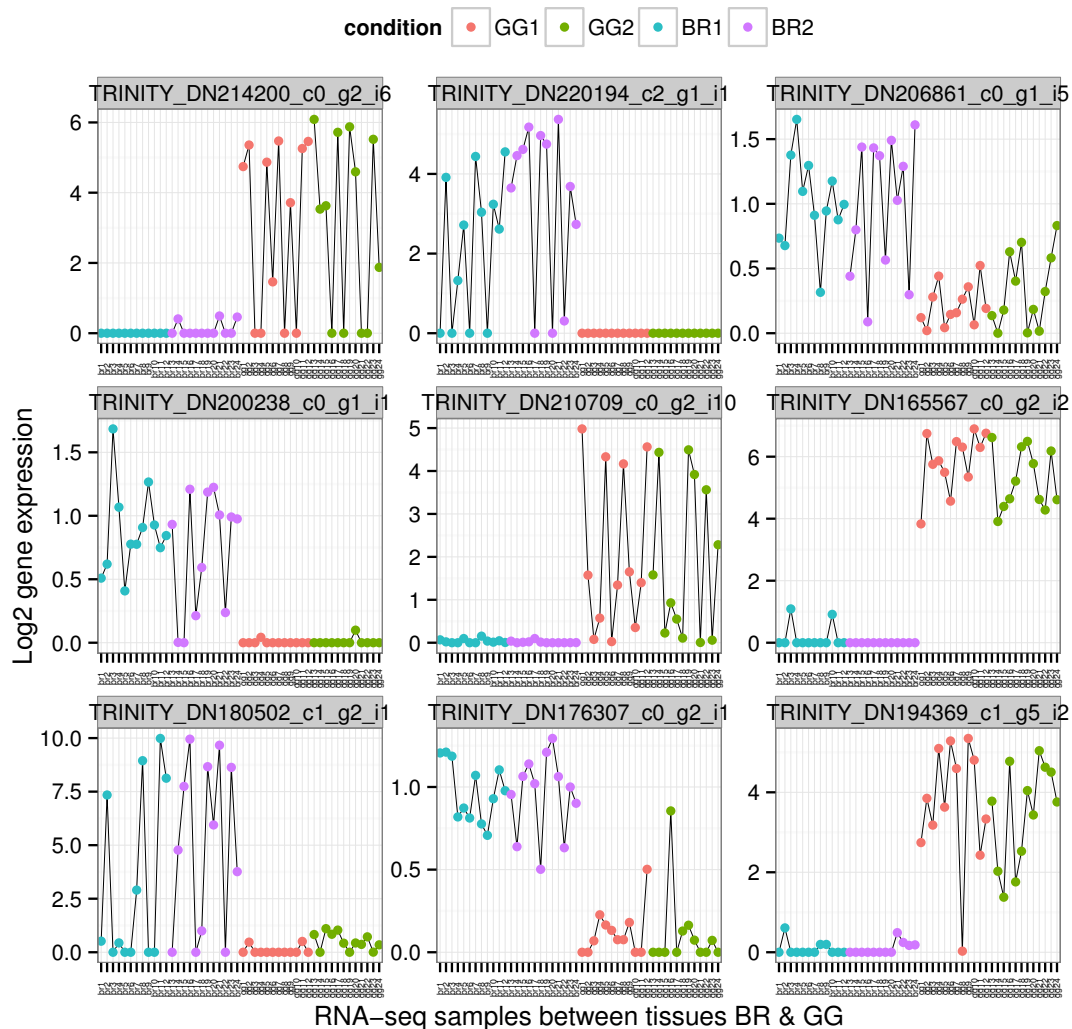
† Not more than 10 genes

```
dat <- t(read.table("./data/test.txt"))
```

```

dat <- data.frame(dat,
                  sample = rownames(dat),
                  condition = gl(4,12,48,
                                labels=c("GG1", "GG2", "BR1", "BR2")))
dat %>%
  gather("genes", "expression", 1:(dim(dat)[2]-2)) %>%
  ggplot(aes(x = factor(sample,
                        c(paste("br", seq(1,24), sep=""),
                          paste("gg", seq(1,24), sep=""))),
            y = expression,
            group = condition)) +
  theme_bw() +
  geom_line(size = .2) +
  geom_point(aes(x = factor(sample),
                       y = expression,
                       colour = condition)) +
  facet_wrap(~ genes,
             ncol = 3,
             scales = "free") +
  labs(x = "RNA-seq samples between tissues BR & GG",
       y = "Log2 gene expression") +
  theme(legend.position = "top",
        axis.text.x = element_text(angle = 90,
                                     vjust = .5,
                                     size = 4)) +
  scale_fill_brewer(type = "qual", palette = 6,
                    name = "Oyster tissues and Diet conditions")

```



19

3 System Information

20 The version number of R and packages loaded for generating the vignette were:

21

```
###save(list=ls(pattern=".*\\.\\.*"),file="PD.Rdata")
```

sessionInfo()

R version 3.2.1 (2015-06-18)

Platform: x86_64-unknown-linux-gnu (64-bit)

Running under: elementary OS Luna

locale:

[1] LC_CTYPE=en_US.UTF-8	LC_NUMERIC=C
[3] LC_TIME=en_US.UTF-8	LC_COLLATE=en_US.UTF-8
[5] LC_MONETARY=en_US.UTF-8	LC_MESSAGES=en_US.UTF-8
[7] LC_PAPER=en_US.UTF-8	LC_NAME=en_US.UTF-8
[9] LC_ADDRESS=en_US.UTF-8	LC_TELEPHONE=en_US.UTF-8
[11] LC_MEASUREMENT=en_US.UTF-8	LC_IDENTIFICATION=en_US.UTF-8

attached base packages:

[1] stats	graphics	grDevices	utils	datasets	methods
[7] base					

other attached packages:

[1] tidyr_0.2.0	dplyr_0.4.2	latticeExtra_0.6-26
[4] RColorBrewer_1.1-2	glmnet_2.0-2	foreach_1.4.2
[7] Matrix_1.2-1	leaps_2.9	caret_6.0-47
[10] ggplot2_1.0.1	lattice_0.20-31	xlsx_0.5.7
[13] xlsxjars_0.6.1	rJava_0.9-6	knitr_1.10.5
[16] RevoUtilsMath_3.2.1		

loaded via a namespace (and not attached):

[1] Rcpp_0.11.6	compiler_3.2.1	formatR_1.2
[4] nloptr_1.0.4	plyr_1.8.3	highr_0.5
[7] iterators_1.0.7	tools_3.2.1	digest_0.6.8
[10] lme4_1.1-8	evaluate_0.7	nlme_3.1-121
[13] gtable_0.1.2	mgcv_1.8-6	DBI_0.3.1
[16] parallel_3.2.1	brglm_0.5-9	SparseM_1.6
[19] proto_0.3-10	BradleyTerry2_1.0-6	stringr_1.0.0
[22] gtools_3.5.0	grid_3.2.1	nnet_7.3-10
[25] R6_2.0.1	minqa_1.2.4	reshape2_1.4.1
[28] car_2.0-25	magrittr_1.5	scales_0.2.5
[31] codetools_0.2-11	MASS_7.3-41	splines_3.2.1
[34] assertthat_0.1	pbkrtest_0.4-2	colorspace_1.2-6
[37] labeling_0.3	quantreg_5.11	stringi_0.5-5
[40] lazyeval_0.1.10	munsell_0.4.2	