Mar 11, 2016

Script R12: Functional Analysis

F1000Research

In 1 collection

DOI
https://dx.doi.org/10.17504/protocols.io.ejibcke

HANNIGAN GD[1], GRICE EA[1], ET AL.[1]

[1]DEPARTMENT OF DERMATOLOGY UNIVERSITY OF PENNSYLVANIA

VERVE Net

Club Grice

Geoffrey D Hannigan
Merck Research Labs

OPEN ACCESS

DOI: https://dx.doi.org/10.17504/protocols.io.ejibcke

External link: http://mbio.asm.org/content/6/5/e01578-15.full

Protocol Citation: HANNIGAN GD, GRICE EA, ET AL. 2016. Script R12: Functional Analysis. **protocols.io** https://dx.doi.org/10.17504/protocols.io.ejibcke

Manuscript citation:
Kindler L, Stoliartchouk A, Teytelman L, Hurwitz BL, Method-centered digital communities on protocols.io for fast-paced scientific innovation. F1000Research doi: 10.12688/f1000research.9453.2

Protocol status: Working

Created: February 16, 2016

Last Modified: November 09, 2017

Protocol Integer ID: 2378

Keywords: kegg functional annotation, functional annotation, kegg, functional analysis this protocol, functional analysis, script r12, analysis

Abstract

This protocol outlines the analysis used to plot KEGG functional annotation.

Guidelines

sessionInfo()

```
## R version 3.2.0 (2015-04-16)
## Platform: x86_64-apple-darwin13.4.0 (64-bit)
## Running under: OS X 10.10.4 (Yosemite)
## ## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats graphics grDevices utils datasets methods base
##
## loaded via a namespace (and not attached):
## [1] magrittr_1.5 formatR_1.2 tools_3.2.0 htmltools_0.2.6
## [5] yaml_2.1.13 stringi_0.4-1 rmarkdown_0.7 knitr_1.10.5
## [9] stringr_1.0.0 digest_0.6.8 evaluate_0.7
```

Before start

Supplemental information available at:

https://figshare.com/articles/The_Human_Skin_dsDNA_Virome_Topographical_and_Temporal_Diversity_Genetic_Enrichment_and_Dynamic_Associations_with_the_Host_Microbiome/1281248

1 Load the required libraries.

Command

```
library(
```

Expected result

```
## [1] '1.8.2'
```

Expected result

```
##
## Attaching package: 'gplots'
##
## The following object is masked from 'package:stats':
##
## lowess
```

Expected result

```
## [1] '2.17.0'
```

2 Read in the metadata.

Command

```
metadata<-read.delim(
```

3 Import the tab delimited file containing the output from HUMAnN. Read in the data for the whole metagenome and standardize sample ID's.

Note

This file was generated using the shell script <>.

Command

```
mpm.s<-read.delim(
```

4 Read in data for the virome and standardize sample ID's.

Command

```
mpm.v<-read.delim(
```

5 The data has information in the first 17 rows that we are not going to use. It needs to be formatted for downstream analysis. The last column is also a mock community sample, which we need to remove.

Command

```
#remove unnecessary information
mpm.s<-mpm.s[-c(1:17),-ncol(mpm.s)]
mpm.v<-mpm.v[-c(1:17),-ncol(mpm.v)]
```

6 Keep only paired samples.

Command

```
metadata<-subset(metadata, metadata$NexteraXT_SampleID %in%
colnames(mpm.s))
metadata<-subset(metadata, metadata$NexteraXT_Virome_SampleID %in%
colnames(mpm.v))
mpm.s<-mpm.s[, colnames(mpm.s) %in% c(metadata$NexteraXT_SampleID,
```

7 Merge whole metagenome and virome output.

Command

```
mpm<-merge(mpm.s, mpm.v, by=c(
```

8 Looking at the KEGG modules, we see they are very specific. We want to categorize them at a higher level.

9 Read in the level information.

Note

I originally downloaed this information from http://www.genome.jp/kegg-bin/ and clicked "Download htext". This gave me the file ko00002.keg which I parsed to get level C information.

Command

```
factor(mpm$NAME[1:5])
mpm_levelC <- read.delim(
```

Expected result

```
## [1] M00001: Glycolysis (Embden-Meyerhof pathway), glucose => pyruvate
## [2] M00002: Glycolysis, core module involving three-carbon compounds
## [3] M00003: Gluconeogenesis, oxaloacetate => fructose-6P
## [4] M00004: Pentose phosphate pathway (Pentose phosphate cycle)
## [5] M00006: Pentose phosphate pathway, oxidative phase, glucose 6P => ribulose 5P
## 5 Levels: M00001: Glycolysis (Embden-Meyerhof pathway), glucose => pyruvate ...
```

10 Aggregate modules that are in the same higher level.

Command

```
mpm_levs<-merge(mpm_levels, mpm, by=c(
```

11 Transpose matrix so that the rows are samples and columns are KEGG module levels.

Command

```
mpmheat<-t(mpm_agg)
```

12 Now that we have categorized the KEGG modules, we want to compare them between the whole metagenome and the virome. We need to format our data frames so that the metagenome and virome samples are in the same order and then do pairwise comparisons. Finally, we want to calculate the log fold change between the metagenome and virome for each KEGG category that is significantly different.

13 Order the skin metagenome samples.

Command

```
mpmheat.s<-merge(metadata[,c(
```

14 Order the virome samples.

Command

```
mpmheat.v<-merge(metadata[,c(
```

15 Do a wilcoxon to determine which modules are significantly different between skinmet and virome samples.

Note

Note this will throw warnings-- some of the modules are almost all 0's which is not optimal when running a wilcoxon. However this does seem to pick up the major differences.

Command

```
wil.test.out<-vector()
for( i in 1:ncol(mpmheat.s) ) { wil<-
wilcox.test(x=as.numeric(mpmheat.s[,i]),y=as.numeric(mpmheat.v[,i]),p.
adj=c(
```

16 Add a small number to the relative abundances so that you can log transform them.

Command

```
c<-1e-25
met<-mpmheat.s+c
vir<-mpmheat.v+c
```

17 Calculate the log fold change.

Command

```
delta<-log2(met/vir)
delta.k<-delta[,colnames(delta) %in% row.names(keep)]
```

18 Finally, lets plot a heatmap to visualize the differences. Green indicates enrichment in the whole metagenome while red indicates enrichment in the virome.

Command

```
heatmap.2(as.matrix(t(delta.k)),notecol=
```

Expected result