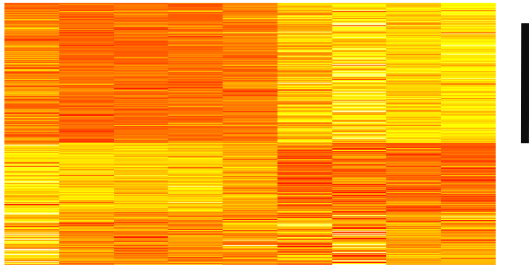


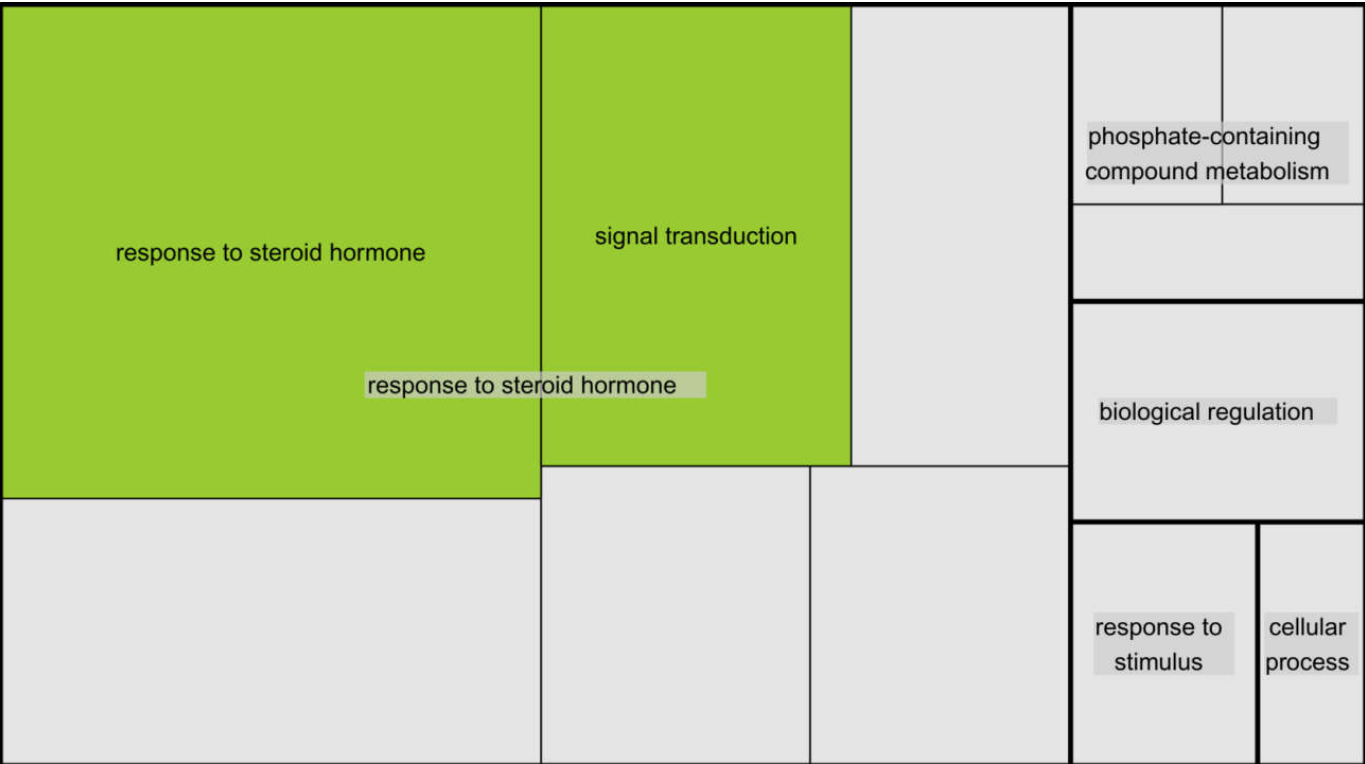
Gene Ontology Enrichment Analysis

Liu Min
2017/03/13

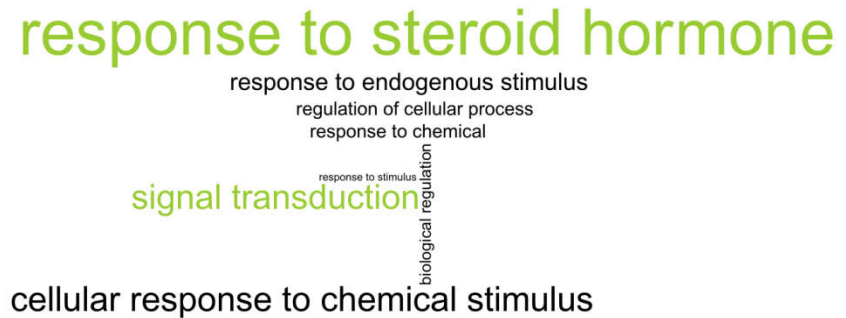
Data

Gene set	Species	Biological function	Data format
1 (27)	A.thaliana	BR-related signaling components	GeneID-GO
		Blastn Blastp ↓ Pfam	
2 (119)	G.max	Homologous genes of BR-related signaling components	GeneID-GO
			
3 (225/187)	G.max		Nucleotide sequences(.fa)
			Blastx ↓ Blast2go GeneID-GO

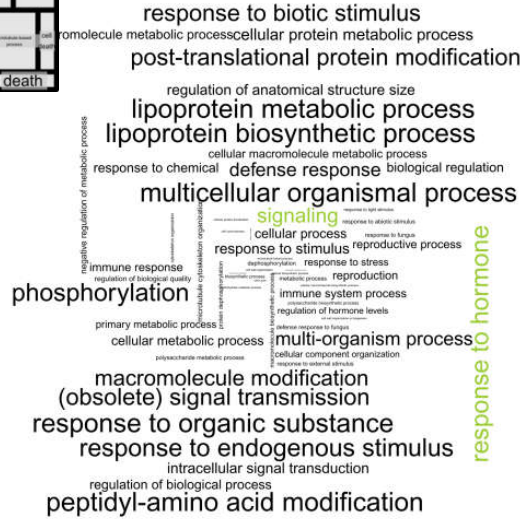
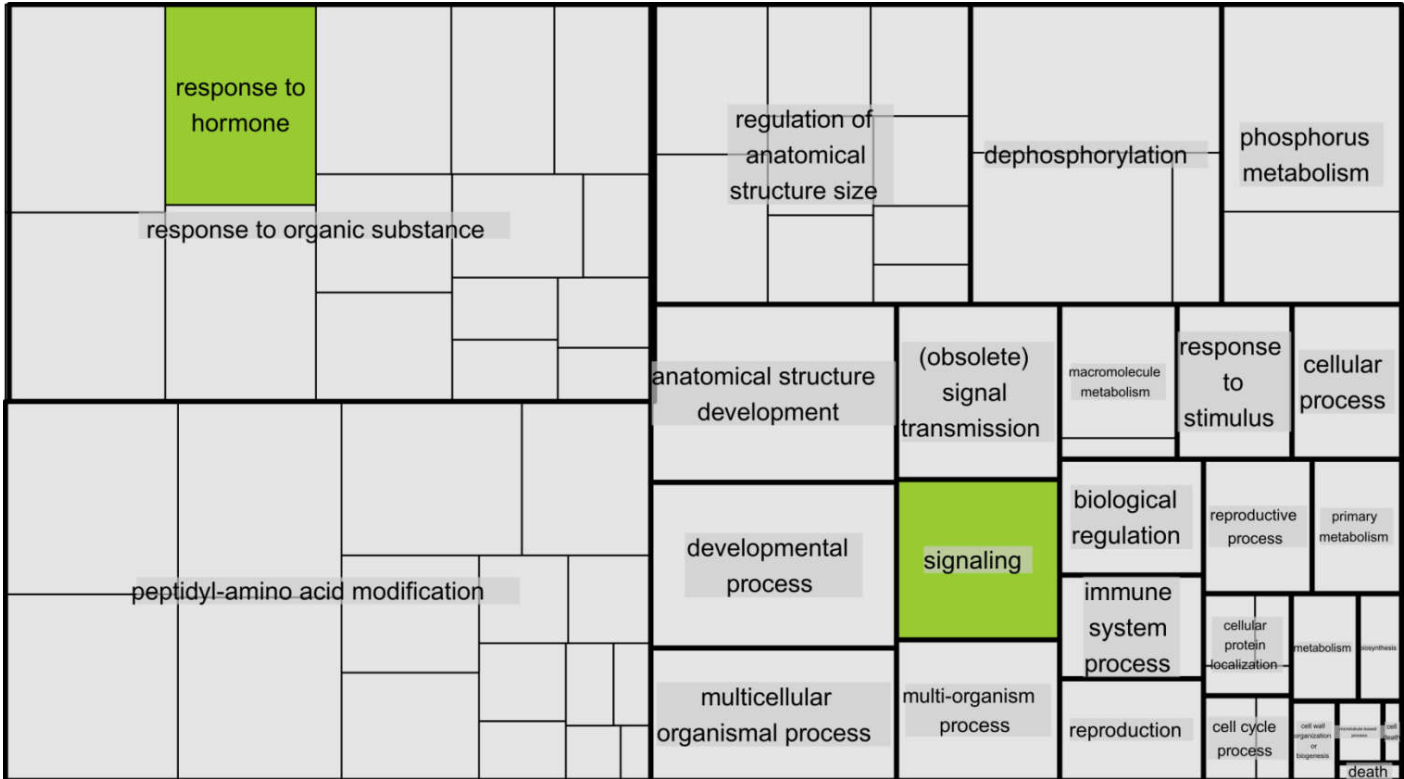
Result-1



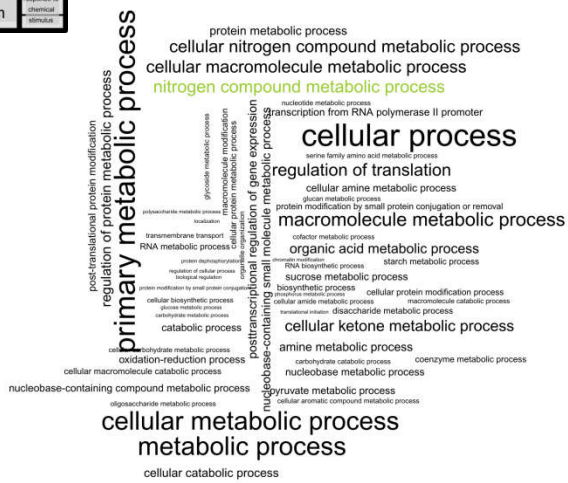
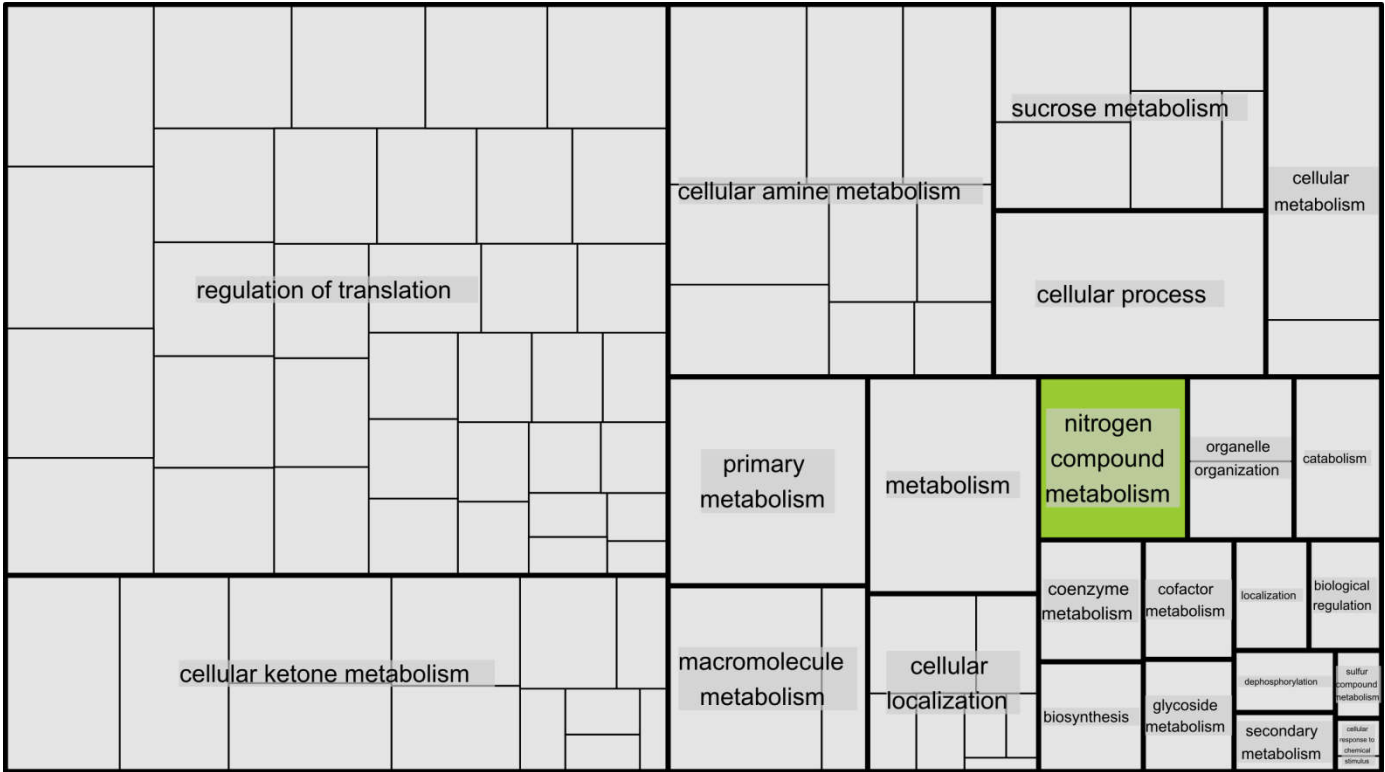
AgriGO: <http://bioinfo.cau.edu.cn/agriGO/>
 REVIGO: <http://revigo.irb.hr/>



Result-2



Result-3



Conclusion

- The function of specific gene set can be identified by gene ontology enrichment analysis
- Validity of GO analysis relies greatly on gene set.

Appendix-R code

```
# A treemap R script produced by the REVIGO server at http://revigo.irb.hr/
# If you found REVIGO useful in your work, please cite the following reference:
# Supek F et al. "REVIGO summarizes and visualizes long lists of Gene Ontology
# terms" PLoS ONE 2011. doi:10.1371/journal.pone.0021800

# author: Anton Kratz <anton.kratz@gmail.com>, RIKEN Omics Science Center, Functional Genomics Technology Team, Japan
# created: Fri, Nov 02, 2012 7:25:52 PM
# last change: Fri, Nov 09, 2012 3:20:01 PM
# -----
# If you don't have the treemap package installed, uncomment the following line:
# install.packages( "treemap" );
library(treemap) # treemap package by Martijn Tennekes
# Set the working directory if necessary
# setwd("C:/Users/username/workingdir");
# -----
# Here is your data from REVIGO. Scroll down for plot configuration options.

revigo.names <- c("term_ID","description","freqInDbPercent","abslog10pvalue","uniqueness","dispensability","representative");
revigo.data <- rbind(c("GO:0006796","phosphate-containing compound metabolic process",16.693,2.2291,0.816,0.000,"phosphate-
containing compound metabolism"),
  c("GO:0016310","phosphorylation",6.300,2.3565,0.814,0.649,"phosphate-containing compound metabolism"),
  c("GO:0006793","phosphorus metabolic process",16.891,2.2291,0.861,0.160,"phosphate-containing compound metabolism"),
  c("GO:0009987","cellular process",65.994,2.0506,0.968,0.000,"cellular process"),
  c("GO:0048545","response to steroid hormone",0.034,20.7959,0.315,0.000,"response to steroid hormone"),
  c("GO:0050794","regulation of cellular process",13.656,6.0655,0.644,0.629,"response to steroid hormone"),
  c("GO:0007165","signal transduction",3.804,11.1427,0.311,0.393,"response to steroid hormone"),
  c("GO:0042221","response to chemical",1.603,6.2596,0.474,0.598,"response to steroid hormone"),
  c("GO:0070887","cellular response to chemical stimulus",0.495,11.2147,0.352,0.611,"response to steroid hormone"),
  c("GO:0009719","response to endogenous stimulus",0.113,7.8861,0.548,0.440,"response to steroid hormone"),
  c("GO:0050896","response to stimulus",8.818,3.5528,0.915,0.000,"response to stimulus"),
  c("GO:0065007","biological regulation",14.918,5.0410,0.921,0.000,"biological regulation"));
```

Appendix-R code

```
stuff <- data.frame(revigo.data);
names(stuff) <- revigo.names;

stuff$abslog10pvalue <- as.numeric( as.character(stuff$abslog10pvalue) );
stuff$freqInDbPercent <- as.numeric( as.character(stuff$freqInDbPercent) );
stuff$uniqueness <- as.numeric( as.character(stuff$uniqueness) );
stuff$dispensability <- as.numeric( as.character(stuff$dispensability) );

# by default, outputs to a PDF file
#pdf( file="revigo_treemap.pdf", width=16, height=9 ) # width and height are in inches
# check the tmPlot command documentation for all possible parameters - there are a lot more
tmPlot(
  stuff,
  index = c("representative","description"),
  vSize = "abslog10pvalue",
  type = "categorical",
  vColor = "representative",
  title = "REVIGO Gene Ontology treemap",
  inflate.labels = FALSE,    # set this to TRUE for space-filling group labels - good for posters
  lowerbound.cex.labels = 0, # try to draw as many labels as possible (still, some small squares may not get a label)
  bg.labels = "#CCCCCCAA",  # define background color of group labels
  # "#CCCCCC00" is fully transparent, "#CCCCCCAA" is semi-transparent grey, NA is opaque
  position.legend = "none"
)
#dev.off()

#wordcloud
library(wordcloud)
wordcloud(stuff$description,stuff$abslog10pvalue,scale=c(2,0))
```