Epistatic Nested Effects Models
Inferring mixed epistatis from indirect measurements of knockout screens.

Madeline Diekmann & Martin Pirkl

November 22, 2016

This package is an extension of the classic Nested Effects Models provided in package *nem*. Nested Effects Models is a pathway reconstruction method, which takes into account effects of downstream genes. Those effects are observed for every knockout of a pathway gene, and the nested structure of observed effects can then be used to reconstruct the pathway structure. However, classic Nested Effects Models do not account for double knockouts. In this package *epiNEM*, one additional layer of complexity is added. For every two genes, acting on one gene together, the relationship is evaluated and added to the model as a logic gate. Genetic relationships are represented by the logics OR (no relationship), AND (functional overlap), NOT (masking or inhibiting) and XOR (mutual prevention from acting on gene C).

Loading epiNEM

```
## install.packages("devtools", verbose = F, quiet = T)

library(devtools)

## install_github("cbg-ethz/epiNEM", quiet = T)

library(epiNEM)
```

```
## Loading required package:  BoolNet
## Loading required package:  e1071
## Loading required package:  gtools
##
## Attaching package:  'gtools'
## The following object is masked from 'package:e1071':
##
##    permutations
## Loading required package:  igraph
##
## Attaching package:  'igraph'
## The following object is masked from 'package:gtools':
##
##    permute
## The following objects are masked from 'package:stats':
##
##    decompose, spectrum
## The following object is masked from 'package:base':
##
##    union
```

Simulations

We compare epiNEM to several network inference methods.

```
library(bnem) #  install_github("MartinFXP/B-NEM/package")

## Loading required package:  CellNOptR
## Loading required package:  RBGL
## Loading required package:  graph
## Loading required package:  BiocGenerics
## Loading required package:  parallel
##
## Attaching package:  'BiocGenerics'
## The following objects are masked from 'package:parallel':
##
##    clusterApply, clusterApplyLB, clusterCall, clusterEvalQ, clusterExport,
##    clusterMap, parApply, parCapply, parLapply, parLapplyLB, parRapply,
##    parSapply, parSapplyLB
## The following objects are masked from 'package:igraph':
##
##    normalize, union
## The following objects are masked from 'package:stats':
##
##    IQR, mad, xtabs
## The following objects are masked from 'package:base':
##
##    Filter, Find, Map, Position, Reduce, anyDuplicated, append, as.data.frame,
##    cbind, colnames, do.call, duplicated, eval, evalq, get, grep, grepl,
##    intersect, is.unsorted, lapply, lengths, mapply, match, mget, order, paste,
##    pmax, pmax.int, pmin, pmin.int, rank, rbind, rownames, sapply, setdiff,
##    sort, table, tapply, union, unique, unsplit, which, which.max, which.min
##
## Attaching package:  'graph'
## The following objects are masked from 'package:igraph':
##
##    degree, edges, intersection
##
## Attaching package:  'RBGL'
## The following objects are masked from 'package:igraph':
##
##    bfs, dfs, transitivity
## The following object is masked from 'package:e1071':
##
##    extractPath
## Loading required package:  hash
## hash-2.2.6 provided by Decision Patterns
## Loading required package:  ggplot2
## Warning:  package 'ggplot2' was built under R version 3.3.2
## Loading required package:  RCurl
## Loading required package:  bitops
## Loading required package:  Rgraphviz
## Loading required package:  grid
## Loading required package:  XML
## Warning:  package 'XML' was built under R version 3.3.2
```

```
##
## Attaching package:  'XML'
## The following object is masked from 'package:graph':
##
##    addNode
## Loading required package:  nem
##
## Attaching package:  'nem'
## The following object is masked from 'package:RBGL':
##
##    transitive.closure
## Loading required package:  matrixStats
## matrixStats v0.51.0 (2016-10-08) successfully loaded.  See ?matrixStats for help.
## Loading required package:  snowfall
## Loading required package:  snow
##
## Attaching package:  'snow'
## The following objects are masked from 'package:BiocGenerics':
##
##    clusterApply, clusterApplyLB, clusterCall, clusterEvalQ, clusterExport,
##    clusterMap, clusterSplit, parApply, parCapply, parLapply, parRapply,
##    parSapply
## The following objects are masked from 'package:parallel':
##
##    clusterApply, clusterApplyLB, clusterCall, clusterEvalQ, clusterExport,
##    clusterMap, clusterSplit, makeCluster, parApply, parCapply, parLapply,
##    parRapply, parSapply, splitIndices, stopCluster
## Loading required package:  latticeExtra
## Loading required package:  lattice
## Loading required package:  RColorBrewer
##
## Attaching package:  'latticeExtra'
## The following object is masked from 'package:ggplot2':
##
##    layer
```

```
library(nem)

library(minet)

library(pcalg)
```

```
runs <- 100

noiselvls <- c(0.01, 0.025, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5)

random <- list(FPrate = 0.1, FNrate = noiselvls, single = 4, double = 1, reporters = 100, replicates = 3

spec <- sens  <- logics <- array(0, dim = c(2, runs, length(noiselvls)))

sens2 <- spec2 <- time <- array(0, dim = c(5, runs, length(noiselvls)))

do <- c("n", "p", "a")
```

```
do <- c("e", "b", do)

popSize <- 100

maxTime <- F

forcelogic <- T

epinemsearch <- "greedy"

nIterations <- 3

bnemsearch <- "genetic"

parallel <- NULL

logicgate <- matrix("", runs, length(noiselvls))

edgenr <- matrix(0, runs, length(noiselvls))

## for (i in 1:runs) {

##     print(paste("run ", i, sep = ""))

##     for (j in 1:length(noiselvls)) {

##         print(paste("noiselvl ", j, sep = ""))

##         topology <- CreateTopology(random$single, random$double, force = forcelogic)

##         topology <- unlist(unique(topology), recursive = FALSE)

##         extTopology <- ExtendTopology(topology$model, random$reporters)

##         sortedData <- GenerateData(topology$model, extTopology, random$FPrate, random$FNrate[j], rand

##         logicgate[i, j] <- paste(topology$logics, collapse = "_")

##         edgenr[i, j] <- sum(topology$origModel == 1)

##         if ("e" %in% do) {
##             print("epiNEM")

##             start <- Sys.time()
##             TriplModel <- epiNEM(filename = sortedData, method = epinemsearch, nIterations = nIterat
##             time[1, i, j] <- difftime(Sys.time(), start, units = "secs")
##             print(time[1, i, j])

##             tp <- sum(topology$model == 1 & TriplModel$model == 1)
##             tn <- sum(topology$model == 0 & TriplModel$model == 0)
##             fp <- sum(topology$model == 0 & TriplModel$model == 1)
##             fn <- sum(topology$model == 1 & TriplModel$model == 0)
##             sens[1, i, j] <- tp/(tp+fn)
```

```
##             spec[1, i, j] <- tn/(tn+fp)
##             tp <- sum(topology$origModel == 1 & TriplModel$origModel == 1)
##             tn <- sum(topology$origModel == 0 & TriplModel$origModel == 0)
##             fp <- sum(topology$origModel == 0 & TriplModel$origModel == 1)
##             fn <- sum(topology$origModel == 1 & TriplModel$origModel == 0)
##             sens2[1, i, j] <- tp/(tp+fn)
##             spec2[1, i, j] <- tn/(tn+fp)
##             tp <- 0
##             for (k in 1:length(topology$column)) {
##                 for (l in 1:length(TriplModel$column)) {
##                     if (topology$column[k] == TriplModel$column[l]) {
##                         if (topology$logics[k] %in% TriplModel$logics[l]) {
##                             tp <- tp + 1
##                         }
##                     }
##                 }
##             }
##             logics[1, i, j] <- tp/(length(topology$logics) + length(TriplModel$logics) - tp)
##             print(sens[1, i, j])
##             print(spec[1, i, j])
##             print(sens2[1, i, j])
##             print(spec2[1, i, j])
##             print(logics[1, i, j])

##         }

##         if ("b" %in% do) {
##             print("B-NEM")

##             gtn <- epi2bg(topology)

##             fc <- cbind(Ctrl_vs_S = -1, epi2bg(sortedData))*(-1)

##             bnemnoise <- sample(1:nrow(fc), floor(nrow(fc)*random$FNrate[j]))

##             fc[bnemnoise, 1] <- 0

##             ers <- t(topology$model)*(-1)
##             colnames(ers) <- paste("S_vs_S_", gsub("\\.", "_", colnames(ers)), sep = "")
##             ers <- cbind(Ctrl_vs_S = 1, ers)
##             ers <- ers[, order(colnames(ers))]

##             CNOlist <- dummyCNOlist(stimuli = "S", inhibitors = LETTERS[1:random$single], maxStim =

##             parents <- unique(unlist(strsplit(colnames(sortedData)[grep("\\.", colnames(sortedData))

##             nodes <- unique(colnames(sortedData)[-grep("\\.", colnames(sortedData))])

##             child <- nodes[-which(nodes %in% parents)]

##             sifMatrix <- NULL
##             for (k in LETTERS[1:random$single]) {
##                sifMatrix <- rbind(sifMatrix, c("S", "1", k))#, c("S", "-1", k)) # bnem can set a pri
```

```
##                 for (l in LETTERS[1:random$single]) {
##                     if (k %in% l) { next() }
##                     if (k %in% parents) {
##                         sifMatrix <- rbind(sifMatrix, c(k, "1", l), c(k, "-1", l))
##                     } else {
##                         sifMatrix <- rbind(sifMatrix, c(k, "1", l))
##                     }

##             randfile <- paste("pkn_", as.numeric(Sys.time()), sep = "")
##             write.table(sifMatrix, file = randfile, sep = "\t",
##                         row.names = FALSE, col.names = FALSE, quote = FALSE)
##             PKN <- readSIF(randfile)
##             unlink(randfile)

##             model <- preprocessing(CNOlist, PKN)

##             initBstring <- absorption(rep(1, length(model$reacID)), model)

##             if (maxTime) { maxTime2 <- time[1, i, j] } else { maxTime2 <- Inf }

##             start <- Sys.time()
##             bga <- bnem(search = bnemsearch,
##                         fc=fc,
##                         CNOlist=CNOlist,
##                         model=model,
##                         initBstring=initBstring,
##                         draw = F,
##                         verbose = F,
##                         popSize = popSize,
##                         maxTime = maxTime2,
##                         parallel = parallel
##                         )
##             time[2, i, j] <- difftime(Sys.time(), start, units = "secs")
##             print(time[2, i, j])

##             ers2 <- computeFc(CNOlist, t(simulateStatesRecursive(CNOlist, model, bga$bString)))
##             ers2 <- ers2[, unique(colnames(fc))]
##             ers2 <- ers2[, order(colnames(ers2))]

##             tp <- sum(ers == -1 & ers2 == -1)
##             tn <- sum(ers == 0 & ers2 == 0)
##             fn <- sum(ers == -1 & ers2 == 0)
##             fp <- sum(ers == 0 & ers2 == -1)
##             sens[2, i, j] <- tp/(tp+fn)
##             spec[2, i, j] <- tn/(tn+fp)
##             gtn2 <- abs(dnf2adj(gtn))
##             if (length(grep("S", rownames(gtn2))) > 0) {
##                 gtn2 <- gtn2[-grep("S", rownames(gtn2)), -grep("S", colnames(gtn2))]
##             }
##             gtn2 <- gtn2[order(rownames(gtn2)), order(colnames(gtn2))]
##             res <- abs(dnf2adj(bga$graph))
##             if (length(grep("S", rownames(res))) > 0) {
##                 res <- as.matrix(res[-grep("S", rownames(res)), -grep("S", colnames(res))])
```

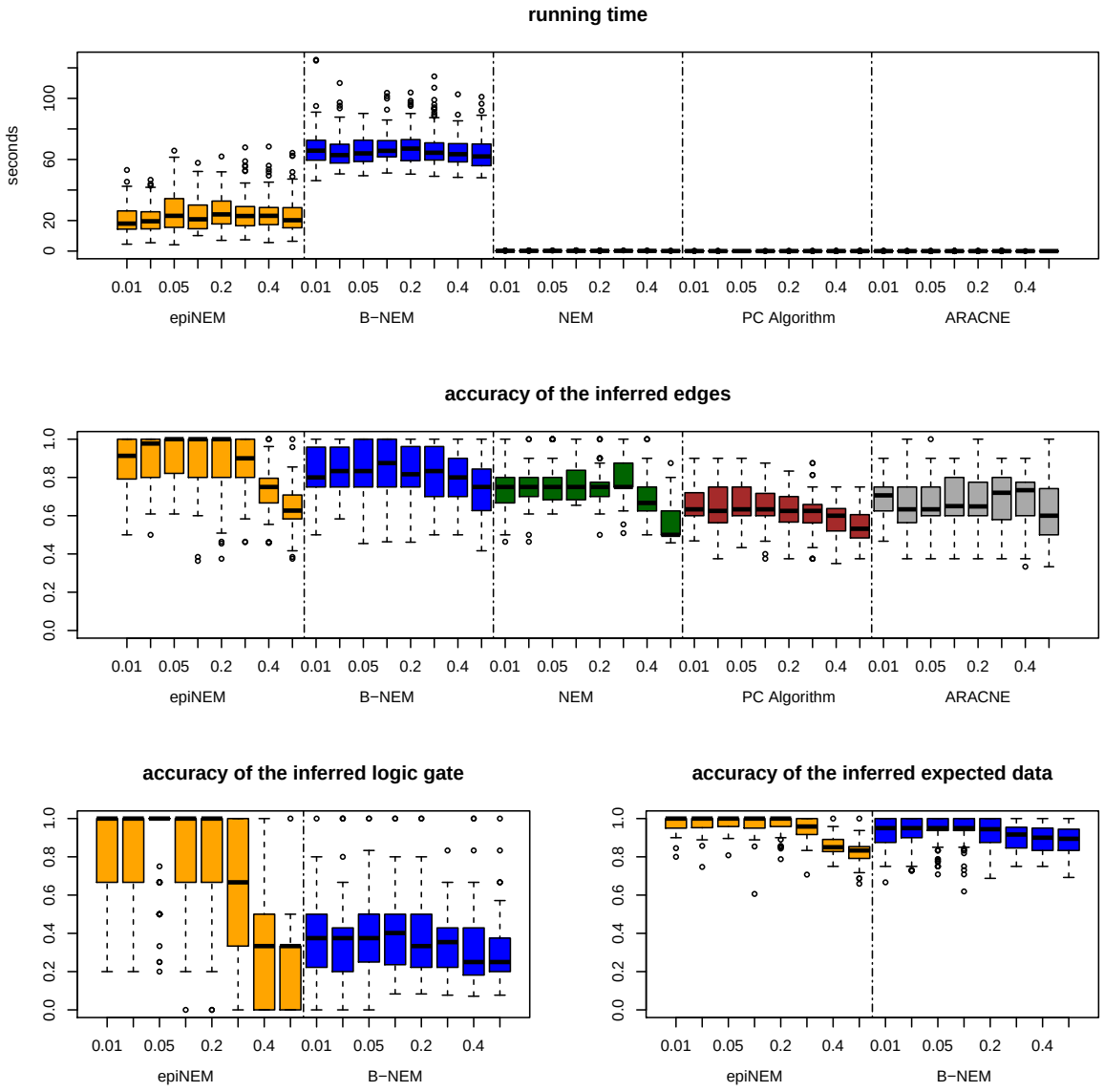
```
##             }
##             if (dim(res)[1] == 1) {
##                 colnames(res) <- rownames(res) <- gsub(".*=", "", bga$graph)
##             } else {
##                 res <- res[order(rownames(res)), order(colnames(res))]
##             }
##             if (nrow(res) < nrow(gtn2)) {
##                 res2 <- rbind(cbind(res, matrix(0, nrow(res), nrow(gtn2) - nrow(res))), matrix(0, nr
##                 colnames(res2)[(ncol(res)+1):ncol(res2)] <- colnames(gtn2)[which(!(colnames(gtn2) %i
##                 rownames(res2)[(nrow(res)+1):nrow(res2)] <- rownames(gtn2)[which(!(rownames(gtn2) %i
##                 res2 <- res2[order(rownames(res2)), order(colnames(res2))]
##                 res <- res2
##             }
##             diag(gtn2) <- diag(res) <- 0
##             tp <- sum(gtn2 == 1 & res == 1)
##             tn <- sum(gtn2 == 0 & res == 0)
##             fn <- sum(gtn2 == 1 & res == 0)
##             fp <- sum(gtn2 == 0 & res == 1)
##             sens2[2, i, j] <- tp/(tp+fn)
##             spec2[2, i, j] <- tn/(tn+fp)
##             tp <- sum(bga$graph %in% gtn)
##             logics[2, i, j] <- tp/(length(gtn) + length(bga$graph) - tp) # (tp/(tp+fn) + tn/(tn+fp))
##             print(sens[2, i, j])
##             print(spec[2, i, j])
##             print(sens2[2, i, j])
##             print(spec2[2, i, j])
##             print(logics[2, i, j])

##             print(bga$graph)
##             print(gtn)

##         }

##         if (any(c("n", "p", "a") %in% do)) {

##             reddata <- sortedData[, -grep("\\.", colnames(sortedData))]
##             gtnadj <- topology$origModel
##             gtnadj <- gtnadj[order(apply(gtnadj, 1, sum), decreasing = T), order(apply(gtnadj, 2, su
##             gtnadj[lower.tri(gtnadj)] <- gtnadj[upper.tri(gtnadj)]
##             gtnadj <- gtnadj[order(rownames(gtnadj)), order(colnames(gtnadj))]
##             eadj <- topology$origModel
##             eadj <- eadj[order(rownames(eadj)), order(colnames(eadj))]
##             reddata2 <- matrix(0, nrow(reddata)*random$replicates, length(unique(colnames(reddata)))
##             for (k in 1:length(unique(colnames(reddata)))) {
##                 reddata2[, k] <- as.vector(reddata[, which(colnames(reddata) %in% unique(colnames(re
##             }
##             colnames(reddata2) <- unique(colnames(reddata))

##         }

##         if ("n" %in% do) {
##             print("NEM")
```

```
##              start <- Sys.time()
##              if (epinemsearch %in% "greedy") {
##                  nemres <- nem(reddata, inference = "nem.greedy")
##              } else {
##                  nemres <- nem(reddata, inference = "search")
##              }
##              nadj <- transitive.reduction(graph2adj(nemres$graph))
##              time[3, i, j] <- difftime(Sys.time(), start, units = "secs")
##              print(time[3, i, j])

##              tp <- sum(eadj  == 1 & nadj == 1)
##              tn <- sum(eadj == 0 & nadj == 0)
##              fp <- sum(eadj == 0 & nadj == 1)
##              fn <- sum(eadj == 1 & nadj == 0)
##              sens2[3, i, j] <- tp/(tp+fn)
##              spec2[3, i, j] <- tn/(tn+fp)
##              print(sens2[3, i, j])
##              print(spec2[3, i, j])

##          }

##          if ("p" %in% do) {
##              print("PCalg")

##              start <- Sys.time()
##              pc.fit <- pc(suffStat = list(C = cor(reddata2), n = nrow(reddata2)),
##                    indepTest = gaussCItest, ## indep.test: partial correlations
##                    alpha=0.05, labels = colnames(reddata2), verbose = F)
##              pcadj <- graph2adj(pc.fit@graph)
##              time[4, i, j] <- difftime(Sys.time(), start, units = "secs")
##              print(time[4, i, j])

##              tp <- sum(gtnadj == 1 & pcadj == 1)
##              tn <- sum(gtnadj  == 0 & pcadj == 0)
##              fp <- sum(gtnadj == 0 & pcadj == 1)
##              fn <- sum(gtnadj == 1 & pcadj == 0)
##              sens2[4, i, j] <- tp/(tp+fn)
##              spec2[4, i, j] <- tn/(tn+fp)
##              print(sens2[4, i, j])
##              print(spec2[4, i, j])

##          }

##          if ("a" %in% do) {
##              print("Aracne")

##              start <- Sys.time()
##              ares <- build.mim(reddata2)
##              ares <- aracne(ares)
##              ares <- disc(ares, 0)
##              ares <- ares[order(rownames(ares)), order(colnames(ares))]
##              nas <- which(is.na(ares) == T)
##              ares[nas] <- 0
```

```
##             diag(ares) <- 0
##             time[5, i, j] <- difftime(Sys.time(), start, units = "secs")
##             print(time[5, i, j])

##             tp <- sum(gtnadj == 1 & ares == 1)
##             tn <- sum(gtnadj == 0 & ares == 0)
##             fp <- sum(gtnadj == 0 & ares == 1)
##             fn <- sum(gtnadj == 1 & ares == 0)
##             sens2[5, i, j] <- tp/(tp+fn)
##             spec2[5, i, j] <- tn/(tn+fp)
##             print(sens2[5, i, j])
##             print(spec2[5, i, j])

##         }

##     }

## }
```

```
data(sim)

colvec <- c(rep("orange", length(noiselvls)), rep("blue", length(noiselvls)), rep("darkgreen", length(no

acc <- (sens + spec)/2

acc2 <- (sens2 + spec2)/2

m <- rbind(c(1,1), c(2,2), c(3,4))

layout(m)

timeframe <- as.data.frame(cbind(data.frame(epiNEM = time[1,,]), data.frame(BNEM = time[2,,]), data.fram

colnames(timeframe) <- rep(noiselvls, 5) # c(paste(rep("epi", length(noiselvls)), noiselvls, sep = "_"),

boxplot(timeframe, col = colvec, main = "running time", ylab = "seconds")

abline(v=(1:(length(do)-1)*length(noiselvls) + 0.5), col = "black", lty = 6)

axis(1, c(3, 11, 19, 28, 36)+1, c("epiNEM", "B-NEM", "NEM", "PC Algorithm", "ARACNE"), tick = F, pos = -

accframe2 <- as.data.frame(cbind(data.frame(epiNEM = acc2[1,,]), data.frame(BNEM = acc2[2,,]), data.fram

colnames(accframe2) <- rep(noiselvls, 5) # c(paste(rep("E", length(noiselvls)), noiselvls, sep = "_"), p

boxplot(accframe2, col = colvec, main = "accuracy of the inferred edges", ylim = c(0,1))

abline(v=(1:(length(do)-1)*length(noiselvls) + 0.5), col = "black", lty = 6)

axis(1, c(3, 11, 19, 28, 36)+1, c("epiNEM", "B-NEM", "NEM", "PC Algorithm", "ARACNE"), tick = F, pos = -

## logical nems:
```

```
colvec2 <- c(rep("orange", length(noiselvls)), rep("blue", length(noiselvls)))

logicsframe <- as.data.frame(cbind(data.frame(epiNEM = logics[1,,]), data.frame(BNEM = logics[2,,])))

colnames(logicsframe) <- rep(noiselvls, 2) # c(paste(rep("E", length(noiselvls)), noiselvls, sep = "_"),

boxplot(logicsframe, col = colvec2, main = "accuracy of the inferred logic gate", ylim = c(0,1))

abline(v=length(noiselvls)+0.5, col = "black", lty = 6)

axis(1, c(3, 11, 19, 28, 36)+1, c("epiNEM", "B-NEM", "NEM", "PC Algorithm", "ARACNE"), tick = F, pos = -

accframe <- as.data.frame(cbind(data.frame(epiNEM = acc[1,,]), data.frame(BNEM = acc[2,,])))

colnames(accframe) <- rep(noiselvls, 2) # c(paste(rep("E", length(noiselvls)), noiselvls, sep = "_"), pa

boxplot(accframe, col = colvec2, main = "accuracy of the inferred expected data", ylim = c(0,1)) # what

abline(v=length(noiselvls)+0.5, col = "black", lty = 6)

axis(1, c(3, 11, 19, 28, 36)+1, c("epiNEM", "B-NEM", "NEM", "PC Algorithm", "ARACNE"), tick = F, pos = -
```



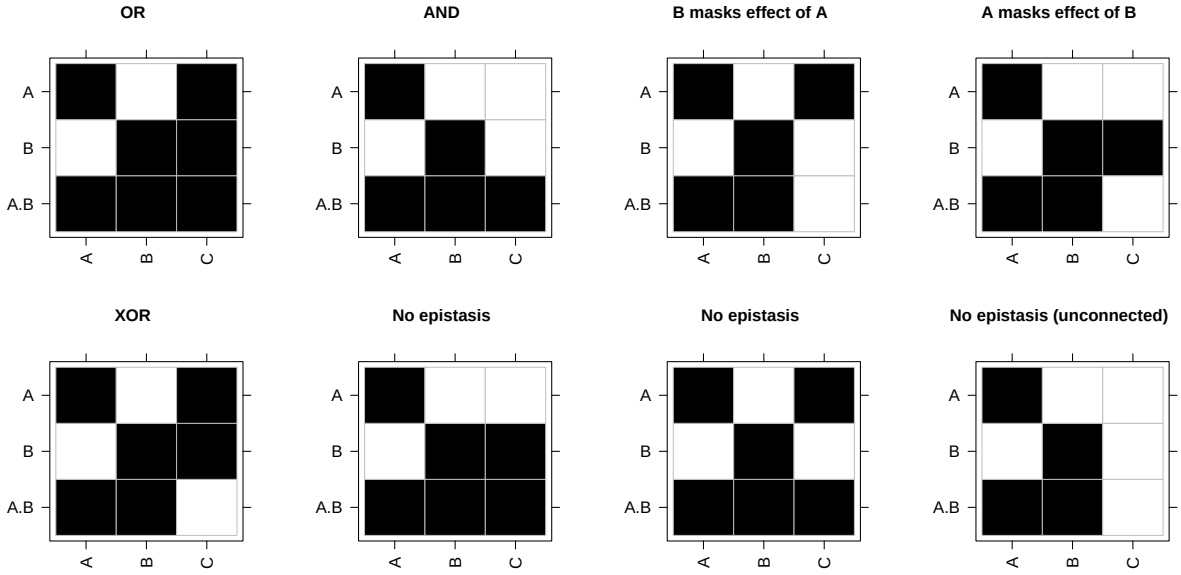
Yeast knockout screens

In this section we analyse previously published yeast knockout screens. The screens consist of gene expression data derived from double and single knockout mutants. We use epiNEM on each double mutant combined with each single mutant.

The results of the knockout screens have been annotated according to the following legend:

```
options(warn=-1)
heatmapOP(matrix(c(1,-1,1,-1,1,1, 1, 1, 1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])), Co
heatmapOP(matrix(c(1,-1,1,-1,1,1, -1, -1, 1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])),
heatmapOP(matrix(c(1,-1,1,-1,1,1, 1, -1, -1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])),
heatmapOP(matrix(c(1,-1,1,-1,1,1, -1, 1, -1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])),
heatmapOP(matrix(c(1,-1,1,-1,1,1, 1, 1, -1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])), C
heatmapOP(matrix(c(1,-1,1,-1,1,1, -1, 1, 1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])), C
```

```
heatmapOP(matrix(c(1,-1,1,-1,1,1, 1, -1, 1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])), C
heatmapOP(matrix(c(1,-1,1,-1,1,1, -1, -1, -1), 3, 3, dimnames = list(c("A", "B", "A.B"), LETTERS[1:3])),
options(warn=0)
```



Wageningen et al., 2010

```
data <- read.delim("http://www.holstegelab.nl/publications/sv/signaling_redundancy/downloads/DataS1.txt"

dataM <- data[-(1:2), (1+(1:(324/2))*2)]

dataP <- data[-(1:2), (2+(1:(324/2))*2)]

dataM <- dataM[-1, ]

dataP <- dataP[-1, ]

dataM <- apply(dataM, c(1,2), as.numeric)

dataP <- apply(dataP, c(1,2), as.numeric)

dataBin <- dataM

sig <- 0.05

cutoff <- log2(1.7)

dataBin[which(dataP < sig & dataP > 0 & abs(dataM) >= cutoff)] <- 1

dataBin[which(dataP >= sig | dataP == 0 | abs(dataM) < cutoff)] <- 0

dataBin <- dataBin[-which(apply(dataBin, 1, max) == 0), ]

genelist <- toupper(c('hsl1', 'cla4', 'gin4', 'swe1', 'hsl1.cla4'))
```

```r
colnames(dataBin) <- gsub(".del.vs..wt", "", colnames(dataBin))

colnames(dataBin) <- gsub(".del", "", colnames(dataBin))

doubles <- colnames(dataBin)[grep("\\.", colnames(dataBin))]

doubles <- sort(doubles[-grep("vs", doubles)])

doubles.genes <- unique(unlist(strsplit(doubles, "\\.")))

singles <- colnames(dataBin)[-grep("\\.", colnames(dataBin))]

singles <- unique(sort(singles))

llmat <- logicmat <- matrix(0, length(singles), length(doubles))

rownames(llmat) <- rownames(logicmat) <- singles

colnames(llmat) <- colnames(logicmat) <- doubles

globalgenes <- which(apply(dataBin, 1, max) == 1)

## for (i in doubles[set]) {
##     if (which(doubles %in% i) == 8) { next() }
##     print(i)
##     doubles.singles <- unlist(strsplit(i, "\\."))
##     egenes <- which(apply(dataBin[, which(colnames(dataBin) %in% c(i, doubles.singles))], 1, max) ==
##     for (j in singles) {
##         print(j)
##         if (j %in% doubles.singles) { next() }

##         dataTmp <- dataBin[, grep(paste(paste("^", c(i, j, doubles.singles), "$", sep = ""), collaps

##         if (path %in% "fixed_set") {
##             dataTmp <- dataTmp[egenes, ]
##         }
##         if (path %in% "global") {
##             dataTmp <- dataTmp[globalgenes, ]
##         }
##         if (path %in% "") {
##             dataTmp <- dataTmp[which(apply(dataTmp, 1, max) == 1), ]
##         }

##         i1 <- which(singles %in% j)
##         i2 <- which(doubles %in% i)

##         if (!(is.null(dim(dataTmp)))) {

##             if (any(dataTmp[, j] != 0)) {

##                 epires <- epiNEM(dataTmp, method = "exhaustive")

##                 tmp <- epires$logics
```

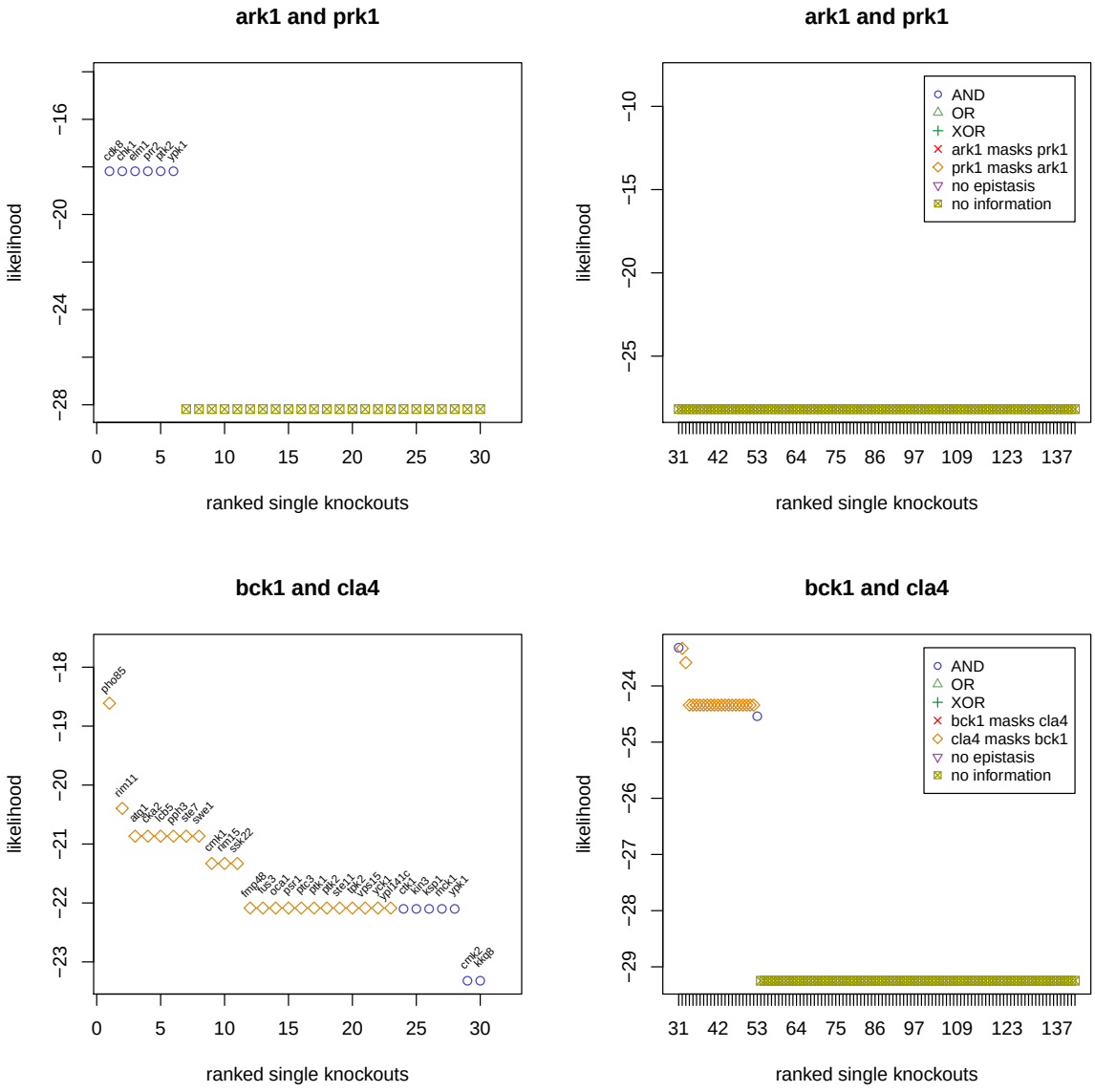
```
##                 if ("OR" %in% tmp) {
##                     if (sum(epires$origModel[, j]) != 2) {
##                         tmp <- "NOEPI"
##                     } else {
##                         if (all(tmp %in% "OR")) {
##                             tmp <- "OR"
##                         } else {
##                             tmp <- tmp[which(!(tmp %in% "OR"))]
##                         }
##                     }
##                 }

##                 logicmat[i1, i2] <- tmp
##                 llmat[i1, i2] <- epires$score

##             } else {

##                 logicmat[i1, i2] <- "UNCON"
##                 llmat[i1, i2] <- -Inf

##             }

##         } else {

##             logicmat[i1, i2] <- "UNCON"
##             llmat[i1, i2] <- -Inf

##         }

##     }

## }
```

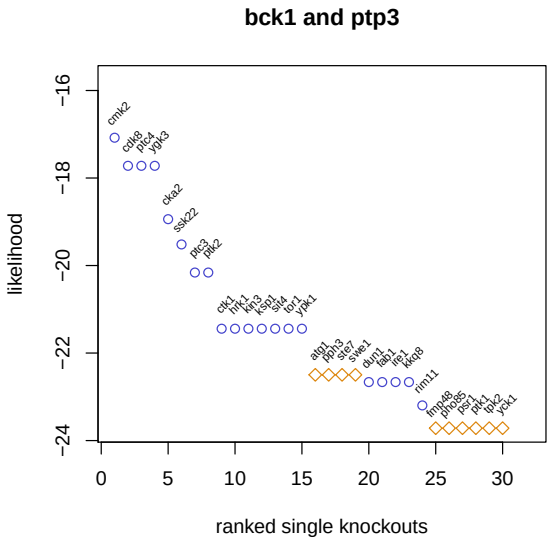
Plot results.

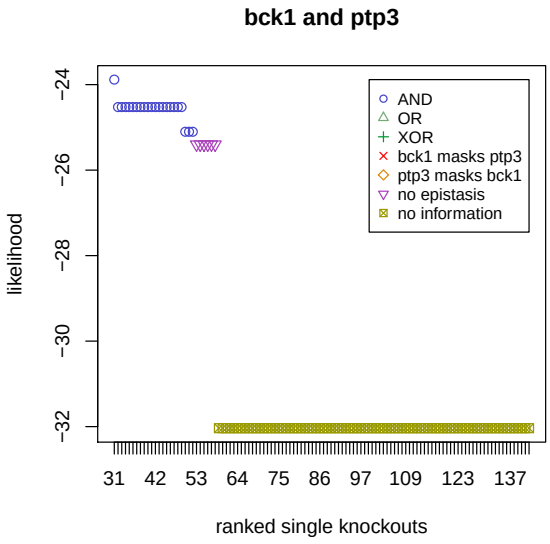
```
palette(c("#4444cc", "#77aa77", "#009933", "#ff0000", "#dd8811", "#aa44bb", "#999900"))

data(wageningen_res)

llmat0 <- wageningen$ll

logicmat0 <- wageningen$logic

for (i in 1:length(doubles)) {

    #if (!(doubles[i] %in% c("ark1.prk1", "prk1.ark1", "ptp2.ptp3", "ptp3.ptp2", "bck1.ptp3", "ptp3.bck

    if (i %in% 8) { next() }

    logicvec <- logicmat0[, i]

    llvec <- llmat0[, i]

    logicvec <- logicvec[order(llvec, decreasing = T)]
```

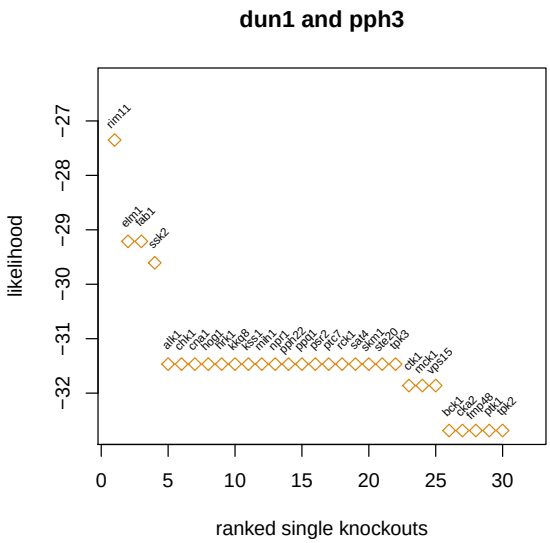
```
    llvec <- llvec[order(llvec, decreasing = T)]

    parents <- unlist(strsplit(doubles[i], "\\."))

    pchvec <- numeric(length(llvec))

    pchvec[which(logicvec %in% "AND")] <- 1
    pchvec[which(logicvec %in% "OR")] <- 2
    pchvec[which(logicvec %in% "XOR")] <- 3
    pchvec[grep(paste("^", parents[1], sep = ""), logicvec)] <- 4
    pchvec[grep(paste("^", parents[2], sep = ""), logicvec)] <- 5
    pchvec[which(logicvec %in% "NOEPI")] <- 6
    pchvec[which(logicvec %in% c("NOINFO", "NOINF"))] <- 7

    logicvec <- logicvec[-which(logicvec %in% "0")]
    pchvec <- pchvec[-which(pchvec == 0)]
    llvec <- llvec[-which(llvec == 0)]

    colvec <- pchvec

    if (all(is.infinite(llvec) == T)) {

        llvec[1:length(llvec)] <- -1000

        margin <- 100

        donames <- 30

    } else {

        llvec[which(is.infinite(llvec) == T)] <- NA

        ## llvec[which(is.infinite(llvec) == T)] <- min(llvec) - 100

        margin <- abs(max(llvec[1:30], na.rm = T) - min(llvec[1:30], na.rm = T))

        offset <- 0.075

        if (margin == 0) { margin <- 10; offset <- 0.0375 }

        donames <- 30 - sum(is.na(llvec[1:30]) == T)

        if (any(is.na(llvec[1:30]) == T)) { margin2 <- margin*2 } else { margin2 <- margin }

        llvec[which(is.na(llvec) == T)] <- min(llvec, na.rm = T) - margin

        margin <- margin2

    }

    if (all(llvec[-(1:30)] - min(llvec[-(1:30)]) == 0)) {

        p2max <- max(llvec[-(1:30)]) + margin
```

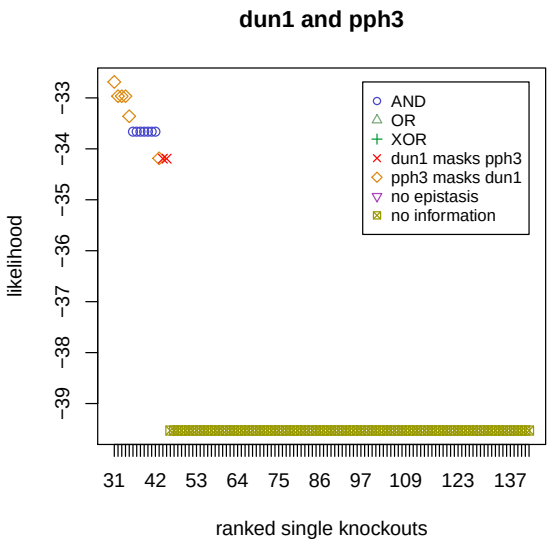
```
    } else {

        p2max <- max(llvec[-(1:30)])

    }
    par = par(mfrow=c(1,2))
    plot = plot(llvec[1:30], pch = pchvec[1:30], col = colvec[1:30], ylab = "likelihood", xlab = "ranked
    text = text((1:30)+0.5, llvec[1:30]+(margin*offset), labels = c(names(llvec)[1:donames], rep("", 30
    plot2 = plot(llvec[-(1:30)], pch = pchvec[-(1:30)], col = colvec[-(1:30)], ylab = "likelihood", xlab
    legend = legend(length(llvec[-(1:30)]), p2max,
                    legend = c("AND", "OR", "XOR", paste(parents[1], " masks ", parents[2], sep = ""), p
    axis = axis(1, at = 1:length(llvec[-(1:30)]), labels = 31:length(llvec))

}
```

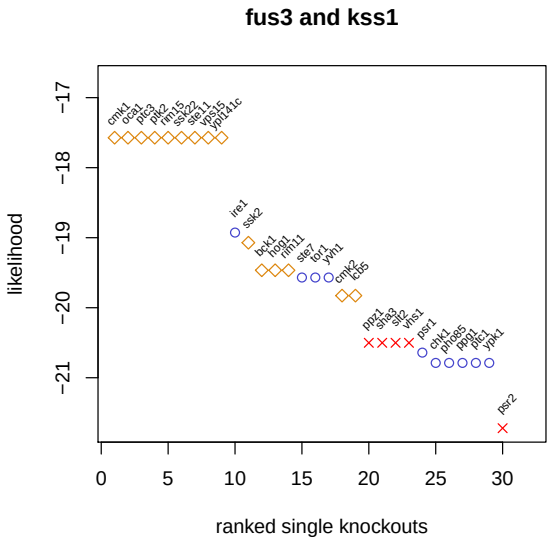


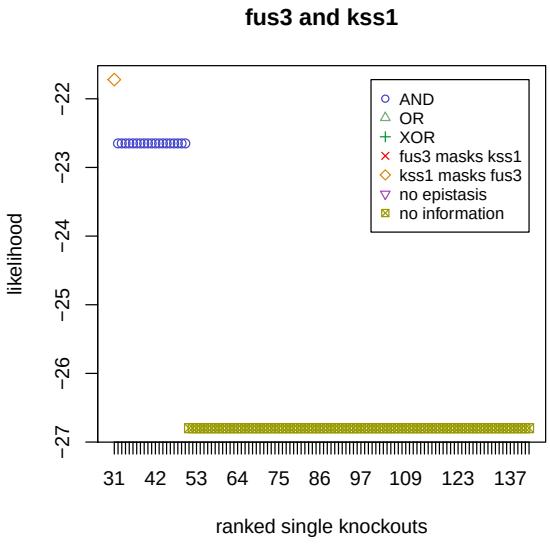

bck1 and ptp3
likelihood
ranked single knockouts
cmk2
cdk8
ptc4
ygk3
cka2
ssk22
ptc3
ptk2
ctk1
hrk1
ken3
ksp1
sid4
tor1
yak1
atg1
pph3
ste7
swe1
dun1
tab1
ire1
kkq8
rim11
fmp48
pho85
ppz1
ptk1
tpk2
ycx1

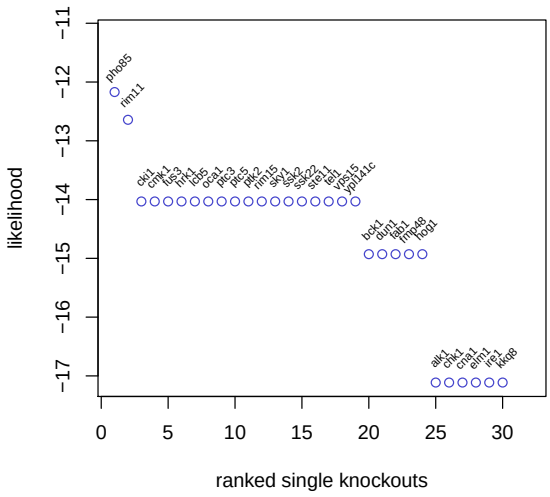

bck1 and ptp3
likelihood
ranked single knockouts
AND
OR
XOR
bck1 masks ptp3
ptp3 masks bck1
no epistasis
no information

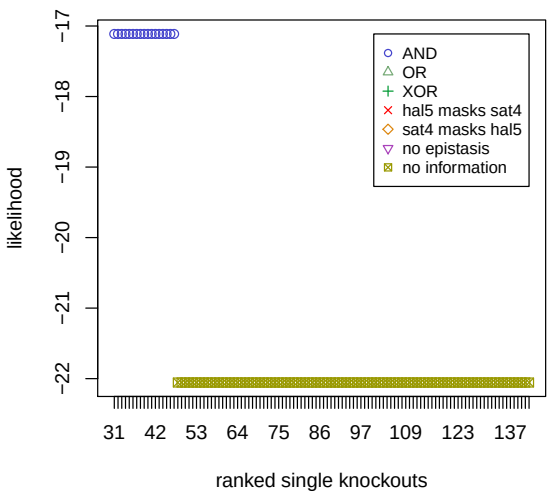

dun1 and pph3
likelihood
ranked single knockouts
rim11
elm1
tab1
ssk2
alk1
chk1
cka1
hog1
kkq8
ksp1
mih1
mpt1
pph22
ppq1
pst2
ptc7
rck1
sid4
skm1
ste20
tpk3
ctk1
mck1
vps15
bck1
cka2
fmp48
ptk1
tpk2

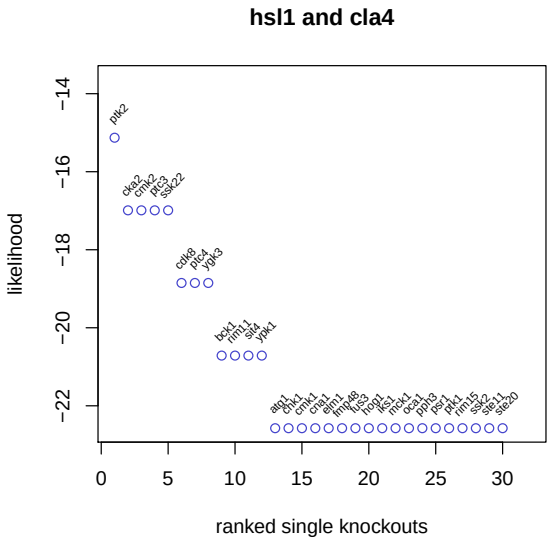

dun1 and pph3
likelihood
ranked single knockouts
AND
OR
XOR
dun1 masks pph3
pph3 masks dun1
no epistasis
no information

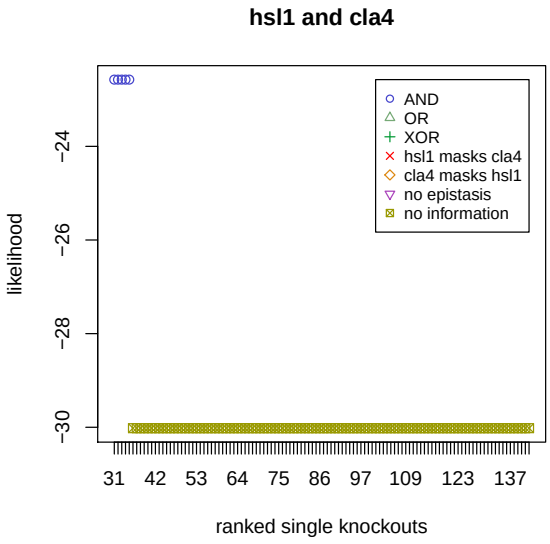

fus3 and kss1
likelihood
ranked single knockouts
cmk1 oca1 ptc3 ptk2 rim15 ssk22 ste11 vps15 ypl141c
ire1 ssk2 bck1 hog1 rim11 ste7 tor1 yvh1 cmk2 cdc5
pzq1 sha3 slt2 vhs1 psr1 cnk1 pho85 pph1 ptc1 ypk1
psr2

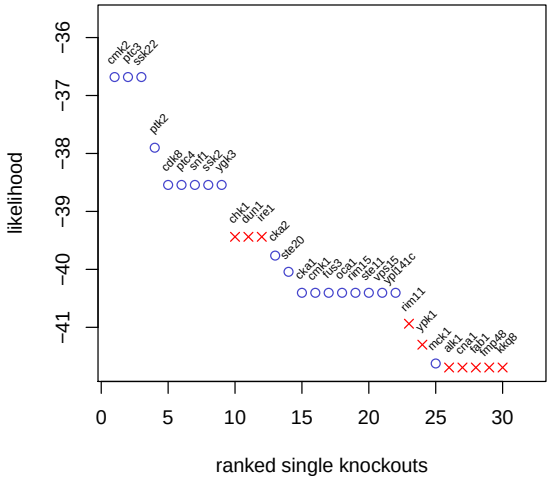

fus3 and kss1
likelihood
ranked single knockouts
AND
OR
XOR
fus3 masks kss1
kss1 masks fus3
no epistasis
no information

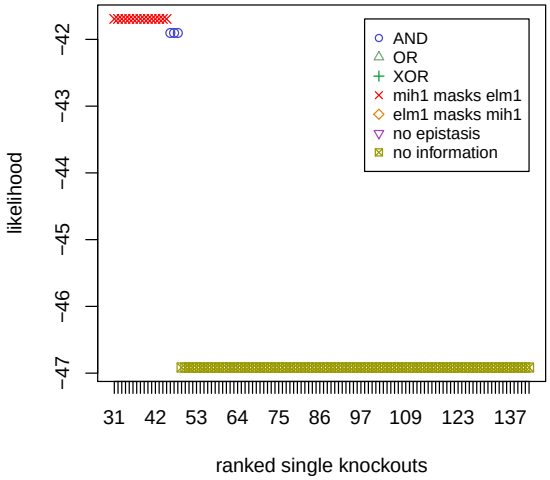

hal5 and sat4
likelihood
ranked single knockouts
pho85 rim11
ckl1 cnk1 hal5 rim1 cdc5 ocs1 rtc3 rtc5 rtk2 rim15 sky1 ssk2 ssk22 ste11 tel1 vps15 ypl141c
bck1 dun1 mbf1 rfm48 hog1
alk1 cka1 csm1 elm1 ire1 kkq8

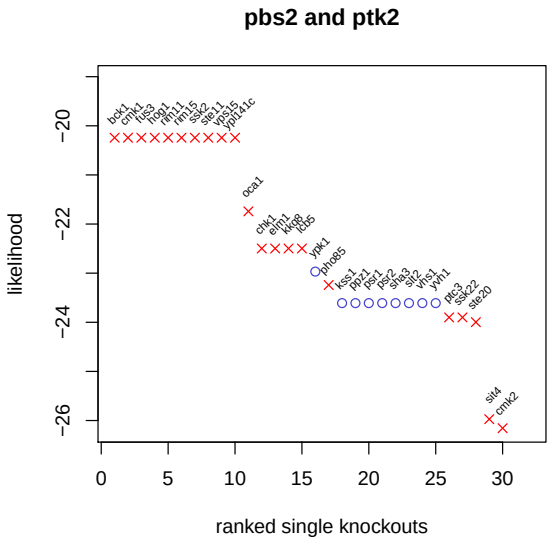

hal5 and sat4
likelihood
ranked single knockouts
AND
OR
XOR
hal5 masks sat4
sat4 masks hal5
no epistasis
no information

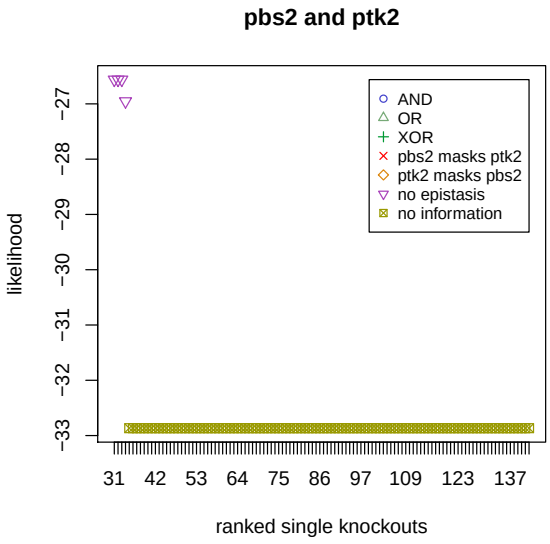

hsl1 and cla4
likelihood
ranked single knockouts
ptk2
cka2
cmk2
ptc3
ssk22
cdt8
ptc4
ypk3
tck1
rim11
sta4
ypk1
atg1
chk1
cmk1
cna1
elm1
imp48
nis3
hog1
hsl1
nck1
oca1
pph3
psr1
pkc1
rim15
ssk2
ste11
ste20
−14
−16
−18
−20
−22
0
5
10
15
20
25
30

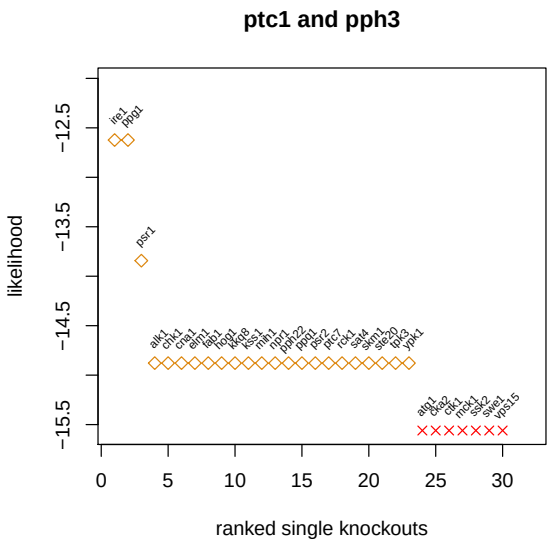

hsl1 and cla4
likelihood
ranked single knockouts
AND
OR
XOR
hsl1 masks cla4
cla4 masks hsl1
no epistasis
no information
−24
−26
−28
−30
31
42
53
64
75
86
97
109
123
137

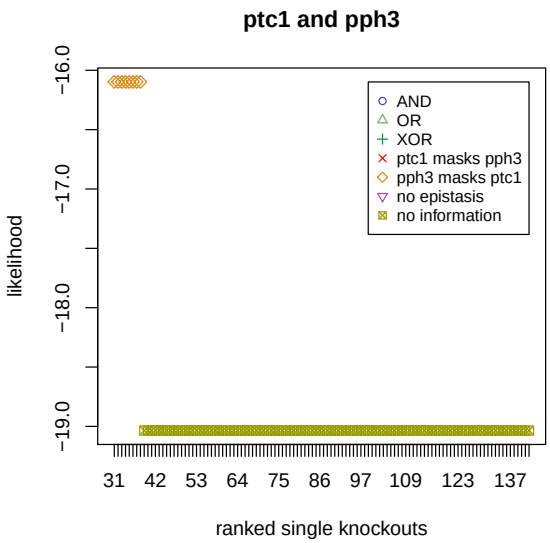

mih1 and elm1
likelihood
ranked single knockouts
cmk2
ptc3
ssk22
ptk2
cdt8
ptc4
snf1
ssk2
ypk3
chk1
dun1
ire1
cka2
ste20
cka1
cmk1
tus3
oca1
rim15
ste11
ypk1
ypk1.c
rim11
ypk1
mck1
atc1
cna1
ire1
imp48
kkq8
−36
−37
−38
−39
−40
−41
0
5
10
15
20
25
30

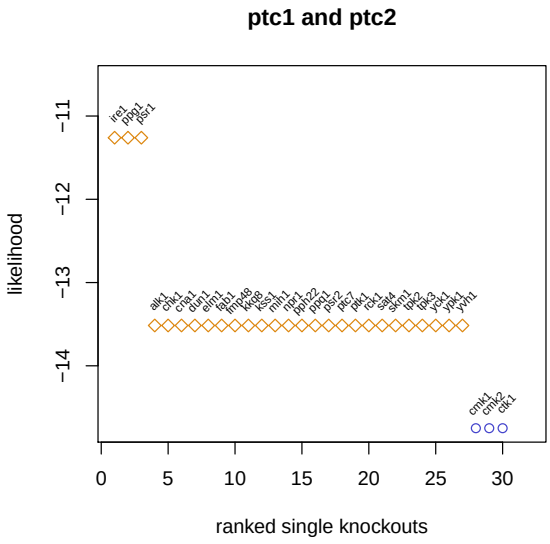

mih1 and elm1
likelihood
ranked single knockouts
AND
OR
XOR
mih1 masks elm1
elm1 masks mih1
no epistasis
no information
−42
−43
−44
−45
−46
−47
31
42
53
64
75
86
97
109
123
137

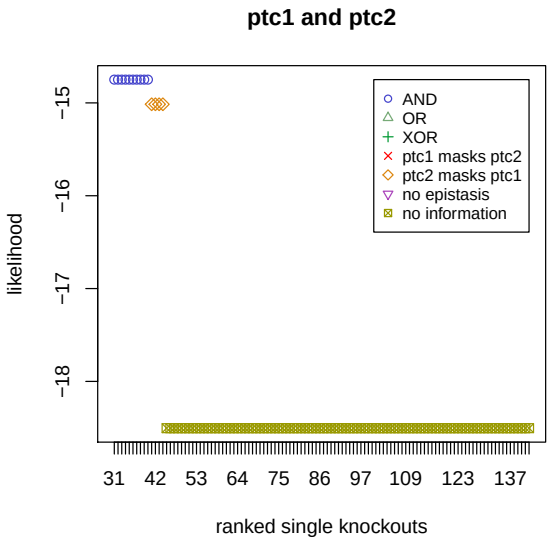

pbs2 and ptk2
likelihood
ranked single knockouts
brk1
cmk1
hks1
hog1
rim11
rim15
ssk2
ste11
vps15
ypl141c
oca1
chk1
eln1
kkq8
tda5
ypk1
pho85
kss1
ppz1
psr1
psr2
sna3
slt2
vhs1
yhi1
ptc3
ssk22
ste20
slt4
cmk2

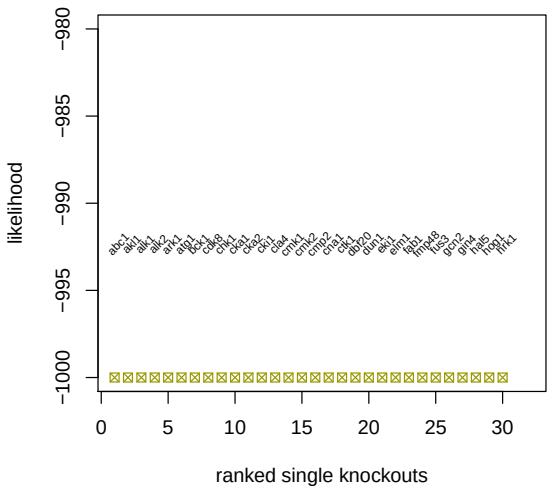

pbs2 and ptk2
likelihood
ranked single knockouts
AND
OR
XOR
pbs2 masks ptk2
ptk2 masks pbs2
no epistasis
no information

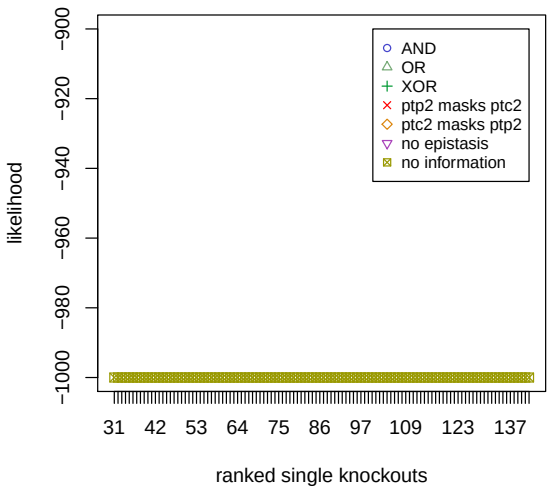

ptc1 and pph3
likelihood
ranked single knockouts
ire1
ppg1
psr1
alk1
chk1
cna1
eln1
fab1
hog1
kkq8
kss1
mih1
npr1
pph22
ppq1
psr2
ptc7
rck1
sat4
skm1
ste20
tpk3
ypk1
atg1
dak2
dbf2
dck1
ms2
ssk2
swe1
vps15

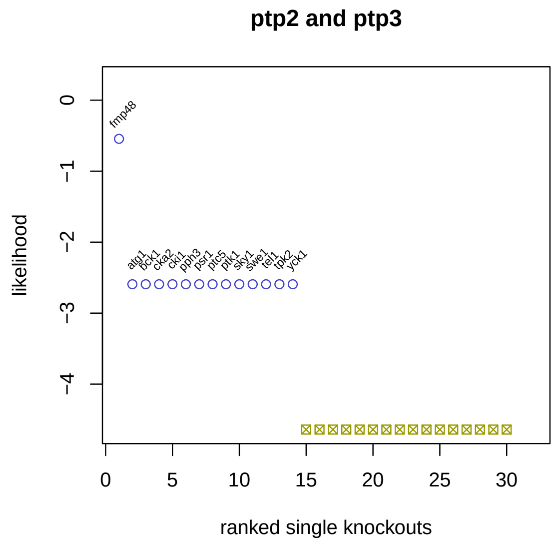

ptc1 and pph3
likelihood
ranked single knockouts
AND
OR
XOR
ptc1 masks pph3
pph3 masks ptc1
no epistasis
no information

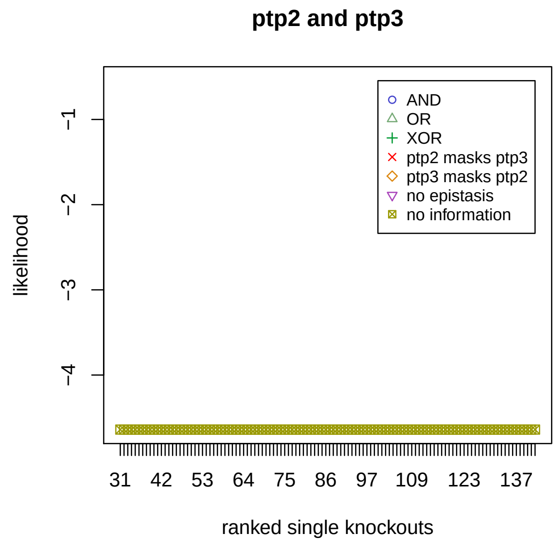

ptc1 and ptc2
likelihood
−11
−12
−13
−14
ire1
ptp1
ptc1
cmk1
cmk2
dk1
0
5
10
15
20
25
30
ranked single knockouts

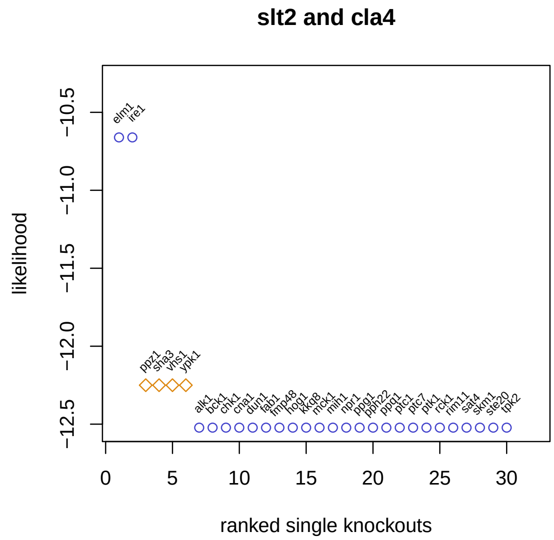

ptc1 and ptc2
likelihood
−15
−16
−17
−18
AND
OR
XOR
ptc1 masks ptc2
ptc2 masks ptc1
no epistasis
no information
31
42
53
64
75
86
97
109
123
137
ranked single knockouts

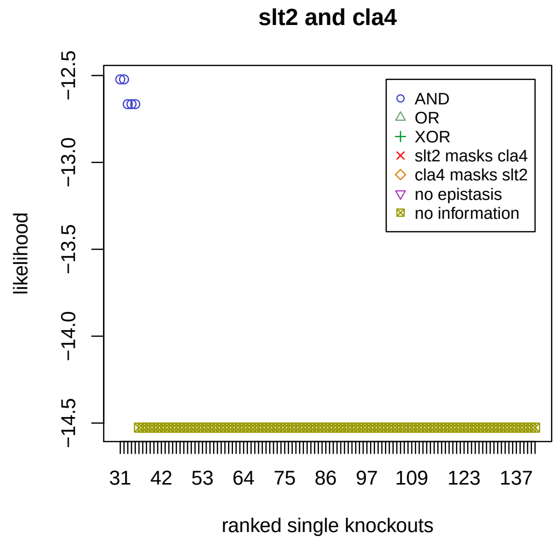

ptp2 and ptc2
likelihood
−980
−985
−990
−995
−1000
0
5
10
15
20
25
30
ranked single knockouts

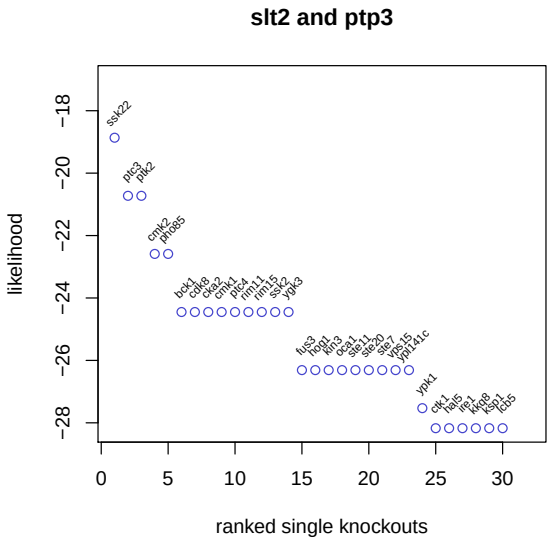

ptp2 and ptc2
likelihood
−900
−920
−940
−960
−980
−1000
AND
OR
XOR
ptp2 masks ptc2
ptc2 masks ptp2
no epistasis
no information
31
42
53
64
75
86
97
109
123
137
ranked single knockouts

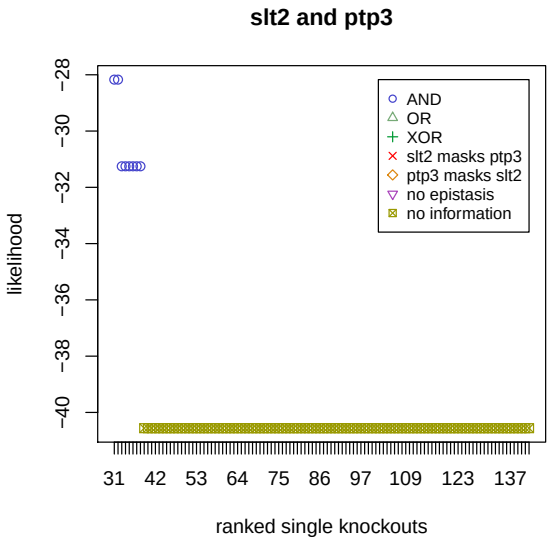

ptp2 and ptp3
likelihood
ranked single knockouts
0
−1
−2
−3
−4
0
5
10
15
20
25
30
fmp48
atg1
bck1
cka2
cki1
pph3
psr1
ptc5
ptk1
sky1
swe1
tel1
tpk2
yck1

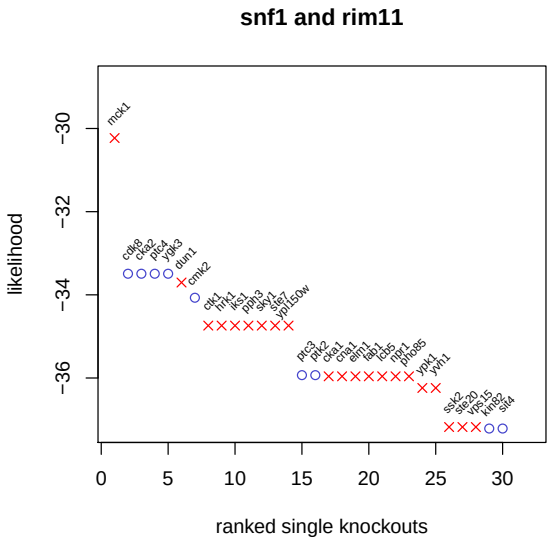

ptp2 and ptp3
likelihood
ranked single knockouts
−1
−2
−3
−4
31
42
53
64
75
86
97
109
123
137
AND
OR
XOR
ptp2 masks ptp3
ptp3 masks ptp2
no epistasis
no information

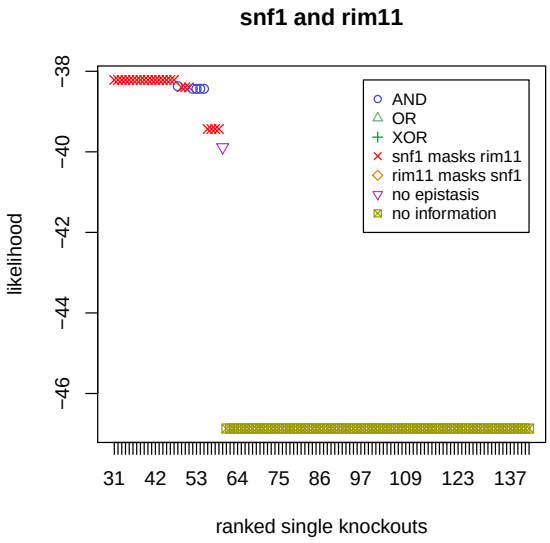

slt2 and cla4
likelihood
ranked single knockouts
−10.5
−11.0
−11.5
−12.0
−12.5
0
5
10
15
20
25
30
elm1
ire1
ppz1
sha3
vhs1
ypk1
alk1
bck1
chk1
cna1
dun1
fab1
fmp48
hog1
hop1
kkq8
mck1
mih1
npr1
ppg1
pph22
ppq1
ptc1
ptc7
ptk1
rck1
rim11
sat4
skm1
ste20
tpk2


slt2 and cla4
likelihood
ranked single knockouts
−12.5
−13.0
−13.5
−14.0
−14.5
31
42
53
64
75
86
97
109
123
137
AND
OR
XOR
slt2 masks cla4
cla4 masks slt2
no epistasis
no information



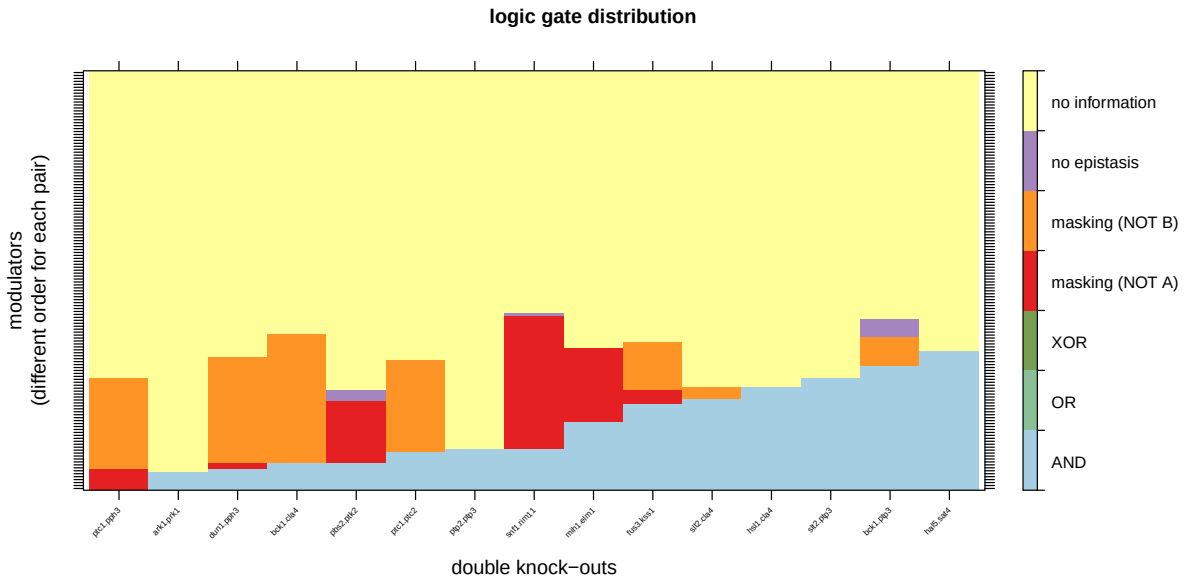







```
distmat <- wageningen$logic

distmat[which(distmat %in% "AND")] <- 1
distmat[which(distmat %in% "OR")] <- 2
distmat[which(distmat %in% "XOR")] <- 3
distmat[which(distmat %in% "NOEPI")] <- 6
distmat[which(distmat %in% c("NOINFO", "NOINF"))] <- 7

for (i in 1:ncol(distmat)) {

    genes <- unlist(strsplit(colnames(distmat)[i], "\\."))

    distmat[which(distmat[, i] %in% paste(genes[1], " masks the effect of ", genes[2], sep = "")), i] <-
```

```
    distmat[which(distmat[, i] %in% paste(genes[2], " masks the effect of ", genes[1], sep = "")), i] <-

}

distmat <- apply(distmat, c(1,2), as.numeric)

for (i in 1:ncol(distmat)) {
    distmat[, i] <- rev(sort(distmat[, i]))
}

rownames(distmat) <- 1:nrow(distmat)

distmat <- distmat[-which(apply(distmat, 1, sum) == 0), ]

distmat <- distmat[, -which(apply(distmat, 2, max) == 0 | apply(distmat, 2, min) == 7)]

library(bnem)

y <- distmat

distmat <- distmat[, order(apply(distmat, 2, function(x) { return(sum(x == 1)) }))]

y[which(y == 5)] <- 4

heatmapOP(distmat, Colv = F, Rowv = F, main = "logic gate distribution", sub = "", col = "Paired", breal
```



Sameith et al., 2015

```
data <- read.delim("http://www.holstegelab.nl/publications/GSTF_geneticinteractions/downloads/del_mutant

data <- apply(data, c(1,2), as.character)

dataM <- data[-1, which(data[1, ] %in% "M")]
```

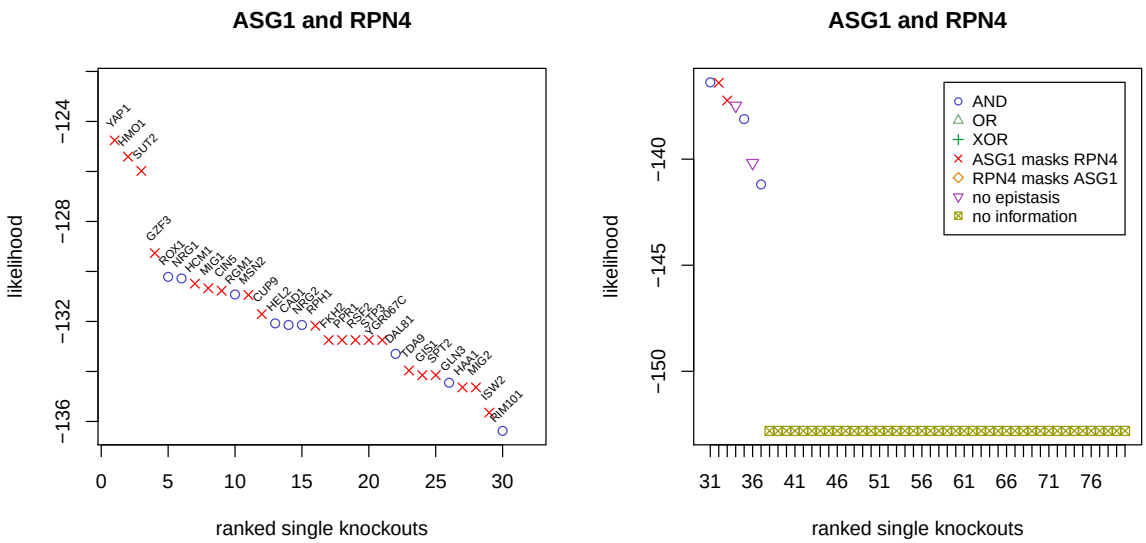
```r
dataM <- apply(dataM, c(1,2), as.numeric)

dataP <- data[-1, which(data[1, ] %in% "p.value")]

dataP <- apply(dataP, c(1,2), as.numeric)

dataBin <- dataM

sig <- 0.01

cutoff <- log2(1.5)

dataBin[which(dataP < sig & dataP > 0 & abs(dataM) >= cutoff)] <- 1

dataBin[which(dataP >= sig | dataP == 0 | abs(dataM) < cutoff)] <- 0

dataBin <- dataBin[-which(apply(dataBin, 1, max) == 0), ]

colnames(dataBin) <- gsub("\\.\\.\\.", "\\.", colnames(dataBin))

## big screen:

doubles <- colnames(dataBin)[grep("\\.", colnames(dataBin))]

doubles.genes <- unique(unlist(strsplit(doubles, "\\.")))

singles <- colnames(dataBin)[-grep("\\.", colnames(dataBin))]

singles <- unique(sort(singles))

llmat <- logicmat <- matrix(0, length(singles), length(doubles))

rownames(llmat) <- rownames(logicmat) <- singles

colnames(llmat) <- colnames(logicmat) <- doubles

globalgenes <- which(apply(dataBin, 1, max) == 1)

## for (i in doubles[set]) {
##     print(i)
##     doubles.singles <- unlist(strsplit(i, "\\."))
##     egenes <- which(apply(dataBin[, which(colnames(dataBin) %in% c(i, doubles.singles))], 1, max) ==
##     for (j in singles) {
##         print(j)
##         if (j %in% doubles.singles) { next() }

##         dataTmp <- dataBin[, grep(paste(paste("^", c(i, j, doubles.singles), "$", sep = ""), collaps

##         if (path %in% "fixed_set") {
##             dataTmp <- dataTmp[egenes, ]
##         }
##         if (path %in% "global") {
##             dataTmp <- dataTmp[globalgenes, ]
```

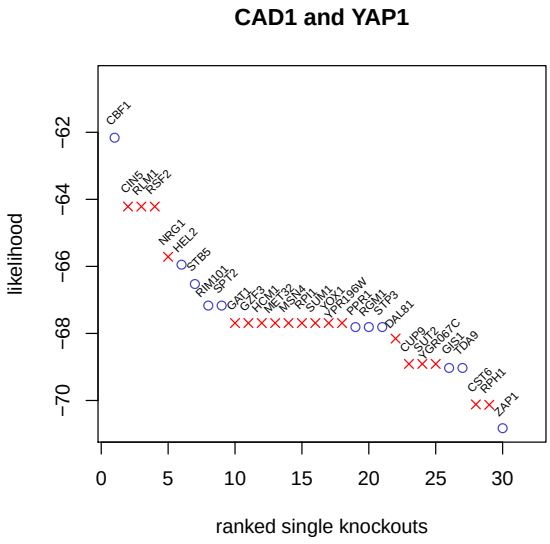
```
##         }
##         if (path %in% "") {
##             dataTmp <- dataTmp[which(apply(dataTmp, 1, max) == 1), ]
##         }

##         i1 <- which(singles %in% j)
##         i2 <- which(doubles %in% i)

##         if (!(is.null(dim(dataTmp)))) {

##             if (any(dataTmp[, j] != 0)) {

##                 epires <- epiNEM(dataTmp, method = "exhaustive")

##                 tmp <- epires$logics
##                 if ("OR" %in% tmp) {
##                     if (sum(epires$origModel[, j]) != 2) {
##                         tmp <- "NOEPI"
##                     } else {
##                         if (all(tmp %in% "OR")) {
##                             tmp <- "OR"
##                         } else {
##                             tmp <- tmp[which(!(tmp %in% "OR"))]
##                         }
##                     }
##                 }

##                 logicmat[i1, i2] <- tmp
##                 llmat[i1, i2] <- epires$score

##             } else {

##                 logicmat[i1, i2] <- "UNCON"
##                 llmat[i1, i2] <- -Inf

##             }

##         } else {

##             logicmat[i1, i2] <- "UNCON"
##             llmat[i1, i2] <- -Inf

##         }

##     }

## }
```

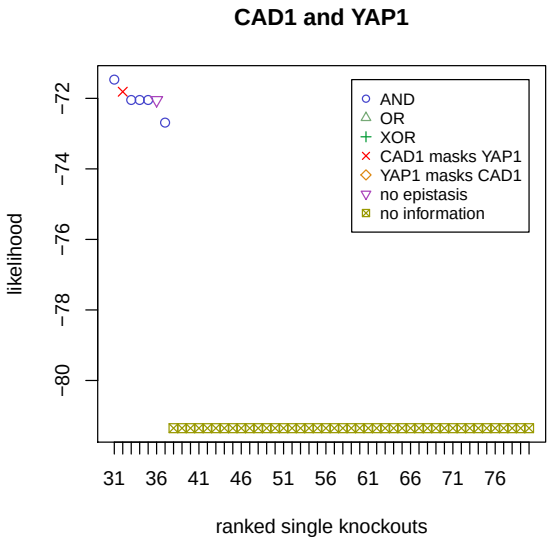
```
data(sameith_res)

llmat0 <- sameith$ll

logicmat0 <- sameith$logic
```

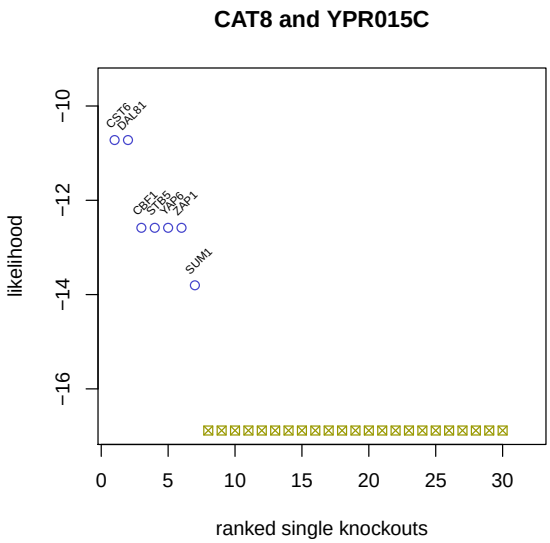
```
paperdoubles <- c(4, 9, 17)

for (i in 1:length(doubles)) {

  ## if (!(doubles[i] %in% c("ECM22.UPC2", "GLN3.GZF3"))) { next() }

  logicvec <- logicmat0[, i]

  llvec <- llmat0[, i]

  logicvec <- logicvec[order(llvec, decreasing = T)]

  llvec <- llvec[order(llvec, decreasing = T)]

  parents <- unlist(strsplit(doubles[i], "\\."))

  pchvec <- numeric(length(llvec))

    pchvec[which(logicvec %in% "AND")] <- 1
    pchvec[which(logicvec %in% "OR")] <- 2
    pchvec[which(logicvec %in% "XOR")] <- 3
    pchvec[grep(paste("^", parents[1], sep = ""), logicvec)] <- 4
    pchvec[grep(paste("^", parents[2], sep = ""), logicvec)] <- 5
    pchvec[which(logicvec %in% "NOEPI")] <- 6
    pchvec[which(logicvec %in% c("NOINFO", "NOINF"))] <- 7

    logicvec <- logicvec[-which(logicvec %in% "0")]
    pchvec <- pchvec[-which(pchvec == 0)]
    llvec <- llvec[-which(llvec == 0)]

    colvec <- pchvec

    if (all(is.infinite(llvec) == T)) {

        llvec[1:length(llvec)] <- -1000

        margin <- 100

        donames <- 30

    } else {

        llvec[which(is.infinite(llvec) == T)] <- NA

        ## llvec[which(is.infinite(llvec) == T)] <- min(llvec) - 100

        margin <- abs(max(llvec[1:30], na.rm = T) - min(llvec[1:30], na.rm = T))

        if (margin == 0) { margin <- 10 }

        donames <- 30 - sum(is.na(llvec[1:30]) == T)

        if (any(is.na(llvec[1:30]) == T)) { margin2 <- margin*2 } else { margin2 <- margin }
```

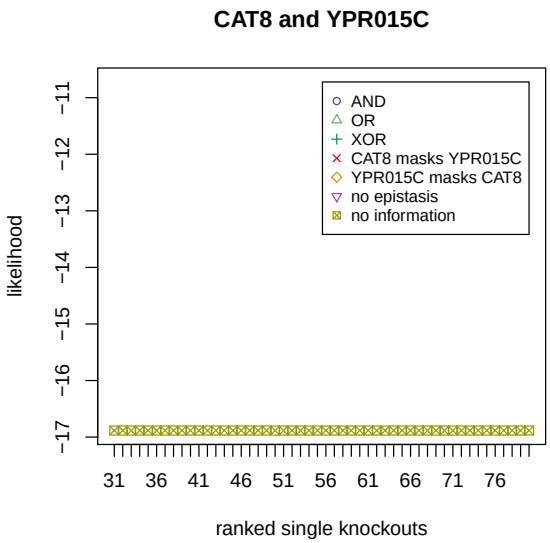
```
        llvec[which(is.na(llvec) == T)] <- min(llvec, na.rm = T) - margin

        margin <- margin2

    }

    if (all(llvec[-(1:30)] - min(llvec[-(1:30)]) == 0)) {

        p2max <- max(llvec[-(1:30)]) + margin

    } else {

        p2max <- max(llvec[-(1:30)])

    }

    par = par(mfrow=c(1,2))
    plot = plot(llvec[1:30], pch = pchvec[1:30], col = colvec[1:30], ylab = "likelihood", xlab = "ranked
    text = text((1:30)+0.5, llvec[1:30]+(margin*0.075), labels = c(names(llvec)[1:donames], rep("", 30 -
    plot2 = plot(llvec[-(1:30)], pch = pchvec[-(1:30)], col = colvec[-(1:30)], ylab = "likelihood", xlab
    legend = legend(length(llvec[-(1:30)]), p2max,
                    legend = c("AND", "OR", "XOR", paste(parents[1], " masks ", parents[2], sep = ""), p
    axis = axis(1, at = 1:length(llvec[-(1:30)]), labels = 31:length(llvec))

}
```

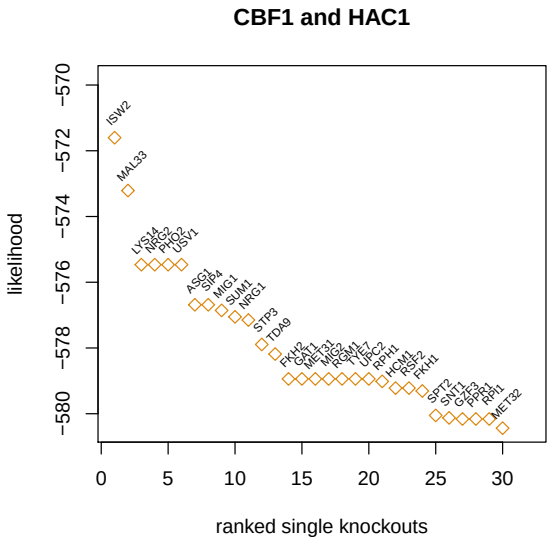


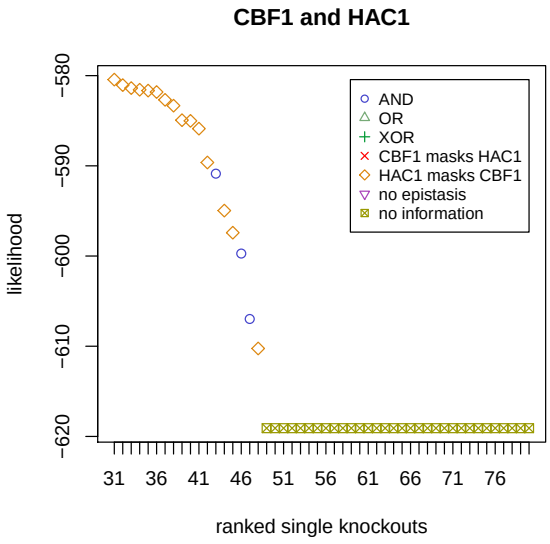

CAD1 and YAP1
likelihood
ranked single knockouts
CBF1
CIN5
RLM1
RSF2
NRG1
HEL2
STB5
RIM101
SPT2
GAT1
GAT3
HOM3
MET32
MSN4
RGM1
SUM1
XOG1
YPR196W
RPD3
RGM1
STP3
DAL81
CUP9
SUT2
GPR67C
GRF1
FDA9
CST6
RPH1
ZAP1

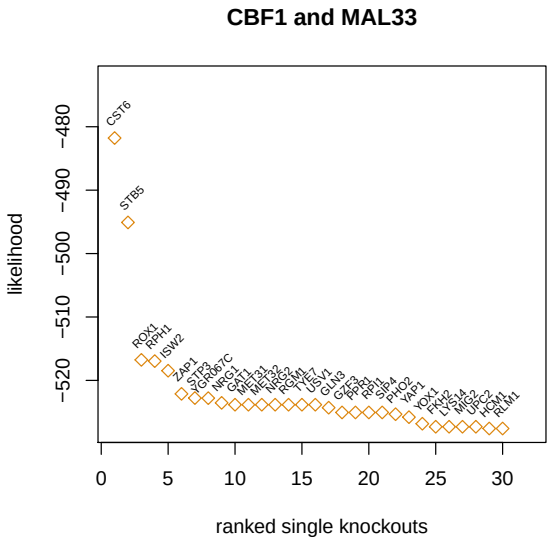

CAD1 and YAP1
likelihood
ranked single knockouts
AND
OR
XOR
CAD1 masks YAP1
YAP1 masks CAD1
no epistasis
no information

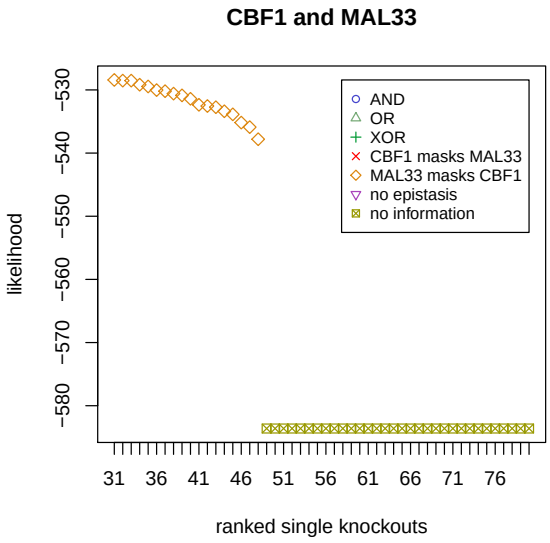

CAT8 and YPR015C
likelihood
ranked single knockouts
CST6
DAL81
CBF1
STB5
YAP6
ZAP1
SUM1

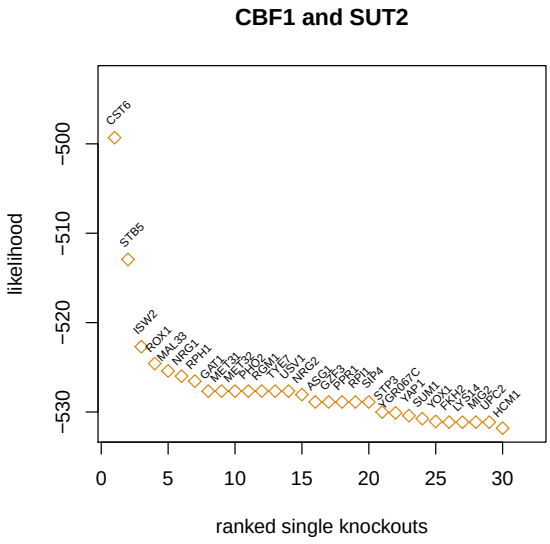

CAT8 and YPR015C
likelihood
ranked single knockouts
AND
OR
XOR
CAT8 masks YPR015C
YPR015C masks CAT8
no epistasis
no information

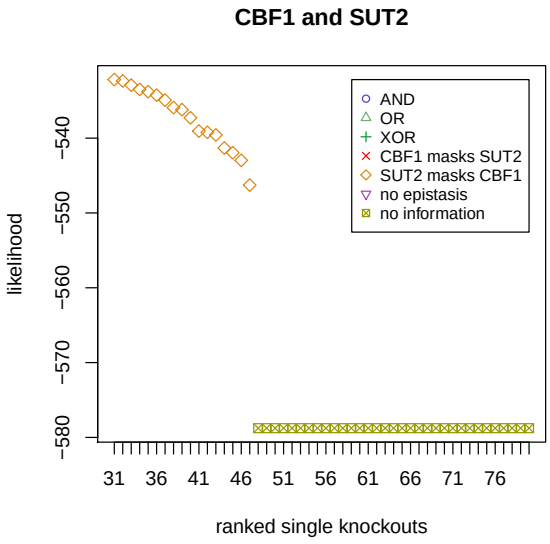

CBF1 and HAC1
likelihood
ranked single knockouts
ISW2
MAL33
LYS14
NRG2
PHO2
USV1
ASG1
SIP4
MIG1
SUM1
NRG1
STP3
TDA9
FKH2
CAT1
MET31
MET32
RGM1
MET7
UPC2
RPH1
HCM1
RGT2
FKH1
SPT2
SNT1
GTS1
PPR1
RPI1
MET32

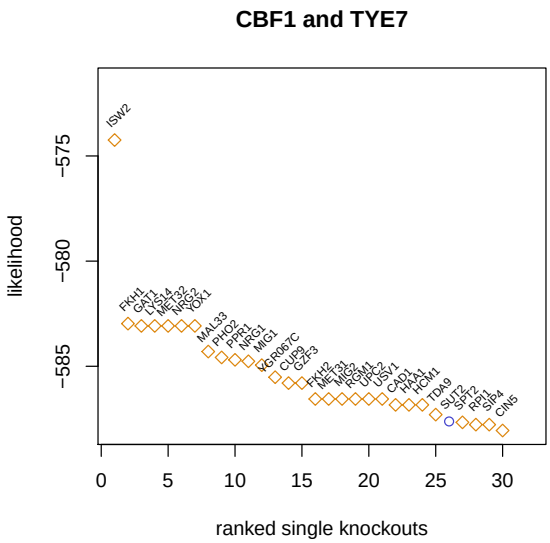

CBF1 and HAC1
likelihood
ranked single knockouts
AND
OR
XOR
CBF1 masks HAC1
HAC1 masks CBF1
no epistasis
no information

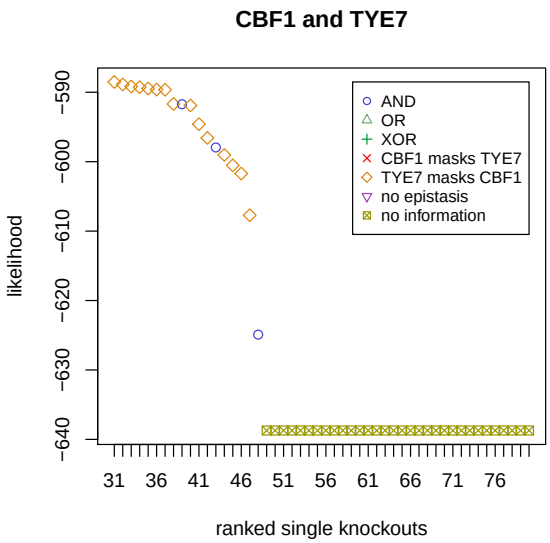

CBF1 and MAL33
likelihood
ranked single knockouts
CST6
STB5
ROX1
RPH1
ISW2
ZAP1
STP3
YGR067C
NRG1
CAT1
MET31
MET32
NRG2
RGM1
MET7
USV1
GLN3
GTS1
PPR1
SIP4
PHO2
YAP1
YOX1
FKH2
LYS14
MIG2
UPC2
HCM1
RLM1

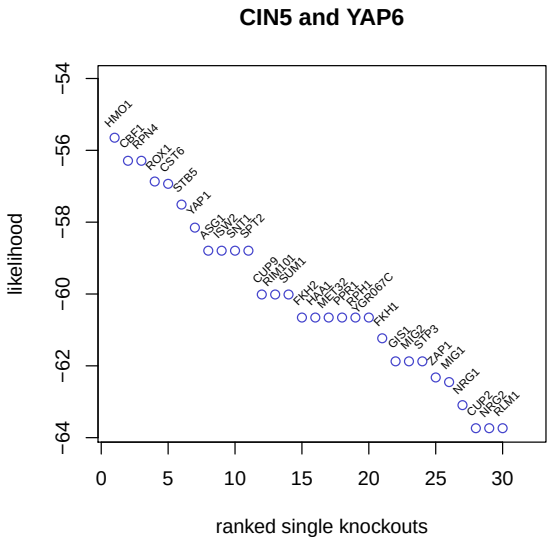

CBF1 and MAL33
likelihood
ranked single knockouts
AND
OR
XOR
CBF1 masks MAL33
MAL33 masks CBF1
no epistasis
no information

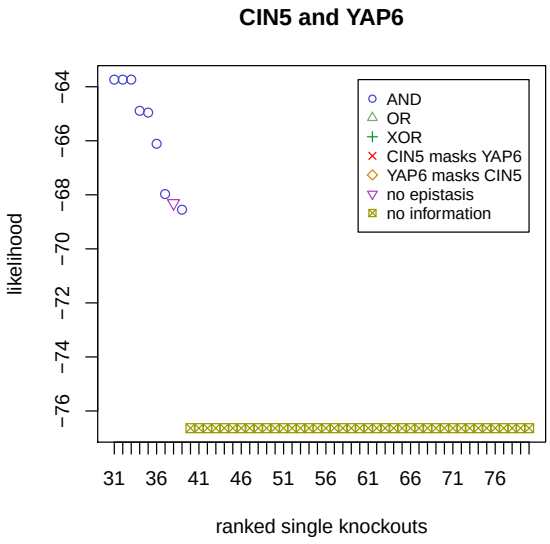

CBF1 and SUT2
likelihood
ranked single knockouts
CST6
STB5
ISW2
ROX1
MAL33
NRG1
RPH1
GAT1
MET31
MET32
PHD2
RGM1
TYE7
USV1
NRG2
ASG1
CZF3
PPR1
RPI1
SIP4
STP3
YGR067C
YAP1
SUM1
YOX1
FKH2
LYS14
MIG2
UPC2
HCM1

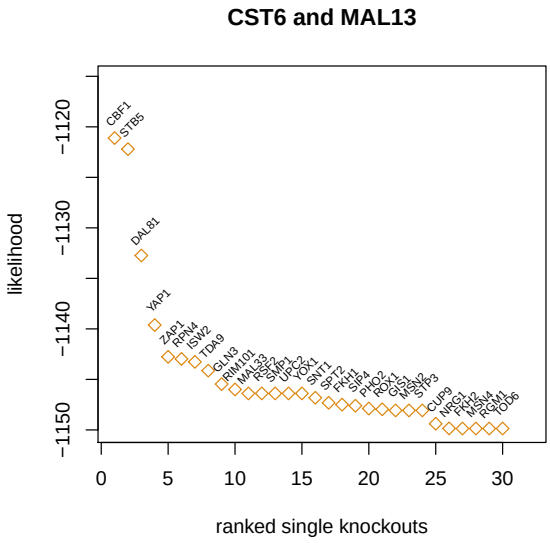

CBF1 and SUT2
likelihood
ranked single knockouts
AND
OR
XOR
CBF1 masks SUT2
SUT2 masks CBF1
no epistasis
no information

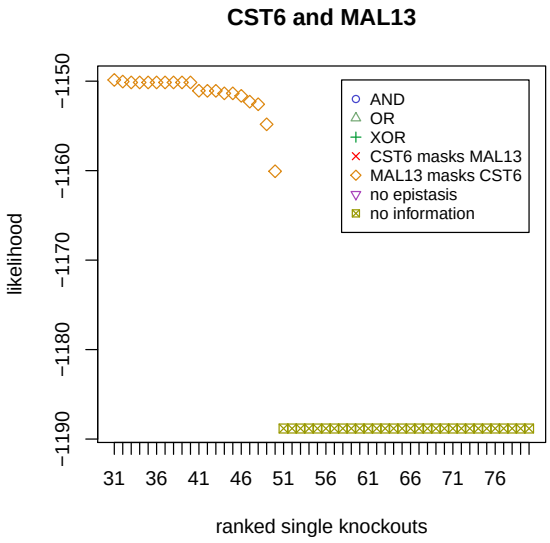

CBF1 and TYE7
likelihood
ranked single knockouts
ISW2
FKH1
GAT1
LYS14
MET32
NRG2
YOX1
MAL33
PHD2
PPR1
NRG1
MIG1
YGR067C
CUP9
CZF3
FKH2
MET31
MIG3
RGM1
UPC2
USV1
CAD1
HAA1
HCM1
TDA9
SUT2
SWI2
RPI1
SIP4
CIN5

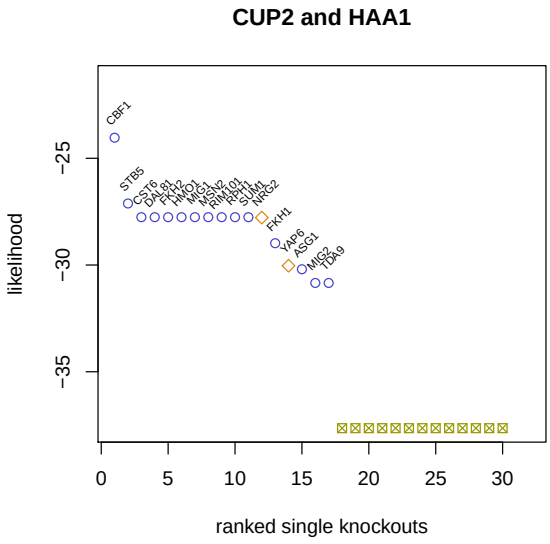

CBF1 and TYE7
likelihood
ranked single knockouts
AND
OR
XOR
CBF1 masks TYE7
TYE7 masks CBF1
no epistasis
no information

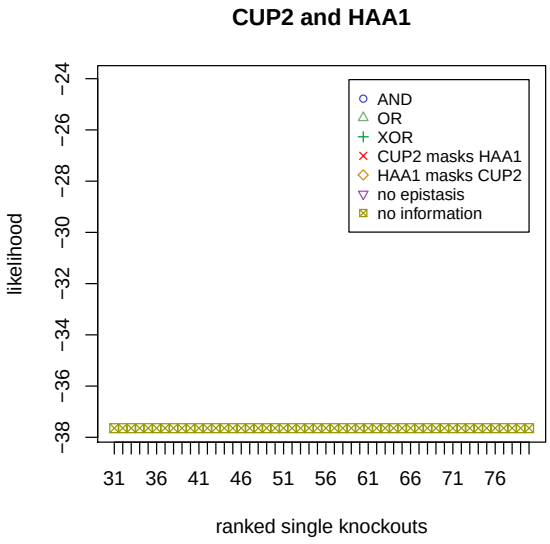

CIN5 and YAP6
likelihood
ranked single knockouts
HMO1
CBF1
RPN4
ROX1
CST6
STB5
YAP1
ASG1
ISW2
SNT1
SPT2
CUP9
RIM101
SUM1
FKH2
HAA1
MET32
PHD1
RGM1
YGR067C
FKH1
GIS1
MTG2
STP3
ZAP1
MIG1
NRG1
CUP2
NRG2
RLM1

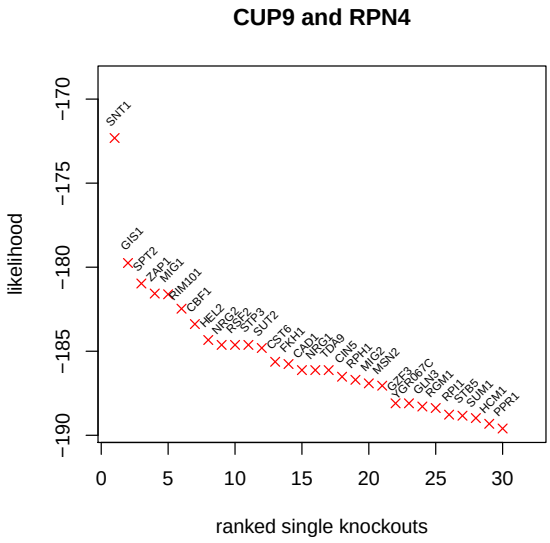

CIN5 and YAP6
likelihood
ranked single knockouts
AND
OR
XOR
CIN5 masks YAP6
YAP6 masks CIN5
no epistasis
no information

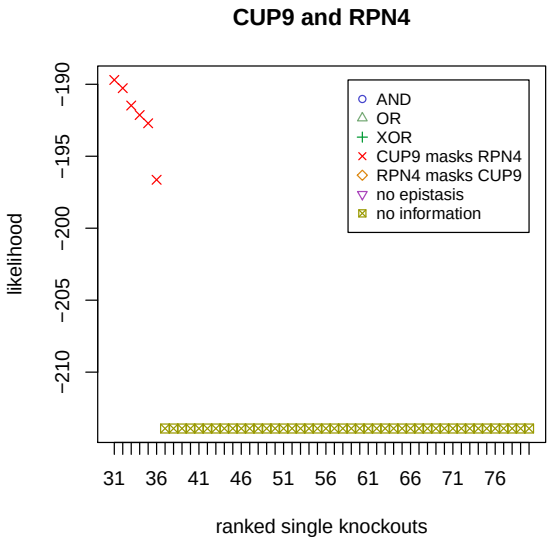

CST6 and MAL13
likelihood
ranked single knockouts
CBF1
STB5
DAL81
YAP1
ZAP1
RPN4
ISW2
TDA9
GLN3
RIM101
MAL13
RSF2
SNF1
UPC2
YOX1
SNT1
SPT2
FKH1
SIP4
PHO2
ROX1
GIS1
MGM1
STP3
CUP9
NRG1
FKH2
MSN4
RGM1
ROD6

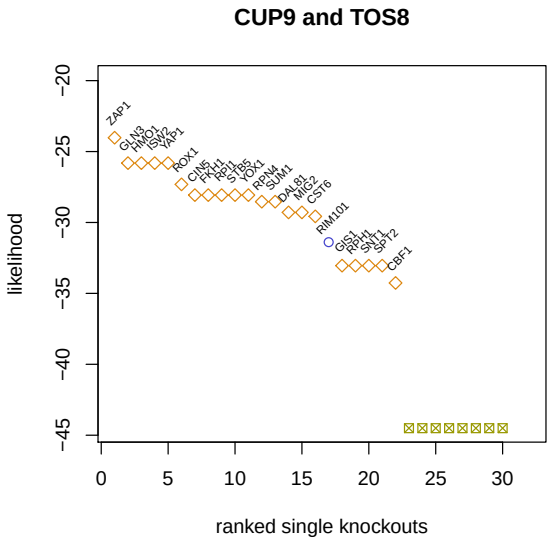

CST6 and MAL13
likelihood
ranked single knockouts
AND
OR
XOR
CST6 masks MAL13
MAL13 masks CST6
no epistasis
no information

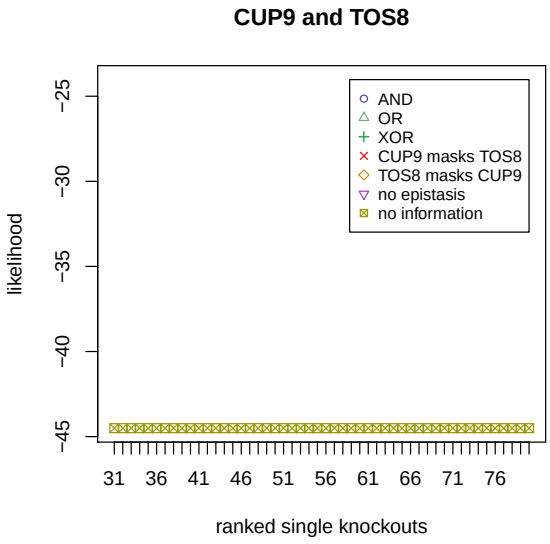

CUP2 and HAA1
likelihood
−25
−30
−35
CBF1
STB5
CST6
DAL81
PKH2
HMO1
MSN2
RIM101
RPH1
SUM1
NRG2
FKH1
YAP6
ASG1
MIG2
TDA9
0
5
10
15
20
25
30
ranked single knockouts

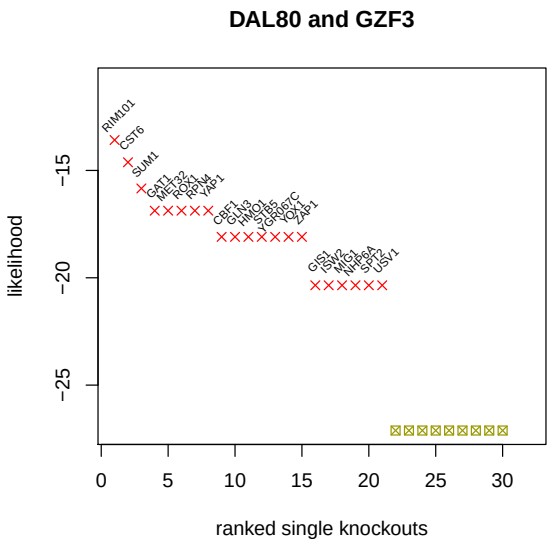

CUP2 and HAA1
likelihood
−24
−26
−28
−30
−32
−34
−36
−38
AND
OR
XOR
CUP2 masks HAA1
HAA1 masks CUP2
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

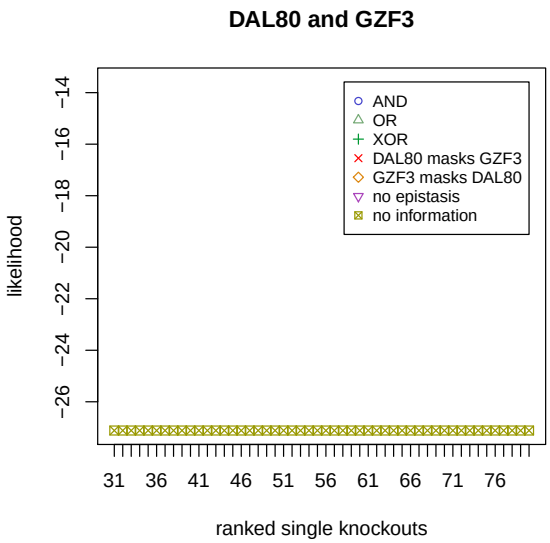

CUP9 and RPN4
likelihood
−170
−175
−180
−185
−190
SNT1
GIS1
SFP2
ZAP1
MIG1
RIM101
CBF1
HEL2
NRG2
RSE2
STP3
SUT2
CST6
FKH1
CAD1
NRG1
TDA9
CIN5
RPH1
MIG2
MSN2
GZF3
YGR067C
GLN3
RGM1
RPI1
STB5
SUM1
HOM1
PPR1
0
5
10
15
20
25
30
ranked single knockouts

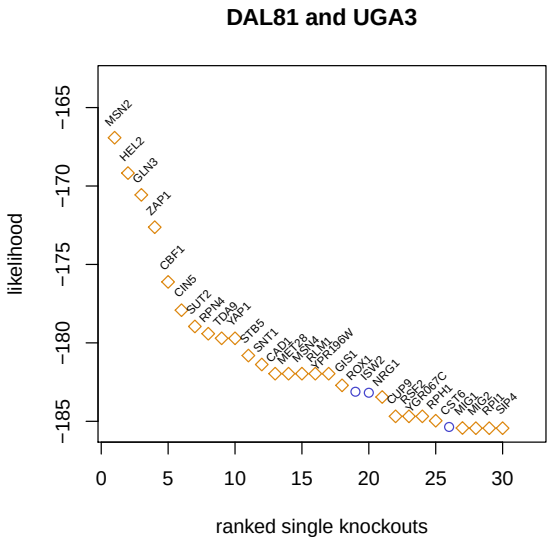

CUP9 and RPN4
likelihood
−190
−195
−200
−205
−210
AND
OR
XOR
CUP9 masks RPN4
RPN4 masks CUP9
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

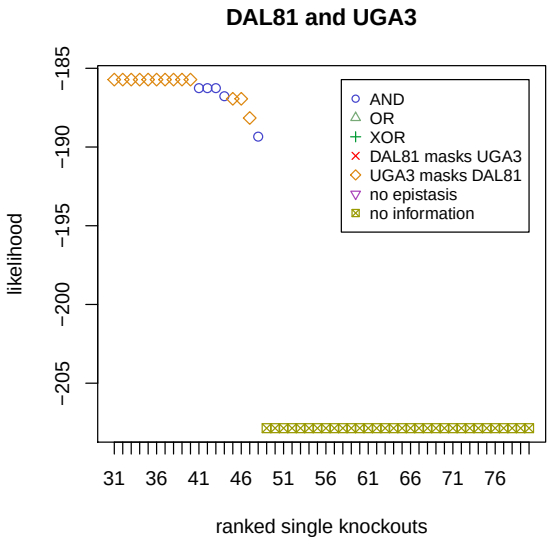

CUP9 and TOS8
likelihood
ranked single knockouts
ZAP1
GLN3
HMO1
ISW2
YAP1
ROX1
CIN5
FKH1
RFX1
STB5
YOX1
RPN4
SUM1
DAL81
MIG2
CST6
RIM101
GIS1
RSF1
RSF1
SPT2
CBF1

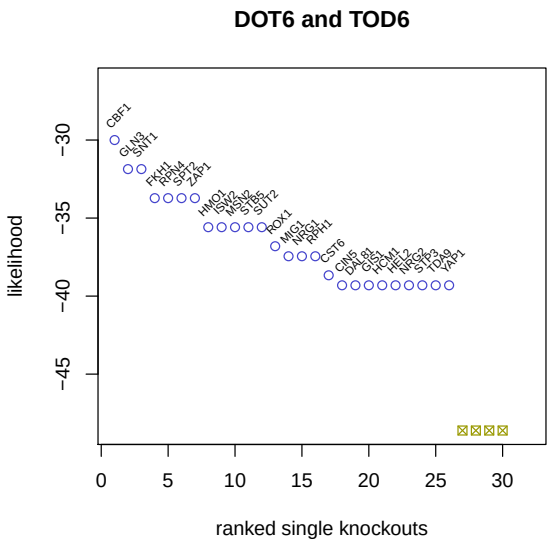

CUP9 and TOS8
likelihood
ranked single knockouts
AND
OR
XOR
CUP9 masks TOS8
TOS8 masks CUP9
no epistasis
no information

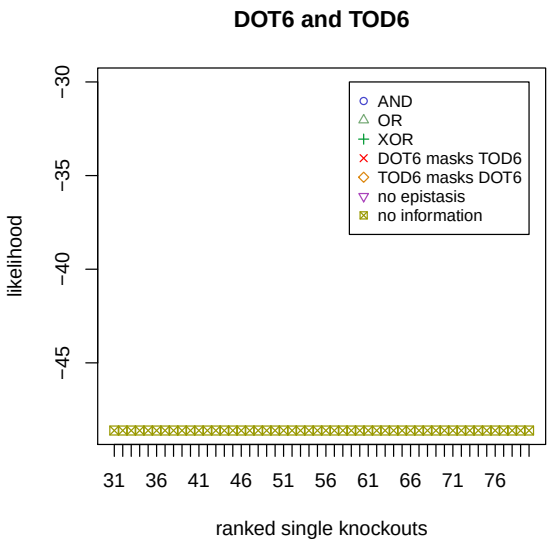

DAL80 and GZF3
likelihood
ranked single knockouts
RIM101
CST6
SUM1
GAT1
MET32
ROX1
RPN4
YAP1
CBF1
GLN3
HMO1
STB5
YGR067C
YOX1
ZAP1
GIS1
ISW2
MIG1
NRG1
SPT2
USV1

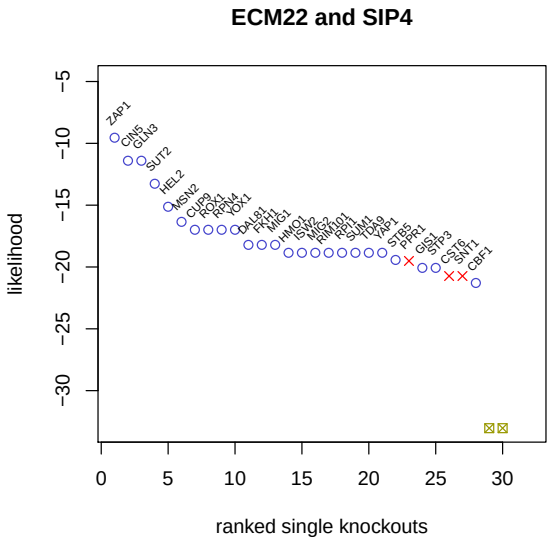

DAL80 and GZF3
likelihood
ranked single knockouts
AND
OR
XOR
DAL80 masks GZF3
GZF3 masks DAL80
no epistasis
no information

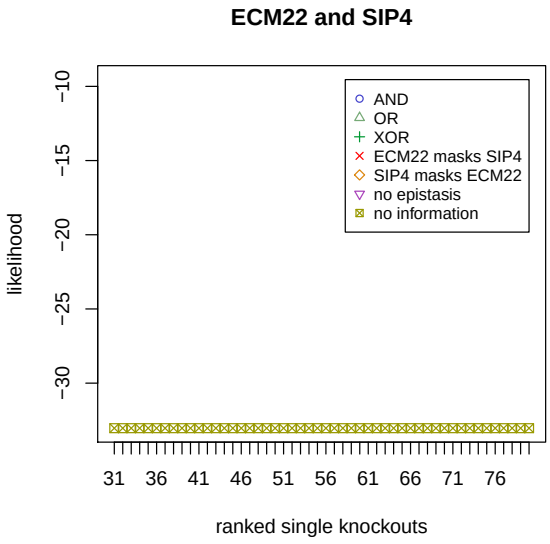

DAL81 and UGA3
likelihood
ranked single knockouts
MSN2
HEL2
GLN3
ZAP1
CBF1
CIN5
SUT2
RPN4
TDA9
YAP1
STB5
SNT1
CAD1
MET18
MSN4
RLM1
YPR196W
GIS1
ROX1
ISW2
NRG1
CUP9
RSF2
YGR067C
RPH1
CST6
MIG1
MIG2
RGT1
SIP4

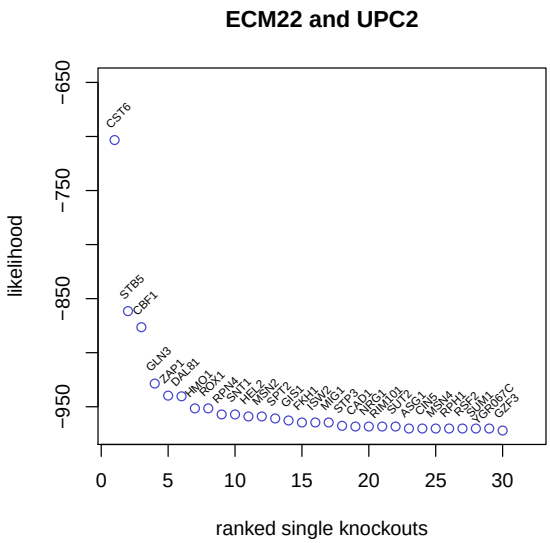

DAL81 and UGA3
likelihood
ranked single knockouts
AND
OR
XOR
DAL81 masks UGA3
UGA3 masks DAL81
no epistasis
no information

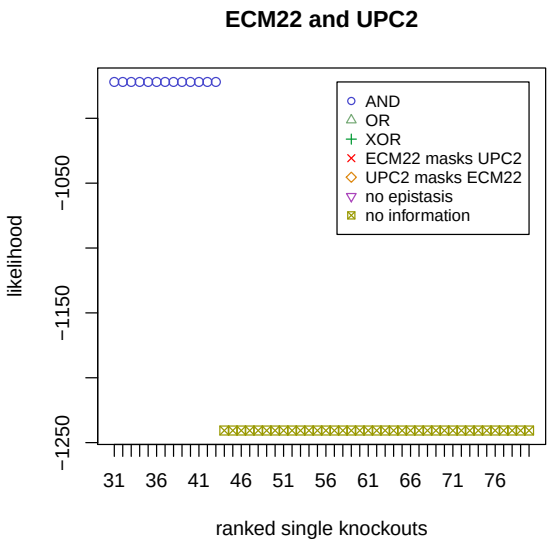

DOT6 and TOD6
likelihood
ranked single knockouts
CBF1
GLN3
SNT1
FKH1
RPN4
SFP1
ZAP1
HMO1
ISW2
MSN2
STB5
SUT2
ROX1
MIG1
NRG1
RPH1
CST6
CIN5
DAL81
HCM1
HEL2
NRG2
STP3
TDA9
YAP1

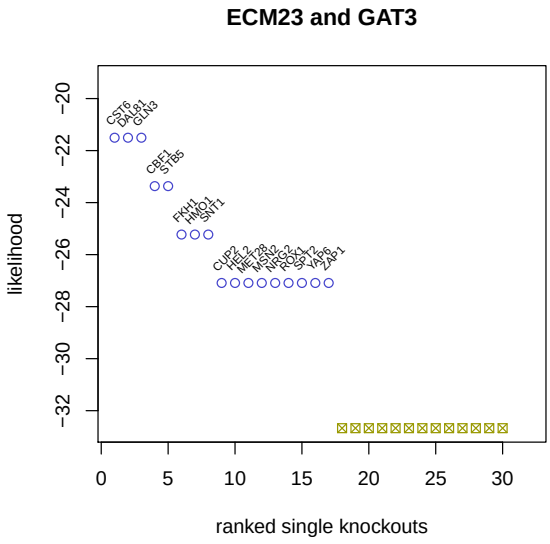

DOT6 and TOD6
likelihood
ranked single knockouts
AND
OR
XOR
DOT6 masks TOD6
TOD6 masks DOT6
no epistasis
no information

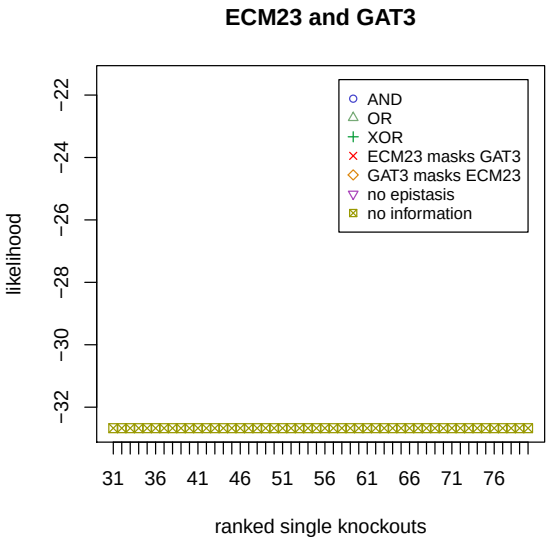

ECM22 and SIP4
likelihood
ranked single knockouts
ZAP1
CIN5
GLN3
SUT2
HEL2
MSN2
CUP9
ROX1
RPN4
NOX1
DAL81
RCH1
MIG1
HMO1
ISW2
RMT101
RPH1
SUM1
DAL9
YAP1
STB5
PPR1
GIS1
STP3
CST6
SNT1
CBF1

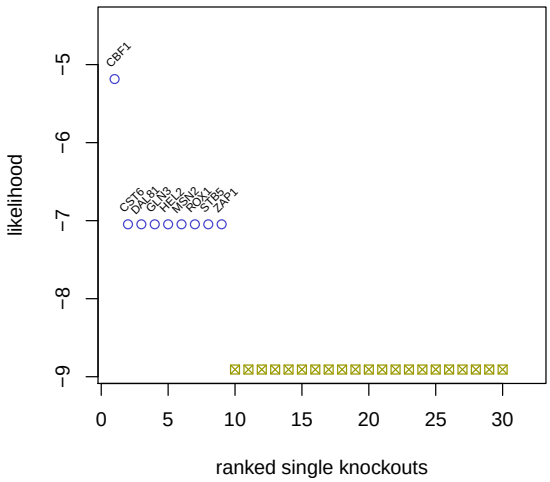

ECM22 and SIP4
likelihood
ranked single knockouts
AND
OR
XOR
ECM22 masks SIP4
SIP4 masks ECM22
no epistasis
no information

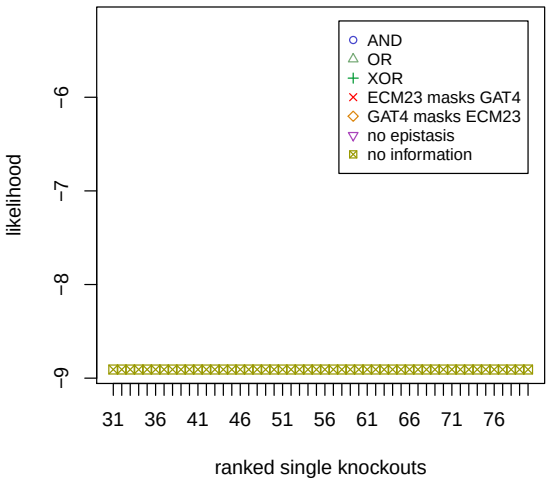

ECM22 and UPC2
likelihood
ranked single knockouts
CST6
STB5
CBF1
GLN3
ZAP1
DAL81
HMO1
ROX1
RPN4
SNT1
HEL2
MSN2
SPT2
GIS1
RCH1
ISW2
MIG1
STP3
CAD1
RMT101
SUT2
ASG1
CIN5
MSN4
RPH1
RSF2
SUM1
YGR067C
GZF3

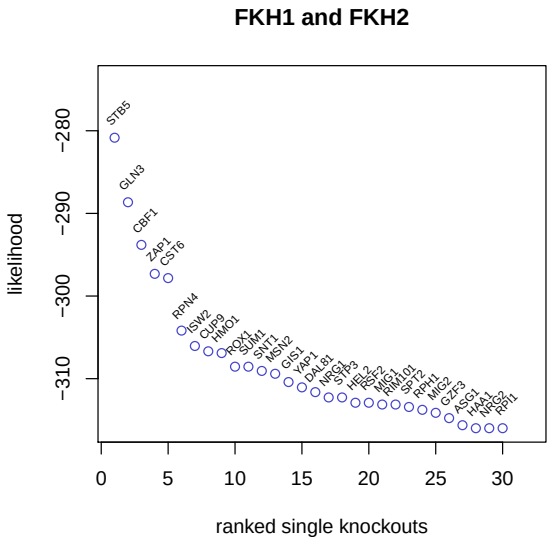

ECM22 and UPC2
likelihood
ranked single knockouts
AND
OR
XOR
ECM22 masks UPC2
UPC2 masks ECM22
no epistasis
no information

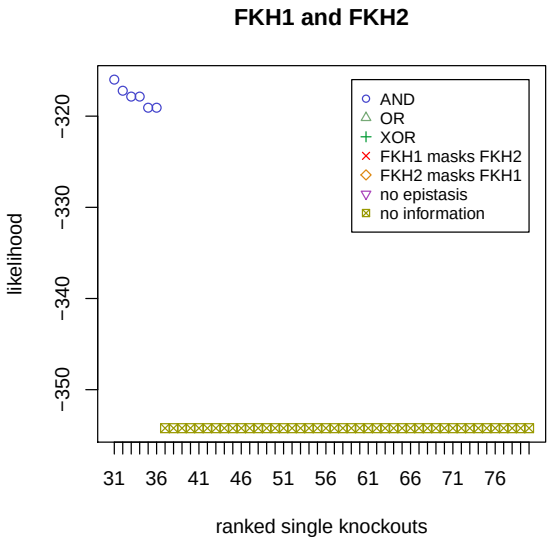

ECM23 and GAT3
likelihood
ranked single knockouts
CST6
DAL81
GLN3
CBF1
STB5
FKH1
HMO1
SFP1
CUP2
HEF3
MET28
MSN2
NRG2
ROX1
SKN7
YAP6
ZAP1

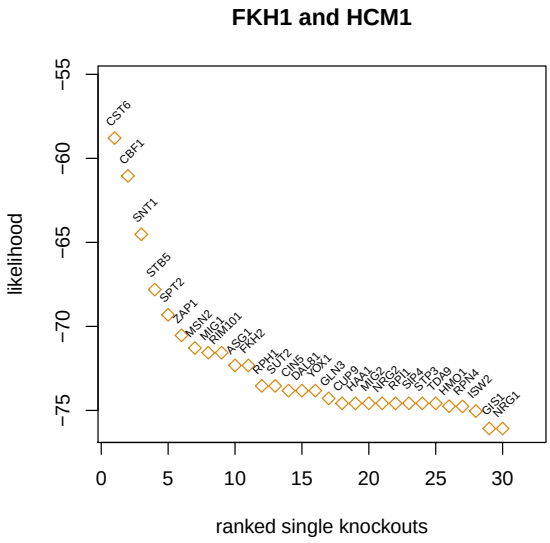

ECM23 and GAT3
likelihood
ranked single knockouts
AND
OR
XOR
ECM23 masks GAT3
GAT3 masks ECM23
no epistasis
no information

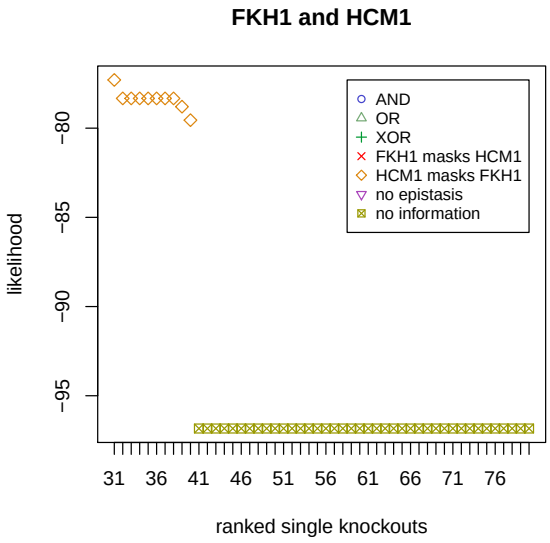

ECM23 and GAT4
likelihood
ranked single knockouts
CBF1
CST6
DAL81
GLN3
HEF3
MSN2
ROX1
STB5
ZAP1

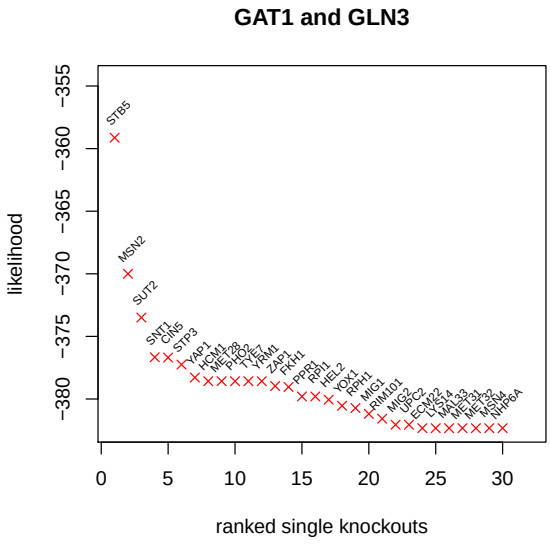

ECM23 and GAT4
likelihood
ranked single knockouts
AND
OR
XOR
ECM23 masks GAT4
GAT4 masks ECM23
no epistasis
no information

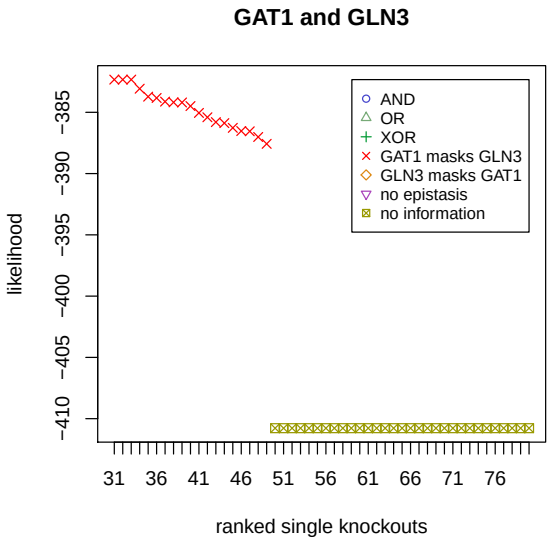

FKH1 and FKH2
likelihood
−280
−290
−300
−310
STB5
GLN3
CBF1
ZAP1
CST6
RPN4
ISW2
CUP9
HMO1
ROX1
SUM1
SNT1
MSN2
GIS1
YAP1
DAL81
NRG1
STP3
HEL2
RGT2
MIG1
RIM101
SPT2
RPH1
MIG2
GZF3
ASG1
HAA1
NRG2
RPI1
0
5
10
15
20
25
30
ranked single knockouts


FKH1 and FKH2
likelihood
−320
−330
−340
−350
AND
OR
XOR
FKH1 masks FKH2
FKH2 masks FKH1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

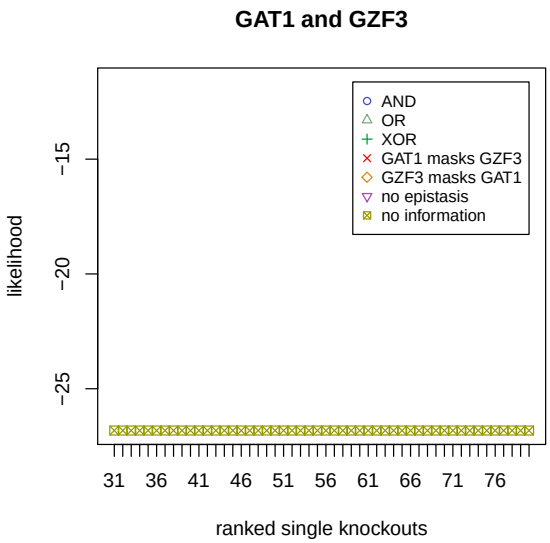

FKH1 and HCM1
likelihood
−55
−60
−65
−70
−75
CST6
CBF1
SNT1
STB5
SPT2
ZAP1
MSN2
MIG1
RIM101
ASG1
FKH2
RPH1
SUT2
CIN5
DAL81
DOT1
GLN3
CUP9
HAA1
MIG2
NRG2
RGT1
SIP4
STP3
TDA9
HMO1
RPN4
ISW2
GIS1
NRG1
0
5
10
15
20
25
30
ranked single knockouts

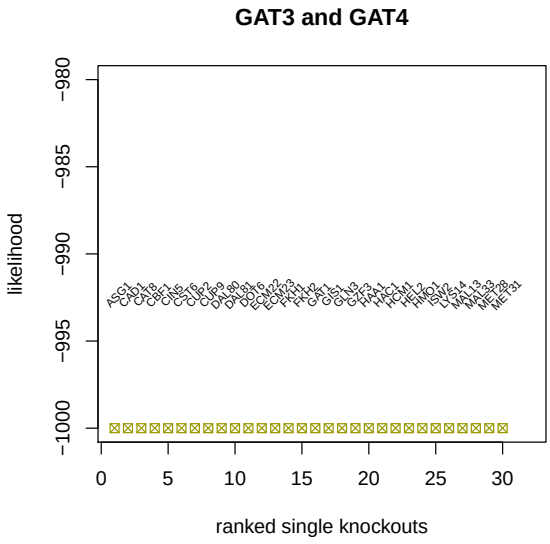

FKH1 and HCM1
likelihood
−80
−85
−90
−95
AND
OR
XOR
FKH1 masks HCM1
HCM1 masks FKH1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

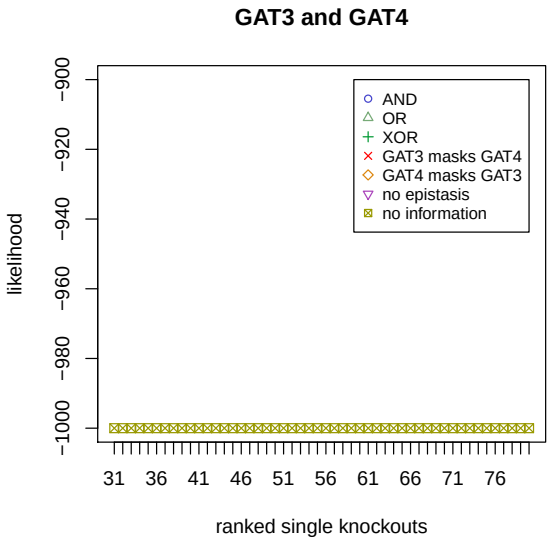

GAT1 and GLN3
likelihood
ranked single knockouts
STB5
MSN2
SUT2
SNT1
CAN5
STP3
YAP1
HOM3
MEP2
HOT1
LYS7
URM1
ZAP1
FKH1
PPR1
RPI1
HEL2
YOX1
RPH1
MIG1
RIM101
MIG2
UBC2
ECM22
LYS14
MAL33
MET31
MET32
MET4
NRG1
NRG2
NRG4
NRG6A

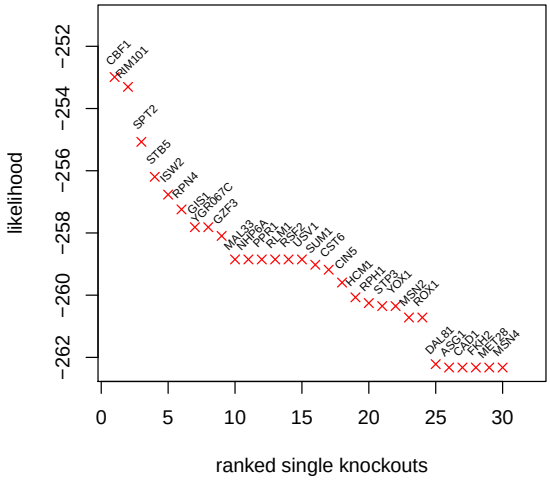

GAT1 and GLN3
likelihood
ranked single knockouts
AND
OR
XOR
GAT1 masks GLN3
GLN3 masks GAT1
no epistasis
no information

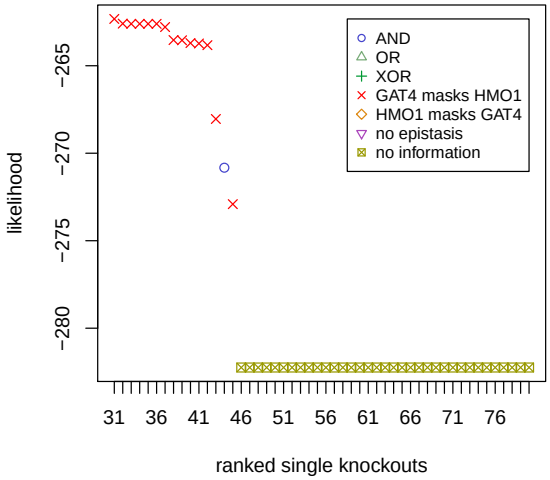

GAT1 and GZF3
likelihood
ranked single knockouts
RIM101
SUM1
CST6
TMO1
CBF1
GLN3
YGR067C
YOX1
ZAP1
GIS1
ISW2
MIG1
NRG2
NRG4
SPT7
UIP1
MET32
FOX1
YAP1
STB5

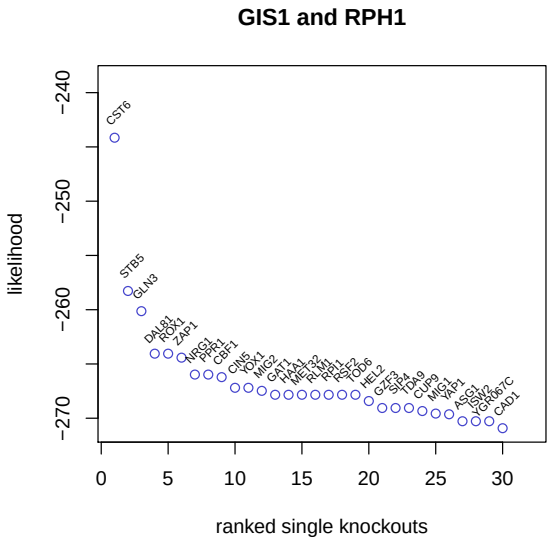

GAT1 and GZF3
likelihood
ranked single knockouts
AND
OR
XOR
GAT1 masks GZF3
GZF3 masks GAT1
no epistasis
no information

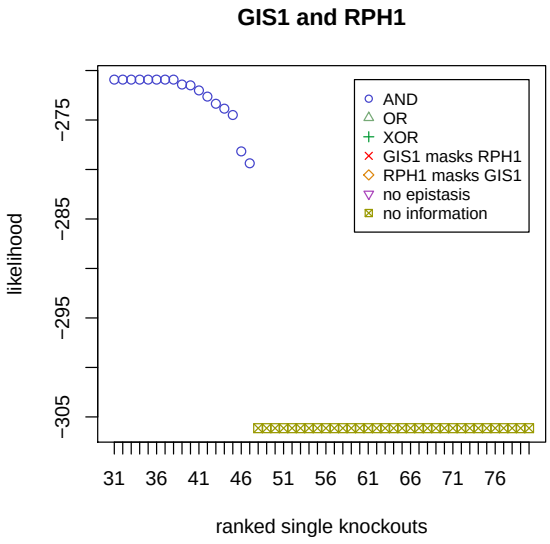

GAT3 and GAT4
likelihood
−980
−985
−990
−995
−1000
0
5
10
15
20
25
30
ranked single knockouts

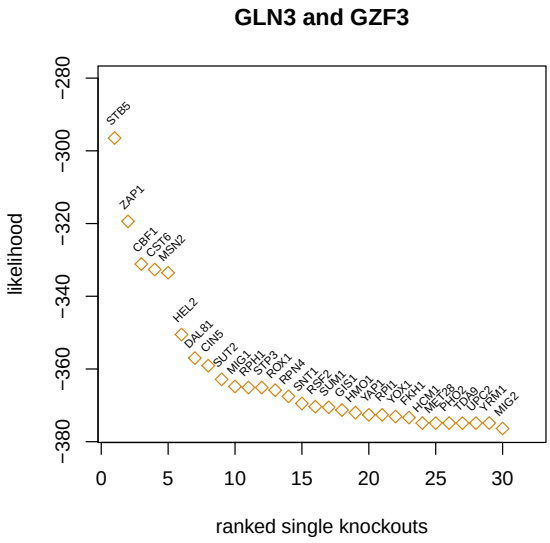

GAT3 and GAT4
likelihood
−900
−920
−940
−960
−980
−1000
AND
OR
XOR
GAT3 masks GAT4
GAT4 masks GAT3
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

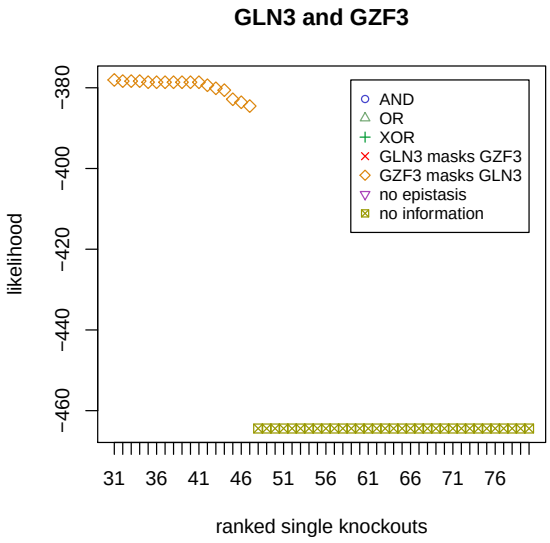

GAT4 and HMO1
likelihood
−252
−254
−256
−258
−260
−262
CBF1
RIM101
SPT2
STB5
ISW2
RPN4
GIS1
YGR067C
GZF3
MAL13
NRG1
PBP1
RLM1
HSF2
USV1
SUM1
CST6
CIN5
HCM1
RPH1
STP3
YOX1
MSN2
ROX1
DAL81
ASG1
CAD1
PDR1
MET28
MSN4
0
5
10
15
20
25
30
ranked single knockouts

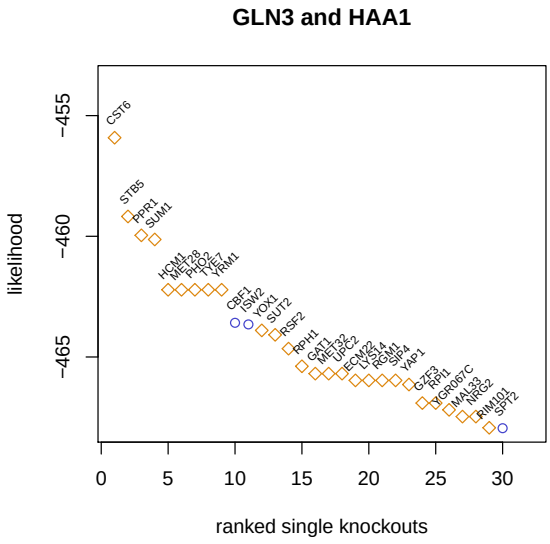

GAT4 and HMO1
likelihood
−265
−270
−275
−280
AND
OR
XOR
GAT4 masks HMO1
HMO1 masks GAT4
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

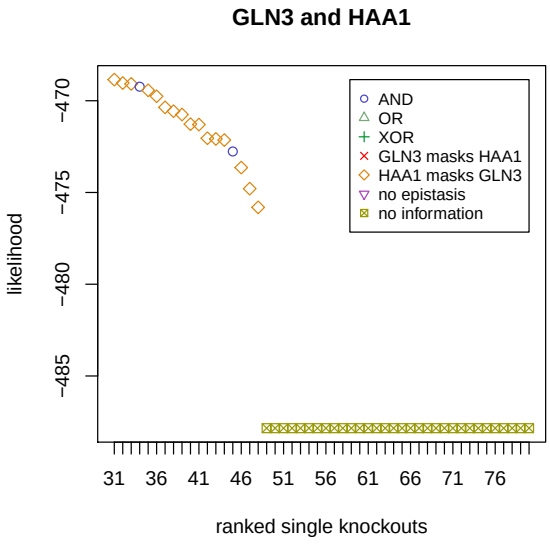

GIS1 and RPH1
likelihood
ranked single knockouts
CST6
STB5
GLN3
DAL81
ROX1
ZAP1
NRG1
PPR1
CBF1
CIN5
YDL1
MIG2
GAT1
HAL1
MET32
RLM1
RPH1
ASF2
TOD6
HEL2
GZF3
STP4
TDA9
CUP9
MIG1
YAP1
ASG1
ISU2
YGR067C
CAD1

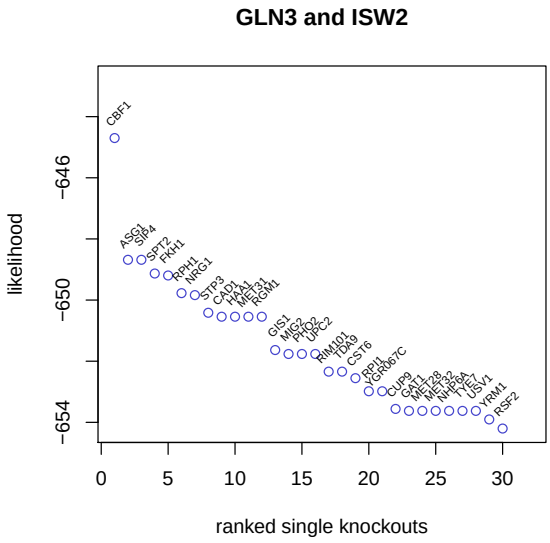

GIS1 and RPH1
likelihood
ranked single knockouts
AND
OR
XOR
GIS1 masks RPH1
RPH1 masks GIS1
no epistasis
no information

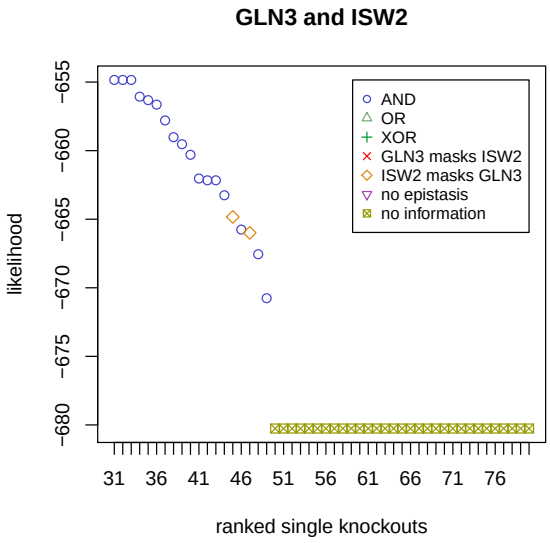

GLN3 and GZF3
likelihood
ranked single knockouts
STB5
ZAP1
CBF1
CST6
MSN2
HEL2
DAL81
CIN5
SUT2
MIG1
RPH1
STP3
ROX1
RPN4
SNT1
RSF2
SUM1
GIS1
HMO1
YAP1
RPB1
YDL1
FKH1
HOM3
MET28
PHD1
TDA9
UPC2
YRM1
MIG2

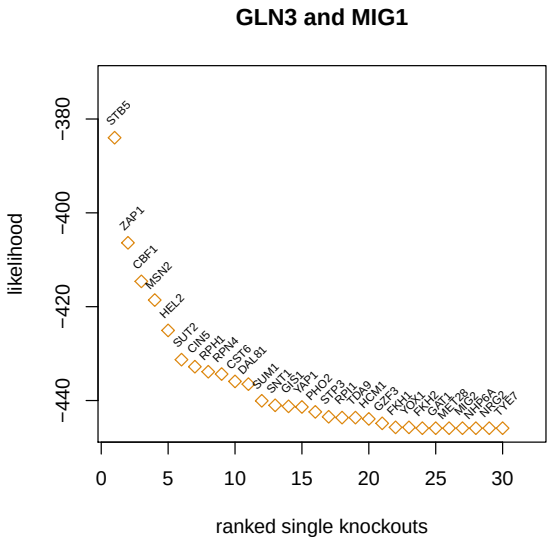

GLN3 and GZF3
likelihood
ranked single knockouts
AND
OR
XOR
GLN3 masks GZF3
GZF3 masks GLN3
no epistasis
no information

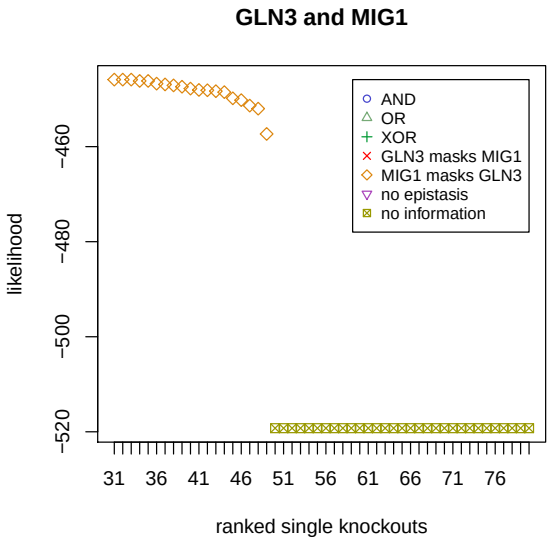

GLN3 and HAA1
likelihood
−455
−460
−465
CST6
STB5
PPR1
SUM1
HCM1
MET28
PHD2
TYE7
YRM1
CBF1
ISW2
YOX1
SUT2
RSF2
RPH1
CAT1
MET32
UPC2
ECM22
LYS14
RGM1
SIP4
YAP1
GZF3
RPI1
YGR067C
MAL33
NRG2
RIM101
SPT2
0
5
10
15
20
25
30
ranked single knockouts

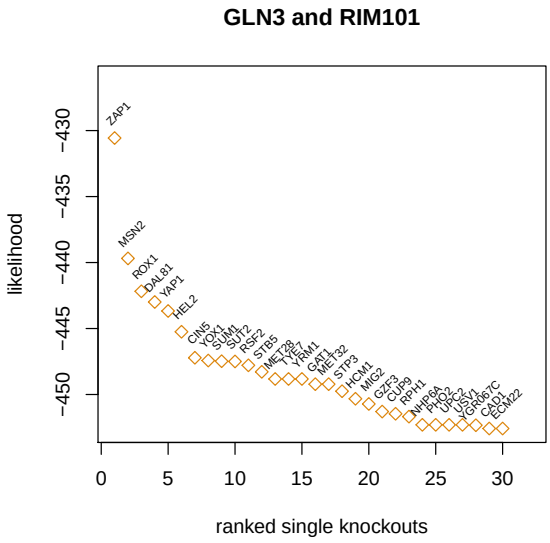

GLN3 and HAA1
likelihood
−470
−475
−480
−485
AND
OR
XOR
GLN3 masks HAA1
HAA1 masks GLN3
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

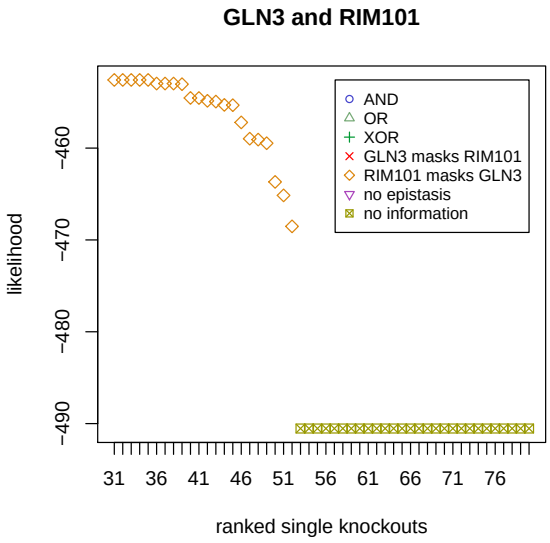

GLN3 and ISW2
likelihood
−646
−650
−654
CBF1
ASG1
SIP4
SPT2
FKH1
RPH1
NRG1
STP3
CAD1
HAL1
MET31
RGM1
GIS1
MIG2
PHD2
UPC2
RIM101
ADA2
CST6
RPI1
YGR067C
CUP9
GAT1
MET28
MET32
NRG1
TYE7
USV1
YRM1
RSF2
0
5
10
15
20
25
30
ranked single knockouts

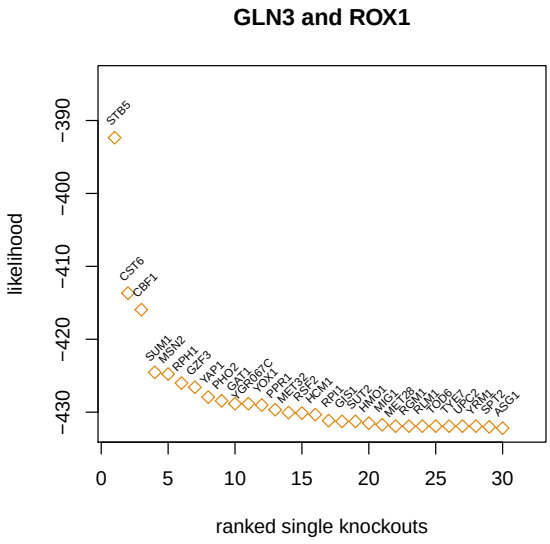

GLN3 and ISW2
likelihood
−655
−660
−665
−670
−675
−680
AND
OR
XOR
GLN3 masks ISW2
ISW2 masks GLN3
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

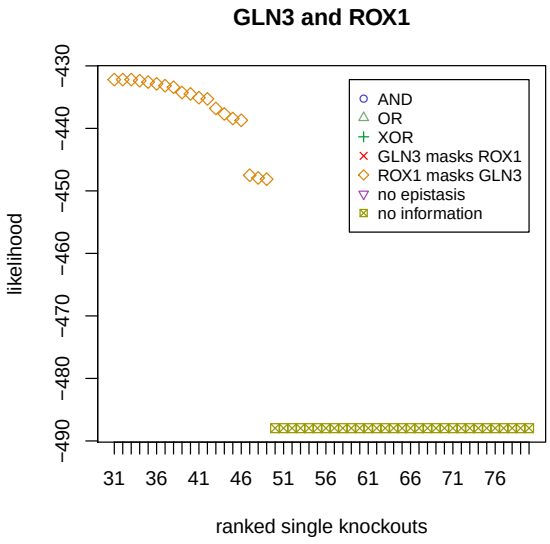

GLN3 and MIG1
likelihood
−380
−400
−420
−440
STB5
ZAP1
CBF1
MSN2
HEL2
SUT2
CIN5
RPI1
RPN4
CST6
DAL81
SUM1
SNT1
GLN1
YAP1
PHO2
STP3
RPI1
RDA9
HCM1
GZF3
FKH1
YOX1
FKH2
GAT1
MET28
MIG2
NHP6A
NRG2
TYE7
0
5
10
15
20
25
30
ranked single knockouts

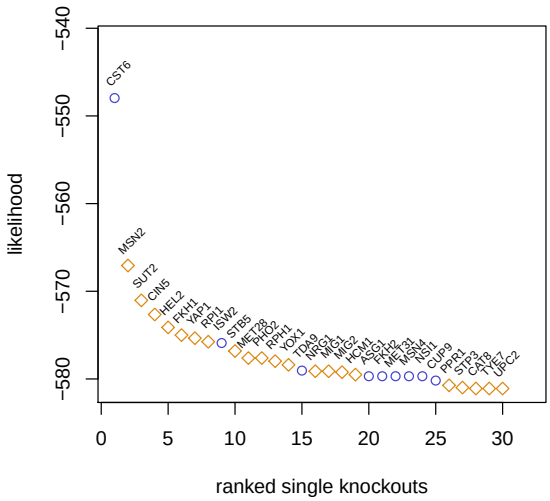

GLN3 and MIG1
likelihood
−460
−480
−500
−520
AND
OR
XOR
GLN3 masks MIG1
MIG1 masks GLN3
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

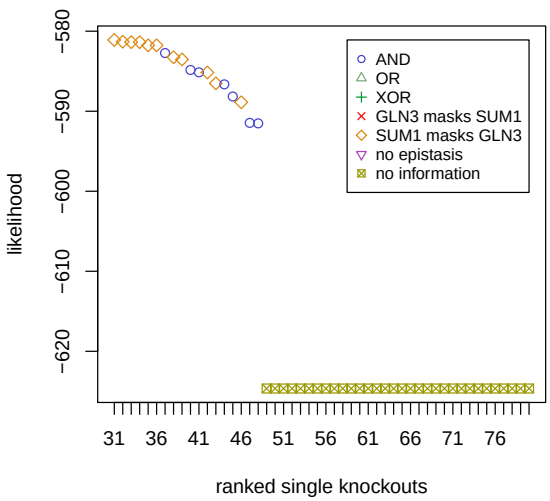

GLN3 and RIM101
likelihood
−430
−435
−440
−445
−450
ZAP1
MSN2
ROX1
DAL81
YAP1
HEL2
CIN5
YOX1
SUM1
SUT2
RSF2
STB5
MET28
TYE7
YRM1
GAT1
MET32
STP3
HCM1
MIG2
GZF3
CUP9
RPH1
NHP6A
FHO2
UPC2
USV1
YGR067C
CAD1
ECM22
0
5
10
15
20
25
30
ranked single knockouts

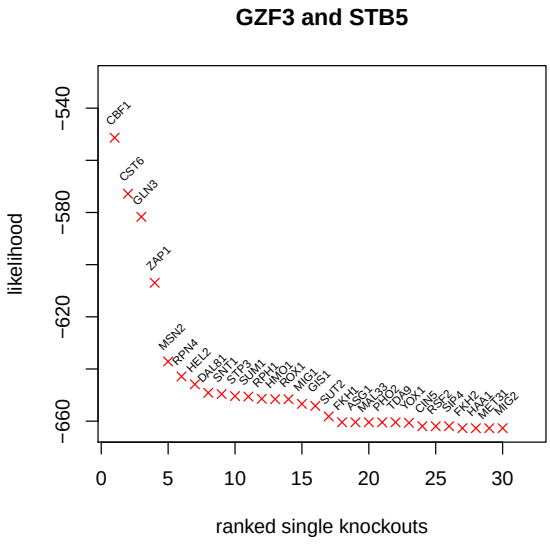

GLN3 and RIM101
likelihood
−460
−470
−480
−490
AND
OR
XOR
GLN3 masks RIM101
RIM101 masks GLN3
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

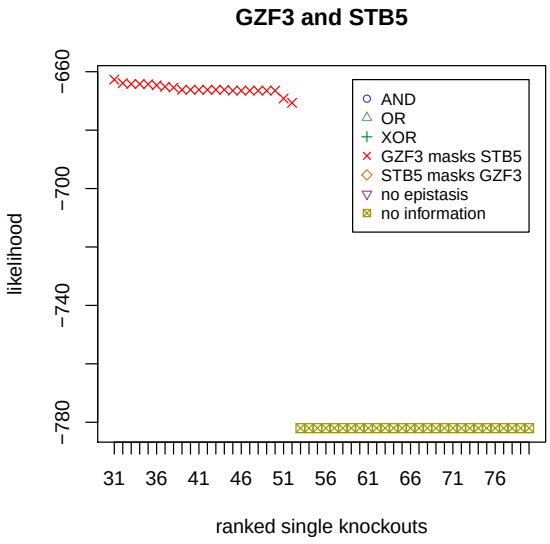

GLN3 and ROX1
likelihood
ranked single knockouts
STB5
CST6
CBF1
SUM1
MSN2
RPH1
GZF3
YAP1
PHO2
GAL1
CRM6/7C
YOX1
PPR1
MET32
HSF2
HCM1
RPI1
GIS1
SUT2
HMO1
MIG2
MET28
RGM1
RLM1
TOD6
TUP2
UPC2
YRM1
SKN7
ASG1

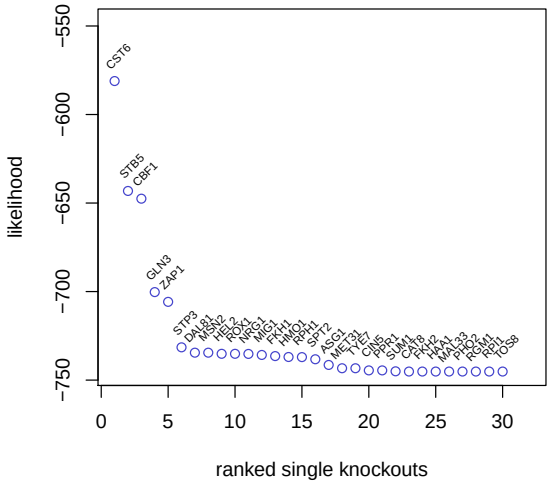

GLN3 and ROX1
likelihood
ranked single knockouts
AND
OR
XOR
GLN3 masks ROX1
ROX1 masks GLN3
no epistasis
no information

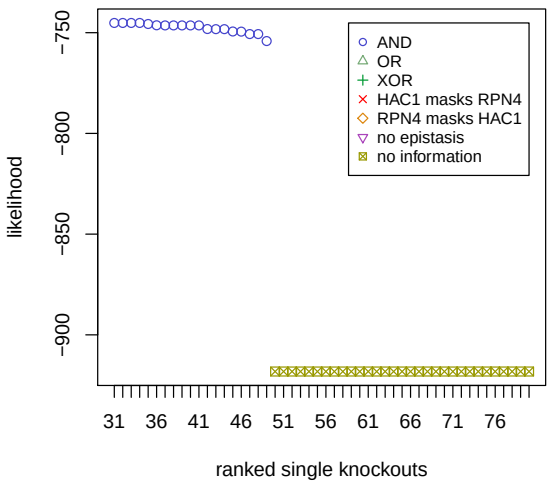

GLN3 and SUM1
likelihood
ranked single knockouts
CST6
MSN2
SUT2
CIN5
HEL2
FKH1
YAP1
RPI1
ISW2
STB5
MET28
PHO2
RPH1
YOX1
TDA9
NRG1
MIG1
MIG2
HCM1
ASG1
FKH2
MET31
MSN4
NSI1
CUP9
PPR1
STP3
CAT8
TYE7
UPC2

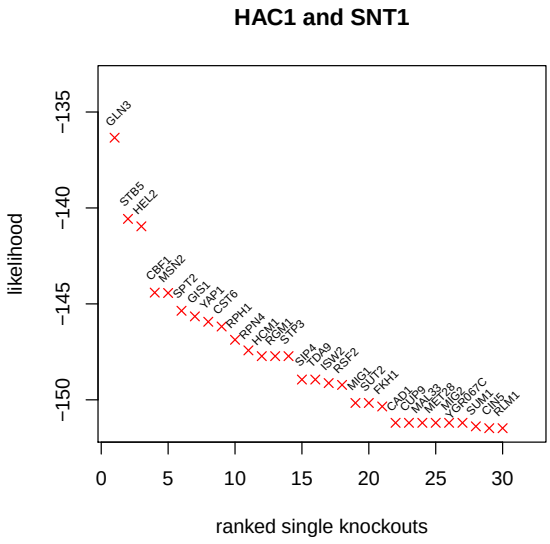

GLN3 and SUM1
likelihood
ranked single knockouts
AND
OR
XOR
GLN3 masks SUM1
SUM1 masks GLN3
no epistasis
no information

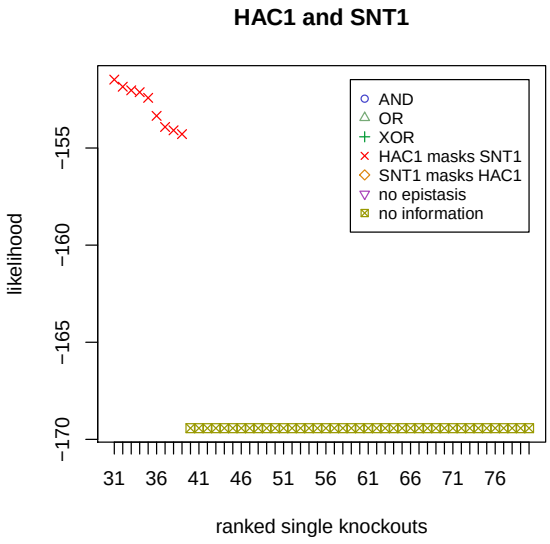

GZF3 and STB5
likelihood
ranked single knockouts
CBF1
CST6
GLN3
ZAP1
MSN2
RPN4
HEL2
DAL81
SNT1
STP3
SUM1
RPH1
HMO1
ROX1
MIG1
GIS1
SUT2
FKH1
NRG1
MAL33
PHD1
YOX1
CIN5
RSF2
SPT4
FKH2
HAA1
MET31
MIG2

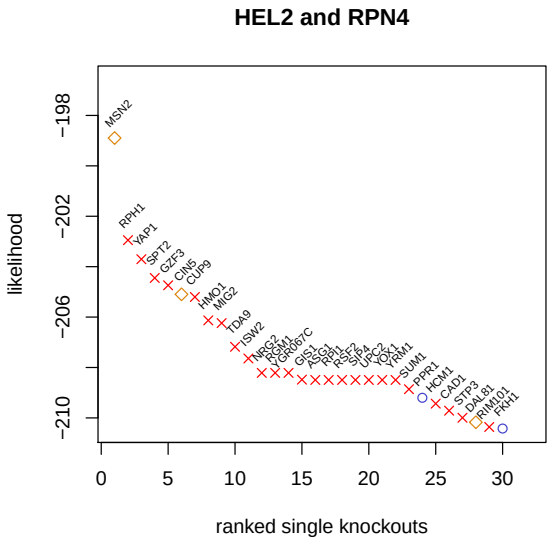

GZF3 and STB5
likelihood
ranked single knockouts
AND
OR
XOR
GZF3 masks STB5
STB5 masks GZF3
no epistasis
no information

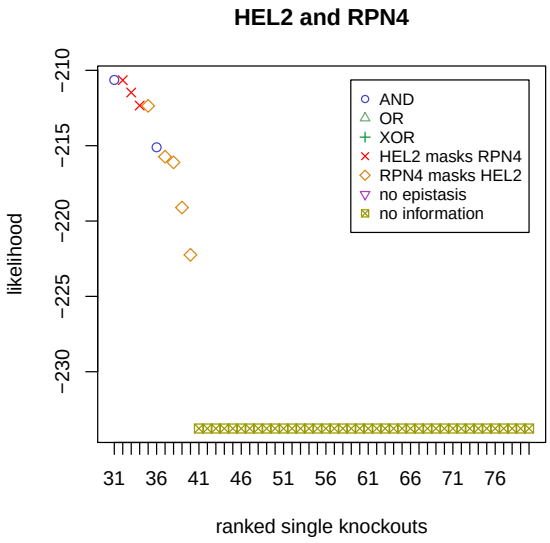

HAC1 and RPN4
likelihood
ranked single knockouts
CST6
STB5
CBF1
GLN3
ZAP1
STP3
DAL81
MSN2
HEL2
ROX1
NRG1
MIG1
FKH1
HMO1
RPH1
SPT2
ASG1
MET31
YAP7
CIN5
PHD1
SUM1
CBF1
FKH2
HAA1
MAL33
PHD2
RGM1
RPI1
TOS8

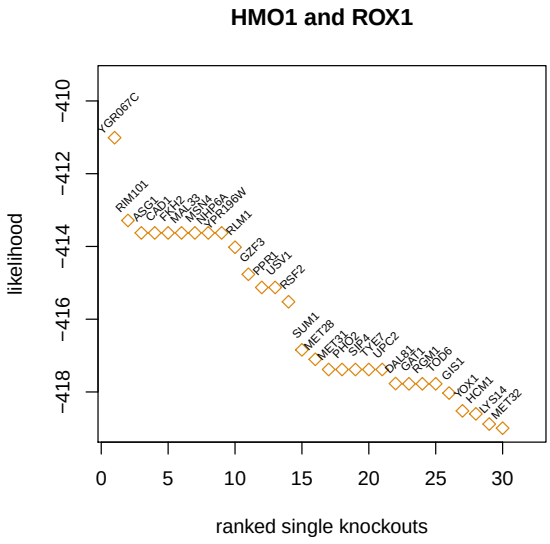

HAC1 and RPN4
likelihood
ranked single knockouts
AND
OR
XOR
HAC1 masks RPN4
RPN4 masks HAC1
no epistasis
no information

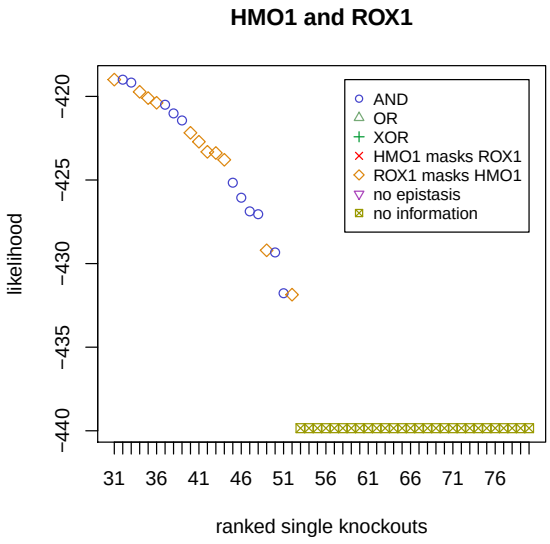

HAC1 and SNT1
likelihood
−135
−140
−145
−150
GLN3
STB5
HEL2
CBF1
MSN2
SPT2
GIS1
YAP1
CST6
RPH1
RPN4
HCM1
RGM1
STP3
SIP4
TDA9
ISW2
RSF2
MIG1
SIT2
FKH1
CAD1
CUP9
MAL33
MET28
MIG3
YGR067C
SUM1
CIN5
RIM1
0
5
10
15
20
25
30
ranked single knockouts


HAC1 and SNT1
likelihood
−155
−160
−165
−170
AND
OR
XOR
HAC1 masks SNT1
SNT1 masks HAC1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

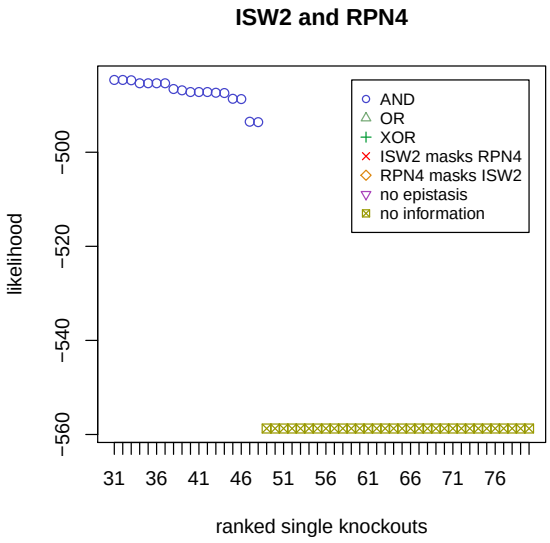

HEL2 and RPN4
likelihood
−198
−202
−206
−210
MSN2
RPH1
YAP1
SPT2
GZF3
CIN5
CUP9
HMO1
MIG2
TDA9
ISW2
NRG2
RGM1
YGR067C
GIS1
ASG1
RPB1
RSF2
SIP4
UPC2
YJL1
YRM1
SUM1
PPR1
HCM1
CAD1
STP3
DAL81
RIM101
FKH1
0
5
10
15
20
25
30
ranked single knockouts

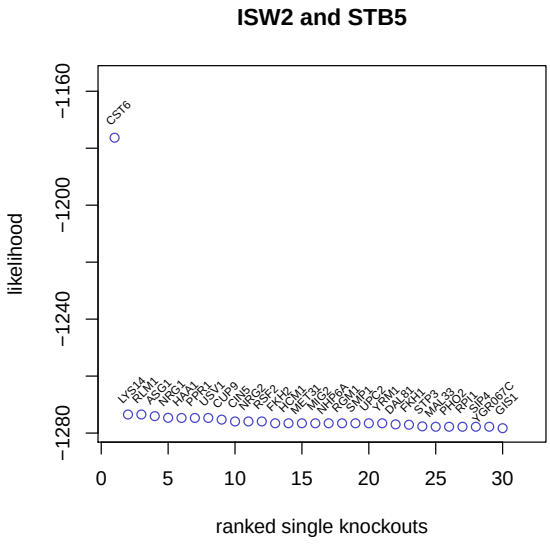

HEL2 and RPN4
likelihood
−210
−215
−220
−225
−230
AND
OR
XOR
HEL2 masks RPN4
RPN4 masks HEL2
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

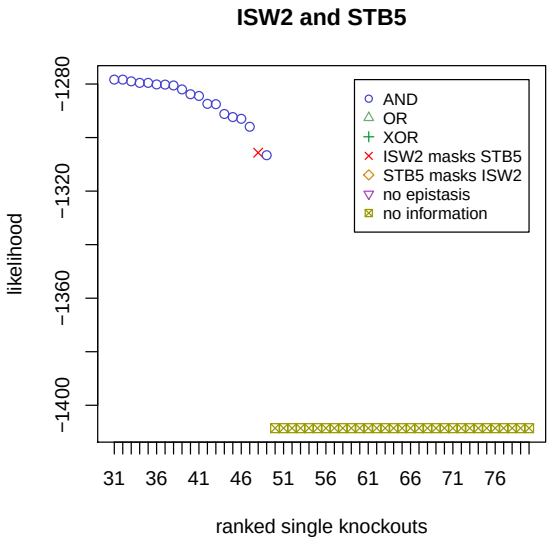

HMO1 and ROX1
likelihood
ranked single knockouts
YGR067C
RIM101
ASG1
CAD1
FKH2
MAC3B
MET4
NRG1
PPR1B6W
RLM1
GZF3
PPT1
USV1
RSF2
SUM1
MET28
MET31
RTG2
SIP4
YJE7
UPC2
DAL81
GAT1
ROX1
TOD6
GIS1
YOX1
HCM1
LYS14
MET32

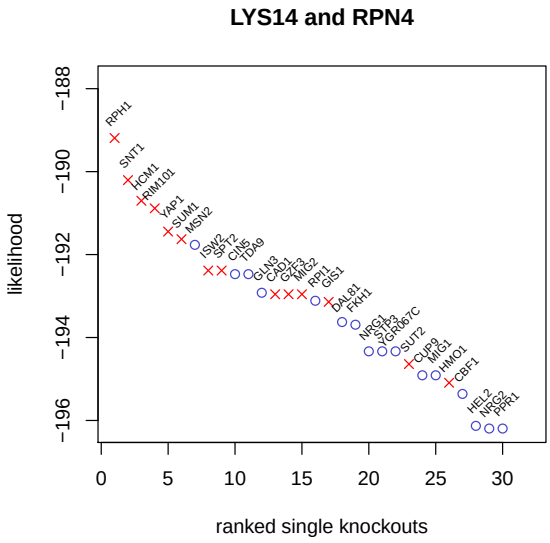

HMO1 and ROX1
likelihood
ranked single knockouts
AND
OR
XOR
HMO1 masks ROX1
ROX1 masks HMO1
no epistasis
no information

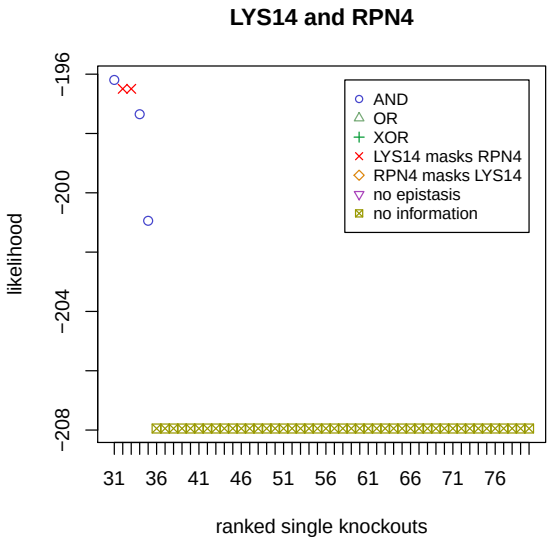

ISW2 and RPN4
likelihood
ranked single knockouts
STB5
GLN3
CBF1
ZAP1
CST6
MSN2
HEL2
DAL81
FKH1
ROX1
CIN5
MIG1
STP3
ASG1
RPN1
SIP4
NRG1
TDA9
YOX1
SUT2
RIM101
CAD1
FKH2
FHL1
MAC3B
MET28
MET31
RLM1
UPC2
HMO1

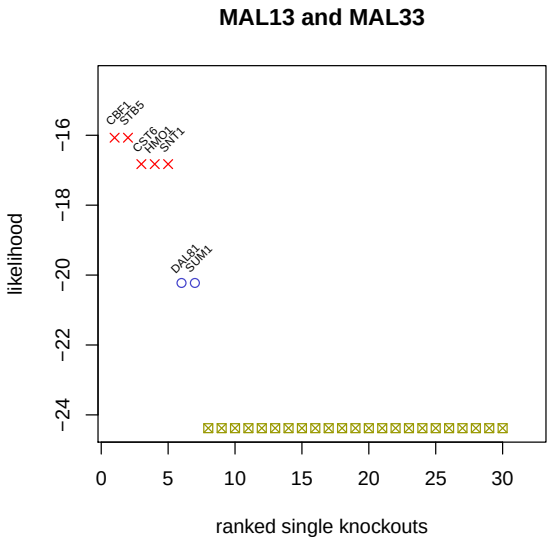

ISW2 and RPN4
likelihood
ranked single knockouts
AND
OR
XOR
ISW2 masks RPN4
RPN4 masks ISW2
no epistasis
no information

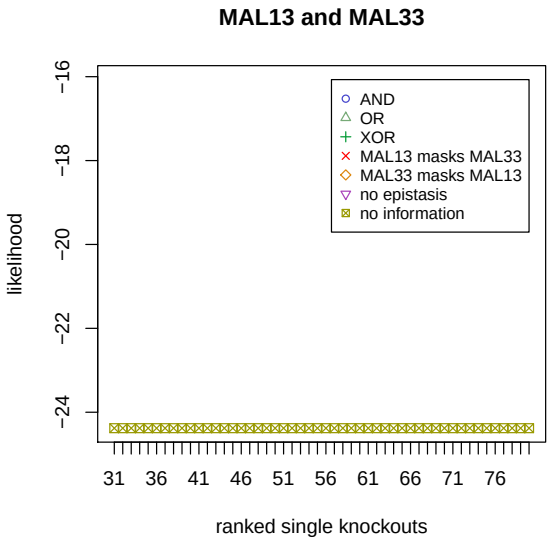

ISW2 and STB5
likelihood
−1160
−1200
−1240
−1280
CST6
LYS14
RLM1
ASG1
NRG1
HAL1
PPR1
USV1
CUP9
CIN5
NRG2
RSF2
FKH2
YCR1
MET31
MCM1
NHP6A
RGM1
SMP1
UGA3
YKR1
DAL81
FKH1
STP3
MAL33
PHD1
RGT1
SIP4
YGR067C
GIS1
0
5
10
15
20
25
30
ranked single knockouts

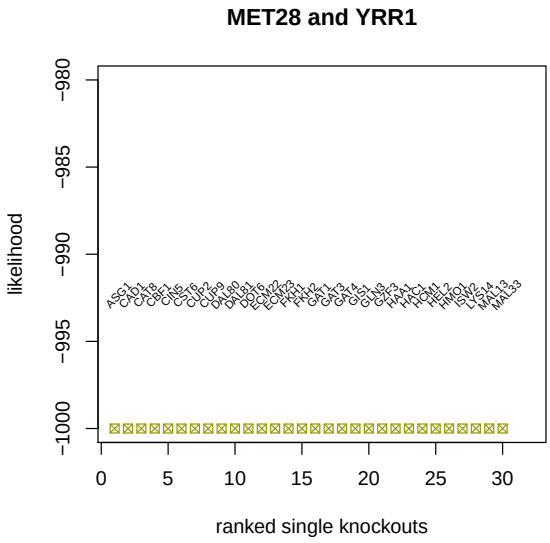

ISW2 and STB5
likelihood
−1280
−1320
−1360
−1400
AND
OR
XOR
ISW2 masks STB5
STB5 masks ISW2
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

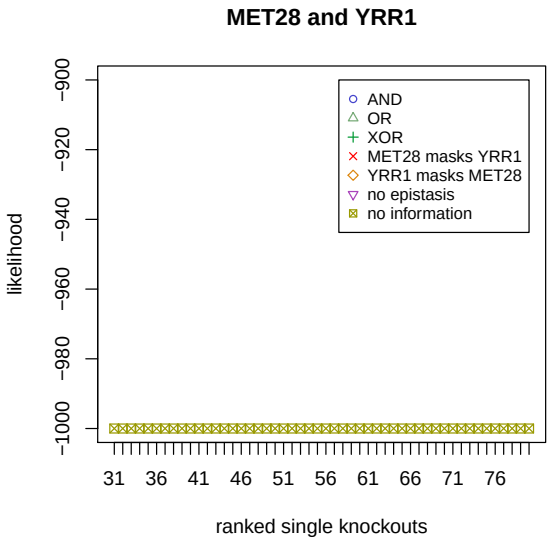

LYS14 and RPN4
likelihood
−188
−190
−192
−194
−196
RPH1
SNT1
HOM1
RIM101
YAP1
SUM1
MSN2
ISW2
SPT2
CIN5
ITDA9
GLN3
CAD1
CST3
MIG2
RPN1
GIS1
DAL81
FKH1
NRG1
STP3
YGR067C
SUT2
CUP9
MIG1
HMO1
CBF1
HEL2
NRG2
PPR1
0
5
10
15
20
25
30
ranked single knockouts

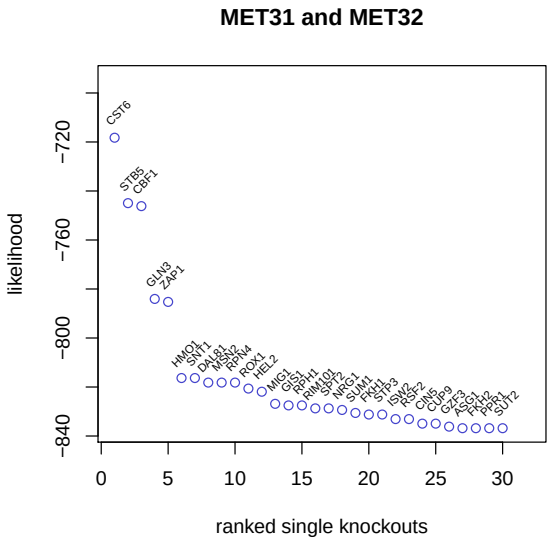

LYS14 and RPN4
likelihood
−196
−200
−204
−208
AND
OR
XOR
LYS14 masks RPN4
RPN4 masks LYS14
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

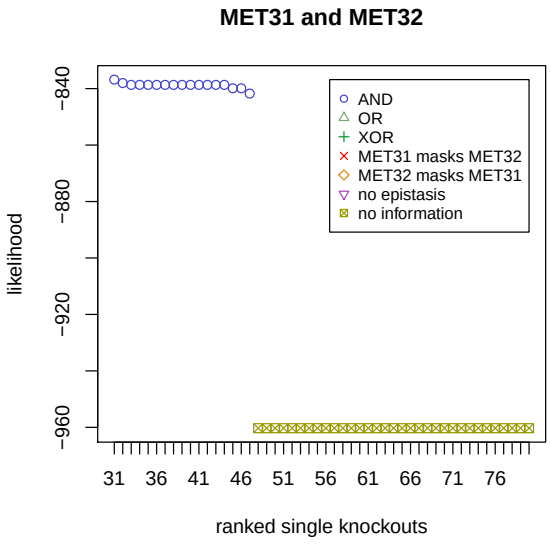

MAL13 and MAL33
likelihood
−16
−18
−20
−22
−24
CRF1
STB5
CST6
TIM10
SWT1
DAL81
SUM1
0
5
10
15
20
25
30
ranked single knockouts

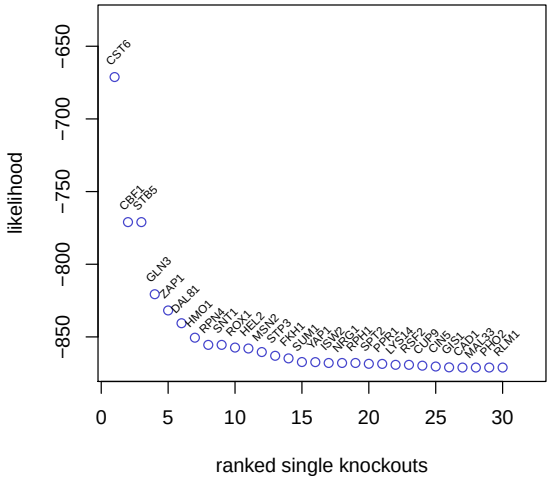

MAL13 and MAL33
likelihood
−16
−18
−20
−22
−24
AND
OR
XOR
MAL13 masks MAL33
MAL33 masks MAL13
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

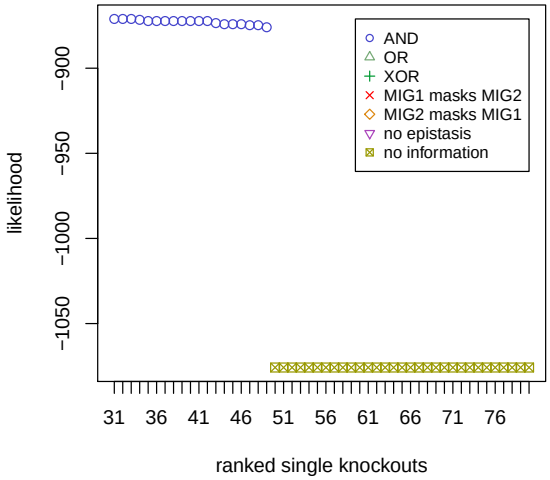

MET28 and YRR1
likelihood
−980
−985
−990
−995
−1000
ASG1
CAD1
CAT8
CBF1
CIN5
CST6
CUP2
CUP9
DAL80
DAL81
DOT6
ECM22
ECM23
FKH1
FKH2
GAT1
GAT3
GAT4
GIS1
GLN3
GZF3
HAA1
HAC1
HCM1
HEL2
HIR1
HIR2
ISW2
LYS14
MAL13
MAL33
0
5
10
15
20
25
30
ranked single knockouts

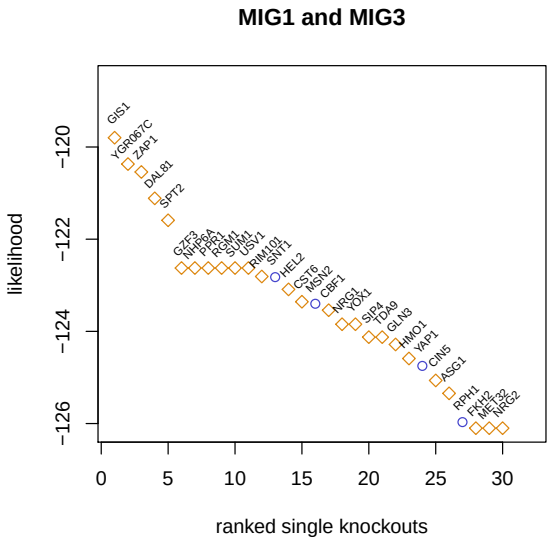

MET28 and YRR1
likelihood
−900
−920
−940
−960
−980
−1000
AND
OR
XOR
MET28 masks YRR1
YRR1 masks MET28
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

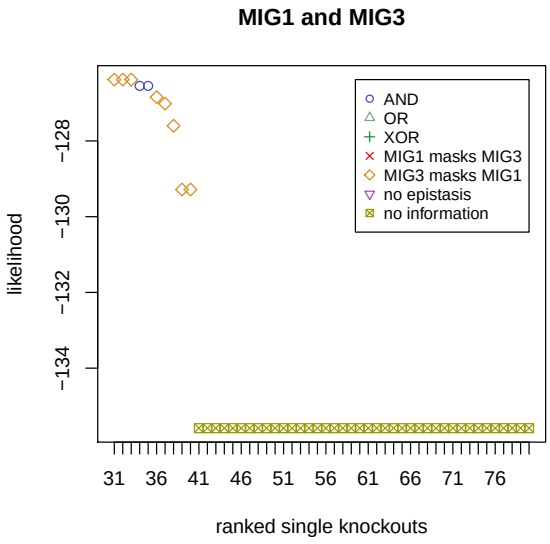

MET31 and MET32
likelihood
−720
−760
−800
−840
0
5
10
15
20
25
30
ranked single knockouts
CST6
STB5
CBF1
GLN3
ZAP1
HMO1
SNT1
DAL81
MSN2
RPN4
ROX1
HEL2
MIG1
GIS1
RPH1
RIM101
NRG1
SUM1
FKH1
STP3
ISW2
REF2
CIN5
CUP9
GAT3
ACS1
FHL1
PPR1
SUT2

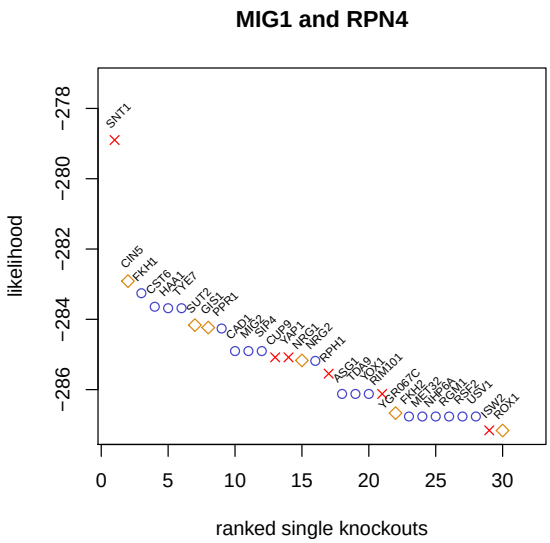

MET31 and MET32
likelihood
−840
−880
−920
−960
31
36
41
46
51
56
61
66
71
76
ranked single knockouts
AND
OR
XOR
MET31 masks MET32
MET32 masks MET31
no epistasis
no information

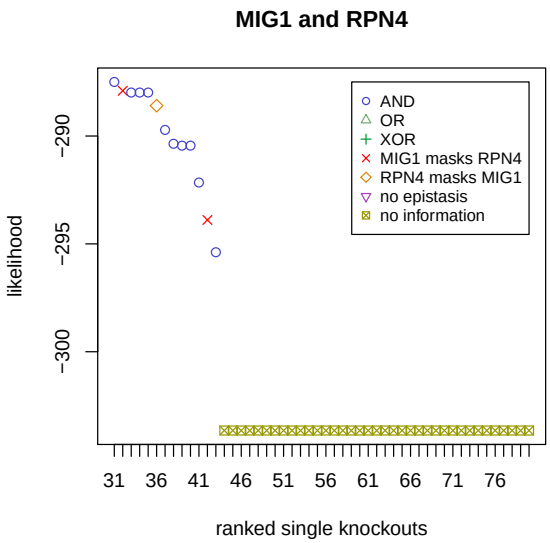

MIG1 and MIG2
likelihood
−650
−700
−750
−800
−850
0
5
10
15
20
25
30
ranked single knockouts
CST6
CBF1
STB5
GLN3
ZAP1
DAL81
HMO1
RPN4
SNT1
ROX1
HEL2
MSN2
STP3
FKH1
SUM1
YAP1
SWI5
NRG1
RPH1
SFL1
PPR1
LYS14
REF2
CUP9
CIN5
GIS1
CAD1
MAL33
PHD1
RLM1

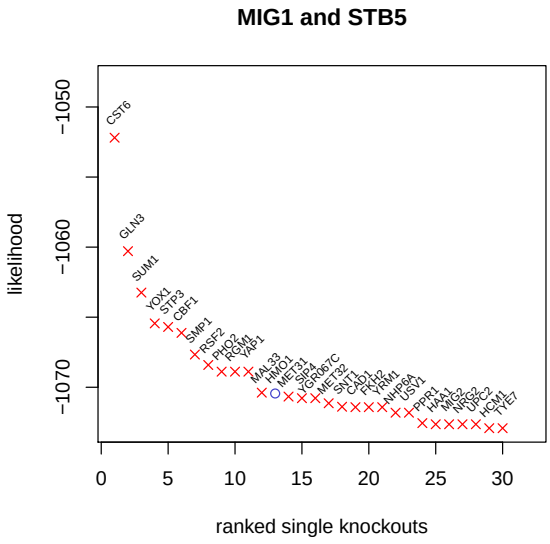

MIG1 and MIG2
likelihood
−900
−950
−1000
−1050
31
36
41
46
51
56
61
66
71
76
ranked single knockouts
AND
OR
XOR
MIG1 masks MIG2
MIG2 masks MIG1
no epistasis
no information

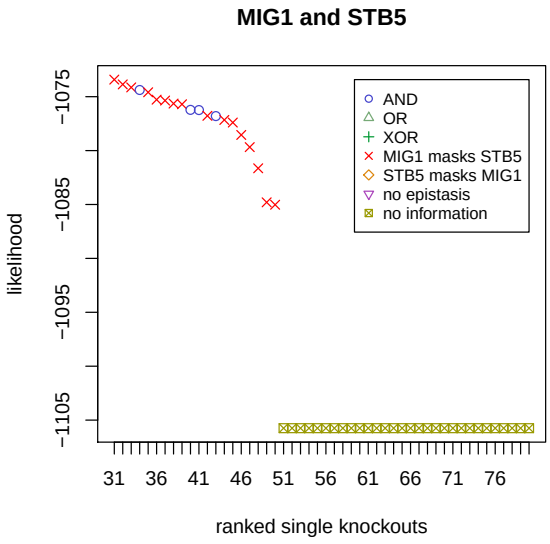

MIG1 and MIG3
likelihood
−120
−122
−124
−126
GIS1
YGR067C
ZAP1
DAL81
SPT2
GZF3
NHP6A
PPR1
ROX1
SUM1
USV1
RIM101
SNT1
HEL2
CST6
MSN2
CBF1
NRG1
YOX1
SIP4
TDA9
GLN3
HMO1
YAP1
CIN5
ASG1
RPH1
FKH2
MET32
NRG2
0
5
10
15
20
25
30
ranked single knockouts

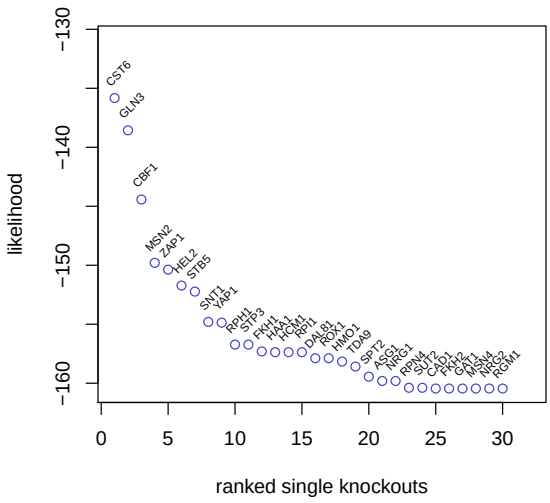

MIG1 and MIG3
likelihood
−128
−130
−132
−134
AND
OR
XOR
MIG1 masks MIG3
MIG3 masks MIG1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

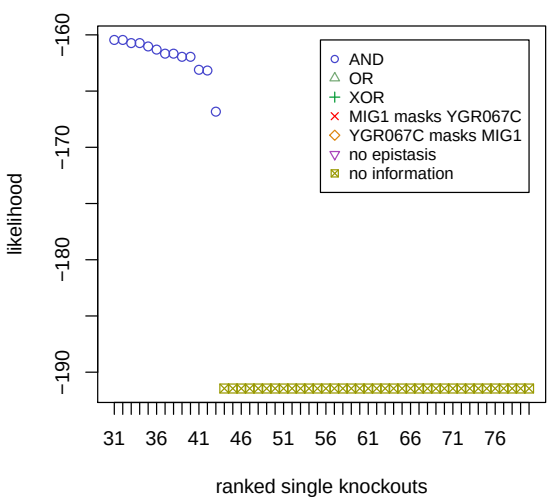

MIG1 and RPN4
likelihood
−278
−280
−282
−284
−286
SNT1
CIN5
FKH1
CST6
HAA1
TYE7
SUT2
GIS1
PPR1
CAD1
MIG2
STP4
CUP9
YAP1
NRG1
NRG2
RPH1
ASG1
TDA9
YOX1
RIM101
YGR067C
FKH2
MET32
NRG1
RGM1
RSF2
USV1
ISW2
ROX1
0
5
10
15
20
25
30
ranked single knockouts

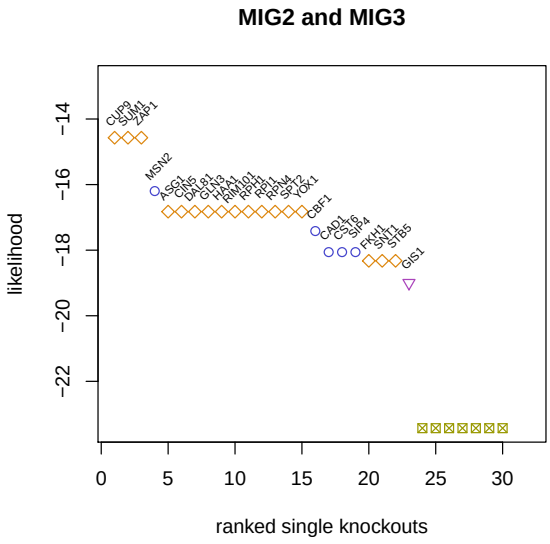

MIG1 and RPN4
likelihood
−290
−295
−300
AND
OR
XOR
MIG1 masks RPN4
RPN4 masks MIG1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

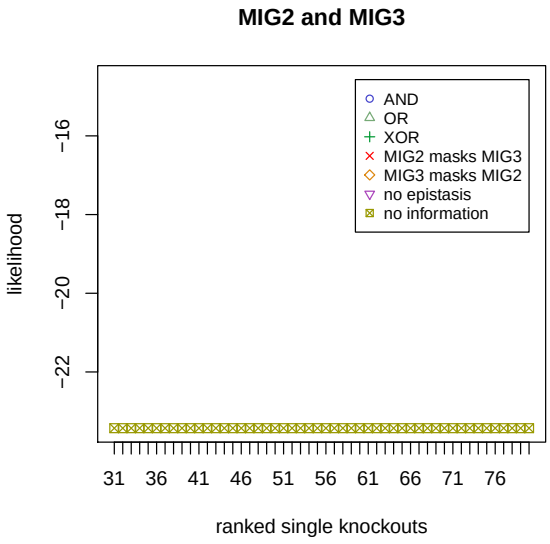

MIG1 and STB5
likelihood
−1050
−1060
−1070
CST6
GLN3
SUM1
YOX1
STP3
CBF1
SMP1
RSP2
PHO2
RGM1
YAP1
MAL33
RMD1
MET31
SIP4
YGR067C
MET32
SNT1
CAD1
RTG2
YRM1
ANP54
USV1
PPR1
HAL1
MIG2
NRG2
UPC2
HCM1
TYE7
0
5
10
15
20
25
30
ranked single knockouts

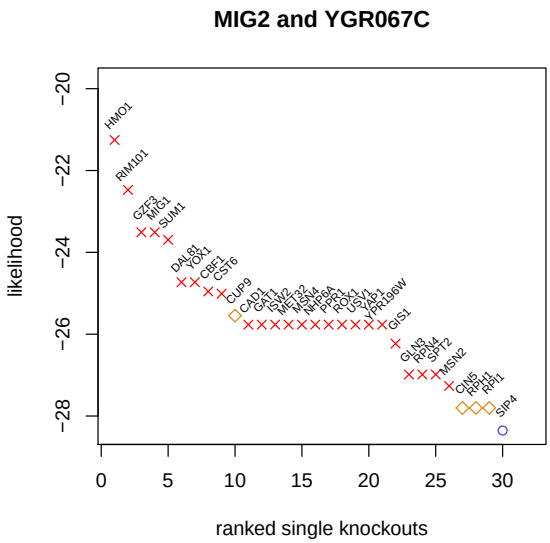

MIG1 and STB5
likelihood
−1075
−1085
−1095
−1105
AND
OR
XOR
MIG1 masks STB5
STB5 masks MIG1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

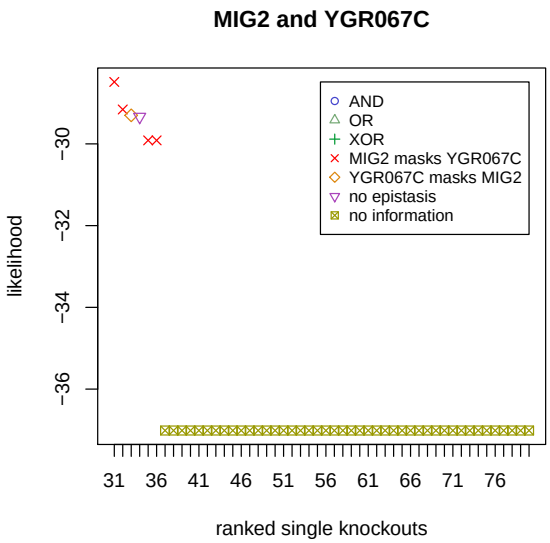

MIG1 and YGR067C
likelihood
−130
−140
−150
−160
CST6
GLN3
CBF1
MSN2
ZAP1
HEL2
STB5
SNT1
YAP1
RPH1
STP3
FKH1
HAL1
HCM1
RPF1
DAL81
ROX1
HMO1
TDA9
SPT2
ASG1
NRG1
RPN4
SUT2
CAD1
RCH1
GAT1
MSN4
NRG2
RGM1
0
5
10
15
20
25
30
ranked single knockouts

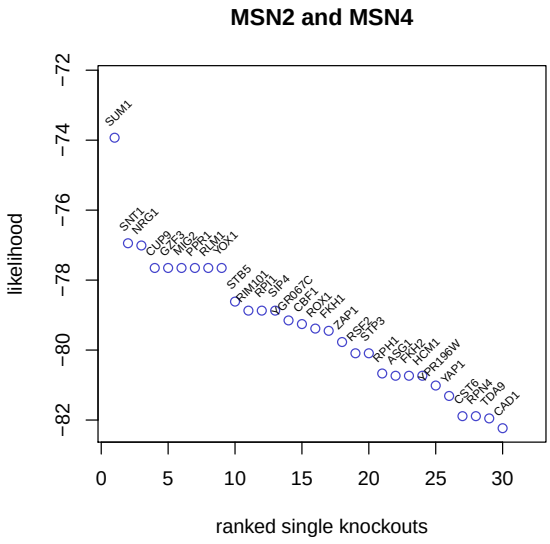

MIG1 and YGR067C
likelihood
−160
−170
−180
−190
AND
OR
XOR
MIG1 masks YGR067C
YGR067C masks MIG1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

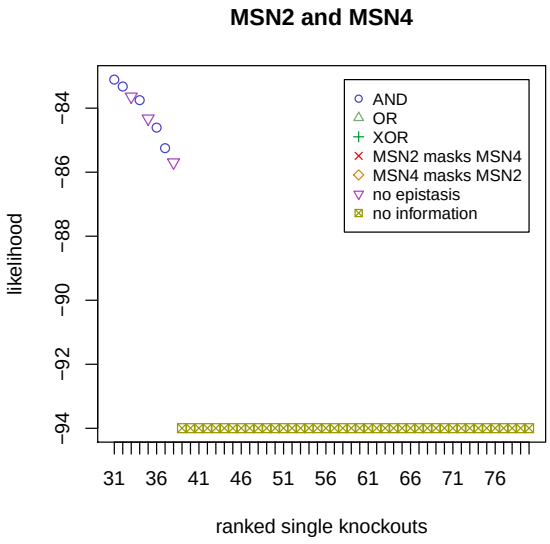

MIG2 and MIG3
likelihood
ranked single knockouts
CUP9
SUM1
ZAP1
MSN2
ASG1
CIN5
DAL81
GLN3
HAL1
RIM101
RPB1
RPB4
SPT2
YOX1
CBF1
CAD1
CST6
SIP4
RXT1
SNT1
STB5
GIS1

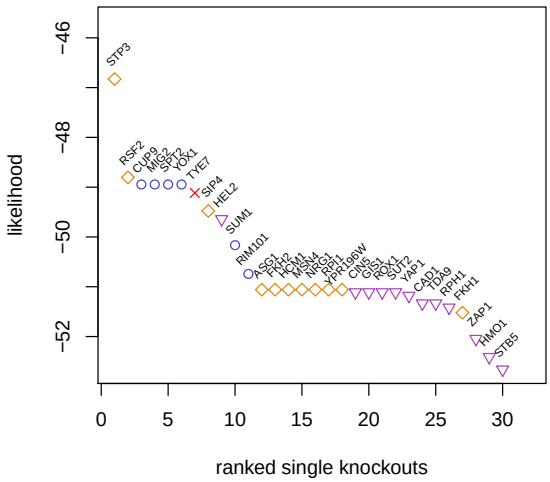

MIG2 and MIG3
likelihood
ranked single knockouts
AND
OR
XOR
MIG2 masks MIG3
MIG3 masks MIG2
no epistasis
no information

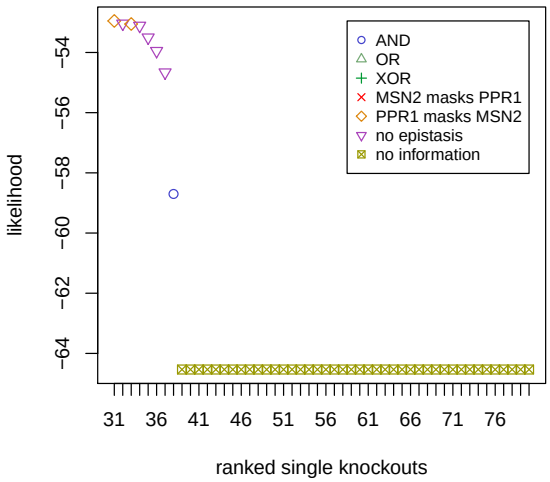

MIG2 and YGR067C
likelihood
ranked single knockouts
HMO1
RIM101
GZF3
MIG1
SUM1
DAL81
YOX1
CBF1
CST6
CUP9
CAD1
GAT1
SWI2
MET31
NRG1
NPR2A
PPR1
ROX1
UGA3
YAP1
YPR196W
GIS1
GLN3
RPB4
SPT2
MSN2
CIN5
RPB1
RPB1
SIP4

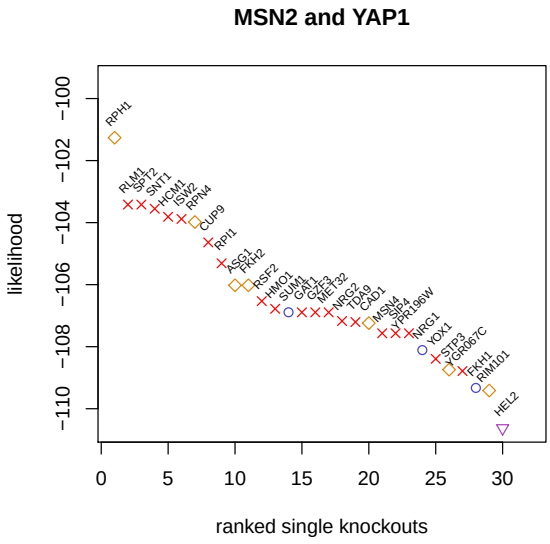

MIG2 and YGR067C
likelihood
ranked single knockouts
AND
OR
XOR
MIG2 masks YGR067C
YGR067C masks MIG2
no epistasis
no information

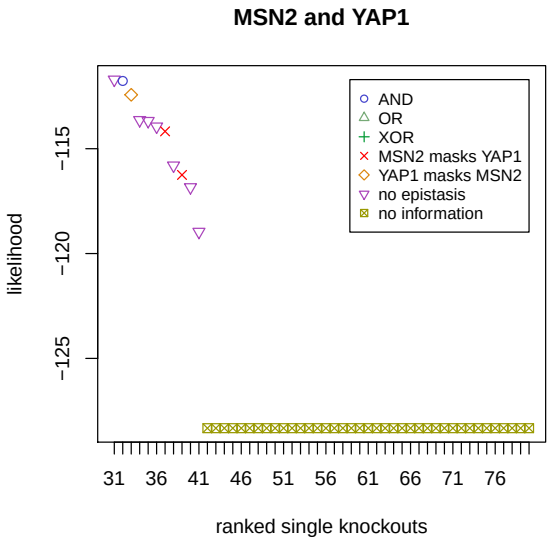

MSN2 and MSN4
likelihood
ranked single knockouts
SUM1
SNT1
NRG1
CUP9
GZF3
MIG2
PPR1
RLM1
YOX1
STB5
RIM101
RPI1
SIP4
YOR067C
CBF1
ROX1
FKH1
ZAP1
RSF2
STP3
RPH1
ASG1
FKH2
HOM1
YPR196W
YAP1
CST6
RPN4
TDA9
CAD1

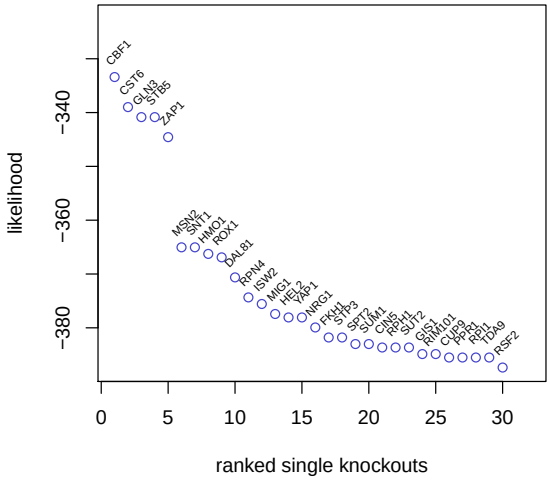

MSN2 and MSN4
likelihood
ranked single knockouts
AND
OR
XOR
MSN2 masks MSN4
MSN4 masks MSN2
no epistasis
no information

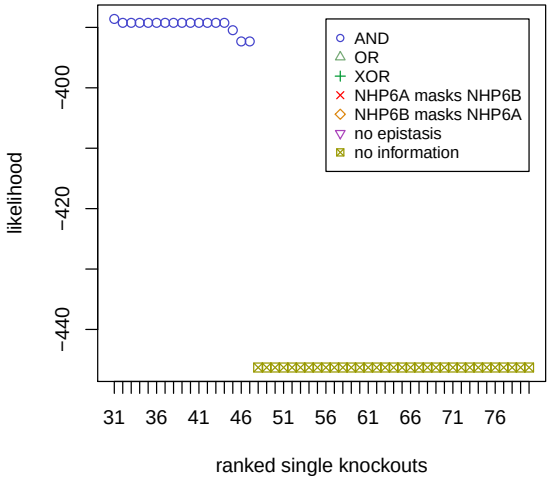

MSN2 and PPR1
likelihood
ranked single knockouts
STP3
RSF2
CUP9
MIG2
SFL2
YOX1
TYE7
SIP4
HEL2
SUM1
RIM101
ASG1
FKH2
FKH1
MSN4
NRG1
RPI1
YPR196W
CIN5
GIS1
ROX1
SUT2
YAP1
CAD1
TDA9
RPH1
FKH1
ZAP1
HMO1
STB5

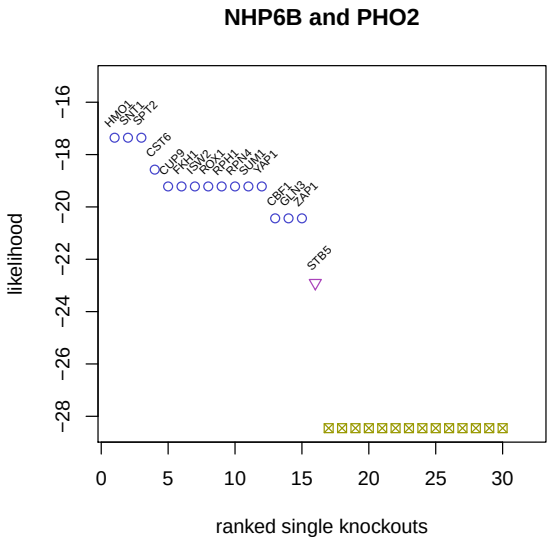

MSN2 and PPR1
likelihood
ranked single knockouts
AND
OR
XOR
MSN2 masks PPR1
PPR1 masks MSN2
no epistasis
no information

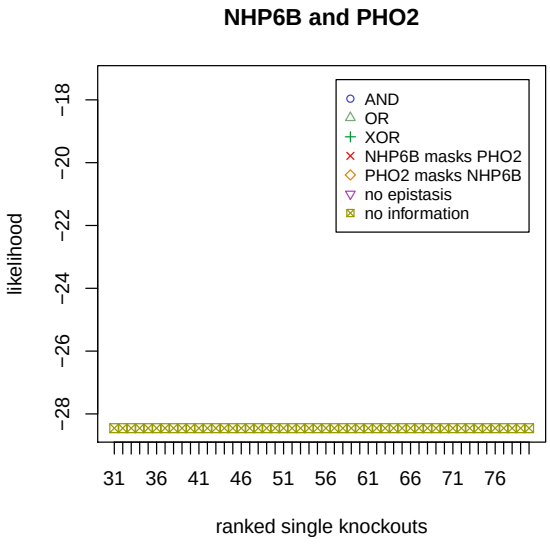

MSN2 and YAP1
likelihood
−100
−102
−104
−106
−108
−110
RPH1
RLM1
SPT2
SNT1
HCM1
ISW2
RPN4
CUP9
RPI1
ASC1
FKH2
RSF2
HMO1
SUM1
CAT1
SZF1
MET32
NRG2
TDA9
CAD1
MSN4
SFP1
YPR196W
NRG1
YOX1
STP3
YGR067C
FKH1
RIM101
HEL2
0
5
10
15
20
25
30
ranked single knockouts

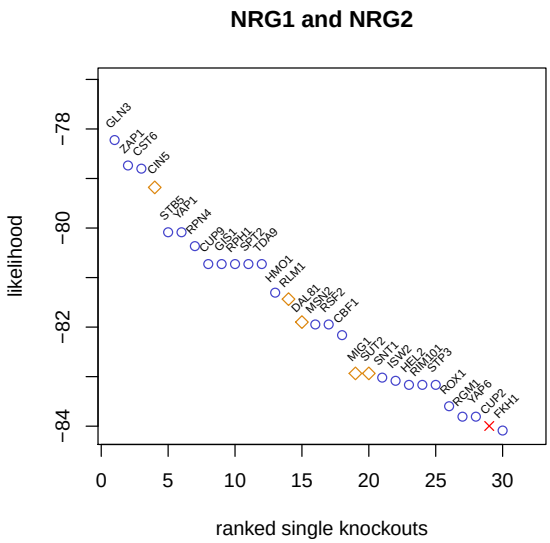

MSN2 and YAP1
likelihood
−115
−120
−125
AND
OR
XOR
MSN2 masks YAP1
YAP1 masks MSN2
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

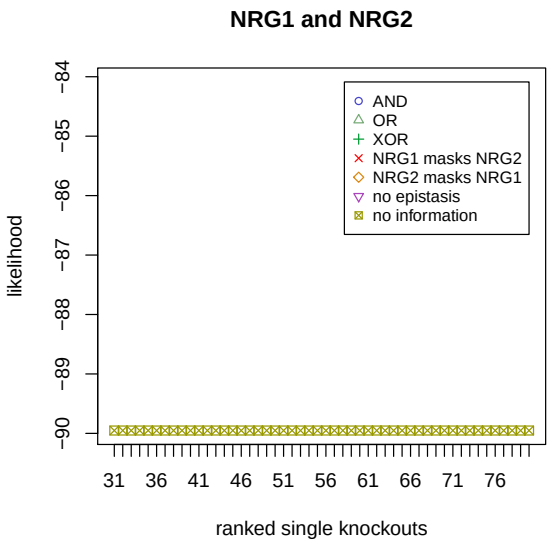

NHP6A and NHP6B
likelihood
−340
−360
−380
CBF1
CST6
GLN3
STB5
ZAP1
MSN2
SNT1
HMO1
ROX1
DAL81
RPN4
ISW2
MIG1
HEL2
YAP1
NRG1
FKH1
STP3
SPT2
SUM1
CIN5
RPH1
SUT2
GIS1
RIM101
CUP9
PPR1
RBS1
TDA9
RSF2
0
5
10
15
20
25
30
ranked single knockouts

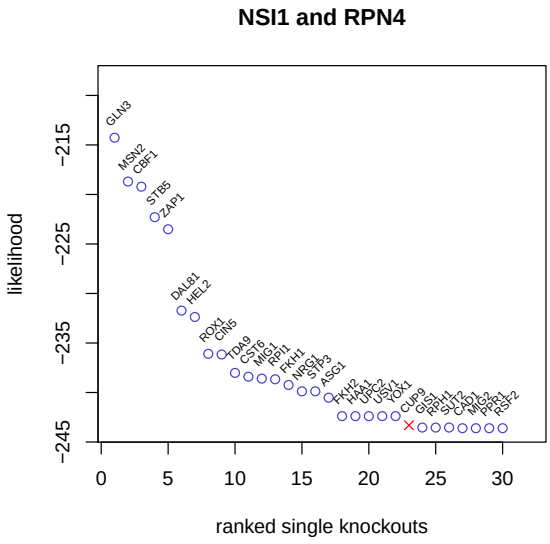

NHP6A and NHP6B
likelihood
−400
−420
−440
AND
OR
XOR
NHP6A masks NHP6B
NHP6B masks NHP6A
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

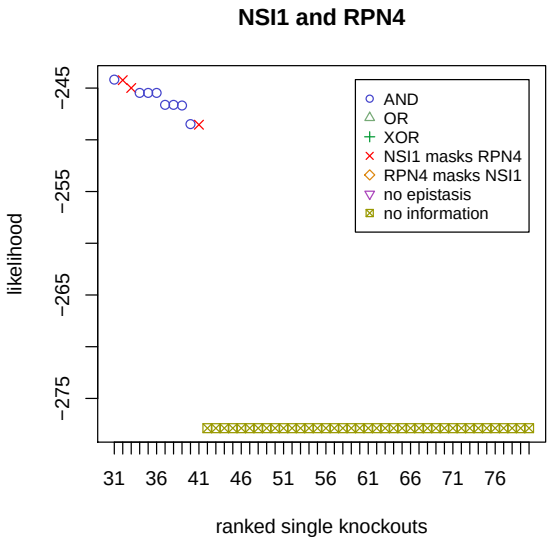

NHP6B and PHO2
likelihood
ranked single knockouts
HMO1
SNT1
SPT2
CST6
CUP9
FKH1
ISW2
ROX1
RPN4
SUM1
YAP1
CBF1
GLN3
ZAP1
STB5

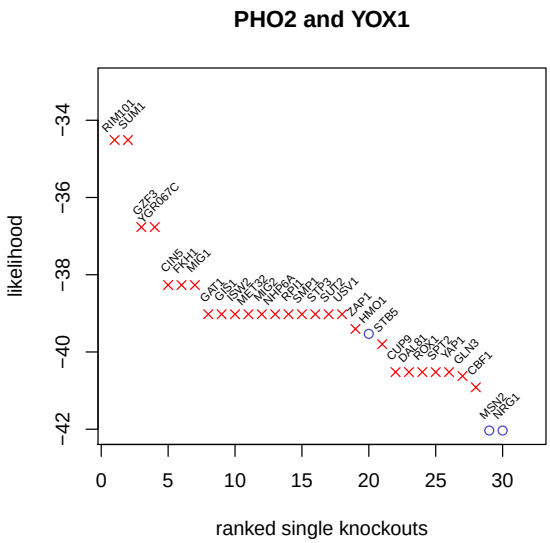

NHP6B and PHO2
likelihood
ranked single knockouts
AND
OR
XOR
NHP6B masks PHO2
PHO2 masks NHP6B
no epistasis
no information

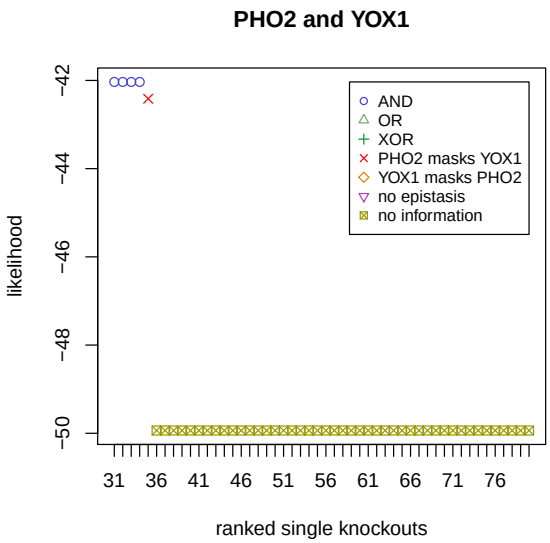

NRG1 and NRG2
likelihood
ranked single knockouts
GLN3
ZAP1
CST6
CIN5
STB5
YAP1
RPN4
CUP9
GIS1
RPH1
SPT2
TDA9
HMC1
RLM1
DAL81
MSN2
RSF2
CBF1
MIG1
SUT2
SNT1
ISW2
HIR2
RIM101
STP3
ROX1
RGM1
YAP6
CUP2
FKH1

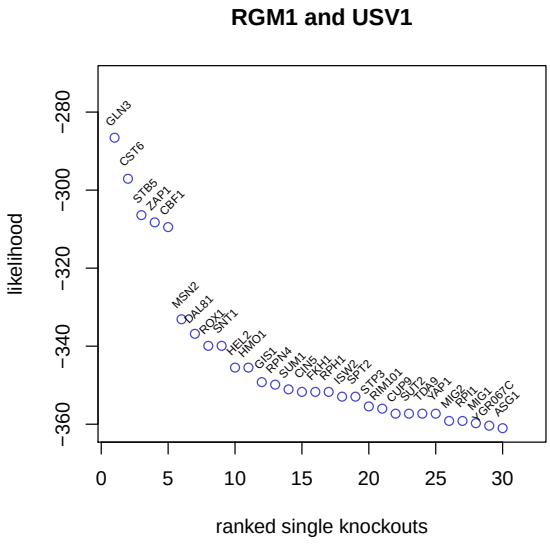

NRG1 and NRG2
likelihood
ranked single knockouts
AND
OR
XOR
NRG1 masks NRG2
NRG2 masks NRG1
no epistasis
no information

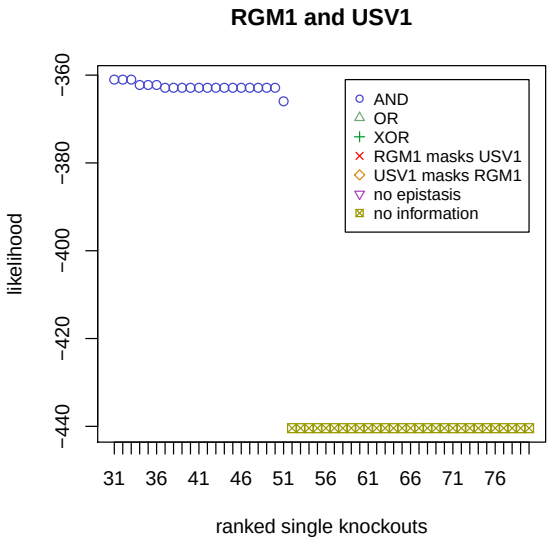

NSI1 and RPN4
likelihood
ranked single knockouts

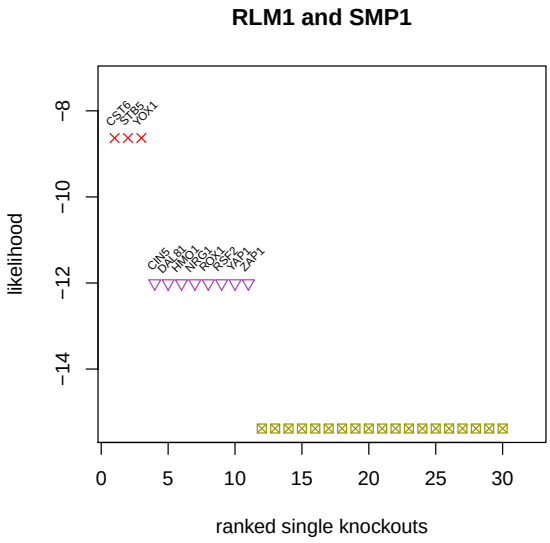

NSI1 and RPN4
AND
OR
XOR
NSI1 masks RPN4
RPN4 masks NSI1
no epistasis
no information
likelihood
ranked single knockouts

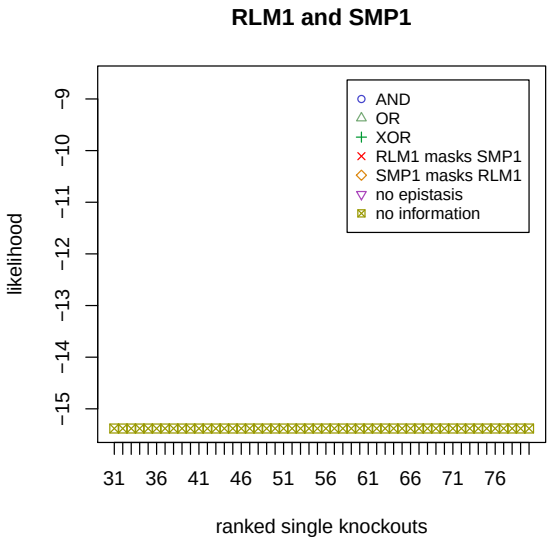

PHO2 and YOX1
likelihood
ranked single knockouts

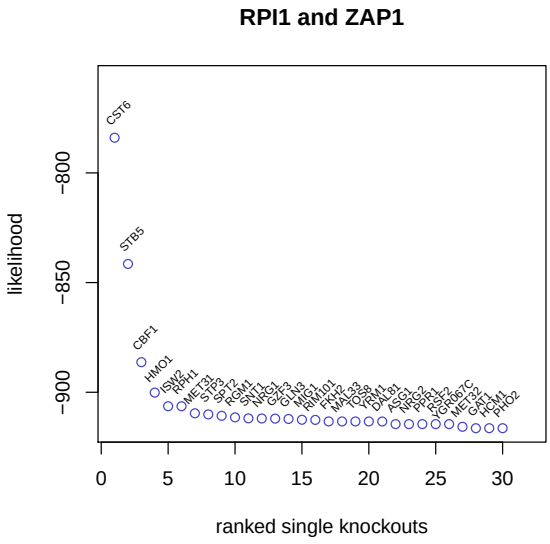

PHO2 and YOX1
AND
OR
XOR
PHO2 masks YOX1
YOX1 masks PHO2
no epistasis
no information
likelihood
ranked single knockouts

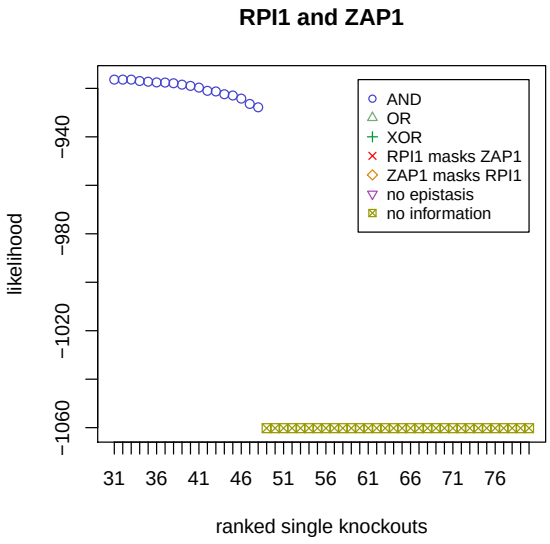

RGM1 and USV1
likelihood
ranked single knockouts
GLN3
CST6
STB5
ZAP1
CBF1
MSN2
DAL81
ROX1
SWI1
HEL2
HMO1
GIS1
RPN4
SUM1
CIN5
NRG1
RPH1
ISW2
SPT2
STP3
RIM101
CUP9
SKN7
TDA9
YAP1
MIG2
RFI1
MIG1
YGR067C
ASG1

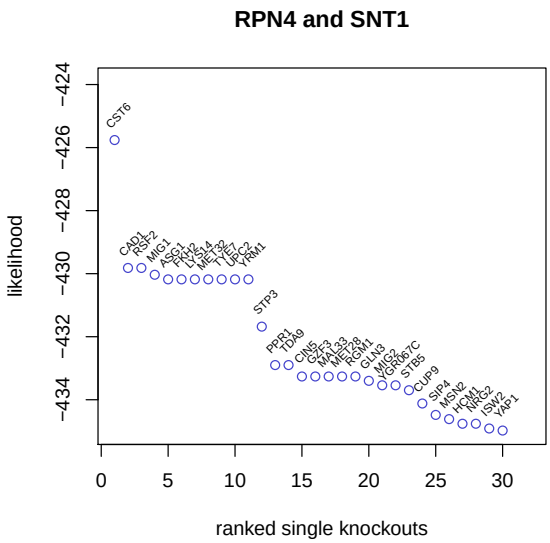

RGM1 and USV1
likelihood
ranked single knockouts
AND
OR
XOR
RGM1 masks USV1
USV1 masks RGM1
no epistasis
no information

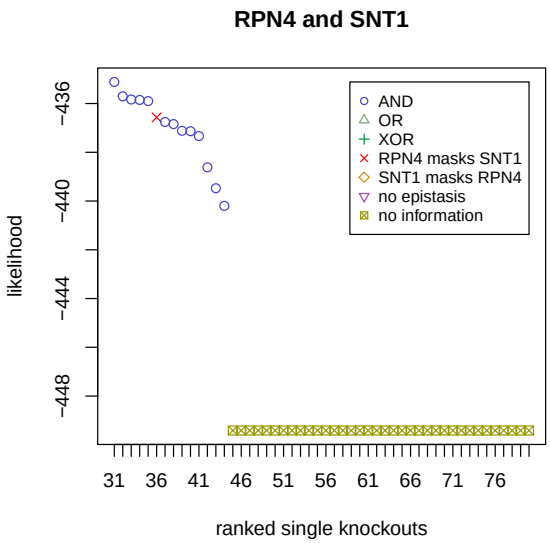

RLM1 and SMP1
likelihood
ranked single knockouts
CST6
STB5
YOX1
CIN5
DAL81
HMO1
NRG1
ROX1
RSF2
VAS1
ZAP1

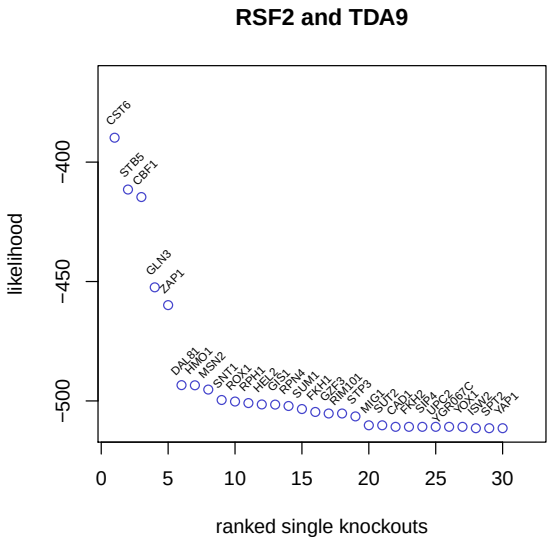

RLM1 and SMP1
likelihood
ranked single knockouts
AND
OR
XOR
RLM1 masks SMP1
SMP1 masks RLM1
no epistasis
no information

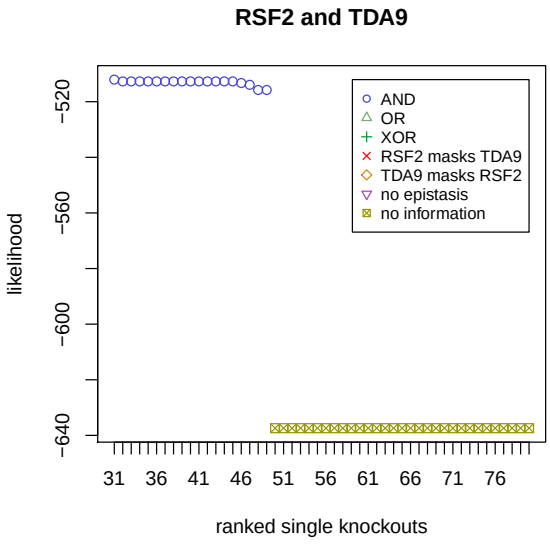

RPI1 and ZAP1
likelihood
−800
−850
−900
CST6
STB5
CBF1
HMO1
ISW2
RPH1
MET31
STP2
RGM1
SNT1
NRG1
GLN3
MIG2
RMI101
ASG1
MET28
TOS8
YRM1
DAL81
ASG1
NRG2
PPR1
RSF2
YGR067C
MET31
GAT1
HCM1
PHD2
0
5
10
15
20
25
30
ranked single knockouts

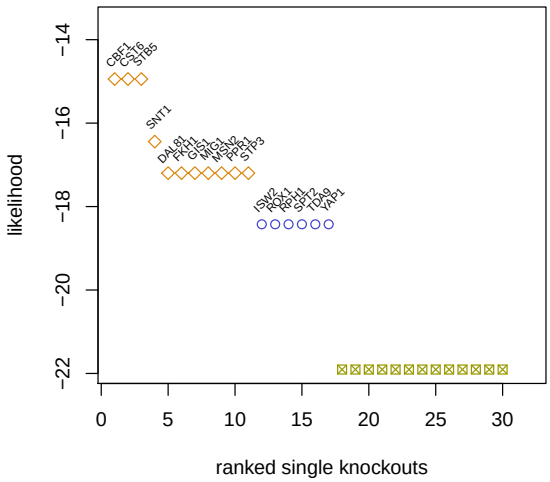

RPI1 and ZAP1
likelihood
−940
−980
−1020
−1060
AND
OR
XOR
RPI1 masks ZAP1
ZAP1 masks RPI1
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

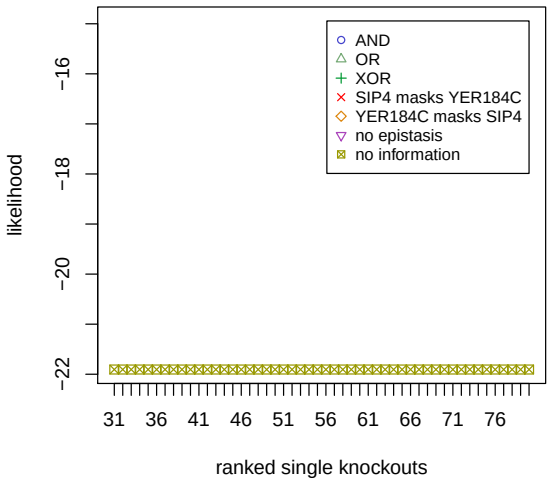

RPN4 and SNT1
likelihood
−424
−426
−428
−430
−432
−434
CST6
CAD1
RSF2
MIG1
ASG1
FKH2
LYS14
MET32
TEC1
UBC2
YRM1
STP3
PPR1
TDA9
CIN5
GZF3
MAL33
MET28
RGM1
GLN3
MIG2
YGR067C
STB5
CUP9
SIP4
MSN2
HCM1
NRG2
ISW2
YAP1
0
5
10
15
20
25
30
ranked single knockouts

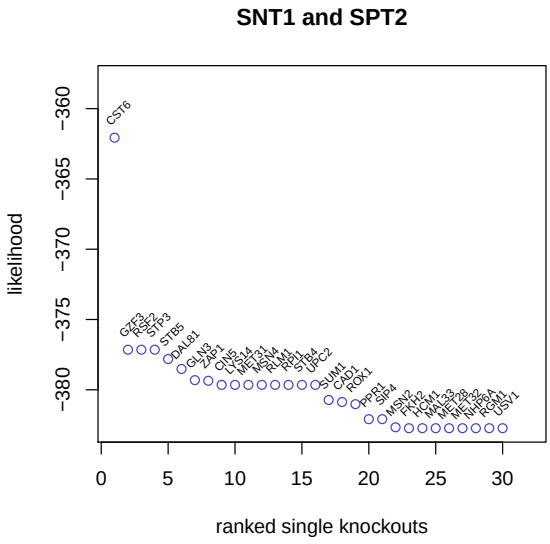

RPN4 and SNT1
likelihood
−436
−440
−444
−448
AND
OR
XOR
RPN4 masks SNT1
SNT1 masks RPN4
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

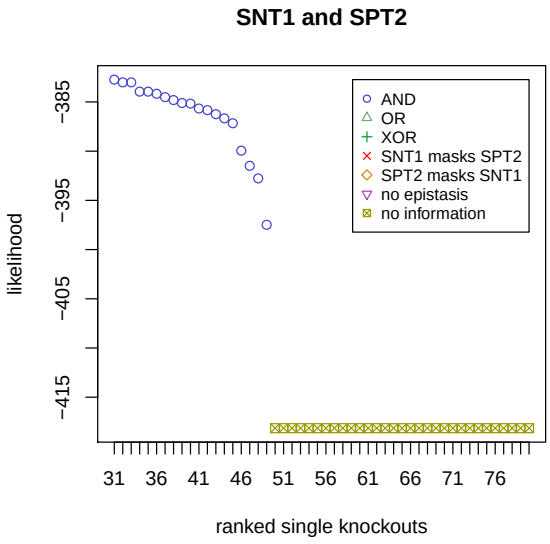

RSF2 and TDA9
likelihood
−400
−450
−500
CST6
STB5
CBF1
GLN3
ZAP1
DAL81
HMO1
MSN2
SNT1
ROX1
RPH1
HEF3
GIS1
RPN4
SUM1
FKH1
GZF3
RIM101
STP3
MIG1
SUT2
CAD1
PKH2
SIP4
YGR067C
YOX1
ISW2
SKO1
YAP1
0
5
10
15
20
25
30
ranked single knockouts

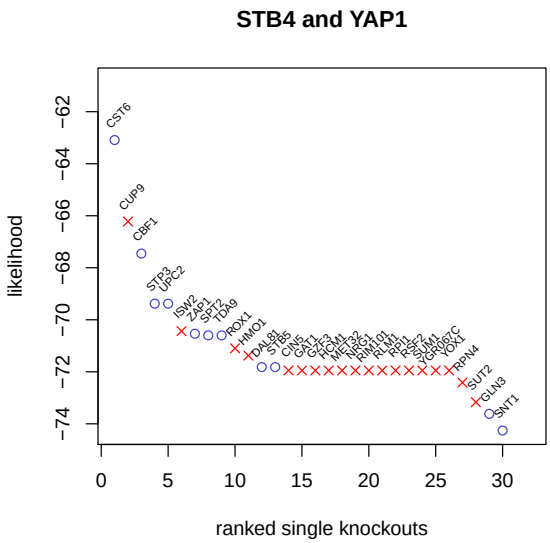

RSF2 and TDA9
likelihood
−520
−560
−600
−640
AND
OR
XOR
RSF2 masks TDA9
TDA9 masks RSF2
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

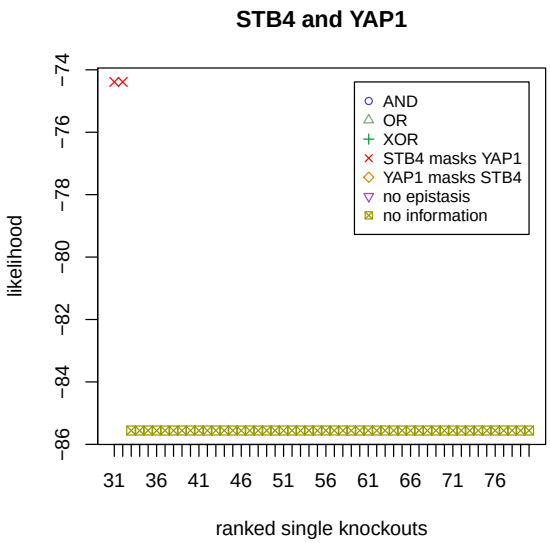

SIP4 and YER184C
likelihood
−14
−16
−18
−20
−22
CRF1
CST6
STB5
SNT1
DAL81
FKH1
GIS1
MIG1
MSN2
MPP1
STP3
ISW2
ROX1
RPH1
SFP1
TDA9
YAP1
0
5
10
15
20
25
30
ranked single knockouts

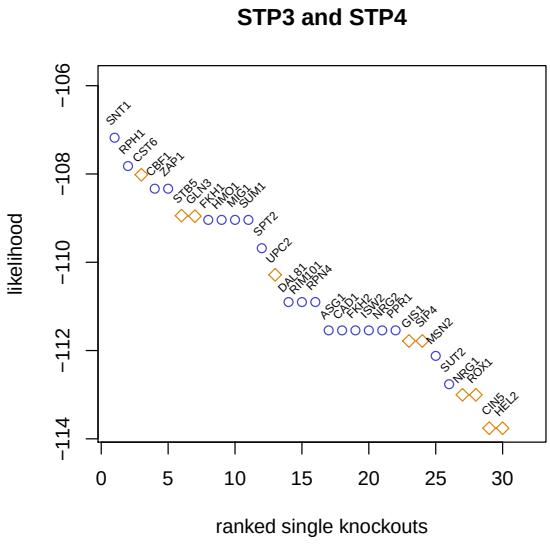

SIP4 and YER184C
likelihood
−16
−18
−20
−22
AND
OR
XOR
SIP4 masks YER184C
YER184C masks SIP4
no epistasis
no information
31
36
41
46
51
56
61
66
71
76
ranked single knockouts

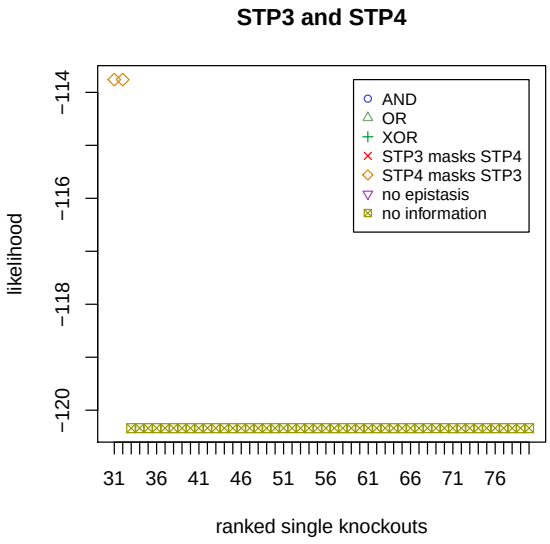

SNT1 and SPT2
likelihood
ranked single knockouts
CST6
GZF3
RGT2
STP3
STB5
DAL81
GLN3
ZAP1
CIN5
LYS14
MET31
MSN4
RLM1
RPI1
STB4
UPC2
SUM1
CAD1
ROX1
PPR1
SIF4
MSN2
FKH2
HCM1
MAL33
MET28
MET32
NRG1
RGM1
USV1

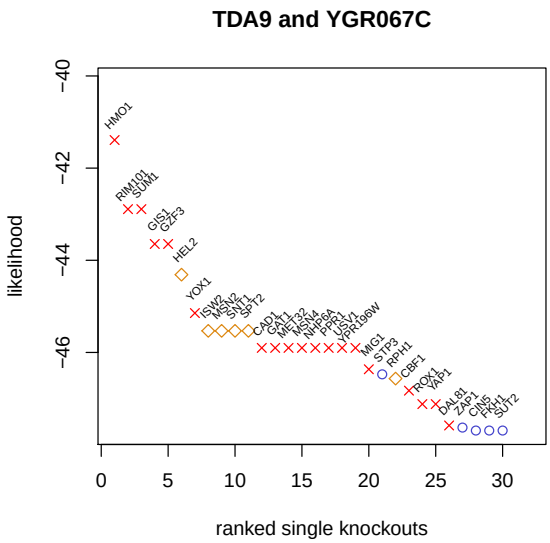

SNT1 and SPT2
likelihood
ranked single knockouts
AND
OR
XOR
SNT1 masks SPT2
SPT2 masks SNT1
no epistasis
no information

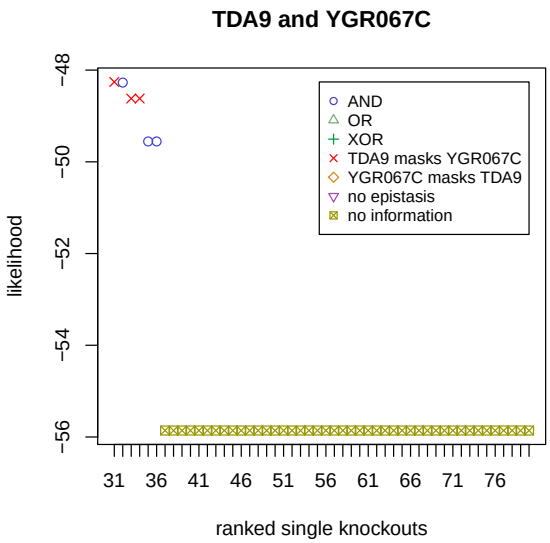

STB4 and YAP1
likelihood
ranked single knockouts
CST6
CUP9
CBF1
STP3
UPC2
ISW2
ZAP1
SPT2
TDA9
ROX1
HMO1
DAL81
STB5
CIN5
CAT1
HCM1
MET32
NRG1
RLM1
RPI1
FKH2
SUM1
GRF10
YOX1
RPN4
SUT2
GLN3
SNT1

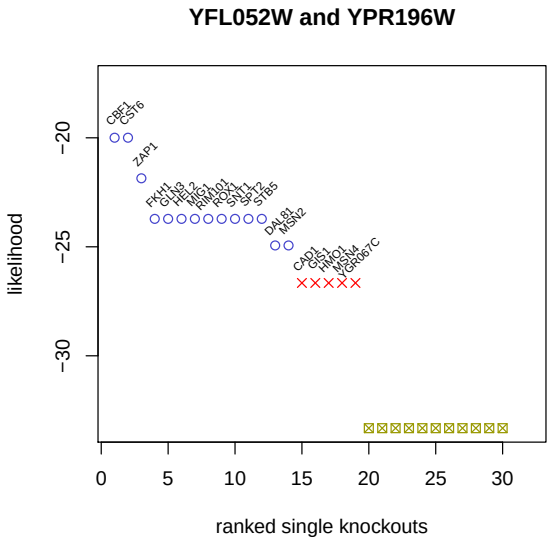

STB4 and YAP1
likelihood
ranked single knockouts
AND
OR
XOR
STB4 masks YAP1
YAP1 masks STB4
no epistasis
no information

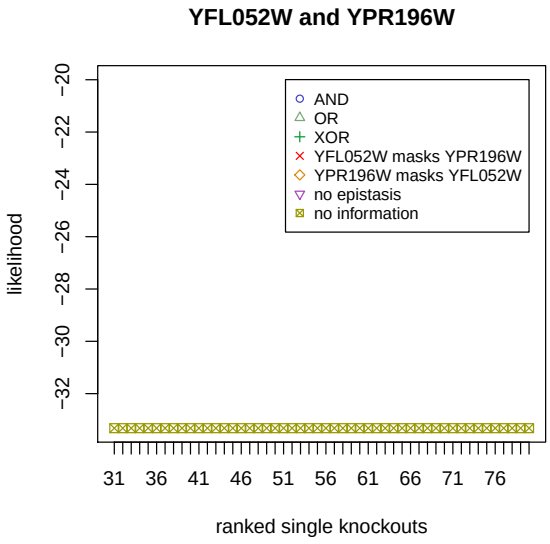

STP3 and STP4
likelihood
ranked single knockouts
SNT1
RPH1
CST6
CBF1
ZAP1
STE5
GLN3
FKH1
HMO1
MIG1
SUM1
SPT2
UPC2
DAL81
RIM101
RPN4
ASG1
CAD1
FKH2
ISW2
NRG2
PHR1
GIS1
SIP4
MSN2
SUT2
NRG1
ROX1
CIN5
HEL2

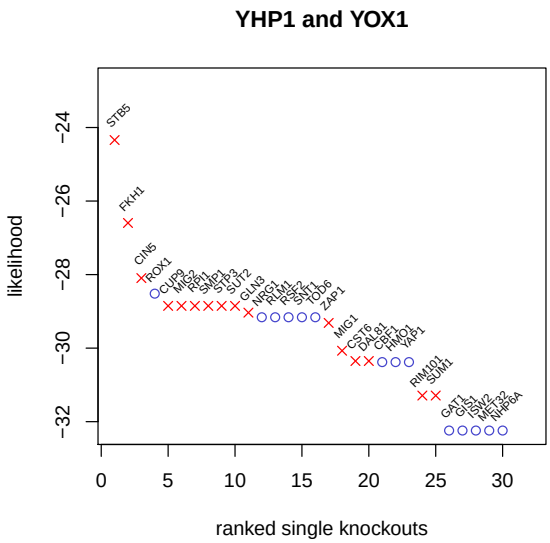

STP3 and STP4
likelihood
ranked single knockouts
AND
OR
XOR
STP3 masks STP4
STP4 masks STP3
no epistasis
no information

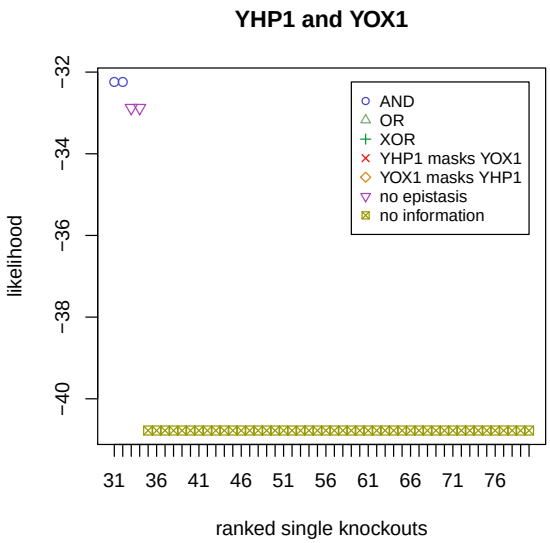

TDA9 and YGR067C
likelihood
ranked single knockouts
HMO1
RIM101
SUM1
GIS1
GZF3
HEL2
YOX1
ISW2
MSN2
SNT1
SPT2
CAD1
GAT1
MET32
NRG1
NHP6A
PHR1
USV1
YPK1
MIG1
STP3
RPH1
CBF1
ROX1
YAP1
DAL81
ZAP1
CIN5
FKH1
SUT2

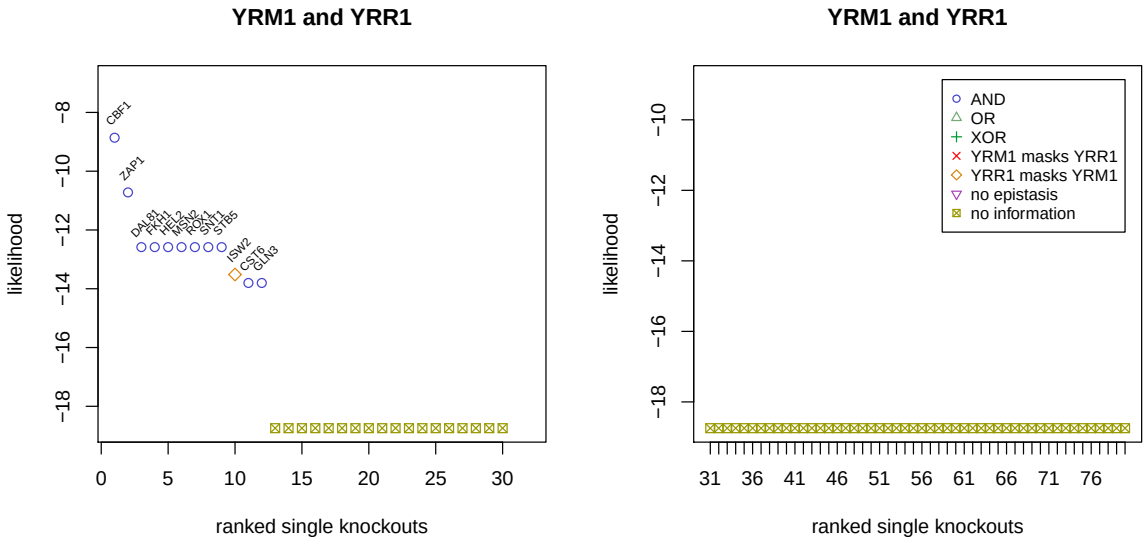

TDA9 and YGR067C
likelihood
ranked single knockouts
AND
OR
XOR
TDA9 masks YGR067C
YGR067C masks TDA9
no epistasis
no information


YFL052W and YPR196W
likelihood
ranked single knockouts


YFL052W and YPR196W
AND
OR
XOR
YFL052W masks YPR196W
YPR196W masks YFL052W
no epistasis
no information
likelihood
ranked single knockouts

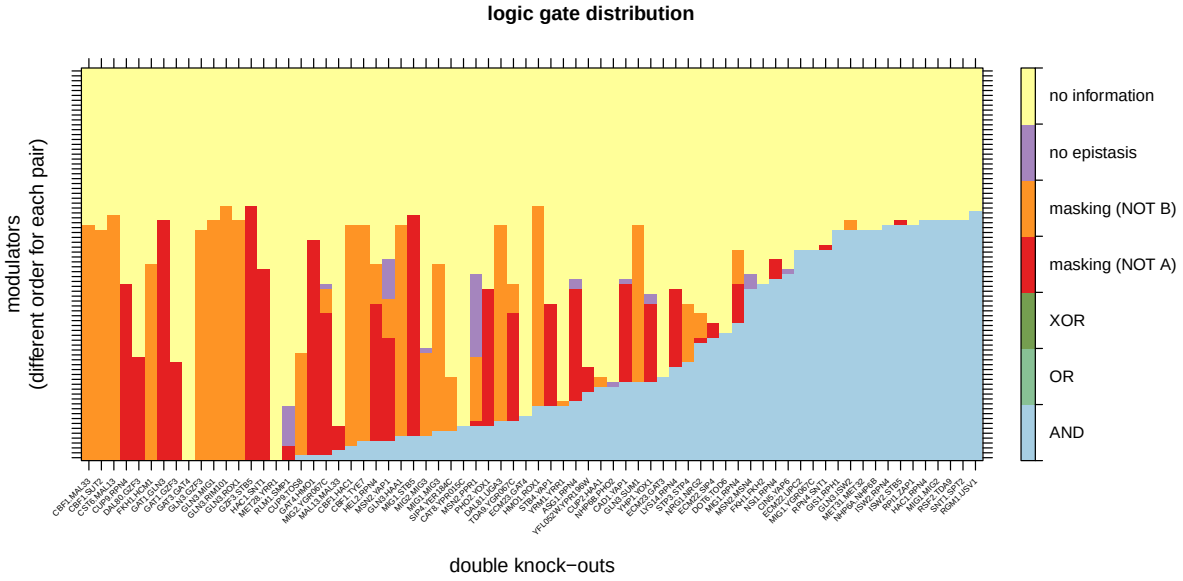

YHP1 and YOX1
likelihood
ranked single knockouts


YHP1 and YOX1
AND
OR
XOR
YHP1 masks YOX1
YOX1 masks YHP1
no epistasis
no information
likelihood
ranked single knockouts




```
distmat <- sameith$logic

distmat[which(distmat %in% "AND")] <- 1
distmat[which(distmat %in% "OR")] <- 2
distmat[which(distmat %in% "XOR")] <- 3
distmat[which(distmat %in% "NOEPI")] <- 6
distmat[which(distmat %in% c("NOINFO", "NOINF"))] <- 7

for (i in 1:ncol(distmat)) {

    genes <- unlist(strsplit(colnames(distmat)[i], "\\."))

    distmat[which(distmat[, i] %in% paste(genes[1], " masks the effect of ", genes[2], sep = "")), i] <-


    distmat[which(distmat[, i] %in% paste(genes[2], " masks the effect of ", genes[1], sep = "")), i] <-

}

distmat <- apply(distmat, c(1,2), as.numeric)

for (i in 1:ncol(distmat)) {
    distmat[, i] <- rev(sort(distmat[, i]))
}

rownames(distmat) <- 1:nrow(distmat)

distmat <- distmat[-which(apply(distmat, 1, sum) == 0), ]

library(bnem)

y <- distmat

distmat <- distmat[, order(apply(distmat, 2, function(x) { return(sum(x == 1)) }))]
```

```
y[which(y == 5)] <- 4

heatmapOP(distmat, Colv = F, Rowv = F, main = "logic gate distribution", sub = "", col = "Paired", breal
```




```
sessionInfo()

## R version 3.3.1 (2016-06-21)
## Platform: x86_64-apple-darwin13.4.0 (64-bit)
## Running under: OS X 10.11.5 (El Capitan)
##
## locale:
## [1] C/UTF-8/C/C/C/C
##
## attached base packages:
## [1] grid      parallel  stats     graphics  grDevices utils     datasets  methods
## [9] base
##
## other attached packages:
##  [1] pcalg_2.4-3         minet_3.32.0        bnem_0.99.0         latticeExtra_0.6-28
##  [5] RColorBrewer_1.1-2  lattice_0.20-34     snowfall_1.84-6.1   snow_0.4-2
##  [9] matrixStats_0.51.0  nem_2.48.0          CellNOptR_1.20.0    XML_3.98-1.5
## [13] Rgraphviz_2.18.0    RCurl_1.95-4.8      bitops_1.0-6        ggplot2_2.2.0
## [17] hash_2.2.6          RBGL_1.50.0         graph_1.52.0        BiocGenerics_0.20.0
## [21] epiNEM_0.99.0       igraph_1.0.1        gtools_3.5.0        e1071_1.6-7
## [25] BoolNet_2.1.1       knitr_1.15          devtools_1.12.0
##
## loaded via a namespace (and not attached):
##  [1] statmod_1.4.26       colorspace_1.3-0     stats4_3.3.1         fastICA_1.2-0
##  [5] gmp_0.5-12           withr_1.0.2          plyr_1.8.4           robustbase_0.92-6
##  [9] stringr_1.1.0        munsell_0.4.3        gtable_0.2.0         bdsmatrix_1.3-2
## [13] memoise_1.0.0        evaluate_0.10        ggm_2.3              BiocInstaller_1.24.0
## [17] class_7.3-14         highr_0.6            DEoptimR_1.0-6       Rcpp_0.12.8
## [21] corpcor_1.6.8        scales_0.4.1         limma_3.30.4         plotrix_3.6-3
## [25] abind_1.4-5          digest_0.6.10        stringi_1.1.2        clue_0.3-51
```

```
## [29] tools_3.3.1          magrittr_1.5         lazyeval_0.2.0       tibble_1.2
## [33] cluster_2.0.5        assertthat_0.1       boot_1.3-18          sfsmisc_1.1-0
```

References: