

R/qtl-related package demo

Overview

This is a quick demo of R/qtl and related packages R/qtlhot, R/qtlnet and R/qtl yeast for workshops on causal networks. For information and tutorials on R and R/qtl, visit <http://www.rqtl.org> . Each package has its own vignettes.

R/qtl

We first show some features of R/qtl. For more info, see <http://rqtl.org/tutorials/rqtltour2.pdf> . This shows a summary of the Brem and Kruglyak (2001) yeast dataset.

```
library(qtl)
library(qtlyeast)
summary(yeast.orf)
```

```
##      Backcross
##
##      No. individuals:      112
##
##      No. phenotypes:      5740
##      Percent phenotyped: 99.2
##
##      No. chromosomes:      16
##      Autosomes:           1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
##
##      Total markers:        2956
##      No. markers:          159 188 49 224 153 113 291 149 132 209 285 273 207
##                          105 309 110
##      Percent genotyped:    98.1
##      Genotypes (%):        AA:50.1  AB:49.9
```

R/qtl genetic map

```
plot.map(yeast.orf)
```

Examine one trait.

We now look at one particular trait, which is located on chr 15.

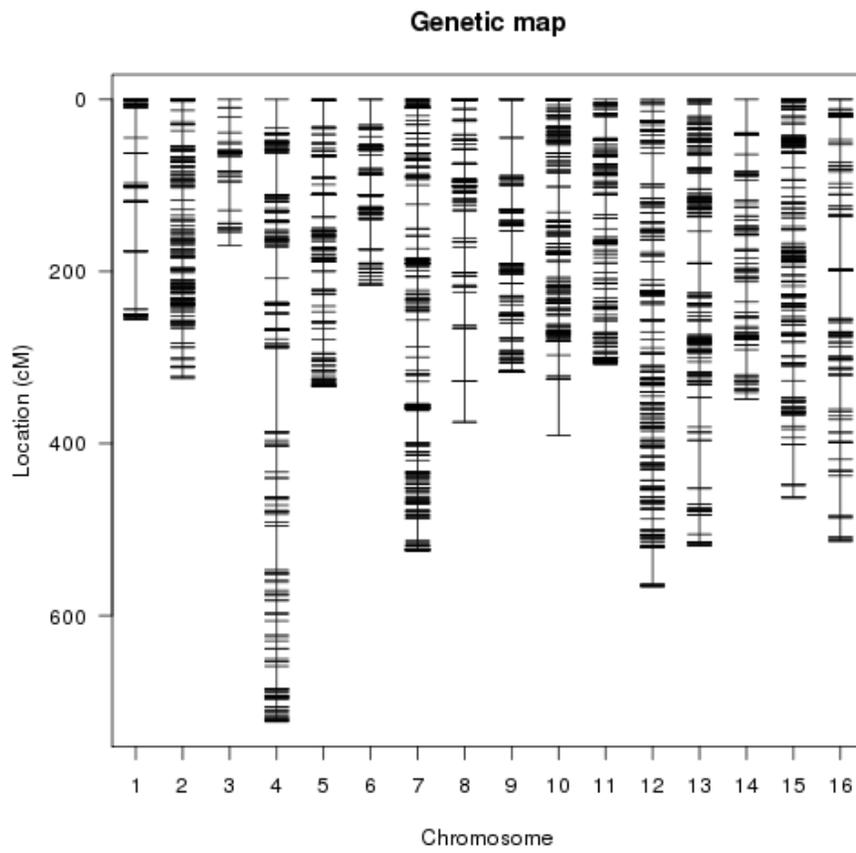


Figure 1: plot of chunk unnamed-chunk-2

```

tf.orf <- "YOL084W"
tf.gene <- yeast.annot[yeast.annot$orf == tf.orf, "gene"]
tf.chr <- yeast.annot[yeast.annot$orf == tf.orf, "chr"]

## Invoke QTL hotspot and causal pair library.
library(qtlhot, quietly = TRUE)
cand <- cis.cand.reg$cis.reg[cis.cand.reg$cis.reg$gene == tf.orf, ]
cand.peak <- cand$peak.pos
cand.pos <- cis.cand.reg$cis.reg$phys.pos[cis.cand.reg$cis.reg$gene == tf.orf]

```

Image map of chr 15.

We examine one particular open reading frame, YOL084W, corresponding to gene, PHM7. Here is an image of the genotypes for chromosome 'r tf.chr' reordered by this ORF.

```

cand.mar <- which.min(abs(cand.pos - pull.map(yeast.orf)[[tf.chr]]))
geno.image(yeast.orf, tf.chr, reorder = find.pheno(yeast.orf, tf.orf))
abline(v = cand.mar, lwd = 4, col = "green")

```

This ORF is located at 58.2166cM; the nearest marker (number 136) is identified with a green line on the genotype image. Now we do a simple interval map scan, which finds the strongest evidence (LOD) for a QTL right at the gene location. The red line is the 0.1% permutation threshold.

1-D scan of genome

```

yeast.orf <- calc.genoprob(yeast.orf, step = 2)
scan.orf <- scanone(yeast.orf, pheno.col = find.pheno(yeast.orf, tf.orf), method = "hk")
plot(scan.orf)

```

1-D scan focused on 15

```

plot(scan.orf, chr = tf.chr)
abline(v = cand.pos, lwd = 3, col = "green")
lod.thr <- c(summary(perm.orf, 0.001))
abline(h = lod.thr, lwd = 3, col = "red", lty = 2)

```

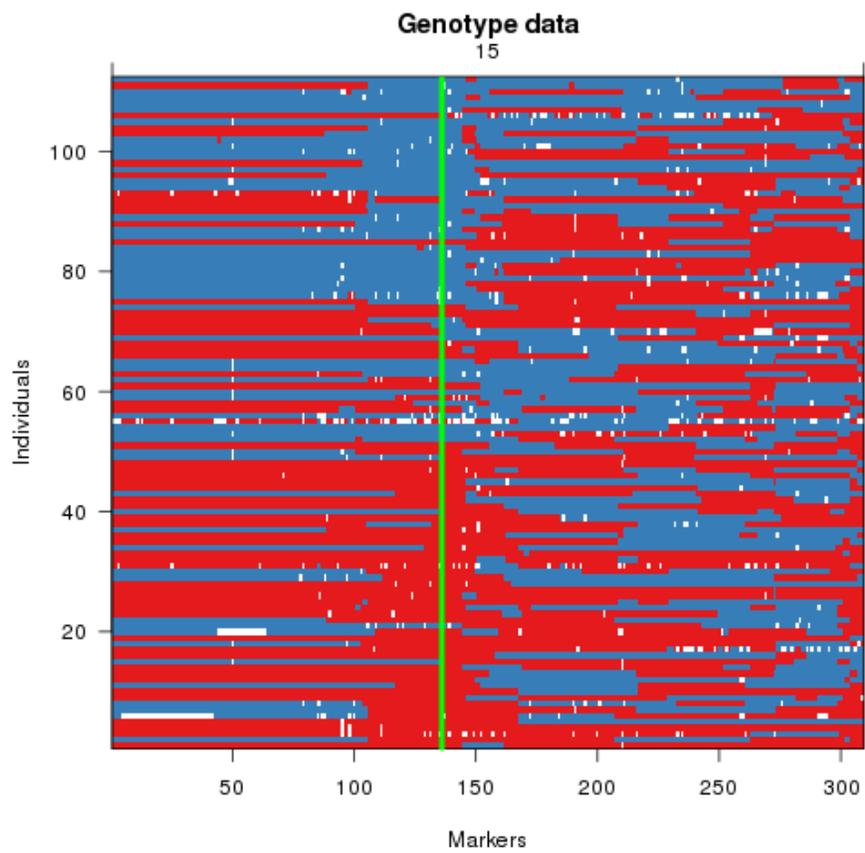


Figure 2: plot of chunk unnamed-chunk-4

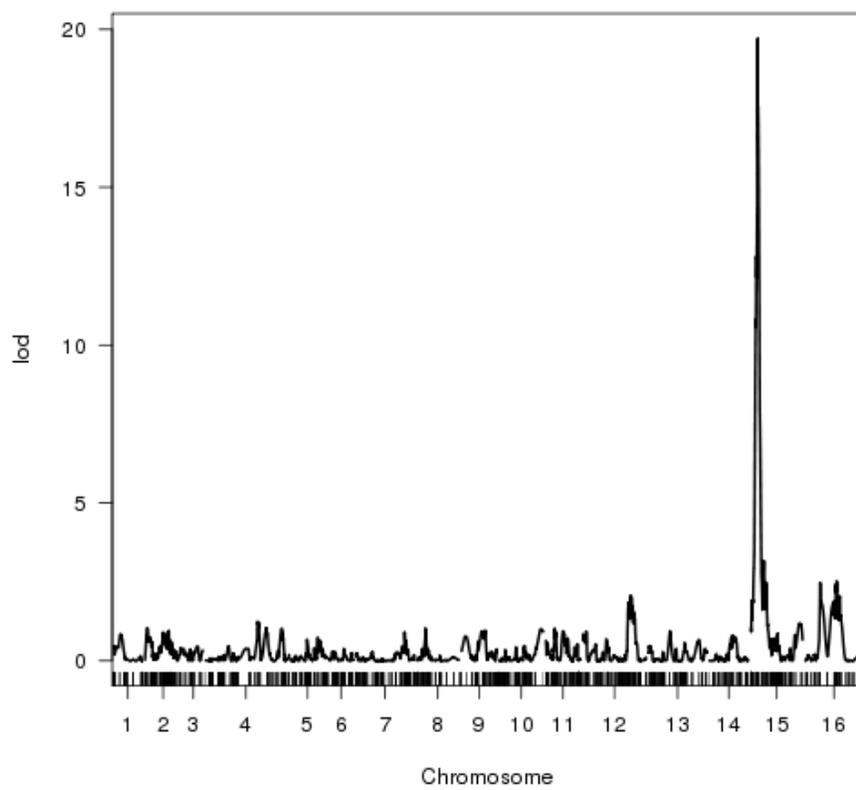


Figure 3: plot of chunk unnamed-chunk-5

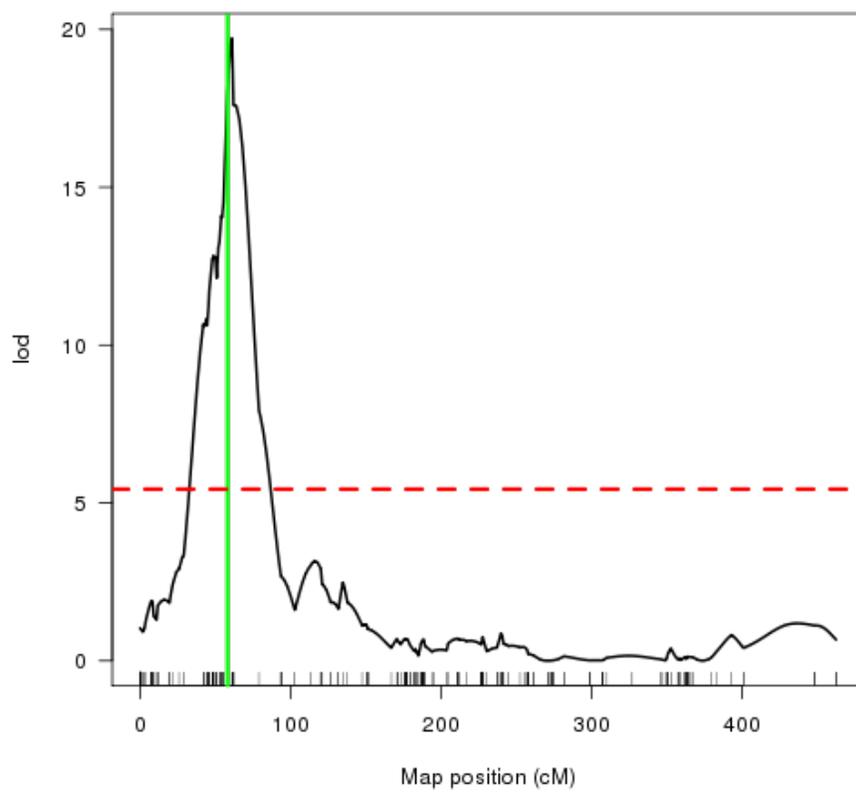


Figure 4: plot of chunk unnamed-chunk-6

R/qtlhot: hotspots

R/qtlhot: hotspots

Now let's look at hotspots. The `vignette("qtlhot", "qtlhot")` walks through a simulated set of hotspots. For the yeast data, see `vignette("yeast_cmst", "qtl yeast")`. Basically, we use the saved high LOD scores (`highlod.orf`) and LOD permutation threshold (`lod.thr`) to find hotspot size across the genome (`hotsize.orf`). The summary shows several significant hotspots, with one of the largest on chr 'r tf.chr', right at YOL084W. The red 5% threshold of 96 is for the number of traits with significant LOD at a position (Chaibub Neto et al. 2012).

```
hotsize.orf <- hotsize(highlod.orf, lod.thr = lod.thr)
hot.thr <- 96
(hot.sum <- summary(hotsize.orf, threshold = hot.thr))

## hotsize elements: chr pos max.N
## LOD threshold: 5.434

##           chr  pos max.N
## c2.loc226      2 226.0  107
## YLR257W_chr12@659357 12 323.6  98
## c14.loc242     14 242.0  259
## gOL02.1_chr15@174364 15  61.1  125

plot(hotsize.orf, chr = hot.sum$chr, bandcol = "gray70")
abline(h = hot.thr, lwd = 2, col = "red", lty = 2)
```

Hotspot on chr 15

```
plot(hotsize.orf, chr = tf.chr)
abline(v = cand.pos, lwd = 3, col = "green")
abline(h = hot.thr, lwd = 2, col = "red", lty = 2)
```

R/qtlhot: causal pairs

R/qtlhot: causal pairs

Find co-mapped traits and get joint-CMST p-values with Benjaminie-Hochberg adjustment (Chaibub Neto et al. 2013).

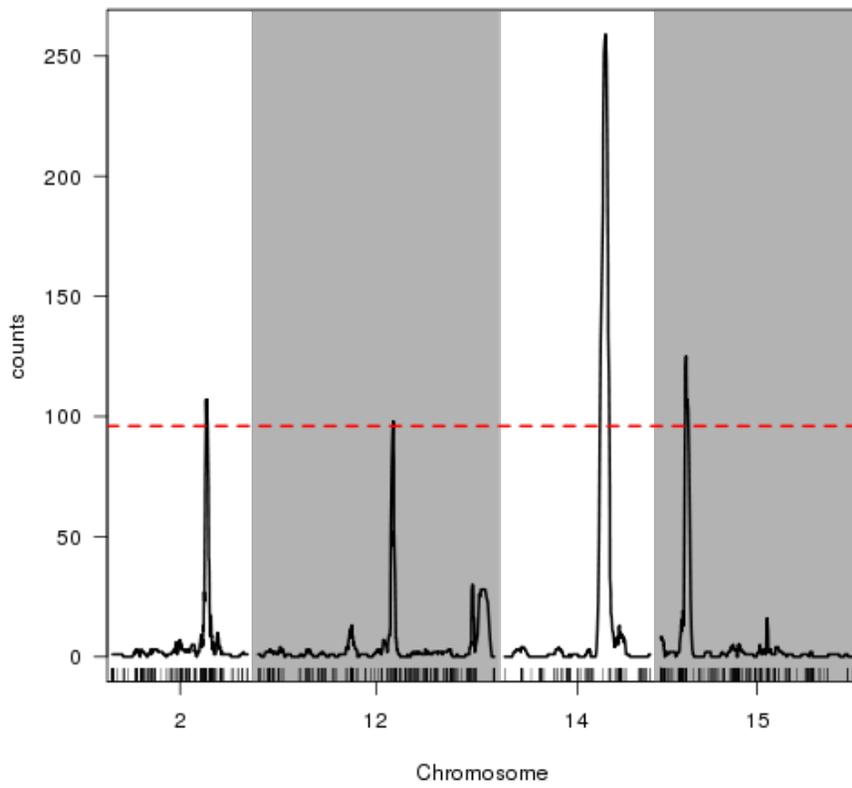


Figure 5: plot of chunk unnamed-chunk-7

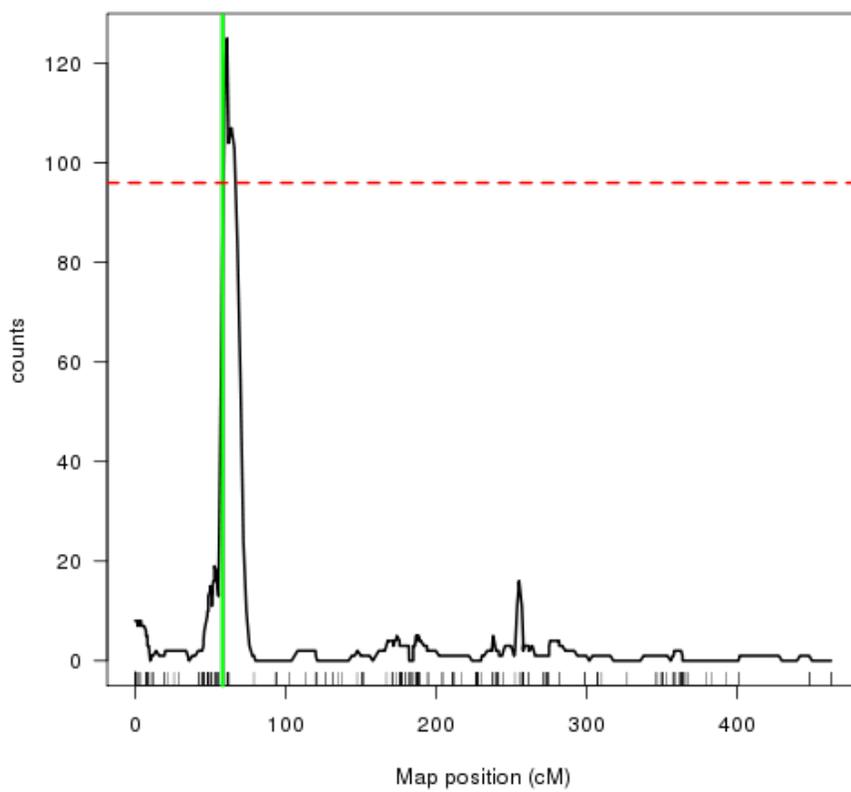


Figure 6: plot of chunk unnamed-chunk-8

```
cand.tf <- cand.reg[cand.reg$gene == tf.orf, , drop = FALSE]
comap.tf <- GetCoMappingTraits(highlod.orf, cand.tf)[[1]]
```

The knock-out experiments in Hughes et al. (2000) and Zhu et al. (2008) provide a database of experimentally validated targets, which are stored in `ko.list` in the `qtlyeast` package. Let's find which of the co-mapping traits are targets of the transcription factor PHM7, and do causal mapping tests for them.

```
match.orf <- !is.na(match(comap.tf, ko.list[[tf.orf]]))
out.peak <- CMSTtests(yeast.orf, tf.orf, comap.tf[match.orf], Q.chr = tf.chr,
  Q.pos = cand.peak, method = "joint", penalty = "bic")
causal.call <- function(x) {
  ordered(c("causal", "react", "indep", "corr")[apply(x, 1, which.min)], c("causal",
    "react", "indep", "corr"))
}
target.peak <- data.frame(pval = apply(out.peak$pvals.j.BIC, 1, min), call = causal.call(out.peak),
row.names(target.peak) <- factor(comap.tf[match.orf]))
```

Focus on Significant causal calls

```
pval.thr <- 0.001
## Look at table by various significance levels.
tmp <- 1 + (target.peak$pval <= 0.05) + (target.peak$pval <= 0.005) + (target.peak$pval <=
  0.001)
lev <- c("n.s.", "0.05", "0.005", "0.001")
target.peak$signif <- ordered(lev[tmp], lev)
rm(tmp, lev)
table(target.peak$signif, target.peak$call)

##
##          causal react indep corr
## n.s.         50     0     9     6
## 0.05         32     0     1     0
## 0.005        10     0     0     0
## 0.001         5     0     0     0

## Now reduce to the more significant targets.
target.peak$gene <- yeast.annot[match(row.names(target.peak), yeast.annot$orf),
  "gene"]
target.peak <- target.peak[order(target.peak$pval), ]
target.peak <- target.peak[target.peak$pval <= 0.005, ]
target.peak
```

```
##           pval   call signif   gene
## YNR014W 6.685e-05 causal  0.001 YNR014W
## YLR178C 2.580e-04 causal  0.001 TFS1
## YNL195C 2.787e-04 causal  0.001 YNL195C
## YJL161W 4.200e-04 causal  0.001 FMP33
## YJL210W 4.674e-04 causal  0.001 PEX2
## YOR028C 1.067e-03 causal  0.005 CIN5
## YJR096W 1.294e-03 causal  0.005 YJR096W
## YJL111W 2.224e-03 causal  0.005 CCT7
## YDR032C 2.382e-03 causal  0.005 PST2
## YMR170C 2.488e-03 causal  0.005 ALD2
## YHR104W 2.553e-03 causal  0.005 GRE3
## YIL113W 3.039e-03 causal  0.005 SDP1
## YPR160W 3.352e-03 causal  0.005 GPH1
## YKL091C 3.513e-03 causal  0.005 YKL091C
## YOL097C 4.341e-03 causal  0.005 WRS1
```

1-D scan of causal trait and targets

Only 5 targets had a highly significant causal call at the 0.001 level. Here are their scans (in blue) along with the YOL084W (in black).

```
plot(scan.orf, chr = tf.chr, xlim = c(0, 100))
targets <- row.names(target.peak)[target.peak$pval <= pval.thr]
scans <- scanone(yeast.orf, pheno.col = find.pheno(yeast.orf, targets), method = "hk")
for (i in seq(targets)) plot(scans, chr = tf.chr, lodcolumn = i, add = TRUE,
    col = "blue")
abline(v = cand.pos, lwd = 3, col = "green")
abline(h = lod.thr, lwd = 3, col = "red", lty = 2)
```

Thus, it appears that the PHM7 is causal to several targets, suggesting the value of inferring a causal network.

R/qtlnet: causal network

R/qtlnet: causal network

Let's take the significant targets of PHM7 and see if we can develop a causal network.

```
net.set <- c(tf.orf, row.names(target.peak)[target.peak$pval <= pval.thr])
library(qtlnet, quietly = TRUE)
```

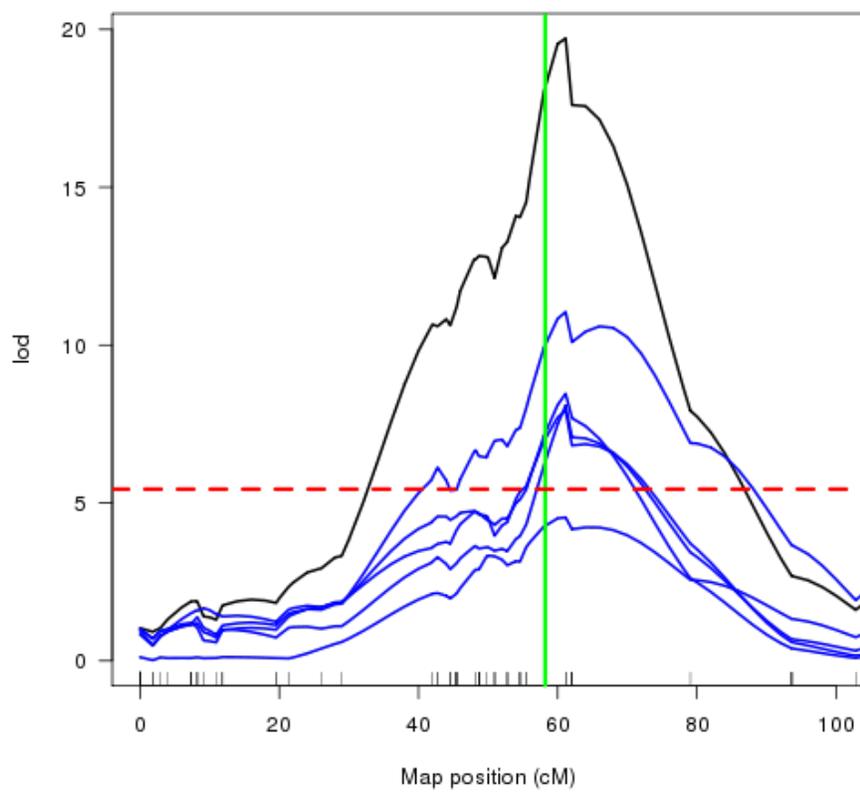


Figure 7: plot of chunk unnamed-chunk-12

```

## Loading required package: MASS

## Loading required package: matrixcalc

## Attaching package: 'matrixcalc'

## The following object(s) are masked from 'package:corpcor':
##
## is.positive.definite

## Attaching package: 'graph'

## The following object(s) are masked from 'package:igraph':
##
## degree, edges

## Loading required package: abind

## Loading required package: sfsmisc

```

The following will create MCMC samples for the QTLnet, but it takes awhile. Instead we read already created data. `phm7.qtlnet <- mcmc.qtlnet(yeast.orf, pheno.col = find.pheno(yeast.orf, net.set), threshold = lod.thr, nSamples = 1000, thinning = 20, random.seed = 123, max.parents = 3) save(phm7.qtlnet, file = "phm7.qtlnet.RData", compress = TRUE)`

Plot of causal network.

```

load("phm7.qtlnet.RData")
phm7.graph <- igraph.qtlnet(phm7.qtlnet)
## Make causal trait yellow.
phm7.graph <- set.vertex.attribute(phm7.graph, "color", 1, "yellow")
plot(phm7.graph)

```

The plot randomly arranges the traits and QTL. Alternatively, use the interactive `tkplot(phm7.graph)` In accordance with the results of the causality tests, ‘r tf.orf’ shows up in the top of the transcriptional network, being causal for all other traits. Note the inclusion of other QTL.

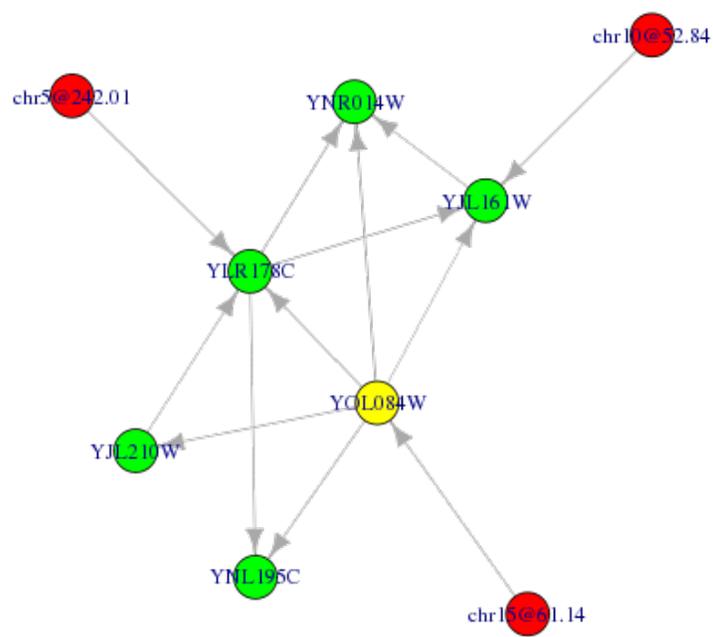


Figure 8: plot of chunk unnamed-chunk-14