Seattle Summer Institute 2009
Advanced QTL
Brian S. Yandell, UW-Madison

www.stat.wisc.edu/~yandell/statgen

- overview: multiple QTL approaches
- Bayesian QTL mapping & model selection
- data examples in detail
- software demos: R/qtl and R/qtlbim

Real knowledge is to know the extent of one's ignorance.
Confucius (on a bench in Seattle)

Overview of Multiple QTL

1. what is the goal of multiple QTL study?
2. gene action and epistasis
3. Bayesian vs. classical QTL
4. QTL model selection
5. QTL software options

1. what is the goal of QTL study?

- uncover underlying biochemistry
  - identify how networks function, break down
  - find useful candidates for (medical) intervention
  - epistasis may play key role
  - statistical goal: maximize number of correctly identified QTL
- basic science/evolution
  - how is the genome organized?
  - identify units of natural selection
  - additive effects may be most important (Wright/Fisher debate)
  - statistical goal: maximize number of correctly identified QTL
- select "elite" individuals
  - predict phenotype (breeding value) using suite of characteristics (phenotypes) translated into a few QTL
  - statistical goal: mimimize prediction error

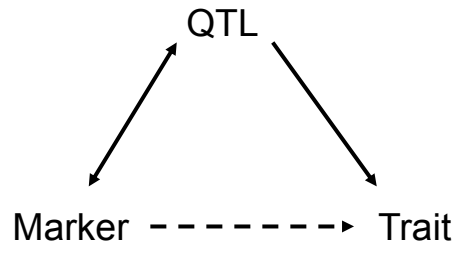
cross two inbred lines

→ linkage disequilibrium

→ associations

→ linked segregating QTL

(after Gary Churchill)



problems of single QTL approach

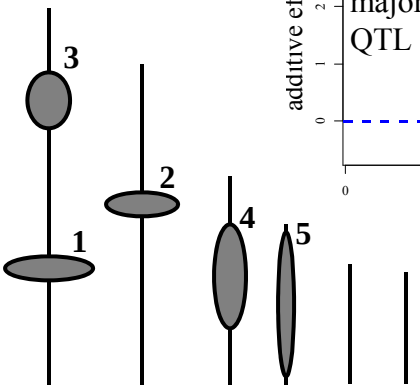
- wrong model: biased view
  - fool yourself: bad guess at locations, effects
  - detect ghost QTL between linked loci
  - miss epistasis completely
- low power
- bad science
  - use best tools for the job
  - maximize scarce research resources
  - leverage already big investment in experiment

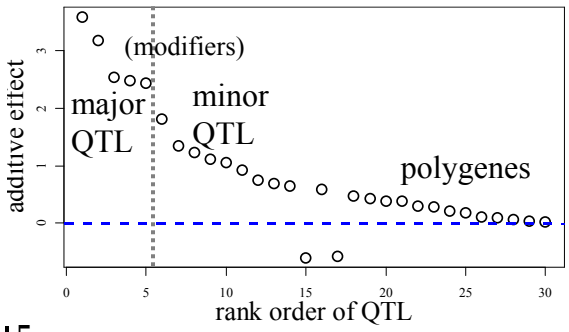
advantages of multiple QTL approach

- improve statistical power, precision
  - increase number of QTL detected
  - better estimates of loci: less bias, smaller intervals
- improve inference of complex genetic architecture
  - patterns and individual elements of epistasis
  - appropriate estimates of means, variances, covariances
    - asymptotically unbiased, efficient
  - assess relative contributions of different QTL
- improve estimates of genotypic values
  - less bias (more accurate) and smaller variance (more precise)
  - mean squared error = MSE = $(\text{bias})^2$ + variance

Pareto diagram of QTL effects

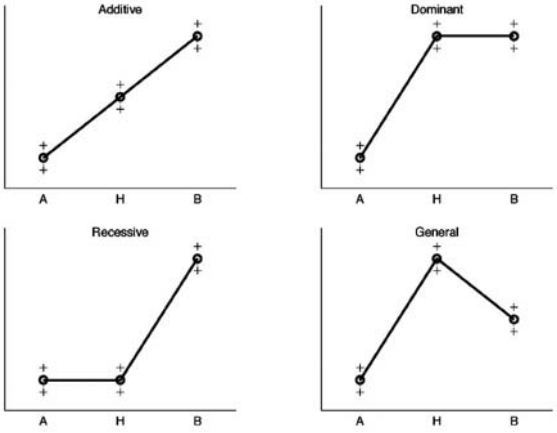
major QTL on linkage map





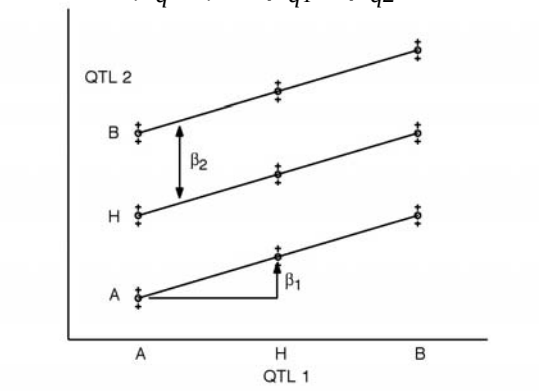
2. Gene Action and Epistasis

additive, dominant, recessive, general effects of a single QTL (Gary Churchill)



additive effects of two QTL (Gary Churchill)

$$\mu_q = \mu + \beta_{q1} + \beta_{q2}$$



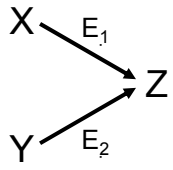
Epistasis (Gary Churchill)

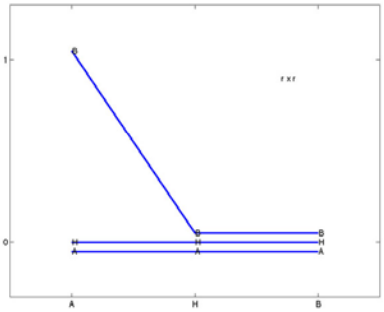
The allelic state at one locus can mask or uncover the effects of allelic variation at another.

- W. Bateson, 1907.

epistasis in parallel pathways (GAC)

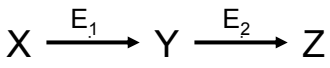
- Z keeps trait value low
- neither E_1 nor E_2 is rate limiting
- loss of function alleles are segregating from parent A at E_1 and from parent B at E_2

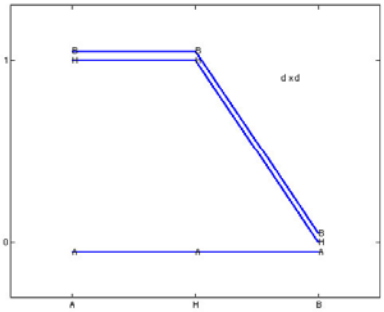




epistasis in a serial pathway (GAC)

- Z keeps trait value high
- either E_1 or E_2 is rate limiting
- loss of function alleles are segregating from parent B at E_1 or from parent A at E_2





epistatic interactions

- model space issues
  - 2-QTL interactions only?
    - or general interactions among multiple QTL?
  - partition of effects
    - Fisher-Cockerham or tree-structured or ?
- model search issues
  - epistasis between significant QTL
    - check all possible pairs when QTL included?
    - allow higher order epistasis?
  - epistasis with non-significant QTL
    - whole genome paired with each significant QTL?
    - pairs of non-significant QTL?
- see papers of Nengjun Yi (2000-7) in *Genetics*

limits of epistatic inference

- power to detect effects
  - epistatic model sizes grow quickly
    - $|A| = 3^{n.qtl}$ for general interactions
  - power tradeoff
    - depends sample size *vs.* model size
    - want $n / |A|$ to be fairly large (say > 5)
    - 3 QTL, $n = 100$ F2: $n / |A| \approx 4$
- rare genotypes may not be observed
  - *aa/BB & AA/bb* rare for linked loci
  - empty cells mess up balance
    - adjusted tests (type III) are wrong
  - confounds main effects & interactions

2 linked QTL
empty cell
with $n = 100$

	bb	*bB*	*BB*
aa	6	15	0
aA	15	25	15
AA	3	15	6

limits of multiple QTL?

- limits of statistical inference
  - power depends on sample size, heritability, environmental variation
  - "best" model balances fit to data and complexity (model size)
  - genetic linkage = correlated estimates of gene effects
- limits of biological utility
  - sampling: only see some patterns with many QTL
  - marker assisted selection (Bernardo 2001 *Crop Sci*)
    - 10 QTL ok, 50 QTL are too many
    - phenotype better predictor than genotype when too many QTL
    - increasing sample size may not give multiple QTL any advantage
  - hard to select many QTL simultaneously
    - 3^m possible genotypes to choose from

QTL below detection level?

- problem of selection bias
  - QTL of modest effect only detected sometimes
  - effects overestimated when detected
  - repeat studies may fail to detect these QTL
- think of probability of detecting QTL
  - avoids sharp in/out dichotomy
  - avoid pitfalls of one "best" model
  - examine "better" models with more probable QTL
- rethink formal approach for QTL
  - directly allow uncertainty in genetic architecture
  - QTL model selection over genetic architecture

3. Bayesian vs. classical QTL study

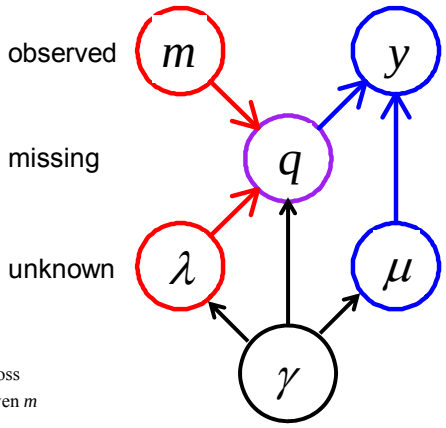
- classical study
  - ***maximize*** over unknown effects
  - ***test*** for detection of QTL at loci
  - model selection in stepwise fashion
- Bayesian study
  - ***average*** over unknown effects
  - ***estimate*** chance of detecting QTL
  - sample all possible models
- both approaches
  - average over missing QTL genotypes
  - scan over possible loci

Bayesian idea

- Reverend Thomas Bayes (1702-1761)
  - part-time mathematician
  - buried in Bunhill Cemetary, Moongate, London
  - famous paper in 1763 *Phil Trans Roy Soc London*
  - was Bayes the first with this idea? (Laplace?)
- basic idea (from Bayes' original example)
  - two billiard balls tossed at random (uniform) on table
  - where is first ball if the second is to its left?
    - prior: anywhere on the table
    - posterior: more likely toward right end of table

QTL model selection: key players

- observed measurements
  - y = phenotypic trait
  - m = markers & linkage map
  - i = individual index (1,…,n)
- missing data
  - missing marker data
  - q = QT genotypes
    - alleles QQ, Qq, or qq at locus
- unknown quantities
  - λ = QT locus (or loci)
  - μ = phenotype model parameters
  - γ = QTL model/genetic architecture
- pr(q|m,λ,γ) genotype model
  - grounded by linkage map, experimental cross
  - recombination yields multinomial for q given m
- pr(y|q,μ,γ) phenotype model
  - distribution shape (assumed normal here)
  - unknown parameters μ (could be non-parametric)



after
Sen Churchill (2001)

Bayes posterior vs. maximum likelihood

- LOD: classical Log ODds
  - maximize likelihood over effects μ
  - `R/qtl scanone/scantwo: method = "em"`
- *LPD*: Bayesian *L*og *P*osterior *D*ensity
  - average posterior over effects μ
  - `R/qtl scanone/scantwo: method = "imp"`

$$\mathrm{LOD}(\lambda) = \log_{10}\{\max_{\mu} \mathrm{pr}(y \mid m, \mu, \lambda)\} + c$$

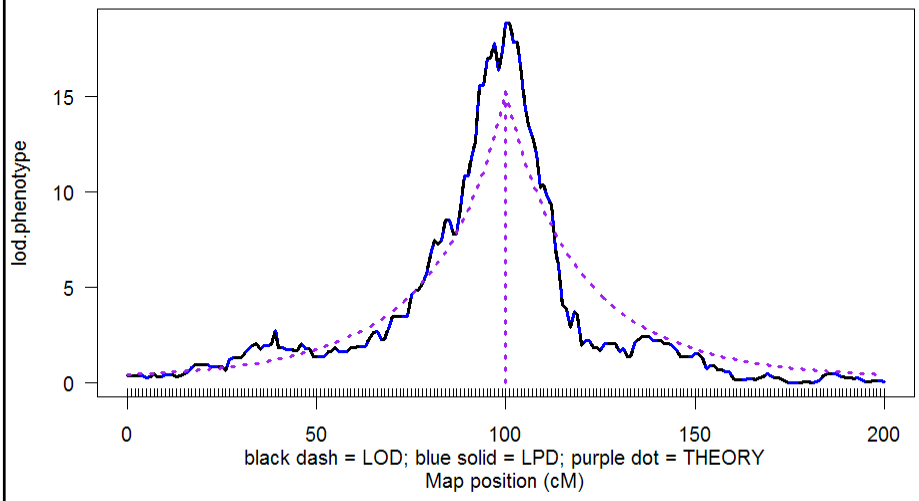
$$\mathrm{LPD}(\lambda) = \log_{10}\{\mathrm{pr}(\lambda \mid m)\int \mathrm{pr}(y \mid m, \mu, \lambda)\mathrm{pr}(\mu)d\mu\} + C$$

likelihood mixes over missing QTL genotypes:

$$\mathrm{pr}(y \mid m, \mu, \lambda) = \sum_q \mathrm{pr}(y \mid q, \mu)\mathrm{pr}(q \mid m, \lambda)$$

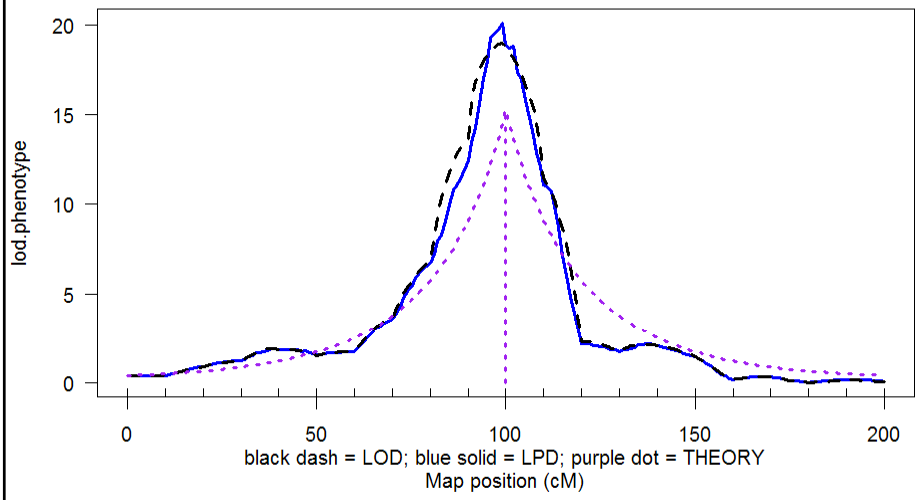
LOD & LPD: 1 QTL

n.ind = 100, 1 cM marker spacing



LOD & LPD: 1 QTL

n.ind = 100, 10 cM marker spacing



marginal LOD or LPD

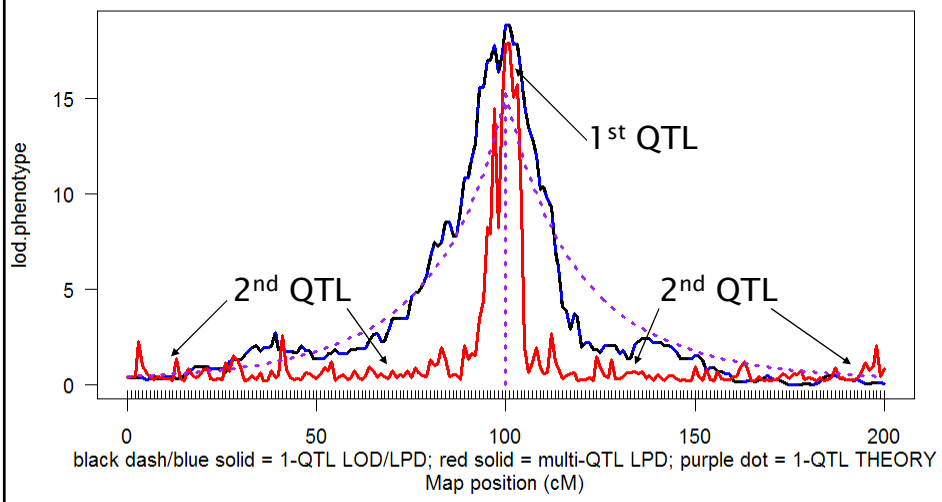
- compare two genetic architectures (γ_2,γ_1) at each locus
  - with (γ_2) or without (γ_1) another QTL at locus λ
    - preserve model hierarchy (e.g. drop any epistasis with QTL at λ)
  - with (γ_2) or without (γ_1) epistasis with QTL at locus λ
  - γ_2 contains γ_1 as a sub-architecture
- allow for multiple QTL besides locus being scanned
  - architectures γ_1 and γ_2 may have QTL at several other loci
  - use marginal LOD, LPD or other diagnostic
  - posterior, Bayes factor, heritability

$$\mathrm{LOD}(\lambda \mid \gamma_2) - \mathrm{LOD}(\lambda \mid \gamma_1)$$

$$\mathrm{LPD}(\lambda \mid \gamma_2) - \mathrm{LPD}(\lambda \mid \gamma_1)$$

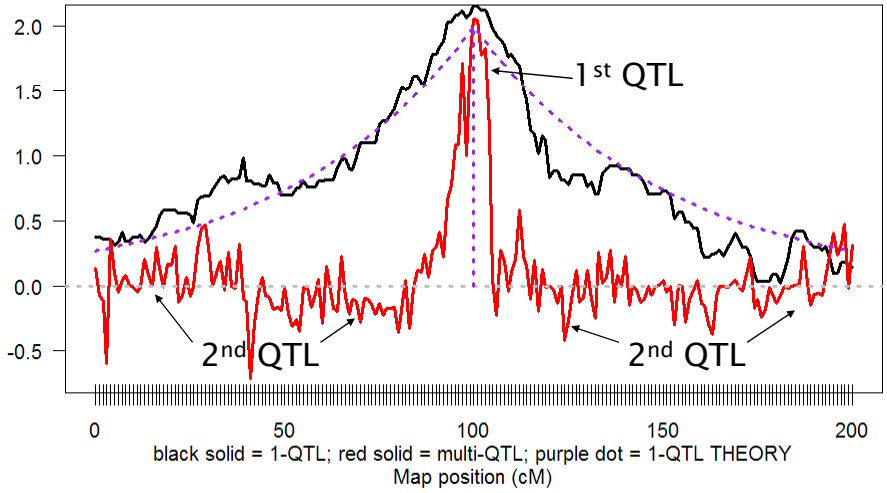
LPD: 1 QTL vs. multi-QTL

marginal contribution to LPD from QTL at λ



substitution effect: 1 QTL vs. multi-QTL

single QTL effect vs. marginal effect from QTL at λ



why use a Bayesian approach?

- first, do *both* classical and Bayesian
  - always nice to have a separate validation
  - each approach has its strengths and weaknesses
- classical approach works quite well
  - selects large effect QTL easily
  - directly builds on regression ideas for model selection
- Bayesian approach is comprehensive
  - samples most probable genetic architectures
  - formalizes model selection within one framework
  - readily (!) extends to more complicated problems

4. QTL model selection

- select class of models
  - see earlier slides above
- decide how to compare models
  - (Bayesian interval mapping talk later)
- search model space
  - (Bayesian interval mapping talk later)
- assess performance of procedure
  - see Kao (2000), Broman and Speed (2002)
  - Manichaukul, Moon, Yandell, Broman (in prep)
  - be wary of HK regression assessments

pragmatics of multiple QTL

- evaluate some objective for model given data
  - classical likelihood
  - Bayesian posterior
- search over possible genetic architectures (models)
  - number and positions of loci
  - gene action: additive, dominance, epistasis
- estimate "features" of model
  - means, variances & covariances, confidence regions
  - marginal or conditional distributions
- art of model selection
  - how select "best" or "better" model(s)?
  - how to search over useful subset of possible models?

comparing models

- balance model fit against model complexity
  - want to fit data well (maximum likelihood)
  - without getting too complicated a model

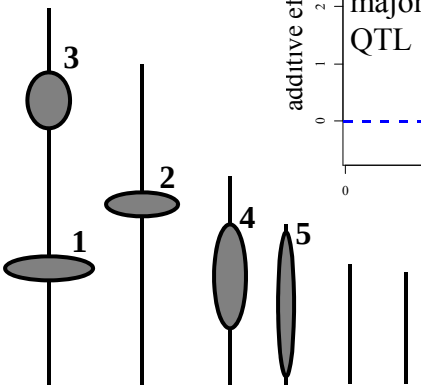
	smaller model	**bigger model**
fit model	miss key features	fits better
estimate phenotype	may be biased	no bias
predict new data	may be biased	no bias
interpret model	easier	more complicated
estimate effects	low variance	high variance

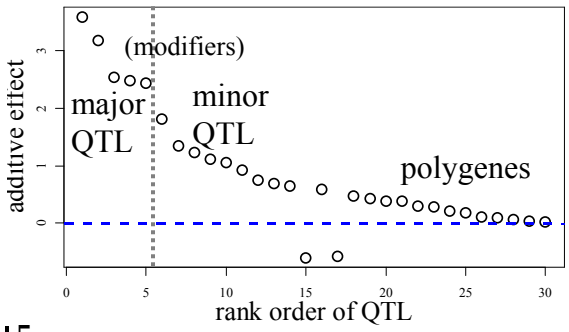
Bayesian model averaging

- average summaries over multiple architectures
- avoid selection of "best" model
- focus on "better" models
- examples in data talk later

Pareto diagram of QTL effects

major QTL on linkage map





5. QTL software options

- methods
  - approximate QTL by markers
  - exact multiple QTL interval mapping
- software platforms
  - MapMaker/QTL (obsolete)
  - QTLCart (statgen.ncsu.edu/qtlcart)
  - R/qtl (www.rqtl.org)
  - R/qtlbim (www.qtlbim.org)
  - Yandell, Bradbury (2007) book chapter

approximate QTL methods

- marker regression
  - locus & effect confounded
  - lose power with missing data
- Haley-Knott (least squares) regression
  - correct mean, wrong variance
  - biased by pattern of missing data (Kao 2000)
- extended HK regression
  - correct mean and variance
  - minimizes bias issue (R/qtl "ehk" method)
- composite interval mapping (QTLCart)
  - use markers to approximate other QTL
  - properties depend on marker spacing, missing data

exact QTL methods

- interval mapping (Lander, Botstein 1989)
  - scan whole genome for single QTL
  - bias for linked QTL, low power
- multiple interval mapping (Kao, Zeng, Teasdale 1999)
  - sequential scan of all QTL
  - stepwise model selection
- multiple imputation (Sen, Churchill 2001)
  - fill in (impute) missing genotypes along genome
  - average over multiple imputations
- Bayesian interval mapping (Yi et al. 2005)
  - sample most likely models
  - marginal scans conditional on other QTL

QTL software platforms

- QTLCart (statgen.ncsu.edu/qtlcart**)**
  - includes features of original MapMaker/QTL
    - not designed for building a linkage map
  - easy to use Windows version WinQTLCart
  - based on Lander-Botstein maximum likelihood LOD
    - extended to marker cofactors (CIM) and multiple QTL (MIM)
    - epistasis, some covariates (GxE)
    - stepwise model selection using information criteria
  - some multiple trait options
  - OK graphics
- R/qtl (www.rqtl.org)
  - includes functionality of classical interval mapping
  - many useful tools to check genotype data, build linkage maps
  - excellent graphics
  - several methods for 1-QTL and 2-QTL mapping
    - epistasis, covariates (GxE)
  - tools available for multiple QTL model selection

Bayesian QTL software options

- Bayesian Haley-Knott approximation: no epistasis
  - Berry C (1998)
    - R/bqtl (www.r-project.org contributed package)
- multiple imputation: epistasis, mostly 1-2 QTL but some multi-QTL
  - Sen and Churchill (2000)
    - matlab/pseudomarker (www.jax.org/staff/churchill/labsite/software)
  - Broman et al. (2003)
    - R/qtl (www.rqtl.org)
- Bayesian interval mapping via MCMC: no epistasis
  - Satagopan et al. (1996); Satagopan, Yandell (1996) Gaffney (2001)
    - R/bim (www.r-project.org contributed package)
    - WinQTLCart/bmapqtl (statgen.ncsu.edu/qtlcart)
  - Stephens & Fisch (1998): no code release
  - Sillanpää Arjas (1998)
    - multimapper (www.rni.helsinki.fi/~mjs)
- Bayesian interval mapping via MCMC: epistasis
  - Yandell et al. (2007)
    - R/qtlbim (www.qtlbim.org)
- Bayesian shrinkage: no epistasis
  - Wang et al. Xu (2005): no code release

R/qtlbim: www.qtlbim.org

- Properties
  - cross-compatible with R/qtl
  - new MCMC algorithms
    - Gibbs with loci indicators; no reversible jump
  - epistasis, fixed & random covariates, GxE
  - extensive graphics
- Software history
  - initially designed (Satagopan Yandell 1996)
  - major revision and extension (Gaffney 2001)
  - R/bim to CRAN (Wu, Gaffney, Jin, Yandell 2003)
  - R/qtlbim to CRAN (Yi, Yandell et al. 2006)
- Publications
  - Yi et al. (2005); Yandell et al. (2007); …

many thanks

U AL Birmingham
Nengjun Yi
Tapan Mehta
Samprit Banerjee
Daniel Shriner
Ram Venkataraman
David Allison

Jackson Labs
Gary Churchill
Hao Wu
Hyuna Yang
Randy von Smith

Alan Attie
Jonathan Stoehr
Hong Lan
Susie Clee
Jessica Byers
Mark Gray-Keller

Tom Osborn
David Butruille
Marcio Ferrera
Josh Udahl
Pablo Quijada

UW-Madison Stats
Yandell lab
Jaya Satagopan
Fei Zou
Patrick Gaffney
Chunfang Jin
Elias Chaibub
W Whipple Neely
Jee Young Moon
Elias Chaibub

Michael Newton
Karl Broman
Christina Kendziorski
Daniel Gianola
Liang Li
Daniel Sorensen

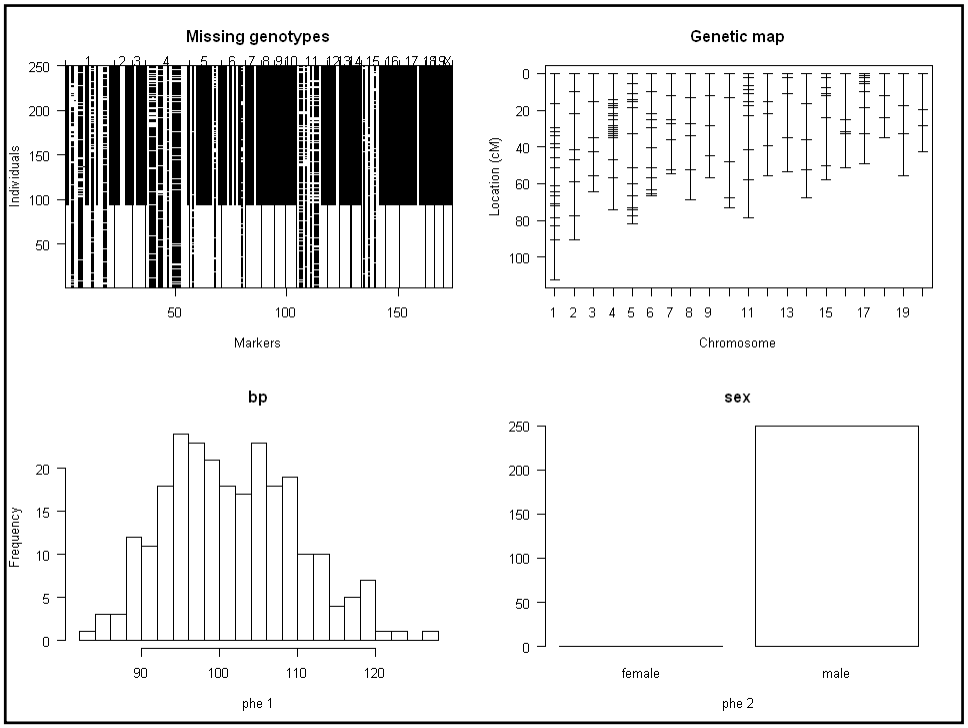
USDA Hatch, NIH/NIDDK (Attie), NIH/R01s (Yi, Broman)

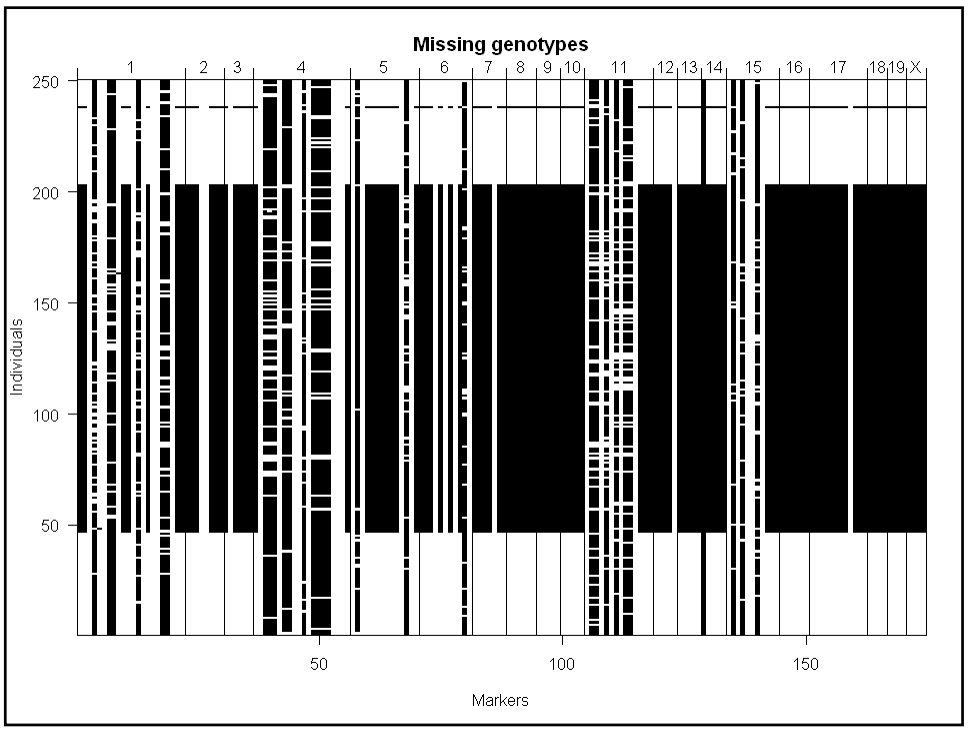
R/qtl & R/qtlbim Tutorials

- R statistical graphics & language system
- R/qtl tutorial
  - R/qtl web site: www.rqtl.org
  - Tutorial: www.rqtl.org/tutorials/rqtltour.pdf
  - R code: www.rqtl.org/tutorials/rqtltour.R
- R/qtlbim tutorial
  - R/qtlbim web site: www.qtlbim.org
  - Tutorial: www.stat.wisc.edu/~yandell/qtlbim/rqtltour.pdf
  - R code: www.stat.wisc.edu/~yandell/qtlbim/rqtltour.R

R/qtl tutorial (www.rqtl.org)

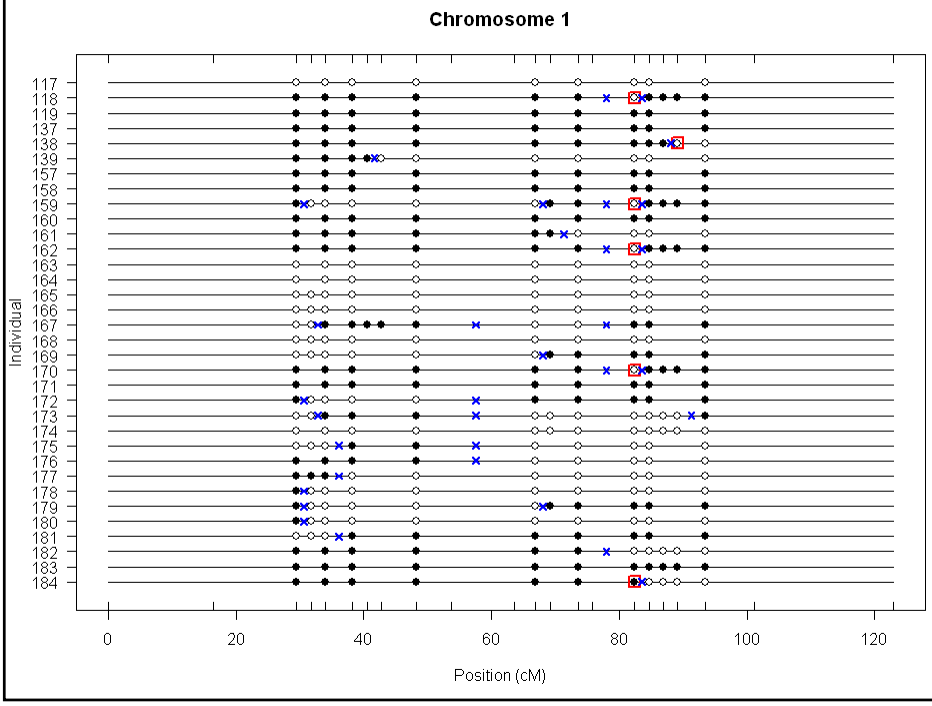
```
> library(qtl)
> data(hyper)
> summary(hyper)
    Backcross

    No. individuals:    250

    No. phenotypes:     2
    Percent phenotyped: 100 100

    No. chromosomes:    20
        Autosomes:      1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
        X chr:          X

    Total markers:      174
    No. markers:        22 8 6 20 14 11 7 6 5 5 14 5 5 5 11 6 12 4 4 4
    Percent genotyped:  47.7
    Genotypes (%):      AA:50.2  AB:49.8
> plot(hyper)
> plot.missing(hyper, reorder = TRUE)
```


Missing genotypes
Individuals
Markers
Genetic map
Location (cM)
Chromosome
bp
Frequency
phe 1
sex
female
male
phe 2


Missing genotypes
Individuals
Markers

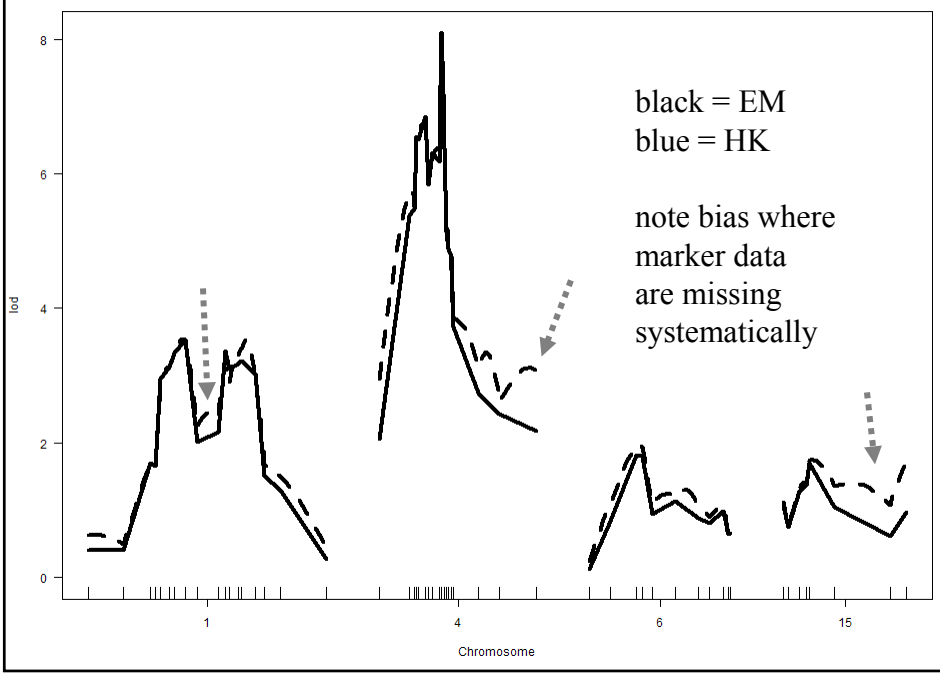
R/qtl: find genotyping errors

```
> hyper <- calc.errorlod(hyper, error.prob=0.01)
> top.errorlod(hyper)

   chr  id    marker errorlod
1    1 118   D1Mit14 8.372794
2    1 162   D1Mit14 8.372794
3    1 170   D1Mit14 8.372794
4    1 159   D1Mit14 8.350341
5    1  73   D1Mit14 6.165395
6    1  65   D1Mit14 6.165395
7    1  88   D1Mit14 6.165395
8    1 184   D1Mit14 6.151606
9    1 241   D1Mit14 6.151606
...
16   1 215  D1Mit267 5.822192
17   1 108  D1Mit267 5.822192
18   1 138  D1Mit267 5.822192
19   1 226  D1Mit267 5.822192
20   1 199  D1Mit267 5.819250
21   1  84  D1Mit267 5.808400

> plot.geno(hyper, chr=1, ind=c(117:119,137:139,157:184))
```

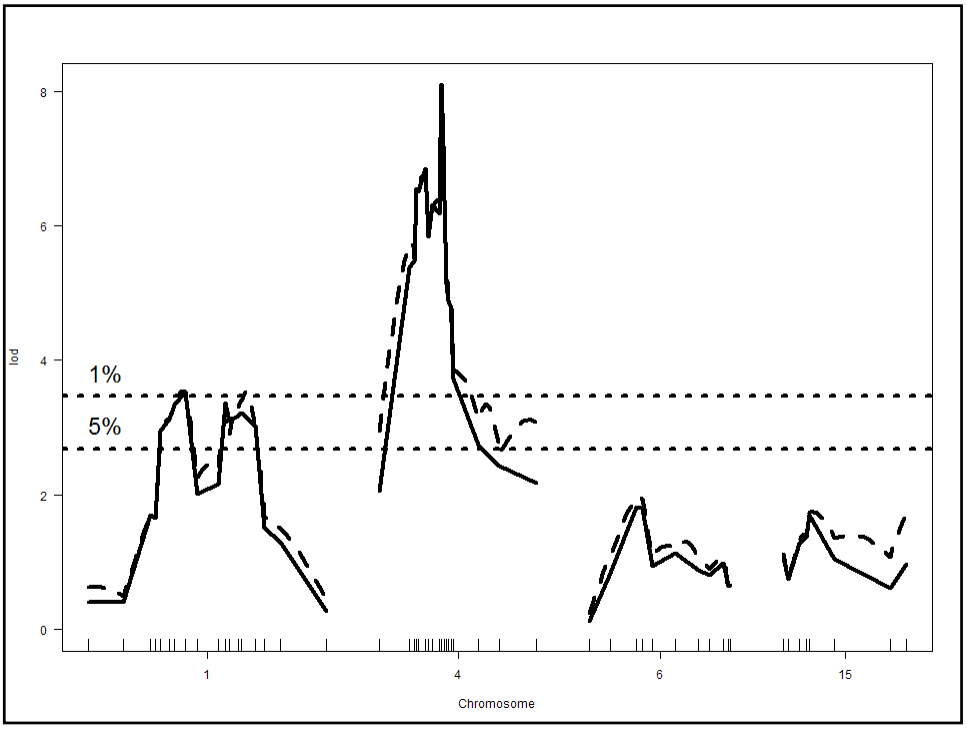
R/qtl: 1 QTL interval mapping

```
> hyper <- calc.genoprob(hyper, step=1,
   error.prob=0.01)
> out.em <- scanone(hyper)
> out.hk <- scanone(hyper, method="hk")
> summary(out.em, threshold=3)
          chr  pos  lod
c1.loc45    1 48.3 3.52
D4Mit164    4 29.5 8.02

> summary(out.hk, threshold=3)
          chr  pos  lod
c1.loc45    1 48.3 3.55
D4Mit164    4 29.5 8.09

> plot(out.em, chr = c(1,4,6,15))
> plot(out.hk, chr = c(1,4,6,15), add = TRUE, lty = 2)
```

R/qtl: permutation threshold

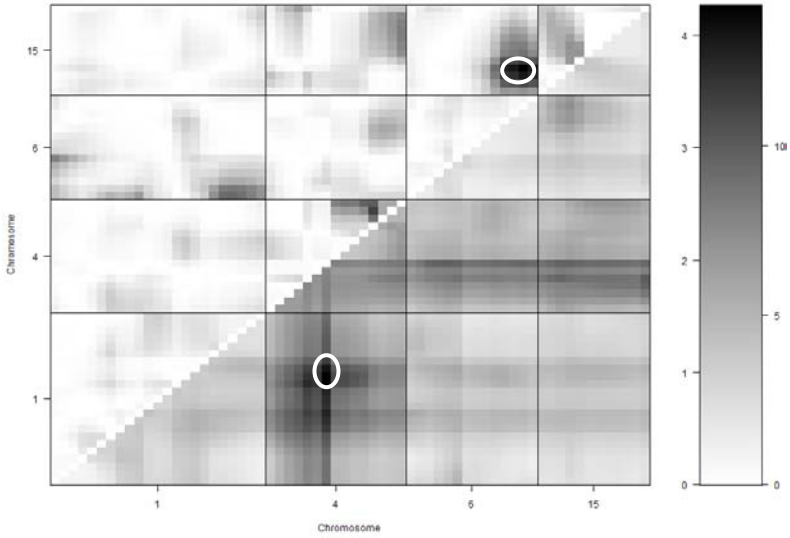
```
> operm.hk <- scanone(hyper, method="hk",
  n.perm=1000)
Doing permutation in batch mode ...
> summary(operm.hk, alpha=c(0.01,0.05))
LOD thresholds (1000 permutations)
    lod
1% 3.79
5% 2.78

> summary(out.hk, perms=operm.hk, alpha=0.05,
  pvalues=TRUE)
  chr  pos  lod  pval
1   1 48.3 3.55 0.015
2   4 29.5 8.09 0.000
```

R/qtl: 2 QTL scan

```
> hyper <- calc.genoprob(hyper, step=5, error.prob=0.01)
>
> out2.hk <- scantwo(hyper, method="hk")
 --Running scanone
 --Running scantwo
 (1,1)
 (1,2)
...
 (19,19)
 (19,X)
 (X,X)
> summary(out2.hk, thresholds=c(6.0, 4.7, 4.4, 4.7, 2.6))

        pos1f pos2f lod.full lod.fv1 lod.int     pos1a pos2a lod.add lod.av1
c1 :c4   68.3  30.0    14.13    6.51   0.225      68.3  30.0   13.90   6.288
c2 :c19  47.7   0.0     6.71    5.01   3.458      52.7   0.0    3.25   1.552
c3 :c3   37.2  42.2     6.10    5.08   0.226      37.2  42.2    5.87   4.853
c6 :c15  60.0  20.5     7.17    5.22   3.237      25.0  20.5    3.93   1.984
c9 :c18  67.0  37.2     6.31    4.79   4.083      67.0  12.2    2.23   0.708
c12:c19   1.1  40.0     6.48    4.79   4.090       1.1   0.0    2.39   0.697

> plot(out2.hk, chr=c(1,4,6,15))
```


upper triangle/left scale: epistasis LOD
lower triangle/right scale: 2-QTL LOD



R/qtl: ANOVA imputation at QTL

```
> hyper <- sim.geno(hyper, step=2, n.draws=16, error.prob=0.01)
> qtl <- makeqtl(hyper, chr = c(1, 1, 4, 6, 15), pos = c(50, 76, 30, 70, 20))

> my.formula <- y ~ Q1 + Q2 + Q3 + Q4 + Q5 + Q4:Q5
> out.fitqtl <- fitqtl(hyper, pheno.col = 1, qtl, formula = my.formula)
> summary(out.fitqtl)

Full model result
----------------------------------
Model formula is:  y ~ Q1 + Q2 + Q3 + Q4 + Q5 + Q4:Q5

       df        SS        MS      LOD     %var Pvalue(Chi2) Pvalue(F)
Model   6  5789.089 964.84822 21.54994 32.76422            0         0
Error 243 11879.847  48.88826
Total 249 17668.936

Drop one QTL at a time ANOVA table:
----------------------------------
                 df Type III SS      LOD     %var F value Pvalue(F)
Chr1@50           1     297.149    1.341    1.682   6.078   0.01438 *
Chr1@76           1     520.664    2.329    2.947  10.650   0.00126 **
Chr4@30           1    2842.089   11.644   16.085  58.134  5.50e-13 ***
Chr6@70           2    1435.721    6.194    8.126  14.684  9.55e-07 ***
Chr15@20          2    1083.842    4.740    6.134  11.085  2.47e-05 ***
Chr6@70:Chr15@20  1     955.268    4.199    5.406  19.540  1.49e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


selected R/qtl publications

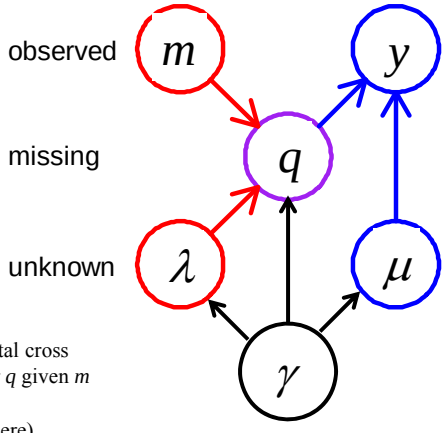
www.stat.wisc.edu/~yandell/statgen

- www.rqtl.org
- tutorials and code at web site
  - www.rqtl.org/tutorials
- Broman et al. (2003 *Bioinformatics*)
  - R/qtl introduction
- Broman (2001 *Lab Animal*)
  - nice overview of QTL issues

Bayesian Interval Mapping

QTL model selection: key players

- observed measurements
  - y = phenotypic trait
  - m = markers & linkage map
  - i = individual index (1,…,n)
- missing data
  - missing marker data
  - q = QT genotypes
    - alleles QQ, Qq, or qq at locus
- unknown quantities
  - λ = QT locus (or loci)
  - μ = phenotype model parameters
  - γ = QTL model/genetic architecture
- pr($q|m,\lambda,\gamma$) genotype model
  - grounded by linkage map, experimental cross
  - recombination yields multinomial for q given m
- pr($y|q,\mu,\gamma$) phenotype model
  - distribution shape (assumed normal here)
  - unknown parameters μ (could be non-parametric)



after
Sen Churchill (2001)

1. Bayesian strategy for QTL study

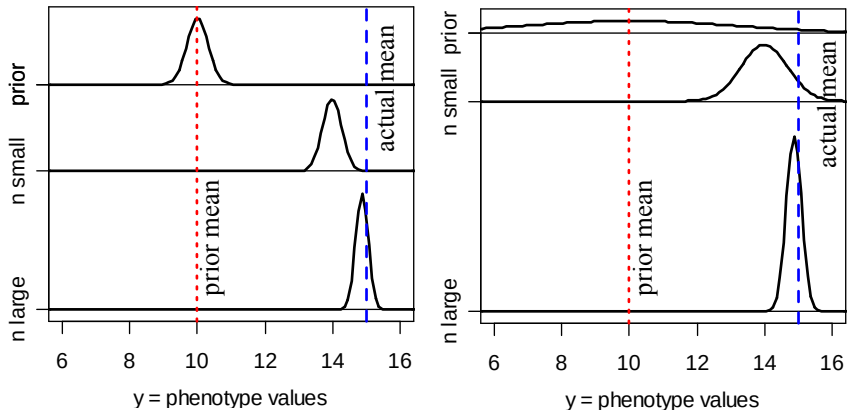
- augment data (y,m) with missing genotypes q
- study unknowns (μ,λ,γ) given augmented data (y,m,q)
  - find better genetic architectures γ
  - find most likely genomic regions = QTL = λ
  - estimate phenotype parameters = genotype means = μ
- sample from posterior in some clever way
  - multiple imputation (Sen Churchill 2002)
  - Markov chain Monte Carlo (MCMC)
    - (Satagopan et al. 1996; Yi et al. 2005, 2007)

$$\text{posterior} = \frac{\text{likelihood} * \text{prior}}{\text{constant}}$$

$$\text{posterior for } q,\mu,\lambda,\gamma = \frac{\text{phenotype likelihood} * [\text{prior for } q,\mu,\lambda,\gamma]}{\text{constant}}$$

$$\text{pr}(q,\mu,\lambda,\gamma \mid y,m) = \frac{\text{pr}(y \mid q,\mu,\gamma) * [\text{pr}(q \mid m,\lambda,\gamma)\text{pr}(\mu \mid \gamma)\text{pr}(\lambda \mid m,\gamma)\text{pr}(\gamma)]}{\text{pr}(y \mid m)}$$

Bayes posterior for normal data



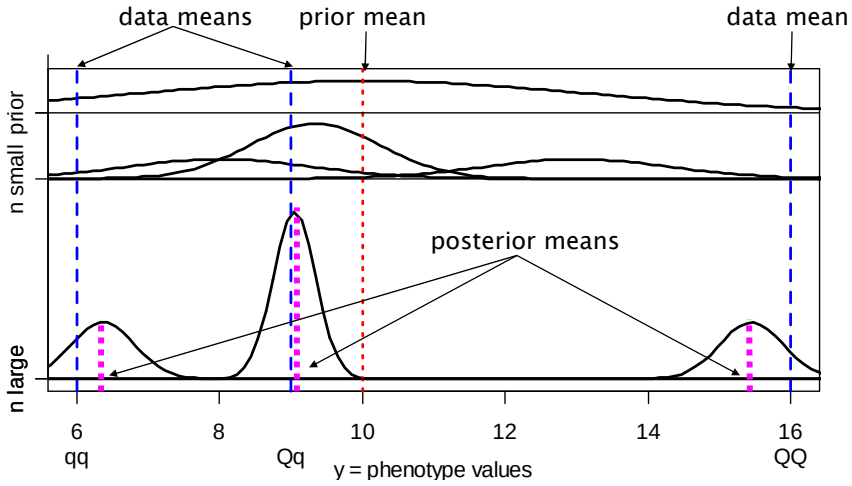
small prior variance large prior variance

Bayes posterior for normal data

model	$y_i = \mu + e_i$
environment	e ~ $N(0, \sigma^2)$, σ^2 known
likelihood	$y \sim N(\mu, \sigma^2)$
prior	$\mu \sim N(\mu_0, \kappa\sigma^2)$, κ known
posterior:	mean tends to sample mean
single individual	$\mu \sim N(\mu_0 + b_1(y_1 - \mu_0), b_1\sigma^2)$
sample of n individuals	$\mu \sim N\left(b_n\bar{y}_\bullet + (1-b_n)\mu_0, b_n\sigma^2/n\right)$ with $\bar{y}_\bullet = \underset{\{i=1,\ldots,n\}}{\text{sum}} y_i/n$
shrinkage factor (shrinks to 1)	$b_n = \frac{\kappa n}{\kappa n + 1} \rightarrow 1$

what values are the genotypic means?

phenotype model pr($y|q,\mu$)



Bayes posterior QTL means

posterior centered on sample genotypic mean
but shrunken slightly toward overall mean

phenotype mean: $E(y \mid q) = \mu_q \qquad V(y \mid q) = \sigma^2$

genotypic prior: $E(\mu_q) = \bar{y}_\bullet \qquad V(\mu_q) = \kappa\sigma^2$

posterior: $E(\mu_q \mid y) = b_q \bar{y}_q + (1 - b_q)\bar{y}_\bullet \qquad V(\mu_q \mid y) = b_q \sigma^2 / n_q$

$n_q = \text{count}\{q_i = q\} \qquad \bar{y}_q = \underset{\{q_i = q\}}{\text{sum}} y_i / n_q$

shrinkage: $b_q = \dfrac{\kappa n_q}{\kappa n_q + 1} \to 1$

partition genotypic effects on phenotype

- phenotype depends on genotype
- genotypic value partitioned into
  - main effects of single QTL
  - epistasis (interaction) between pairs of QTL

$$\mu_q = \beta_0 + \beta_q = E(Y;q)$$
$$\beta_q = \beta(q_2) + \beta(q_2) + \beta(q_1, q_2)$$

partitition genotypic variance

- consider same 2 QTL + epistasis

- centering variance $V(\beta_0) = \kappa_0 \sigma^2 = s^2$

- genotypic variance $V(\beta_q) = \kappa_1 \sigma^2 = \sigma_q^2 = \sigma_1^2 + \sigma_2^2 + \sigma_{12}^2$

- heritability $h_q^2 = \frac{\sigma_q^2}{\sigma_q^2 + \sigma^2} = h_1^2 + h_2^2 + h_{12}^2$

posterior mean ≈ LS estimate

$$\beta_q \mid y \sim N(b_q \hat{\beta}_q, b_q C_q \sigma^2)$$

$$\approx N(\hat{\beta}_q, C_q \sigma^2)$$

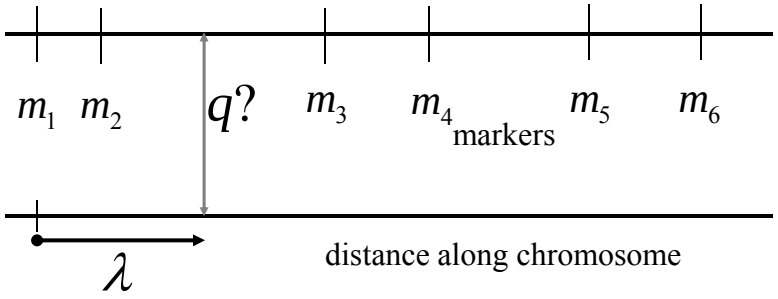
LS estimate $\hat{\beta}_q = \text{sum}_i[\text{sum}_j \hat{\beta}(q_{ij})] = \text{sum}_i w_{qi} y_i$

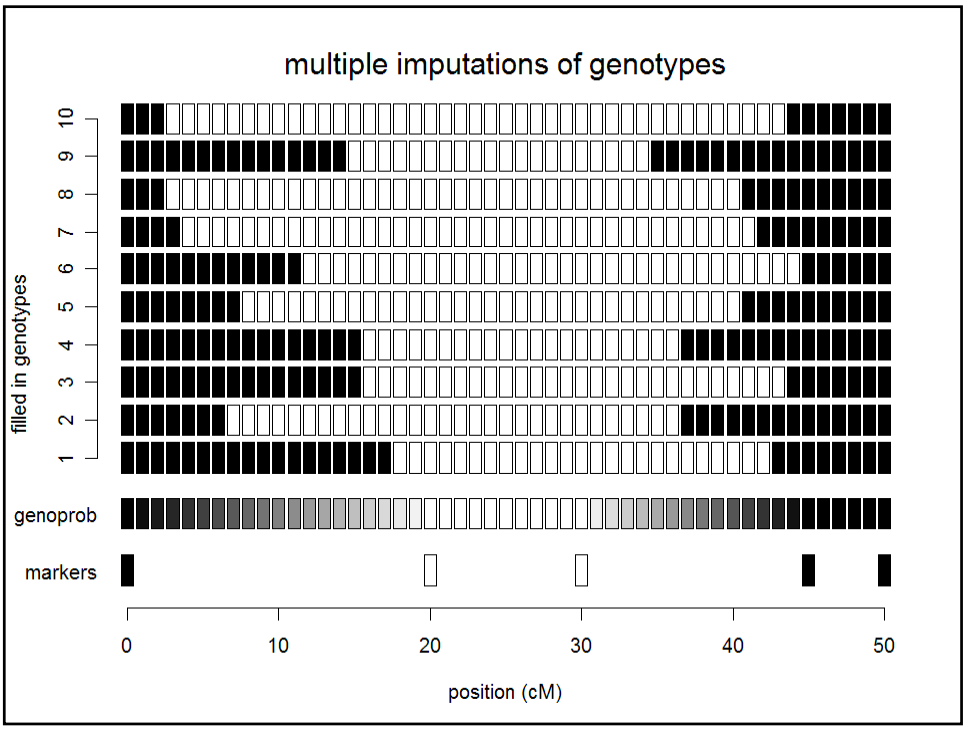
variance $V(\hat{\beta}_q) = \text{sum}_i w_{qi}^2 \sigma^2 = C_q \sigma^2$

shrinkage $b_q = \kappa_1 / (\kappa_1 + C_q) \to 1$

pr($q|m,\lambda$) recombination model

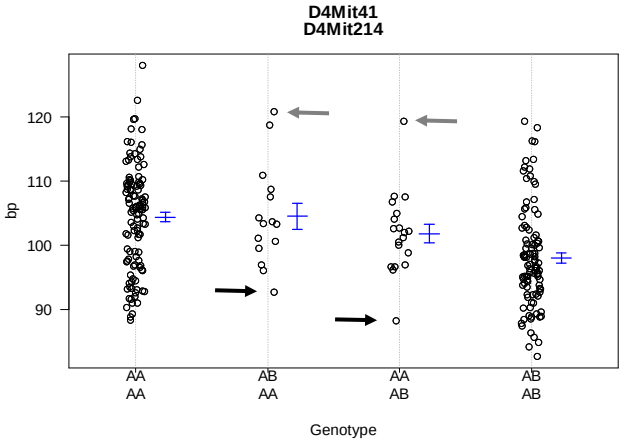
pr($q|m,\lambda$) = pr(geno | map, locus) ≈
pr(geno | flanking markers, locus)





what are likely QTL genotypes *q*?

how does phenotype *y* improve guess?



what are probabilities for genotype *q* between markers?

recombinants AA:AB

all 1:1 if ignore *y* and if we use *y*?

posterior on QTL genotypes *q*

- full conditional of *q* given data, parameters
  - proportional to prior pr($q \mid m, \lambda$)
    - weight toward *q* that agrees with flanking markers
  - proportional to likelihood pr($y \mid q, \mu$)
    - weight toward *q* with similar phenotype values
  - posterior recombination model balances these two
- this *is* the E-step of EM computations

$$\text{pr}(q \mid y, m, \mu, \lambda) = \frac{\text{pr}(y \mid q, \mu) * \text{pr}(q \mid m, \lambda)}{\text{pr}(y \mid m, \mu, \lambda)}$$

Where are the loci λ on the genome?

- prior over genome for QTL positions
  - flat prior = no prior idea of loci
  - or use prior studies to give more weight to some regions
- posterior depends on QTL genotypes q

  $\text{pr}(\lambda \mid m,q) = \text{pr}(\lambda)\ \text{pr}(q \mid m,\lambda) / \text{constant}$
  - constant determined by averaging
    - over all possible genotypes q
    - over all possible loci λ on entire map
- no easy way to write down posterior

what is the genetic architecture γ?

- which positions correspond to QTLs?
  - priors on loci (previous slide)
- which QTL have main effects?
  - priors for presence/absence of main effects
    - same prior for all QTL
    - can put prior on each d.f. (1 for BC, 2 for F2)
- which pairs of QTL have epistatic interactions?
  - prior for presence/absence of epistatic pairs
    - depends on whether 0,1,2 QTL have main effects
    - epistatic effects less probable than main effects

γ = genetic architecture:

loci:

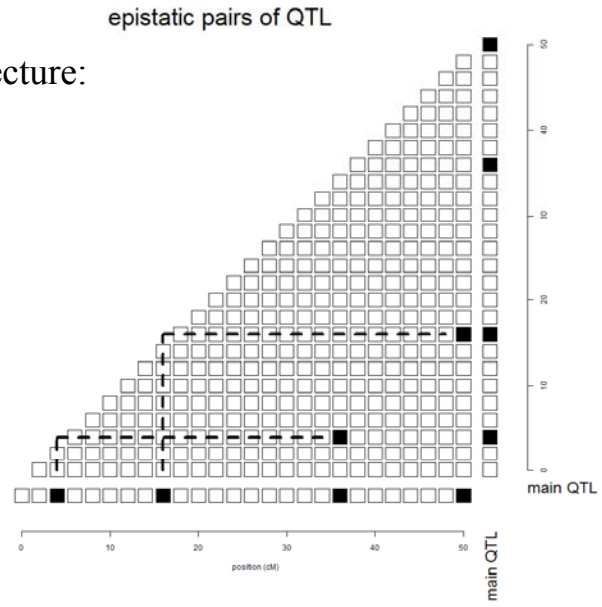
main QTL

epistatic pairs

effects:

add, dom

aa, ad, dd



Bayesian priors & posteriors

- augmenting with missing genotypes q
  - prior is recombination model
  - posterior is (formally) E step of EM algorithm
- sampling phenotype model parameters μ
  - prior is "flat" normal at grand mean (no information)
  - posterior shrinks genotypic means toward grand mean
  - (details for unexplained variance omitted here)
- sampling QTL loci λ
  - prior is flat across genome (all loci equally likely)
- sampling QTL genetic architecture model γ
  - number of QTL
    - prior is Poisson with mean from previous IM study
  - genetic architecture of main effects and epistatic interactions
    - priors on epistasis depend on presence/absence of main effects

2. Markov chain sampling

- construct Markov chain around posterior
  - want posterior as stable distribution of Markov chain
  - in practice, the chain tends toward stable distribution
    - initial values may have low posterior probability
    - burn-in period to get chain mixing well
- sample QTL model components from full conditionals
  - sample locus λ given q,γ (using Metropolis-Hastings step)
  - sample genotypes q given λ,μ,y,γ (using Gibbs sampler)
  - sample effects μ given q,y,γ (using Gibbs sampler)
  - sample QTL model γ given λ,μ,y,q (using Gibbs or M-H)

$$(\lambda,q,\mu,\gamma) \sim \text{pr}(\lambda,q,\mu,\gamma \mid y,m)$$

$$(\lambda,q,\mu,\gamma)_1 \rightarrow (\lambda,q,\mu,\gamma)_2 \rightarrow \cdots \rightarrow (\lambda,q,\mu,\gamma)_N$$

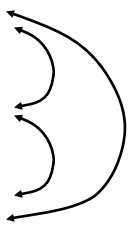
MCMC sampling of unknowns (q,μ,λ)
for given genetic architecture γ

- Gibbs sampler
  - genotypes q
  - effects μ
  - *not* loci λ

$$q \sim \text{pr}(q \mid y_i, m_i, \mu, \lambda)$$

$$\mu \sim \frac{\text{pr}(y \mid q,\mu)\text{pr}(\mu)}{\text{pr}(y \mid q)}$$

$$\lambda \sim \frac{\text{pr}(q \mid m,\lambda)\text{pr}(\lambda \mid m)}{\text{pr}(q \mid m)}$$



- Metropolis-Hastings sampler
  - extension of Gibbs sampler
  - does not require normalization
    - pr(q | m) = sum$_\lambda$ pr(q | m, λ) pr(λ)

Gibbs sampler
for two genotypic means

- want to study two correlated effects
  - could sample directly from their bivariate distribution
  - assume correlation ρ is known
- instead use Gibbs sampler:
  - sample each effect from its full conditional given the other
  - pick order of sampling at random
  - repeat many times

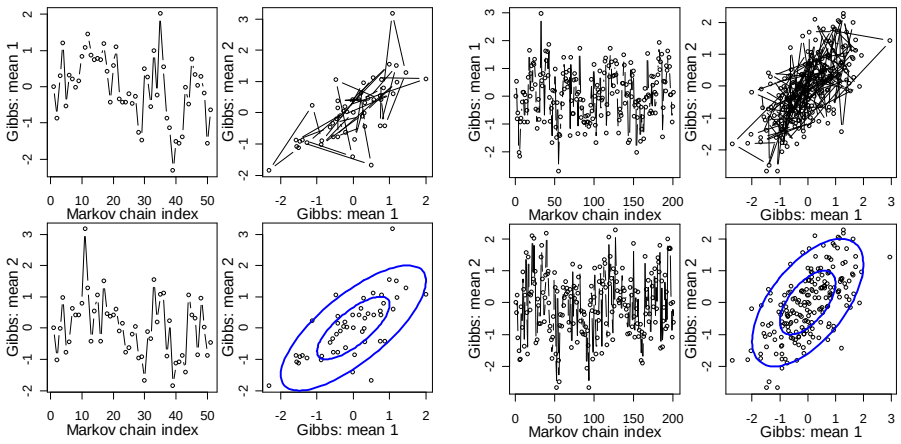
$$\begin{pmatrix}\mu_1\\ \mu_2\end{pmatrix} \sim N\left(\begin{pmatrix}0\\ 0\end{pmatrix}, \begin{pmatrix}1 & \rho\\ \rho & 1\end{pmatrix}\right)$$

$$\mu_1 \sim N\left(\rho\mu_2, 1-\rho^2\right)$$

$$\mu_2 \sim N\left(\rho\mu_1, 1-\rho^2\right)$$

Gibbs sampler samples: $\rho = 0.6$

$N = 50$ samples $N = 200$ samples



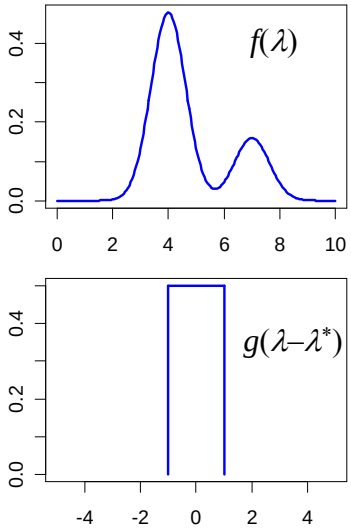
full conditional for locus

- cannot easily sample from locus full conditional
  $\text{pr}(\lambda \mid y,m,\mu,q) = \text{pr}(\lambda \mid m,q)$
  $= \text{pr}(q \mid m, \lambda)\ \text{pr}(\lambda) / \text{constant}$
- constant is very difficult to compute explicitly
  - must average over all possible loci λ over genome
  - must do this for every possible genotype q
- Gibbs sampler will not work in general
  - but can use method based on ratios of probabilities
  - Metropolis-Hastings is extension of Gibbs sampler

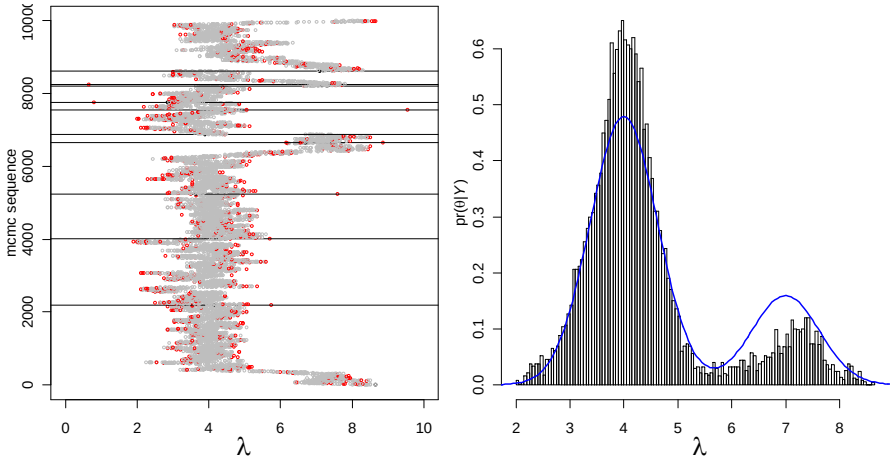
Metropolis-Hastings idea

- want to study distribution $f(\lambda)$
  - take Monte Carlo samples
    - unless too complicated
  - take samples using ratios of f
- Metropolis-Hastings samples:
  - propose new value λ^*
    - near (?) current value λ
    - from some distribution g
  - accept new value with prob a
    - Gibbs sampler: $a = 1$ always

$$a = \min\left(1, \frac{f(\lambda^*)g(\lambda^* - \lambda)}{f(\lambda)g(\lambda - \lambda^*)}\right)$$

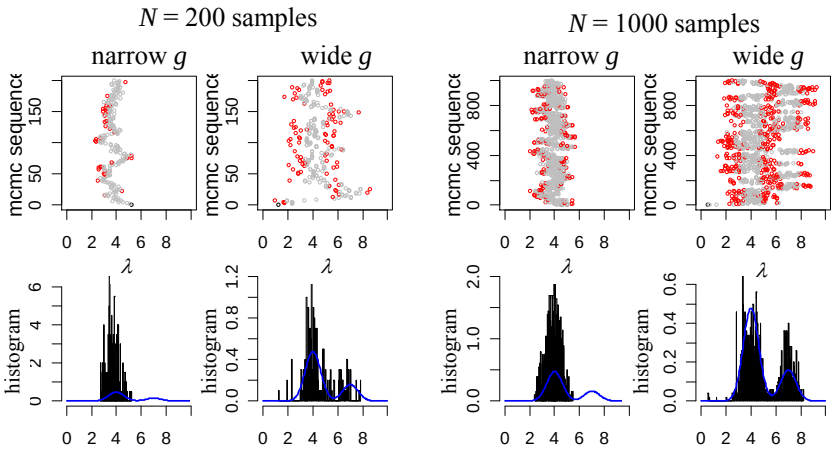


Metropolis-Hastings for locus λ



added twist: occasionally propose from entire genome

Metropolis-Hastings samples

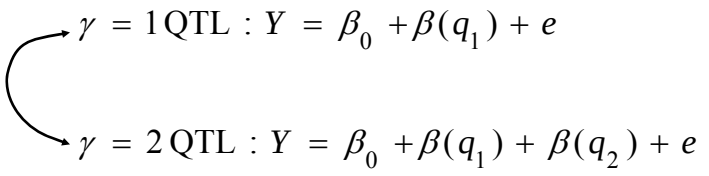


3. sampling genetic architectures

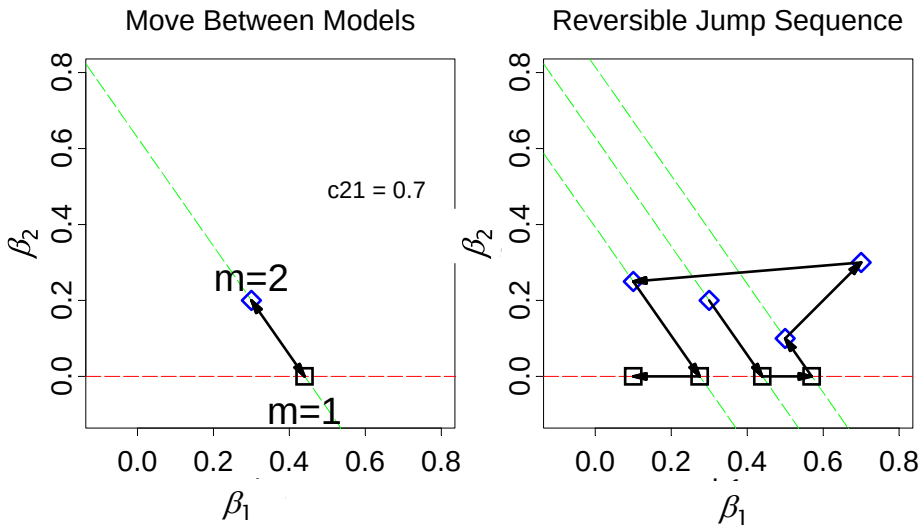
- search across genetic architectures *A* of various sizes
  - allow change in number of QTL
  - allow change in types of epistatic interactions
- methods for search
  - reversible jump MCMC
  - Gibbs sampler with loci indicators
- complexity of epistasis
  - Fisher-Cockerham effects model
  - general multi-QTL interaction & limits of inference

reversible jump MCMC

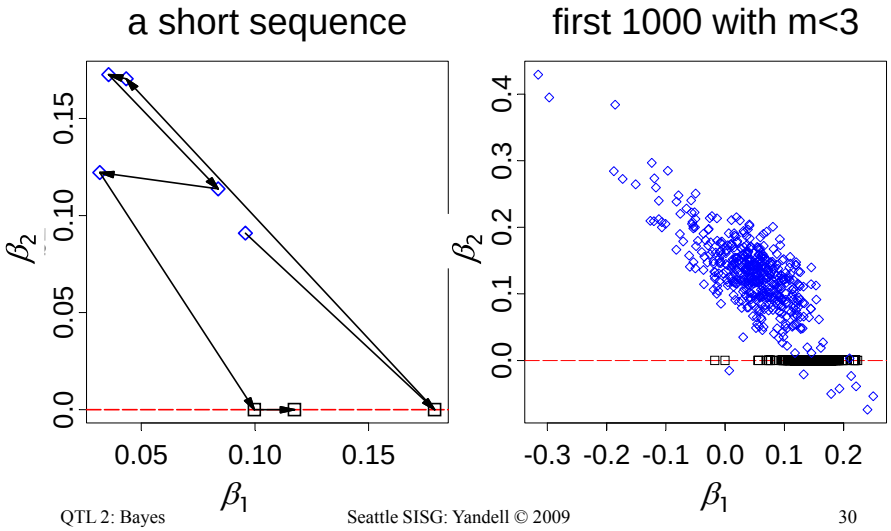
- consider known genotypes q at 2 known loci λ
  - models with 1 or 2 QTL
- M-H step between 1-QTL and 2-QTL models
  - model changes dimension (via careful bookkeeping)
  - consider mixture over QTL models *H*



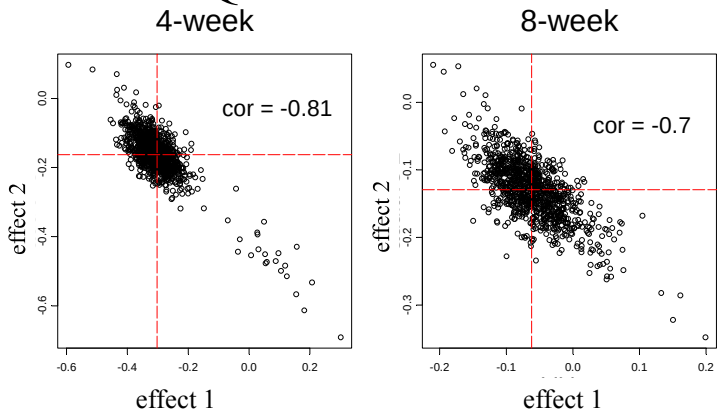
geometry of reversible jump



geometry allowing q and λ to change

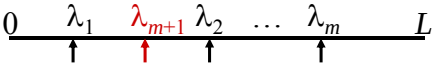


collinear QTL = correlated effects



- linked QTL = collinear genotypes
  - ➢ correlated estimates of effects (negative if in coupling phase)
  - ➢ sum of linked effects usually fairly constant

sampling across QTL models γ



action steps: draw one of three choices

- update QTL model γ with probability 1-$b(\gamma)$-$d(\gamma)$
  - update current model using full conditionals
  - sample QTL loci, effects, and genotypes
- add a locus with probability $b(\gamma)$
  - propose a new locus along genome
  - innovate new genotypes at locus and phenotype effect
  - decide whether to accept the "birth" of new locus
- drop a locus with probability $d(\gamma)$
  - propose dropping one of existing loci
  - decide whether to accept the "death" of locus

Gibbs sampler with loci indicators

- consider only QTL at pseudomarkers
  - every 1-2 cM
  - modest approximation with little bias
- use loci indicators in each pseudomarker
  - $\gamma = 1$ if QTL present
  - $\gamma = 0$ if no QTL present
- Gibbs sampler on loci indicators γ
  - relatively easy to incorporate epistasis
  - Yi, Yandell, Churchill, Allison, Eisen, Pomp (2005 *Genetics*)
    - (see earlier work of Nengjun Yi and Ina Hoeschele)

$$\mu_q = \mu + \gamma_1 \beta(q_1) + \gamma_2 \beta(q_2), \quad \gamma_k = 0,1$$

Bayesian shrinkage estimation

- soft loci indicators
  - strength of evidence for λ_j depends on γ
  - $0 \le \gamma \le 1$ (grey scale)
  - shrink most γs to zero
- Wang et al. (2005 *Genetics*)
  - Shizhong Xu group at U CA Riverside

$$\mu_q = \beta_0 + \gamma_1 \beta_1(q_1) + \gamma_2 \beta_2(q_1), \quad 0 \le \gamma_k \le 1$$

4. criteria for model selection

balance fit against complexity

- classical information criteria
  - penalize likelihood L by model size $|\gamma|$
  - IC = $-2 \log L(\gamma \mid y)$ + penalty(γ)
  - maximize over unknowns
- Bayes factors
  - marginal posteriors pr($y \mid \gamma$)
  - average over unknowns

classical information criteria

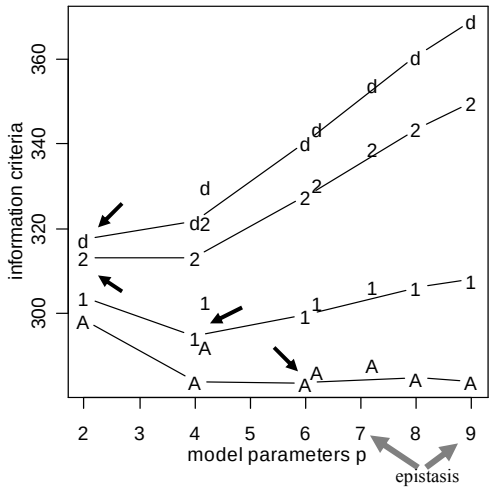
- start with likelihood $L(\gamma \mid y, m)$
  - measures fit of architecture (γ) to phenotype (y)
    - given marker data (m)
  - genetic architecture (γ) depends on parameters
    - have to estimate loci (μ) and effects (λ)
- complexity related to number of parameters
  - $|\gamma|$ = size of genetic architecture
    - BC: $|\gamma| = 1 + n.qtl + n.qtl(n.qtl - 1) = 1 + 4 + 12 = 17$
    - F2: $|\gamma| = 1 + 2n.qtl + 4n.qtl(n.qtl - 1) = 1 + 8 + 48 = 57$

classical information criteria

- construct information criteria
  - balance fit to complexity
  - Akaike $\text{AIC} = -2\log(L) + 2|\gamma|$
  - Bayes/Schwartz $\text{BIC} = -2\log(L) + |\gamma|\log(n)$
  - Broman $\text{BIC}_\delta = -2\log(L) + \delta|\gamma|\log(n)$
  - general form: $\text{IC} = -2\log(L) + |\gamma| D(n)$
- compare models
  - hypothesis testing: designed for one comparison
    - $2\log[LR(\gamma_1, \gamma_2)] = L(y|m, \gamma_2) - L(y|m, \gamma_1)$
  - model selection: penalize complexity
    - $\text{IC}(\gamma_1, \gamma_2) = 2\log[LR(\gamma_1, \gamma_2)] + (|\gamma_2| - |\gamma_1|) D(n)$

information criteria vs. model size

- WinQTL 2.0
- SCD data on F2
- A=AIC
- 1=BIC(1)
- 2=BIC(2)
- d=BIC(δ)
- models
  - 1,2,3,4 QTL
    - 2+5+9+2
  - epistasis
    - 2:2 AD



Bayes factors

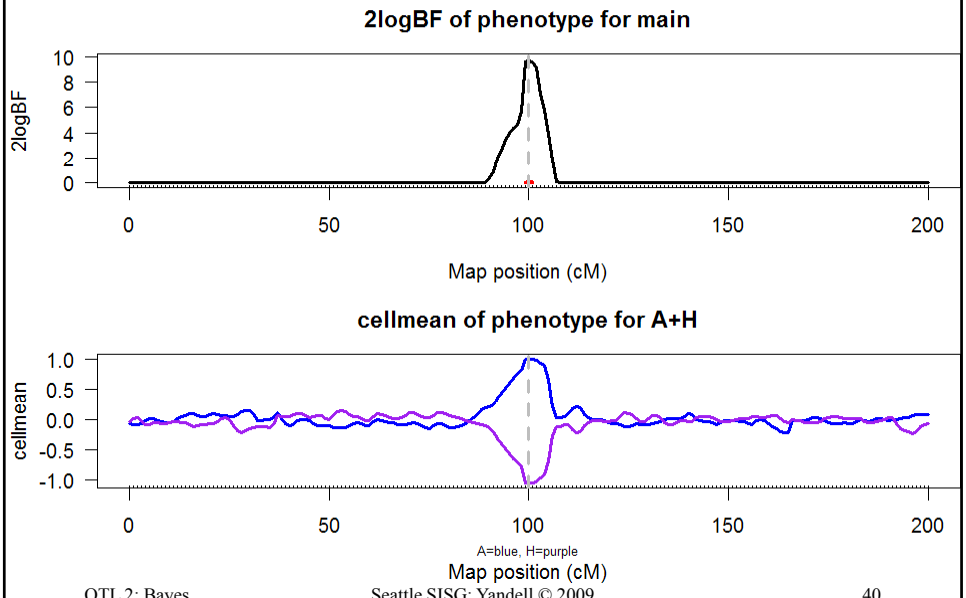
- ratio of model likelihoods
  - ratio of posterior to prior odds for architectures
  - averaged over unknowns

$$B_{12} = \frac{\mathrm{pr}(\gamma_1 \mid y,m)/\mathrm{pr}(\gamma_2 \mid y,m)}{\mathrm{pr}(\gamma_1)/\mathrm{pr}(\gamma_2)} = \frac{\mathrm{pr}(y \mid m,\gamma_1)}{\mathrm{pr}(y \mid m,\gamma_2)}$$

- roughly equivalent to BIC
  - BIC maximizes over unknowns
  - BF averages over unknowns

$$-2\log(B_{12}) = -2\log(LR) - (|\gamma_2| - |\gamma_1|)\log(n)$$

scan of marginal Bayes factor & effect



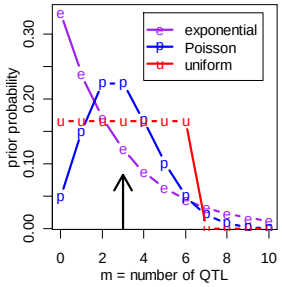
issues in computing Bayes factors

- *BF* insensitive to shape of prior on γ
  - geometric, Poisson, uniform
  - precision improves when prior mimics posterior
- *BF* sensitivity to prior variance on effects θ
  - prior variance should reflect data variability
  - resolved by using hyper-priors
    - automatic algorithm; no need for user tuning
- easy to compute Bayes factors from samples
  - sample posterior using MCMC
  - posterior pr($\gamma \mid y, m$) is marginal histogram

Bayes factors & genetic architecture γ

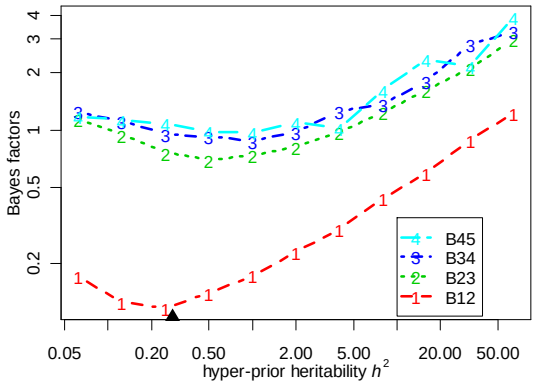
- $|\gamma|$ = number of QTL
  - prior pr(γ) chosen by user
  - posterior pr($\gamma \mid y,m$)
    - sampled marginal histogram
    - shape affected by prior pr(A)

$$BF_{\gamma_1,\gamma_2} = \frac{\Pr(\gamma_1|y,m)/\Pr(\gamma_1)}{\Pr(\gamma_2|y,m)/\Pr(\gamma_2)}$$



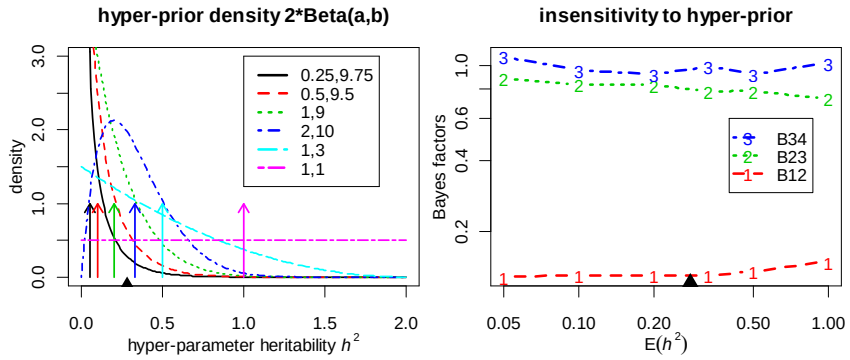
- pattern of QTL across genome
- gene action and epistasis

BF sensitivity to fixed prior for effects



$$\beta_{qj} \sim \mathrm{N}\left(0, \sigma_G^2 / m\right), \sigma_G^2 = h^2 \sigma_{\text{total}}^2, h^2 \text{ fixed}$$

BF insensitivity to random effects prior



$$\beta_{qj} \sim \mathrm{N}\left(0, \sigma_G^2 / m\right), \sigma_G^2 = h^2 \sigma_{\text{total}}^2, \tfrac{1}{2} h^2 \sim \mathrm{Beta}(a, b)$$

R/qtlbim (www.qtlbim.org)

- cross-compatible with R/qtl
- model selection for genetic architecture
  - epistasis, fixed & random covariates, GxE
  - samples multiple genetic architectures
  - examines summaries over nested models
- extensive graphics

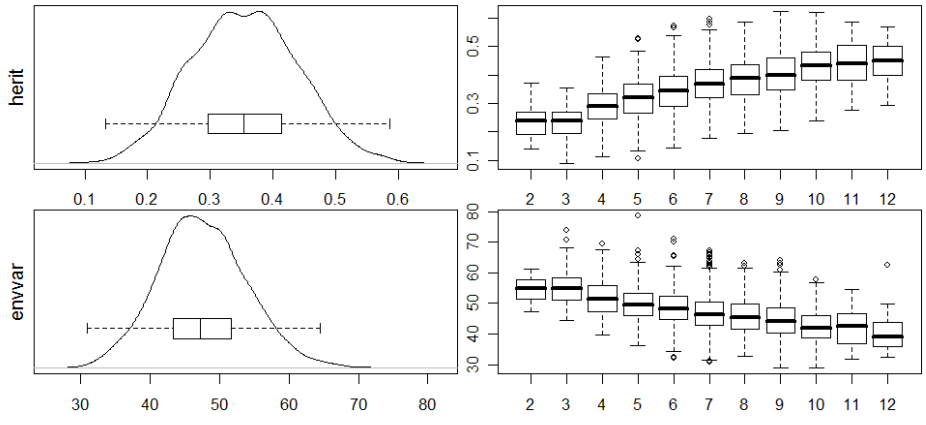
R/qtlbim: tutorial
(www.stat.wisc.edu/~yandell/qtlbim)

```
> data(hyper)
## Drop X chromosome (for now).
> hyper <- subset(hyper, chr=1:19)
> hyper <- qb.genoprob(hyper, step=2)
## This is the time-consuming step:
> qbHyper <- qb.mcmc(hyper, pheno.col = 1)
## Here we get stored samples.
> qb.load(hyper, qbHyper)
> summary(qbHyper)
```

R/qtlbim: initial summaries

```
> summary(qbHyper)

Bayesian model selection QTL mapping object qbHyper on cross object hyper
had 3000 iterations recorded at each 40 steps with 1200 burn-in steps.

Diagnostic summaries:
          nqtl   mean envvar varadd  varaa    var
Min.     2.000  97.42  28.07  5.112  0.000  5.112
1st Qu.  5.000 101.00  44.33 17.010  1.639 20.180
Median   7.000 101.30  48.57 20.060  4.580 25.160
Mean     6.543 101.30  48.80 20.310  5.321 25.630
3rd Qu.  8.000 101.70  53.11 23.480  7.862 30.370
Max.    13.000 103.90  74.03 51.730 34.940 65.220

Percentages for number of QTL detected:
 2  3  4  5  6  7  8  9 10 11 12 13
 2  3  9 14 21 19 17 10  4  1  0  0

Percentages for number of epistatic pairs detected:
pairs
 1  2  3  4  5  6
29 31 23 11  5  1

Percentages for common epistatic pairs:
 6.15  4.15   4.6   1.7 15.15   1.4   1.6   4.9  1.15  1.17   1.5  5.11   1.2  7.15   1.1
   63    18    10     6     6     5     4     4     3     3     3     2     2     2     2

> plot(qb.diag(qbHyper, items = c("herit", "envvar")))
```

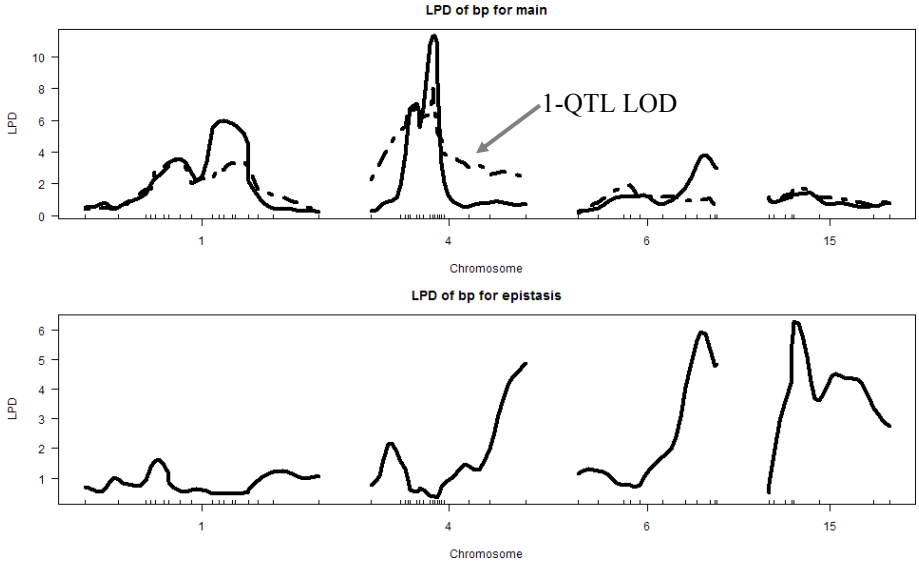

diagnostic summaries



R/qtlbim: 1-D (*not* 1-QTL!) scan

```
> one <- qb.scanone(qbHyper, chr = c(1,4,6,15), type =
   "LPD")
> summary(one)

LPD of bp for main,epistasis,sum

    n.qtl  pos m.pos e.pos  main epistasis   sum
c1  1.331 64.5  64.5  67.8  6.10     0.442  6.27
c4  1.377 29.5  29.5  29.5 11.49     0.375 11.61
c6  0.838 59.0  59.0  59.0  3.99     6.265  9.60
c15 0.961 17.5  17.5  17.5  1.30     6.325  7.28

> plot(one, scan = "main")
> plot(out.em, chr=c(1,4,6,15), add = TRUE, lty = 2)
> plot(one, scan = "epistasis")
```

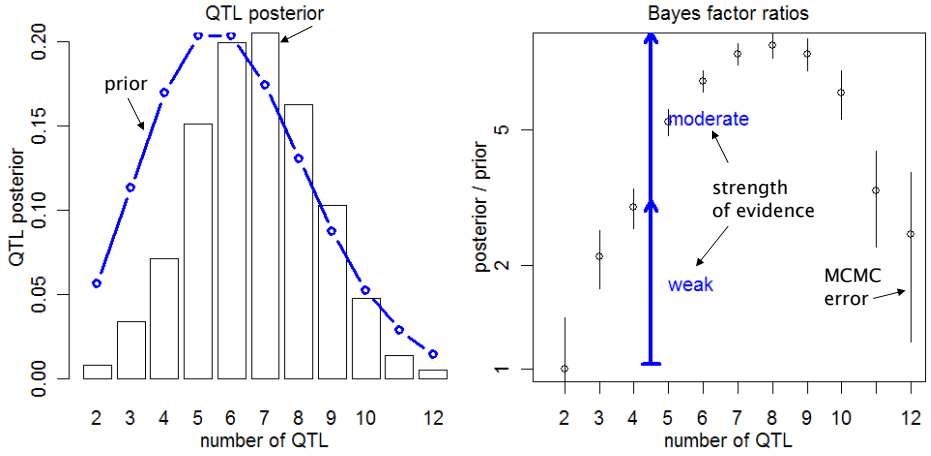

1-QTL LOD vs. marginal LPD



most probable patterns

```
> summary(qb.BayesFactor(qbHyper, item = "pattern"))

                   nqtl posterior    prior    bf  bfse
1,4,6,15,6:15         5   0.03400 2.71e-05 24.30 2.360
1,4,6,6,15,6:15       6   0.00467 5.22e-06 17.40 4.630
1,1,4,6,15,6:15       6   0.00600 9.05e-06 12.80 3.020
1,1,4,5,6,15,6:15     7   0.00267 4.11e-06 12.60 4.450
1,4,6,15,15,6:15      6   0.00300 4.96e-06 11.70 3.910
1,4,4,6,15,6:15       6   0.00300 5.81e-06 10.00 3.330
1,2,4,6,15,6:15       6   0.00767 1.54e-05  9.66 2.010
1,4,5,6,15,6:15       6   0.00500 1.28e-05  7.56 1.950
1,2,4,5,6,15,6:15     7   0.00267 6.98e-06  7.41 2.620
1,4                   2   0.01430 1.51e-04  1.84 0.279
1,1,2,4               4   0.00300 3.66e-05  1.59 0.529
1,2,4                 3   0.00733 1.03e-04  1.38 0.294
1,1,4                 3   0.00400 6.05e-05  1.28 0.370
1,4,19                3   0.00300 5.82e-05  1.00 0.333

> plot(qb.BayesFactor(qbHyper, item = "nqtl"))
```


hyper: number of QTL posterior, prior, Bayes factors

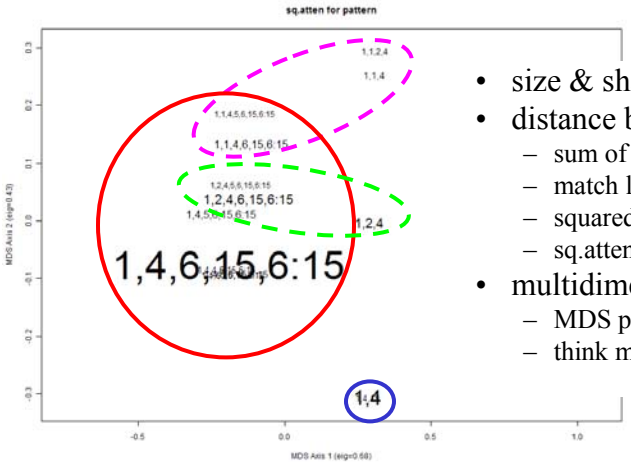


what is best estimate of QTL?

- find most probable pattern
  - 1,4,6,15,6:15 has posterior of 3.4%
- estimate locus across all nested patterns
  - Exact pattern seen ~100/3000 samples
  - Nested pattern seen ~2000/3000 samples
- estimate 95% confidence interval using quantiles

```
> best <- qb.best(qbHyper)
> summary(best)$best

    chrom locus locus.LCL locus.UCL     n.qtl
247     1  69.9  24.44875   95.7985 0.8026667
245     4  29.5  14.20000   74.3000 0.8800000
248     6  59.0  13.83333   66.7000 0.7096667
246    15  19.5  13.10000   55.7000 0.8450000

> plot(best)
```


what patterns are "near" the best?

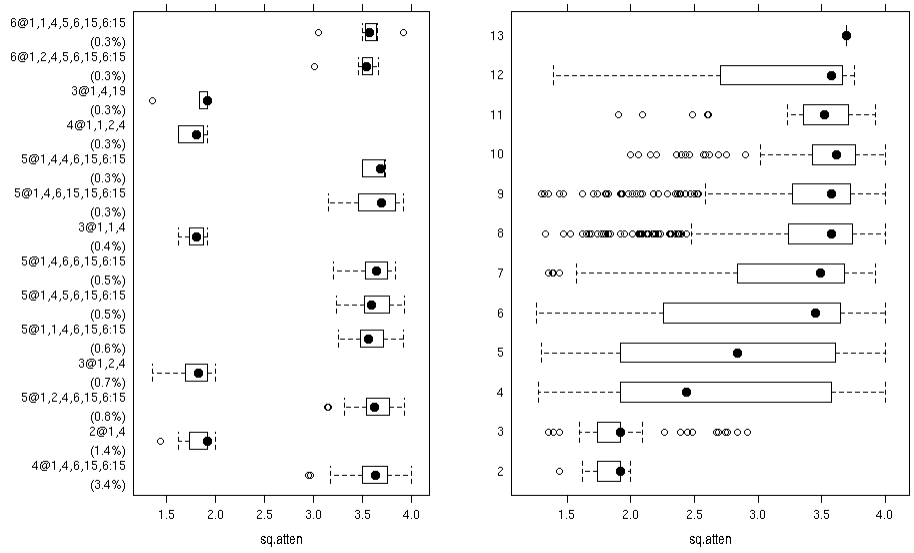


- size & shade ~ posterior
- distance between patterns
  - sum of squared attenuation
  - match loci between patterns
  - squared attenuation = $(1\text{-}2r)^2$
  - sq.atten in scale of LOD & LPD
- multidimensional scaling
  - MDS projects distance onto 2-D
  - think mileage between cities

how close are other patterns?

```
> target <- qb.best(qbHyper)$model[[1]]
> summary(qb.close(qbHyper, target))

 score by sample number of qtl
    Min. 1st Qu. Median  Mean 3rd Qu.  Max.
2  1.437   1.735  1.919 1.834   1.919 2.000
3  1.351   1.735  1.916 1.900   1.919 2.916
4  1.270   1.916  2.437 2.648   3.574 4.000
5  1.295   1.919  2.835 2.798   3.611 4.000
6  1.257   2.254  3.451 3.029   3.648 4.000
...
13 3.694   3.694  3.694 3.694   3.694 3.694

 score by sample chromosome pattern
                     Percent  Min. 1st Qu. Median  Mean 3rd Qu.  Max.
4@1,4,6,15,6:15          3.4 2.946   3.500  3.630 3.613   3.758 4.000
2@1,4                    1.4 1.437   1.735  1.919 1.832   1.919 2.000
5@1,2,4,6,15,6:15        0.8 3.137   3.536  3.622 3.611   3.777 3.923
3@1,2,4                  0.7 1.351   1.700  1.821 1.808   1.919 2.000
5@1,1,4,6,15,6:15        0.6 3.257   3.484  3.563 3.575   3.698 3.916
5@1,4,5,6,15,6:15        0.5 3.237   3.515  3.595 3.622   3.777 3.923
5@1,4,6,6,15,6:15        0.5 3.203   3.541  3.646 3.631   3.757 3.835
...

> plot(close)
> plot(close, category = "nqtl")
```

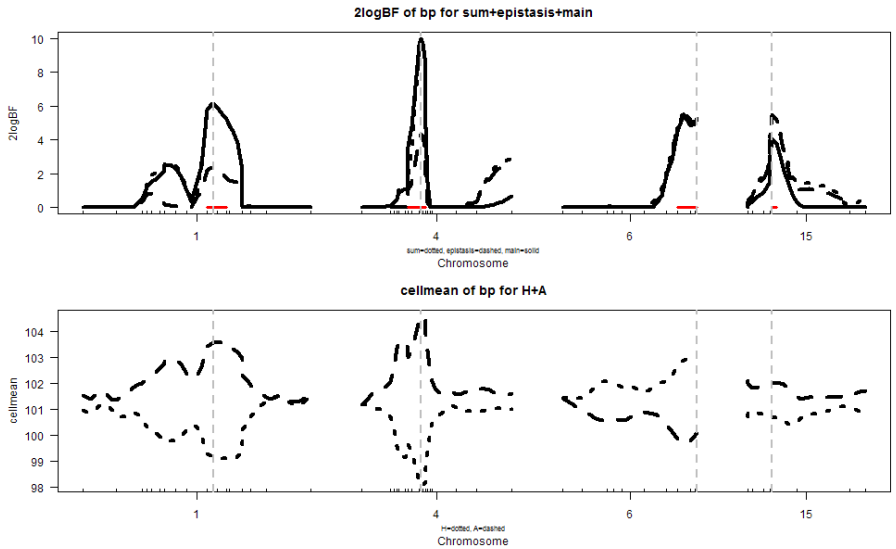

how close are other patterns?



R/qtlbim: automated QTL selection

```
> hpd <- qb.hpdone(qbHyper, profile = "2logBF")
> summary(hpd)

   chr n.qtl  pos lo.50% hi.50% 2logBF       A       H
1    1 0.829 64.5   64.5   72.1  6.692 103.611  99.090
4    4 3.228 29.5   25.1   31.7 11.169 104.584  98.020
6    6 1.033 59.0   56.8   66.7  6.054  99.637 102.965
15  15 0.159 17.5   17.5   17.5  5.837 101.972 100.702

> plot(hpd)
```

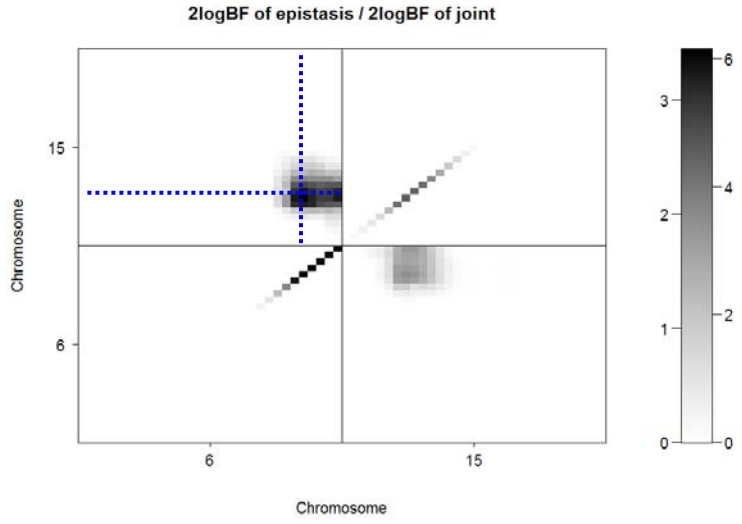

2log(BF) scan with 50% HPD region



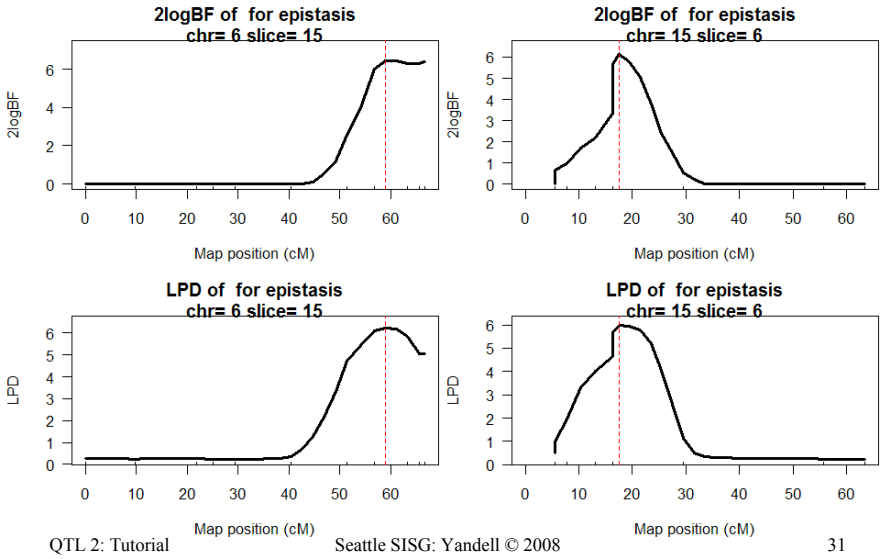
R/qtlbim: 2-D (*not* 2-QTL) scans

```
> two <- qb.scantwo(qbHyper, chr = c(6,15),
  type = "2logBF")
> plot(two)

> plot(two, chr = 6, slice = 15)
> plot(two, chr = 15, slice = 6)

> two.lpd <- qb.scantwo(qbHyper, chr = c(6,15),
  type = "LPD")
> plot(two.lpd, chr = 6, slice = 15)
> plot(two.lpd, chr = 15, slice = 6)
```


2-D plot of 2logBF: chr 6 & 15

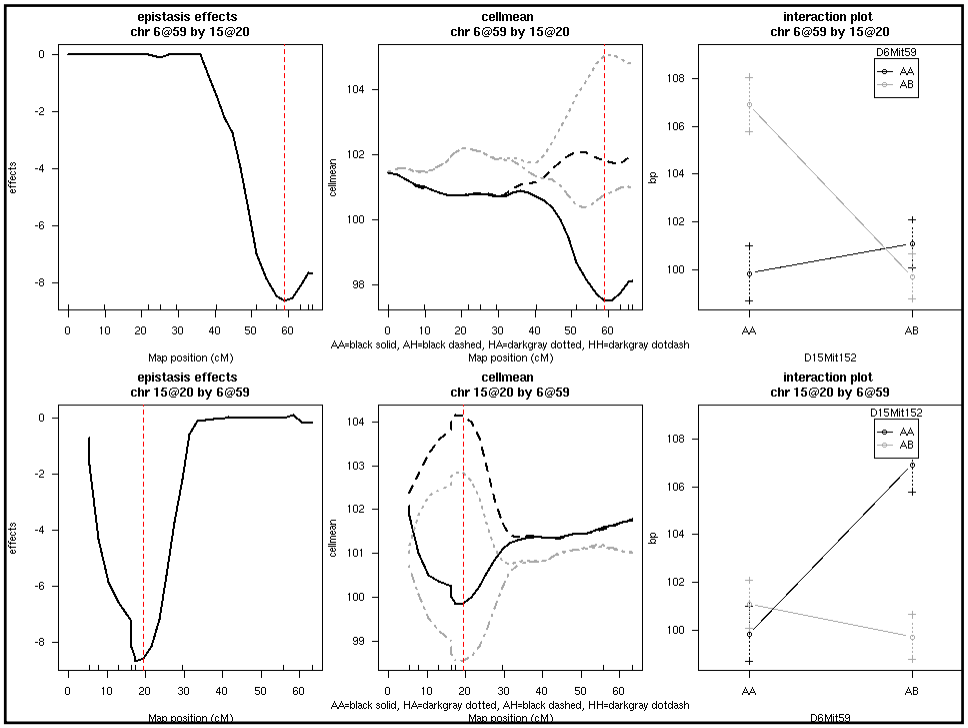


1-D Slices of 2-D scans: chr 6 & 15



R/qtlbim: slice of epistasis

```
> slice <- qb.slicetwo(qbHyper, c(6,15), c(59,19.5))
> summary(slice)

2logBF of bp for epistasis

    n.qtl  pos m.pos e.pos epistasis slice
c6  0.838 59.0  59.0  66.7      15.8  18.1
c15 0.961 17.5  17.5  17.5      15.5  60.6

cellmean of bp for AA,HA,AH,HH

    n.qtl  pos m.pos   AA  HA  AH    HH slice
c6  0.838 59.0  59.0 97.4 105 102 100.8  18.1
c15 0.961 17.5  17.5 99.8 103 104  98.5  60.6

estimate of bp for epistasis

    n.qtl  pos m.pos e.pos epistasis slice
c6  0.838 59.0  59.0  66.7     -7.86  18.1
c15 0.961 17.5  17.5  17.5     -8.72  60.6

> plot(slice, figs = c("effects", "cellmean", "effectplot"))
```



selected publications

www.stat.wisc.edu/~yandell/statgen

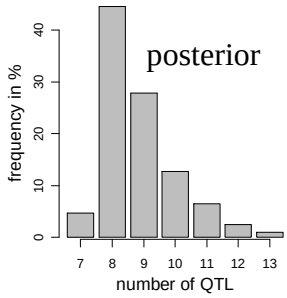
- www.qtlbim.org
- vignettes in R/qtlbim package
- Yandell, Bradbury (2007) *Plant Map* book chapter
  - overview/comparison of QTL methods
- Yandell et al. (2007 *Bioinformatics*)
  - R/qtlbim introduction
- Yi et al. (2005 *Genetics,* 2007 *Genetics*)
  - methodology of R/qtlbim

examples in detail

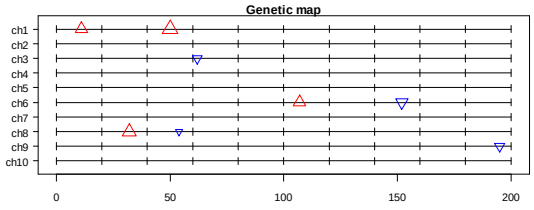
- simulation study (after Stephens & Fisch (1998) 2-3
- obesity in mice (n = 421) 4-12
  - epistatic QTLs with no main effects
- expression phenotype (SCD1) in mice (n = 108) 13-22
  - multiple QTL and epistasis
- mapping two correlated phenotypes 23-35
  - Jiang & Zeng 1995 paper
  - *Brassica napus* vernalization
- gonad shape in *Drosophila* spp. (insect) (n = 1000) 36-42
  - multiple traits reduced by PC
  - many QTL and epistasis

simulation with 8 QTL

- simulated F2 intercross, 8 QTL
  - (Stephens, Fisch 1998)
  - n=200, heritability = 50%
  - detected 3 QTL
- increase to detect all 8
  - n=500, heritability to 97%



QTL	chr	loci	effect
1	1	11	–3
2	1	50	–5
3	3	62	+2
4	6	107	–3
5	6	152	+3
6	8	32	–4
7	8	54	+1
8	9	195	+2



loci pattern across genome

- notice which chromosomes have persistent loci
- best pattern found 42% of the time

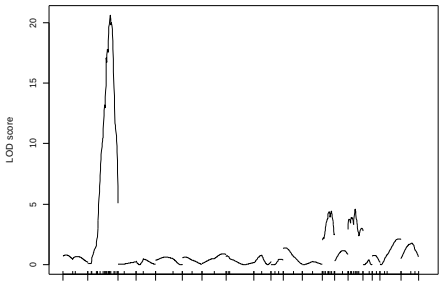
Chromosome

m	**1**	2	**3**	4	5	**6**	7	**8**	**9**	10	**Count of 8000**
8	**2**	**0**	**1**	**0**	**0**	**2**	**0**	**2**	**1**	**0**	3371
9	*3*	0	1	0	0	2	0	2	1	0	751
7	2	0	1	0	0	2	0	*1*	1	0	377
9	2	0	1	0	0	2	0	2	1	0	218
9	2	0	1	0	0	*3*	0	2	1	0	218
9	2	0	1	0	0	2	0	2	*2*	0	198

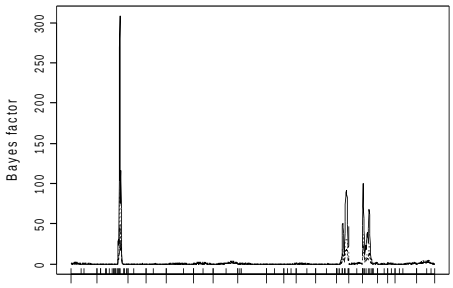
obesity in CAST/Ei BC onto M16i

- 421 mice (Daniel Pomp)
  - (213 male, 208 female)
- 92 microsatellites on 19 chromosomes
  - 1214 cM map
- subcutaneous fat pads
  - pre-adjusted for sex and dam effects
- Yi, Yandell, Churchill, Allison, Eisen, Pomp (2005) *Genetics*

non-epistatic analysis

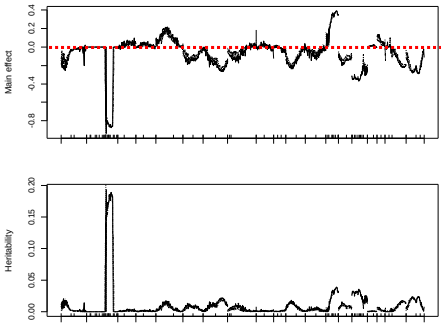


single QTL LOD profile

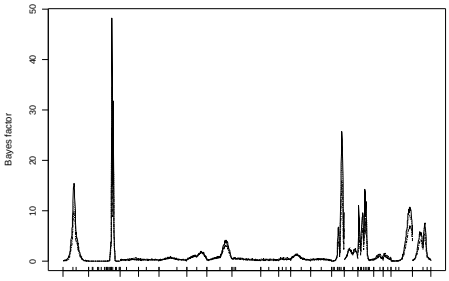


multiple QTL
Bayes factor profile

posterior profile of main effects in epistatic analysis

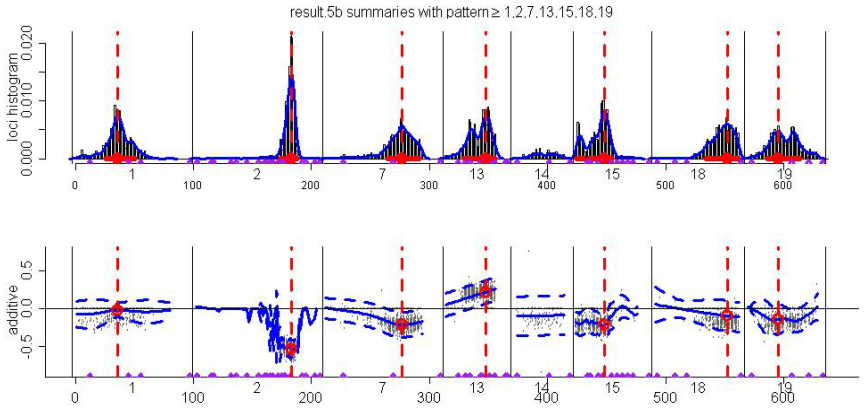


main effects & heritability profile



Bayes factor profile

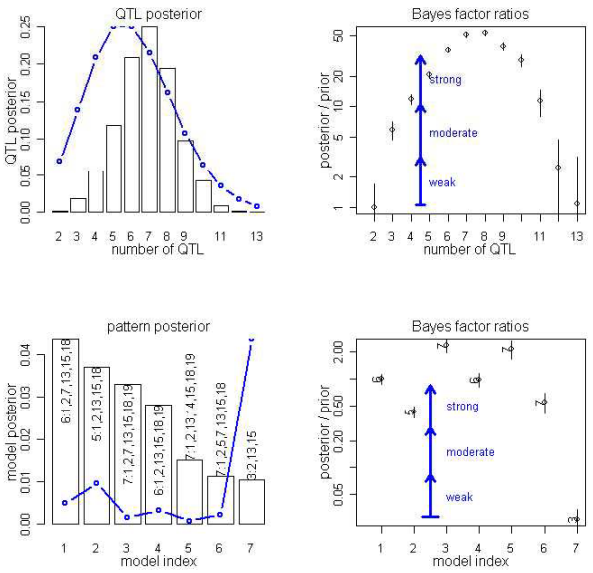
posterior profile of main effects in epistatic analysis



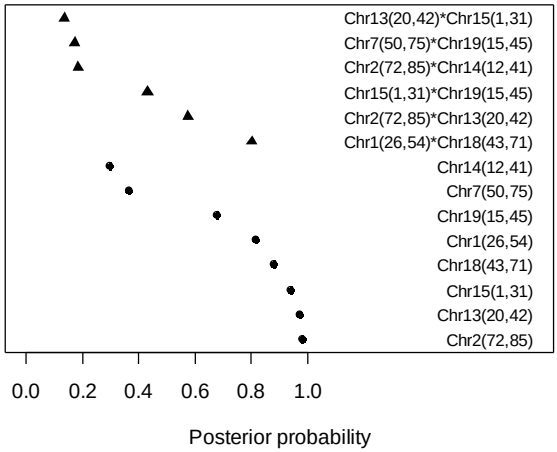
model selection via Bayes factors for epistatic model

number of QTL

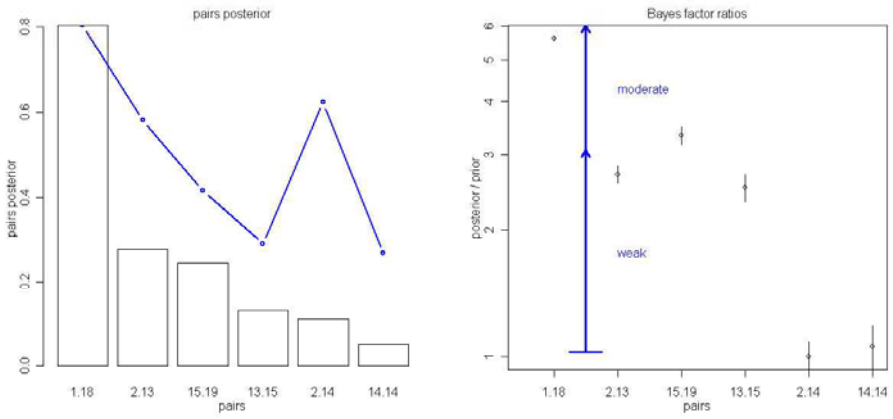
QTL pattern



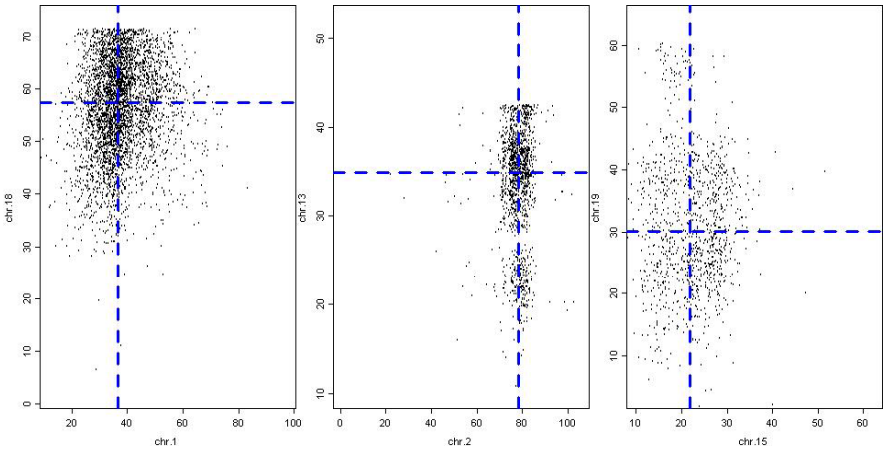
posterior probability of effects



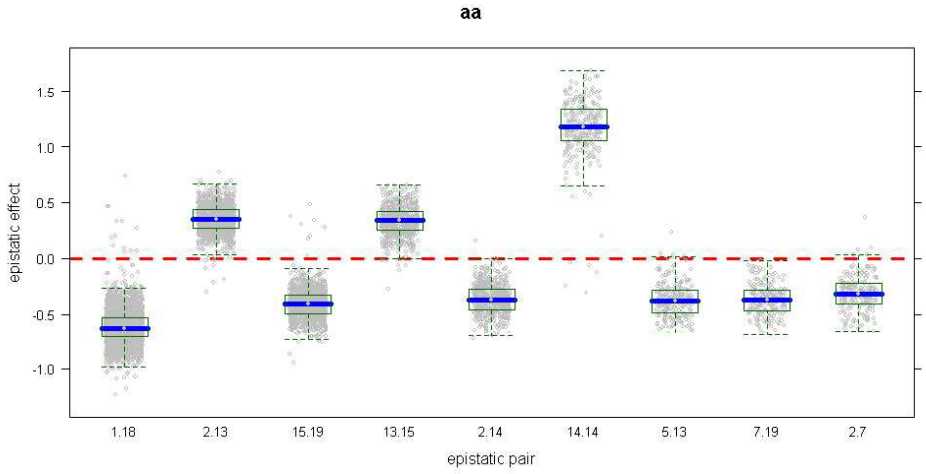
model selection for pairs



scatterplot estimates of epistatic loci



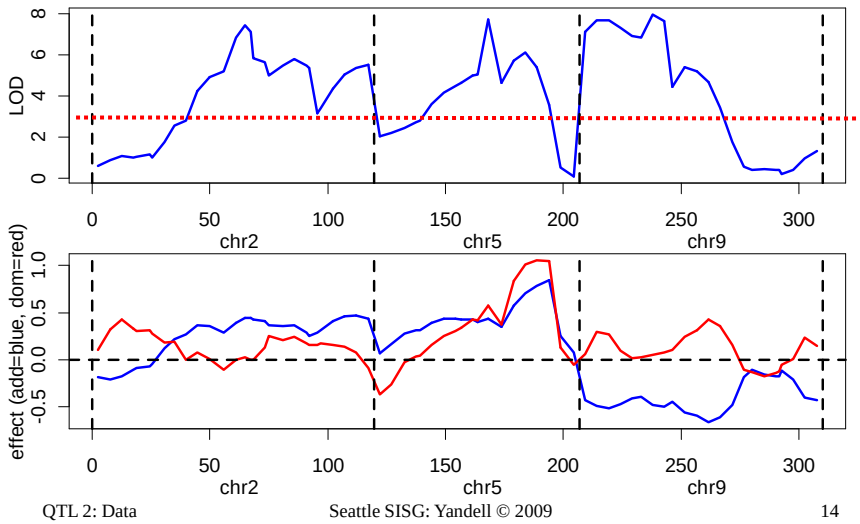
stronger epistatic effects



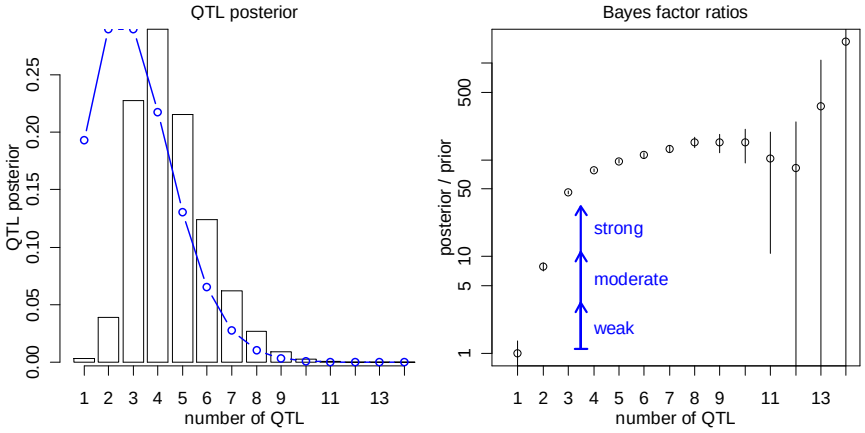
studying diabetes in an F2

- segregating cross of inbred lines
  - B6.ob x BTBR.ob → F1 → F2
  - selected mice with ob/ob alleles at leptin gene (chr 6)
  - measured and mapped body weight, insulin, glucose at various ages (Stoehr et al. 2000 Diabetes)
  - sacrificed at 14 weeks, tissues preserved
- gene expression data
  - Affymetrix microarrays on parental strains, F1
    - key tissues: adipose, liver, muscle, β-cells
    - novel discoveries of differential expression (Nadler et al. 2000 PNAS; Lan et al. 2002 in review; Ntambi et al. 2002 PNAS)
  - RT-PCR on 108 F2 mice liver tissues
    - 15 genes, selected as important in diabetes pathways
    - SCD1, PEPCK, ACO, FAS, GPAT, PPARgamma, PPARalpha, G6Pase, PDI,…

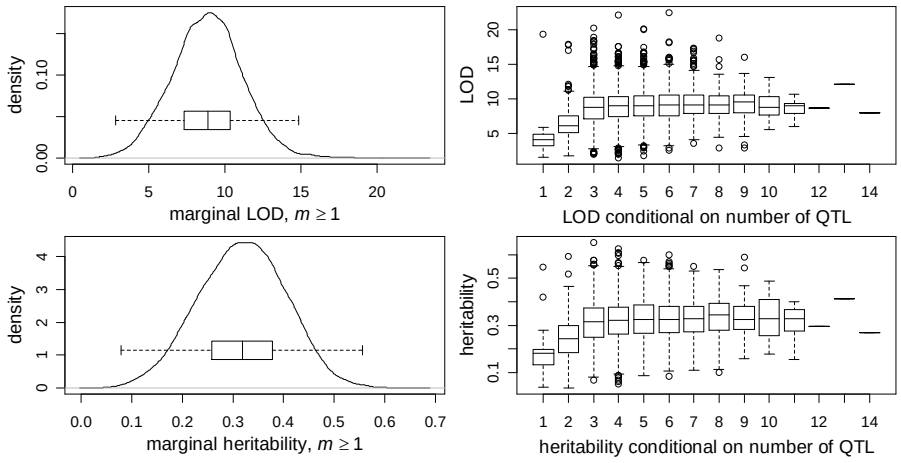
Multiple Interval Mapping (QTLCart)
SCD1: multiple QTL plus epistasis!



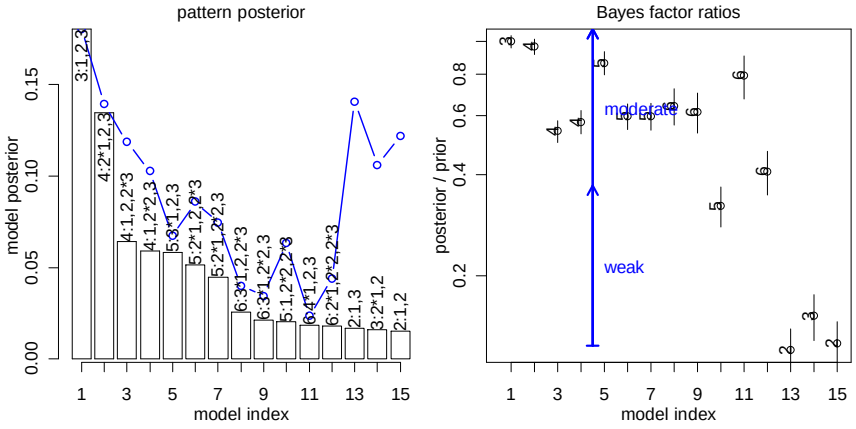
Bayesian model assessment: number of QTL for SCD1



Bayesian LOD and h^2 for SCD1

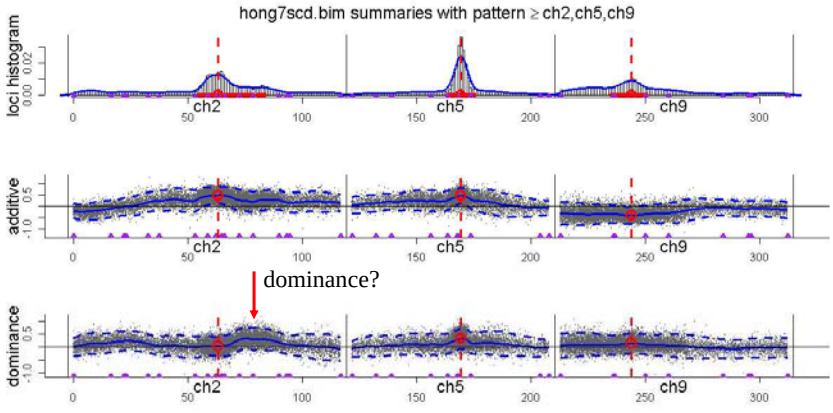


Bayesian model assessment: chromosome QTL pattern for SCD1

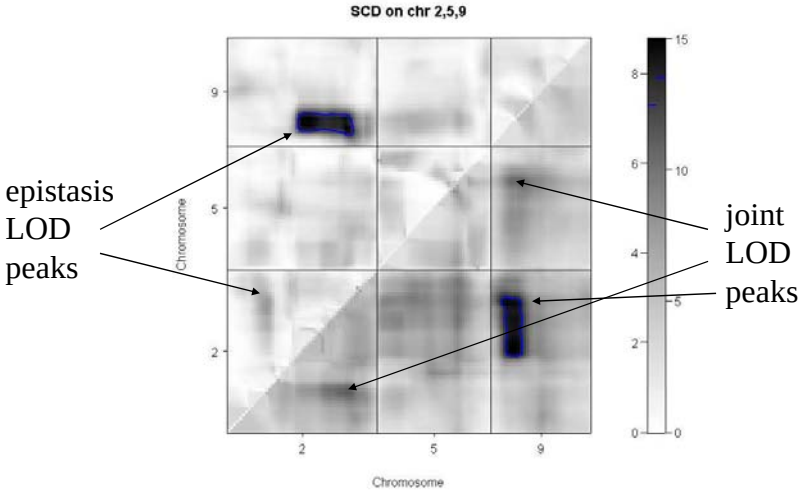


trans-acting QTL for SCD1

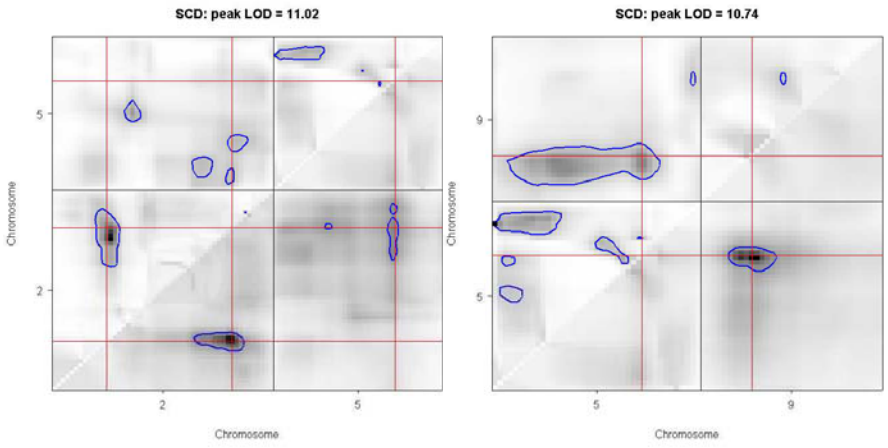
(no epistasis yet: see Yi, Xu, Allison 2003)



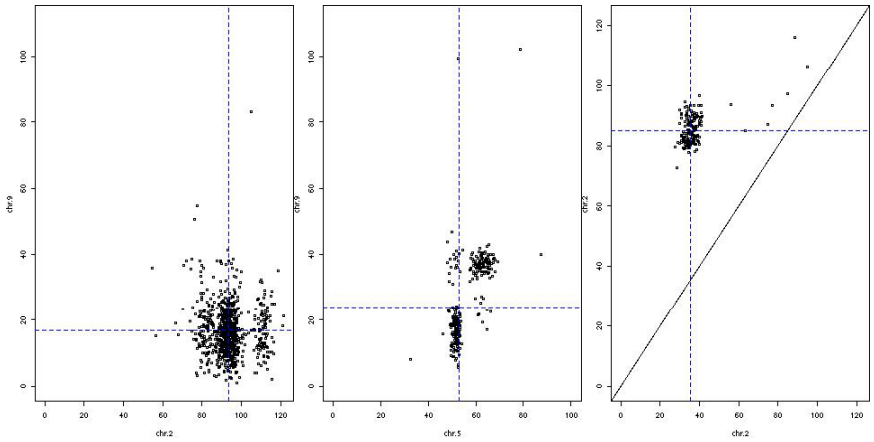
2-D scan: assumes only 2 QTL!



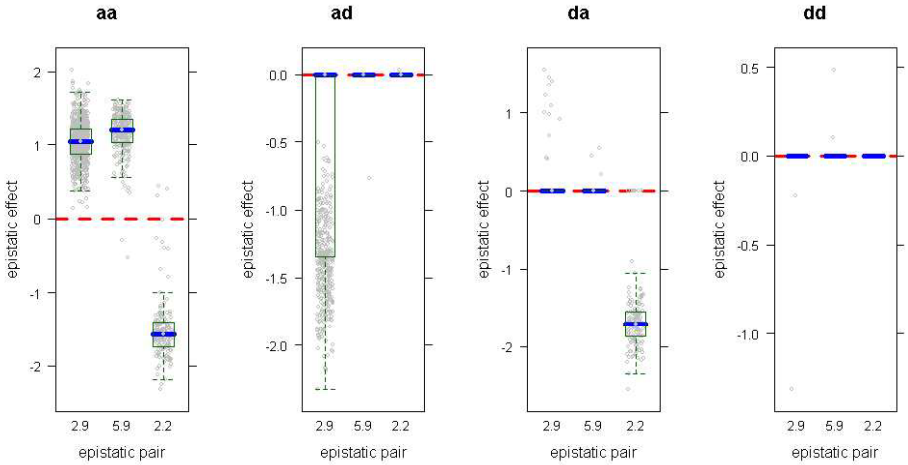
sub-peaks can be easily overlooked!



epistatic model fit



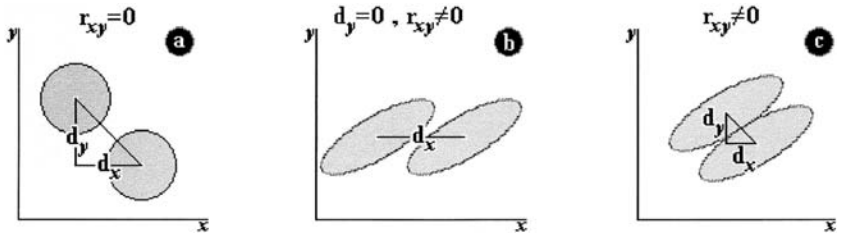
Cockerham epistatic effects



co-mapping multiple traits

- avoid reductionist approach to biology
  - address physiological/biochemical mechanisms
  - Schmalhausen (1942); Falconer (1952)
- separate close linkage from pleiotropy
  - 1 locus or 2 linked loci?
- identify epistatic interaction or canalization
  - influence of genetic background
- establish QTL x environment interactions
- decompose genetic correlation among traits
- increase power to detect QTL

interplay of pleiotropy & correlation



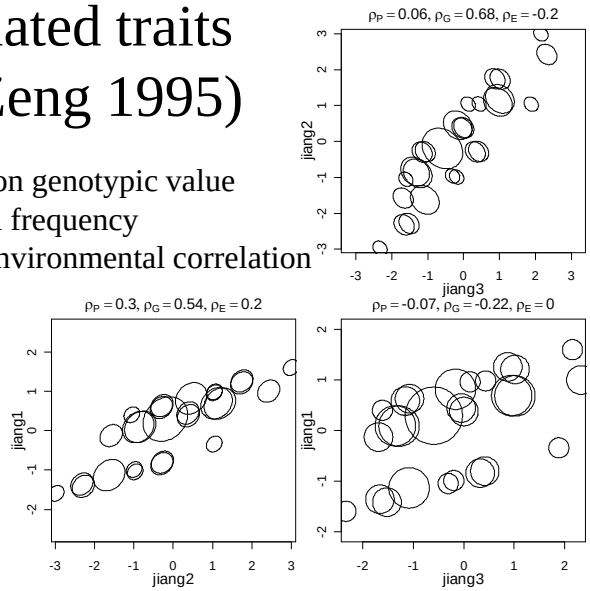
pleiotropy only correlation only both

Korol et al. (2001)

3 correlated traits (Jiang Zeng 1995)

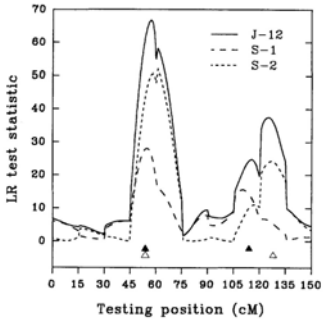
ellipses centered on genotypic value
width for nominal frequency
main axis angle environmental correlation
3 QTL, F2
27 genotypes

note signs of genetic and environmental correlation



pleiotropy or close linkage?

2 traits, 2 qtl/trait
pleiotropy @ 54cM
linkage @ 114,128cM
Jiang Zeng (1995)



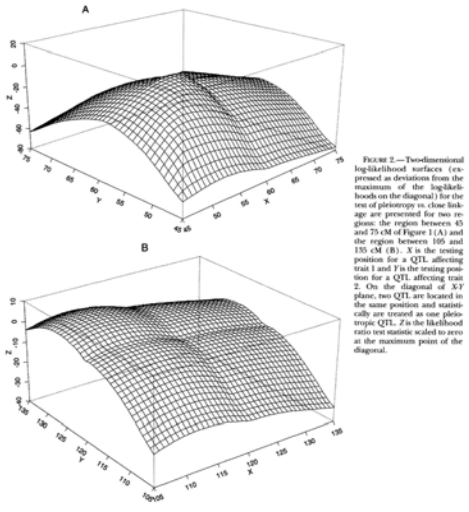


FIGURE 2.—Two-dimensional log-likelihood surfaces (expressed as deviations from the maximum of the log-likelihoods on the diagonal) for the test of pleiotropy vs. close linkage are presented for two regions: the region between 45 and 75 cM of Figure 1(A) and the region between 105 and 135 cM (B). X is the testing position for a QTL affecting trait 1 and Y is the testing position for a QTL affecting trait 2. On the diagonal of XY plane, two QTL are located in the same position and statistically are treated as one pleiotropic QTL. Z is the likelihood ratio test statistic scaled to zero at the maximum point of the diagonal.

Brassica napus: 2 correlated traits

- 4-week & 8-week vernalization effect
  - log(days to flower)
- genetic cross of
  - Stellar (annual canola)
  - Major (biennial rapeseed)
- 105 F1-derived double haploid (DH) lines
  - homozygous at every locus (*QQ* or *qq*)
- 10 molecular markers (RFLPs) on LG9
  - two QTLs inferred on LG9 (now chromosome N2)
  - corroborated by Butruille (1998)
  - exploiting synteny with *Arabidopsis thaliana*

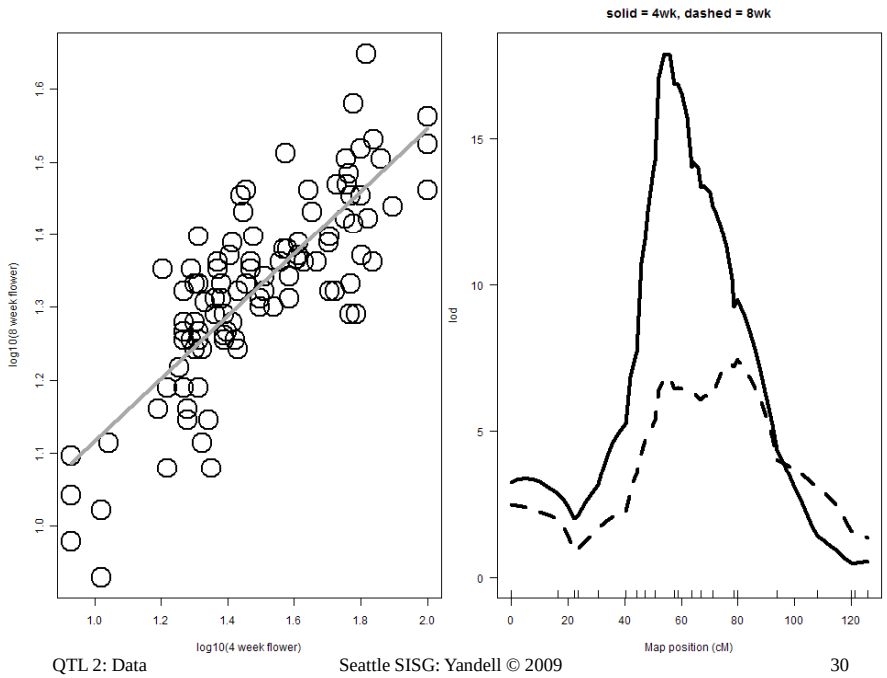
QTL with GxE or Covariates

- adjust phenotype by covariate
  - covariate(s) = environment(s) or other trait(s)
- additive covariate
  - covariate adjustment same across genotypes
  - "usual" analysis of covariance (ANCOVA)
- interacting covariate
  - address GxE
  - capture genotype-specific relationship among traits
- another way to think of multiple trait analysis
  - examine single phenotype adjusted for others

R/qtl & covariates

- additive and/or interacting covariates
- test for QTL after adjusting for covariates

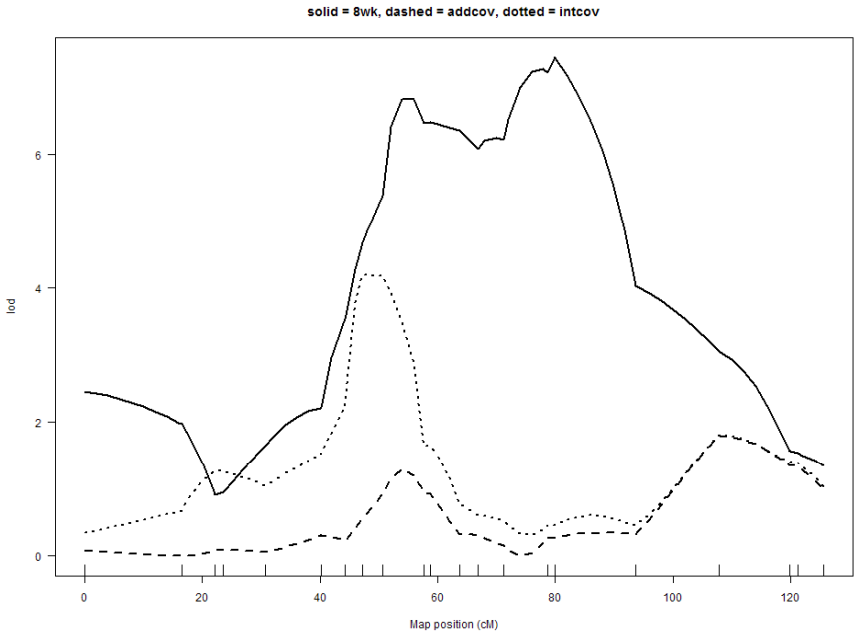
```
## Get Brassica data.
library(qtlbim)
data(Bnapus)
Bnapus <- calc.genoprob(Bnapus, step = 2, error = 0.01)

## Scatterplot of two phenotypes: 4wk & 8wk flower time.
plot(Bnapus$pheno$log10flower4,Bnapus$pheno$log10flower8)

## Unadjusted IM scans of each phenotype.
fl8 <- scanone(Bnapus,, find.pheno(Bnapus, "log10flower8"))
fl4 <- scanone(Bnapus,, find.pheno(Bnapus, "log10flower4"))
plot(fl4, fl8, chr = "N2", col = rep(1,2), lty = 1:2,
  main = "solid = 4wk, dashed = 8wk", lwd = 4)
```

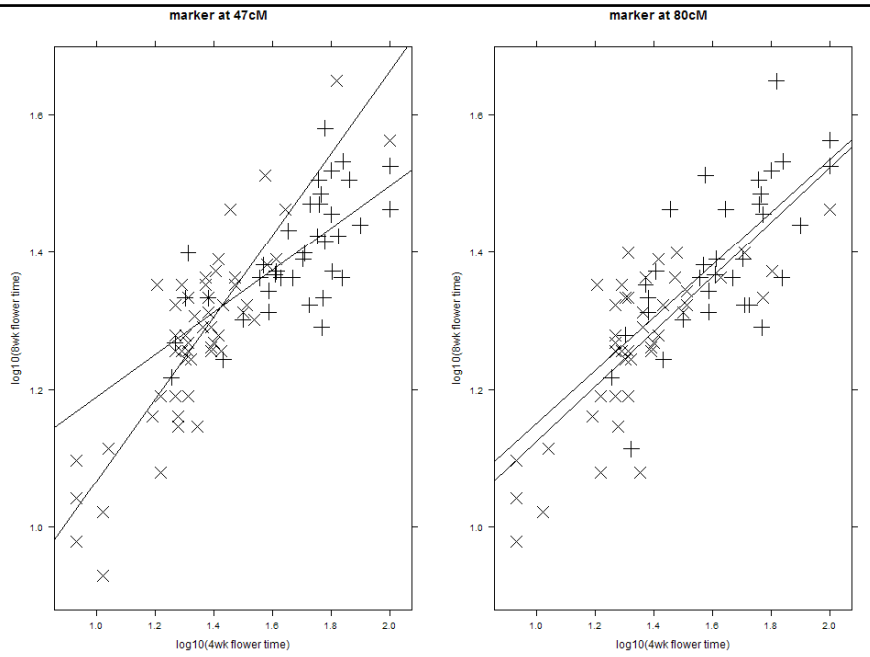



R/qtl & covariates

- additive and/or interacting covariates
- test for QTL after adjusting for covariates

```
## IM scan of 8wk adjusted for 4wk.
## Adjustment independent of genotype
fl8.4 <- scanone(Bnapus,, find.pheno(Bnapus, "log10flower8"),
  addcov = Bnapus$pheno$log10flower4)

## IM scan of 8wk adjusted for 4wk.
## Adjustment changes with genotype.
fl8.4 <- scanone(Bnapus,, find.pheno(Bnapus, "log10flower8"),
  intcov = Bnapus$pheno$log10flower4)

plot(fl8, fl8.4a, fl8.4, chr = "N2",
  main = "solid = 8wk, dashed = addcov, dotted = intcov")
```



scatterplot adjusted for covariate

```
## Set up data frame with peak markers, traits.
markers <- c("E38M50.133","ec2e5a","wg7f3a")
tmpdata <- data.frame(pull.geno(Bnapus)[,markers])
tmpdata$fl4 <- Bnapus$pheno$log10flower4
tmpdata$fl8 <- Bnapus$pheno$log10flower8

## Scatterplots grouped by marker.
library(lattice)
xyplot(fl8 ~ fl4, tmpdata, group = wg7f3a,
  col = "black", pch = 3:4, cex = 2, type = c("p","r"),
  xlab = "log10(4wk flower time)",
  ylab = "log10(8wk flower time)",
  main = "marker at 47cM")
xyplot(fl8 ~ fl4, tmpdata, group = E38M50.133,
  col = "black", pch = 3:4, cex = 2, type = c("p","r"),
  xlab = "log10(4wk flower time)",
  ylab = "log10(8wk flower time)",
  main = "marker at 80cM")
```

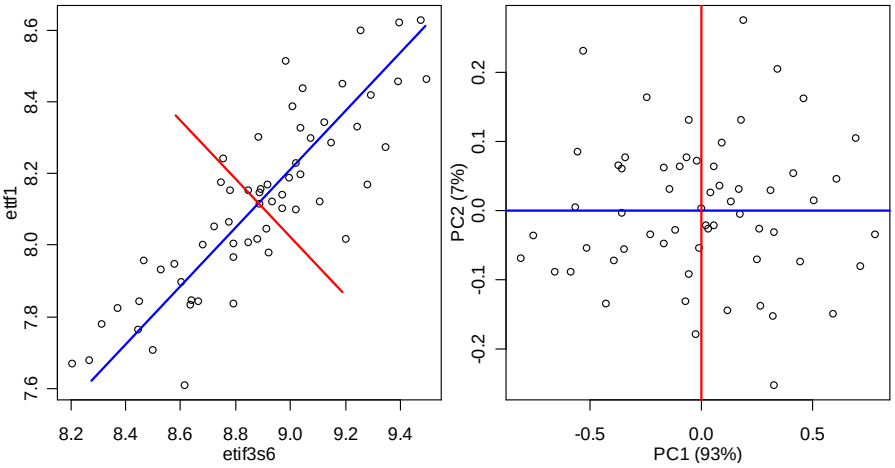
R/qtlbim and GxE

- similar idea to R/qtl
  - fixed and random additive covariates
  - GxE with fixed covariate
- multiple trait analysis tools coming soon
  - theory & code mostly in place
  - properties under study
  - expect in R/qtlbim later this year
  - Samprit Banerjee (N Yi, advisor)

reducing many phenotypes to 1

- *Drosophila mauritiana* x *D. simulans*
  - reciprocal backcrosses, ~500 per bc
- response is "shape" of reproductive piece
  - trace edge, convert to Fourier series
  - reduce dimension: first principal component
- many linked loci
  - brief comparison of CIM, MIM, BIM

PC for two correlated phenotypes



shape phenotype via PC

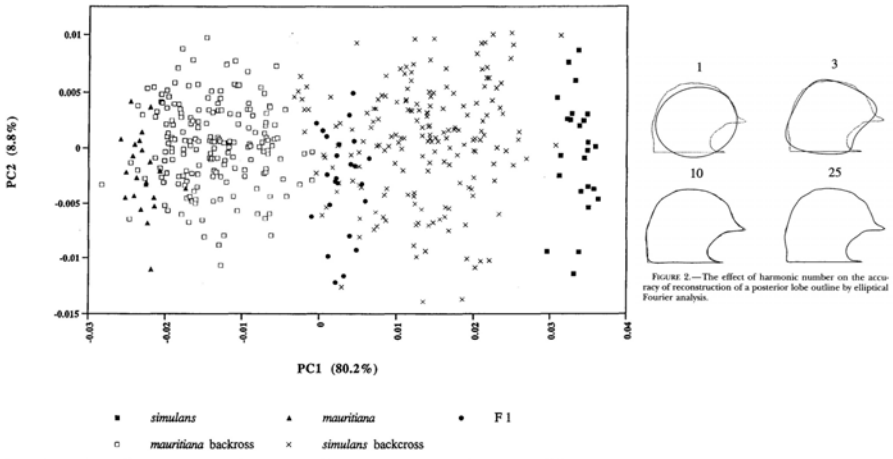


FIGURE 2.—The effect of harmonic number on the accuracy of reconstruction of a posterior lobe outline by elliptical Fourier analysis.

FIGURE 5.—A plot of the first two principal components of the Fourier coefficients from posterior lobe outlines. Many individuals from each of five genotypic classes are represented. Each point represents an average of scores from the left and right sides of an individual (with a few exceptions for which the score is from one side only). The percentage of variation in the Fourier coefficients accounted for by each principal component is given in parentheses. Liu et al. (1996) *Genetics*

shape phenotype in BC study indexed by PC1

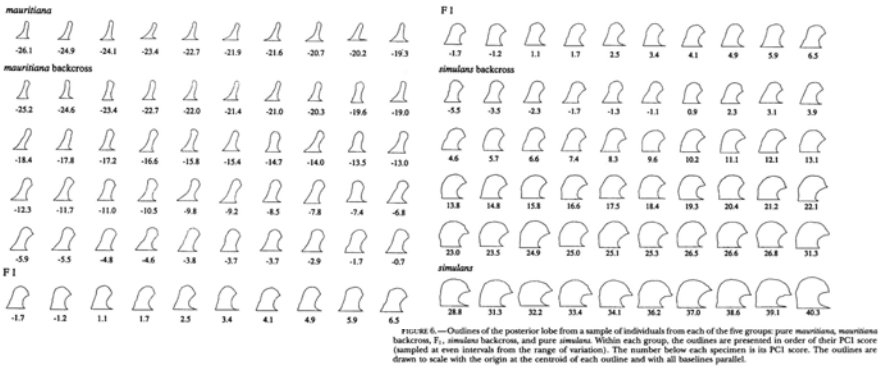


FIGURE 6.—Outlines of the posterior lobe from a sample of individuals from each of the five groups: pure *mauritiana*, *mauritiana* backcross, F_1, *simulans* backcross, and pure *simulans*. Within each group, the outlines are presented in order of their PC1 score (sampled at even intervals from the range of variation). The number below each specimen is its PC1 score. The outlines are drawn to scale with the origin at the centroid of each outline and with all baselines parallel.

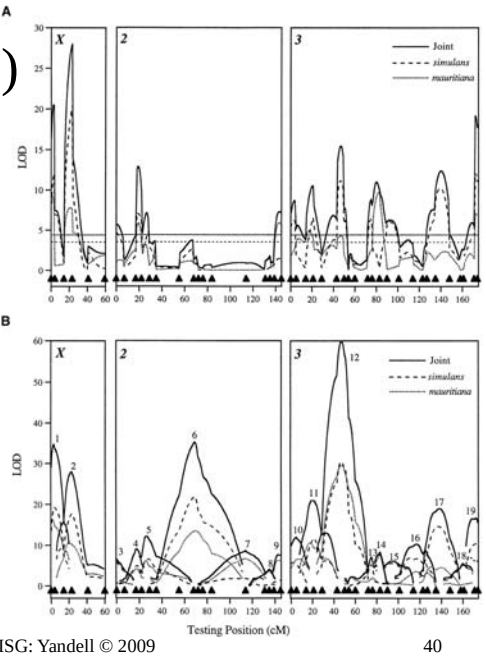
Liu et al. (1996) *Genetics*

Zeng et al. (2000) CIM vs. MIM

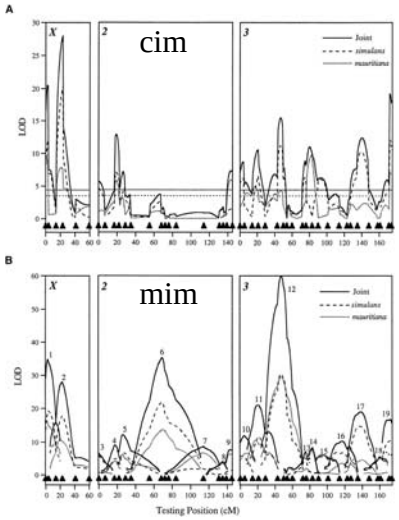
composite interval mapping
- (Liu et al. 1996)
- narrow peaks
- miss some QTL

multiple interval mapping
- (Zeng et al. 2000)
- triangular peaks

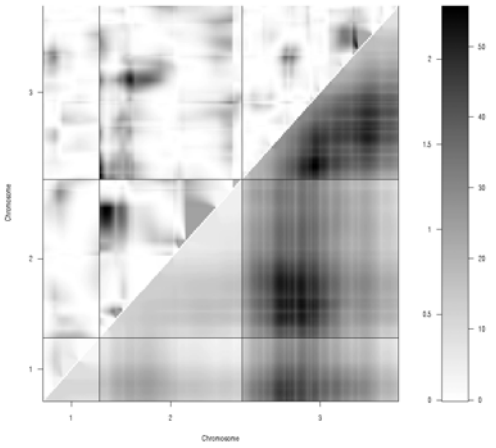
both conditional 1-D scans
- fixing all other "QTL"



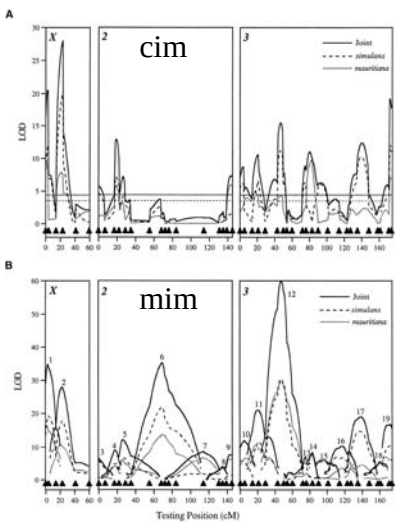
CIM, MIM and IM pairscan

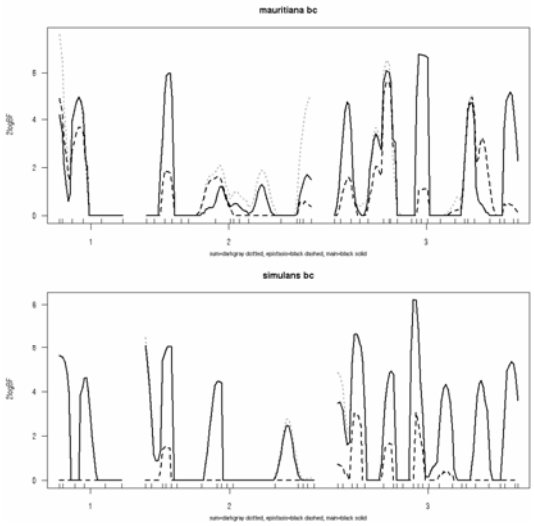


2-D im



multiple QTL: CIM, MIM and BIM





eQTL Tools
a collaboration in progress

Brian Yandell & Bioinformatics Team
Attie Lab, UW-Madison
1 jul 2009

experimental context

- B6 x BTBR obese mouse cross
  - model for diabetes and obesity
  - 500+ mice from intercross (F2)
  - collaboration with Rosetta/Merck
- genotypes
  - 5K SNP Affymetrix mouse chip
  - care in curating genotypes! (map version, errors, …)
- phenotypes
  - clinical phenotypes (>100 / mouse)
  - gene expression traits (>40,000 / mouse / tissue)
  - other molecular phenotypes

how does one filter traits?

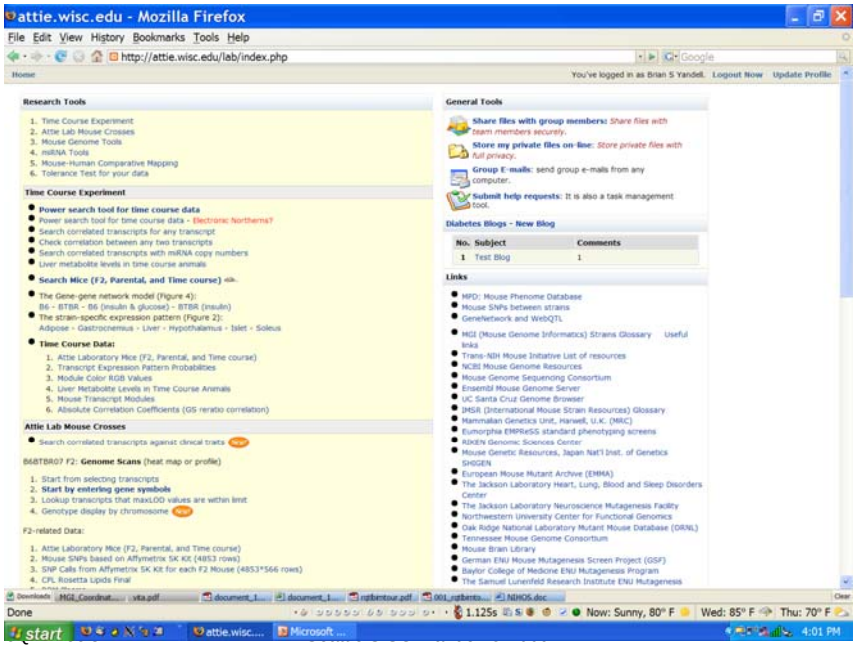
- want to reduce to “manageable” set
  - 10/100/1000: depends on needs/tools
  - How many can the biologist handle?
- how can we create such sets?
  - data-driven procedures
    - correlation-based modules
      - Zhang & Horvath 2005 *SAGMB*, Keller et al. 2008 *Genome Res*
      - Li et al. 2006 *Hum Mol Gen*
    - mapping-based focus on genome region
  - function-driven selection with database tools
    - GO, KEGG, etc
    - Incomplete knowledge leads to bias
  - random sample

why build Web eQTL tools?

- common storage/maintainence of data
  - one well-curated copy
  - central repository
  - reduce errors, ensure analysis on same data
- automate commonly used methods
  - biologist gets immediate feedback
  - statistician can focus on new methods
  - codify standard choices

how does one build tools?

- no one solution for all situations
- use existing tools wherever possible
  - new tools take time and care to build!
  - downloaded databases must be updated regularly
- human component is key
  - need informatics expertise
  - need continual dialog with biologists
- build bridges (interfaces) between tools
  - Web interface uses PHP
  - commands are created dynamically for R
- continually rethink & redesign organization

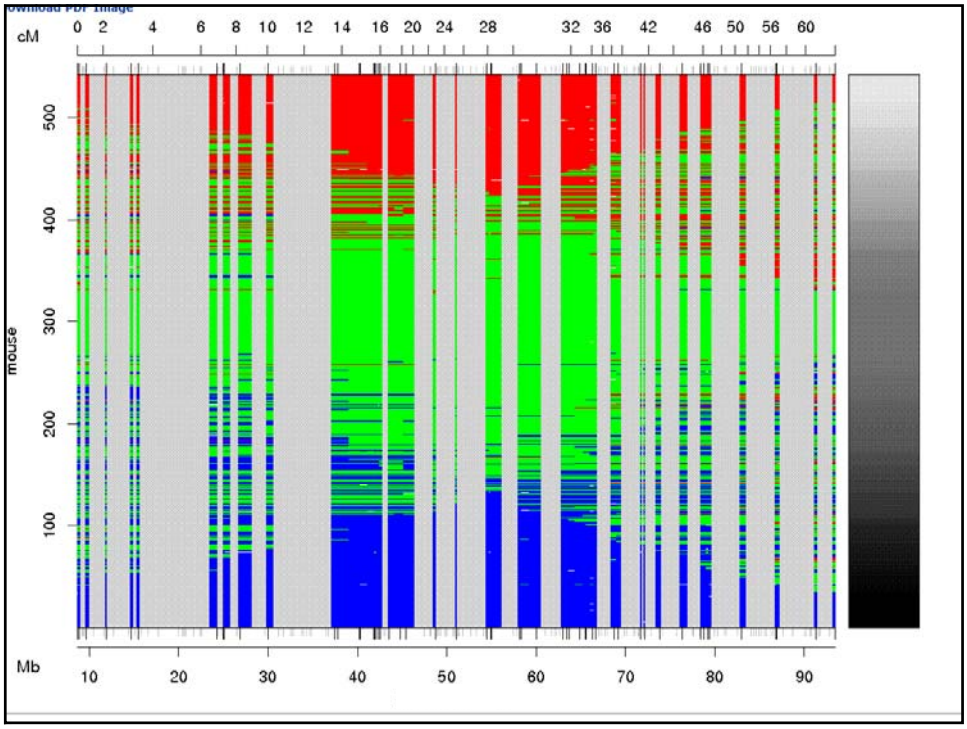


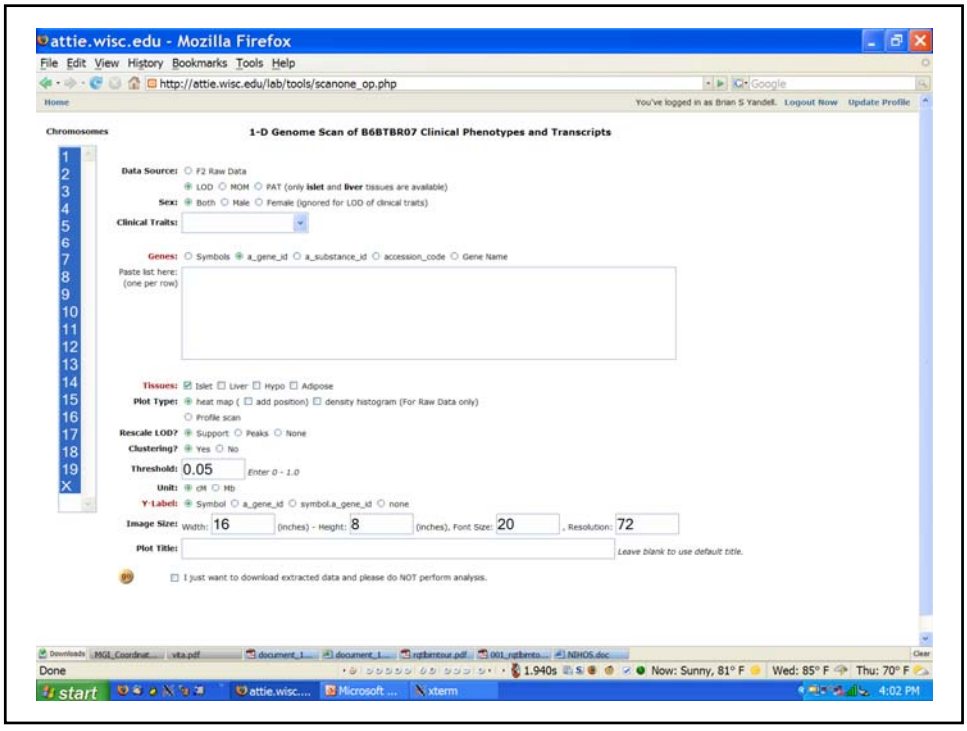
steps in using Web tools

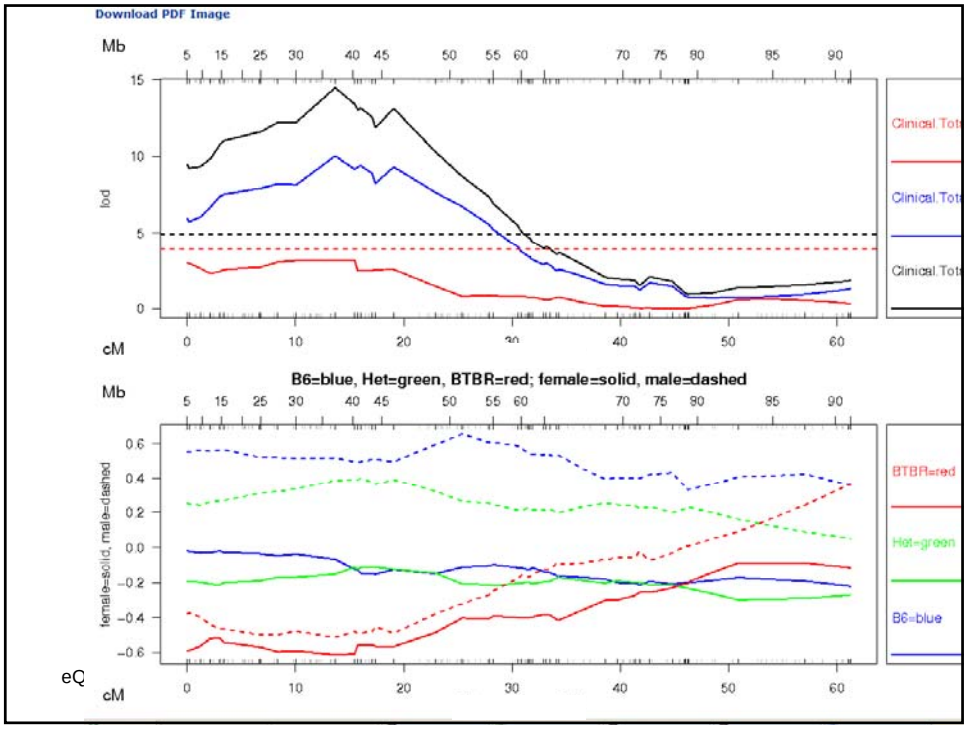
- user enters data on Web page
- PHP tool interprets user data
- PHP builds R script
- R run on script
  - creates plots, summaries, warnings
- PHP grabs results & displays on page
- user examines, saves
- user modifies data and reruns

raw data or fancy results?

- raw data flexible but slow
  - LOD profiles for 100 (1000) traits?
- fancy results from sophisticated analysis
  - IM, MIM, BIM, MOM analysis
  - too complicated to put in biologists' hands?
    - methods are unrefined, state-of-art, research tools
    - use of methods involved many subtle choices
  - batch computation over weeks
    - compute once, save, display many times

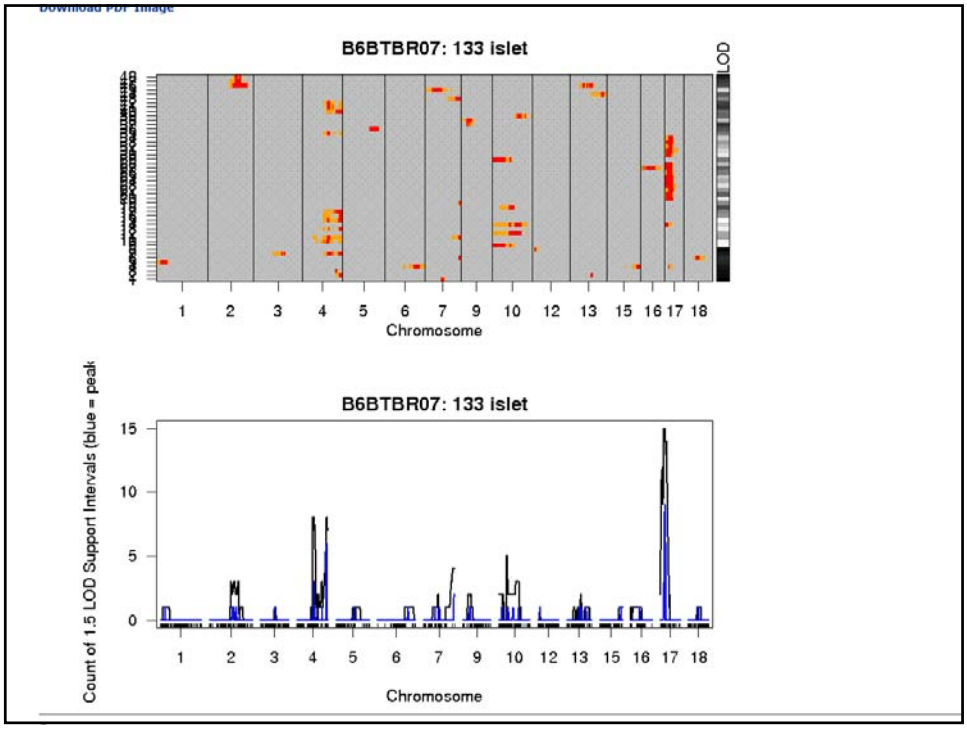

cM
0 2 4 6 8 10 12 14 16 20 24 28 32 36 42 46 50 56 60
mouse
100 200 300 400 500
Mb
10 20 30 40 50 60 70 80 90

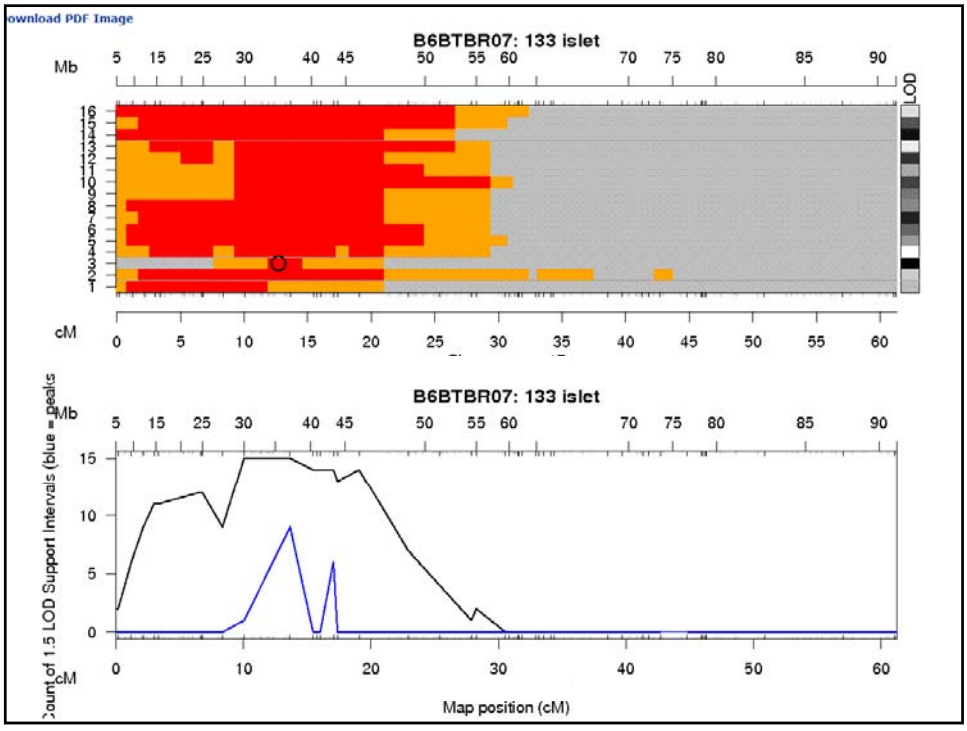

attie.wisc.edu - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://attie.wisc.edu/lab/tools/scanone_op.php
Home
You've logged in as Brian S Yandell. Logout Now Update Profile
Chromosomes
1-D Genome Scan of B6BTBR07 Clinical Phenotypes and Transcripts
Data Source:
Sex:
Clinical Traits:
Genes:
Paste list here: (one per row)
Tissues:
Plot Type:
Rescale LOD?
Clustering?
Threshold: 0.05
Unit:
Y-Label:
Image Size: Width: 16 (inches) - Height: 8 (inches), Font Size: 20 , Resolution: 72
Plot Title:
Leave blank to use default title.
I just want to download extracted data and please do NOT perform analysis.
Done
Now: Sunny, 81° F
Wed: 85° F
Thu: 70° F
start
4:02 PM

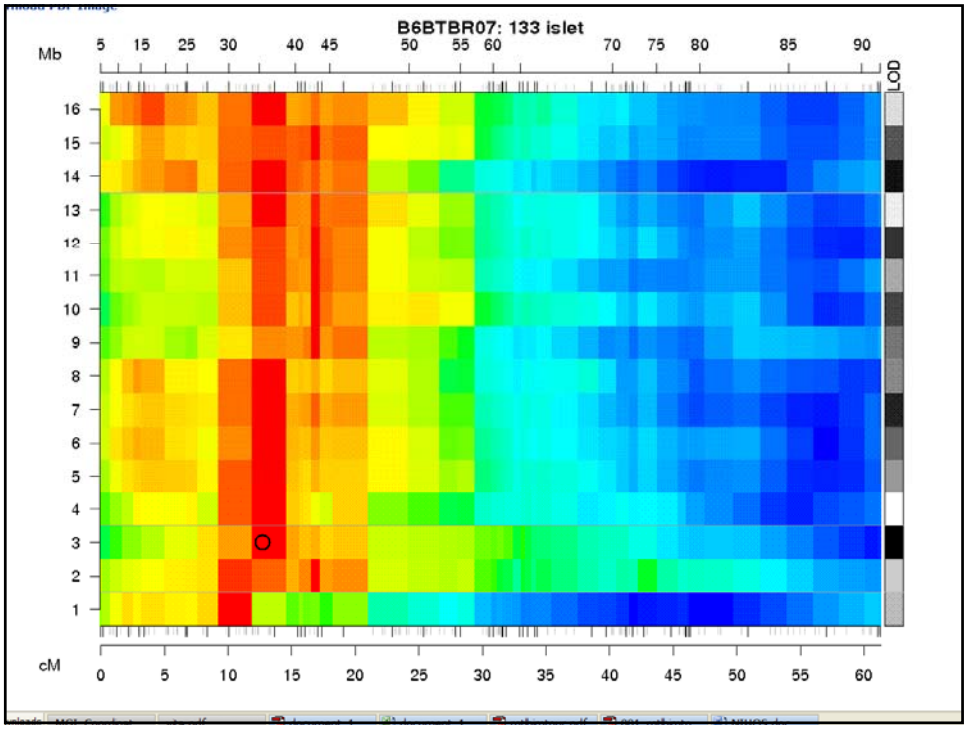


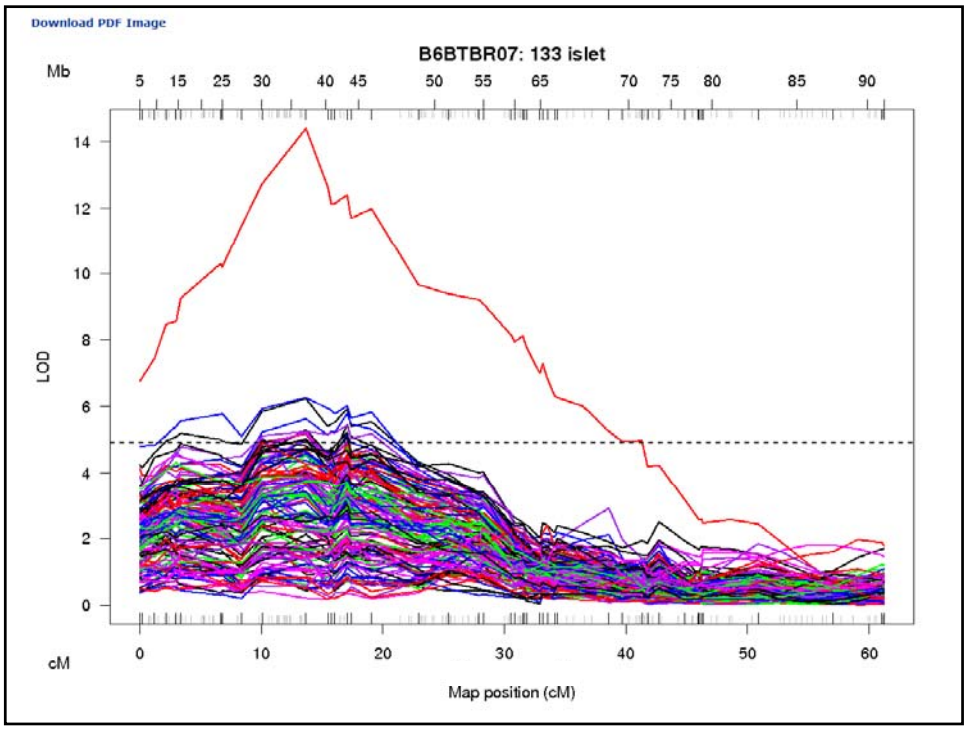
automated R script

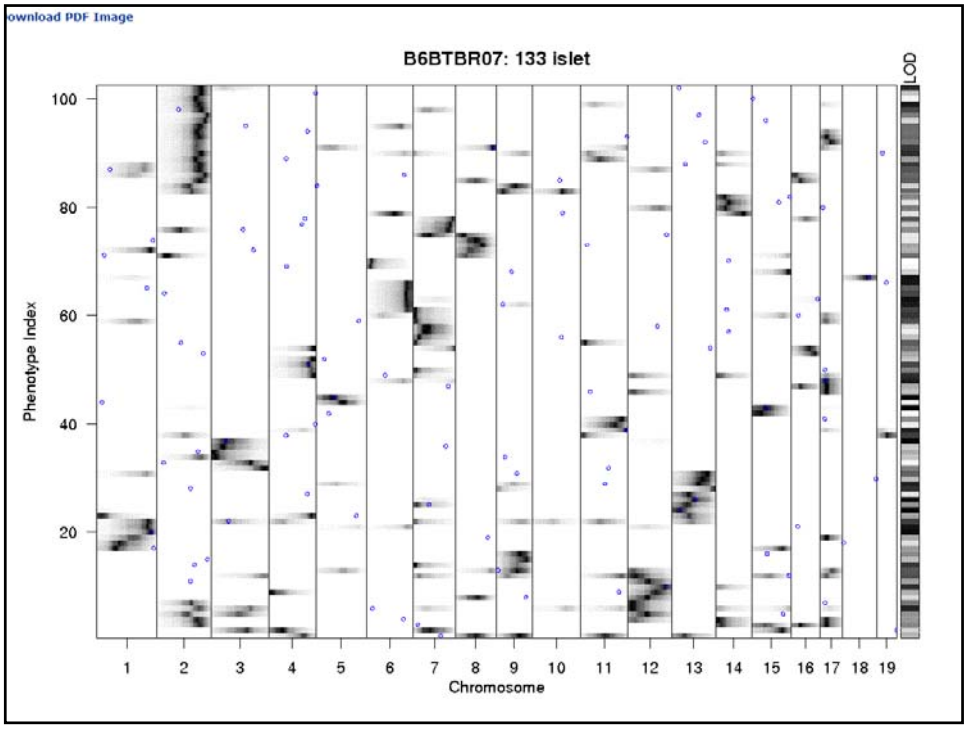
```
library('B6BTBR07')

out <- multtrait(cross.name='B6BTBR07',
  filename = 'scanone_1214952578.csv',
  category = 'islet', chr = c(17),
  threshold.level = 0.05, sex = 'both',)

sink('scanone_1214952578.txt')
print(summary(out))
sink()

bitmap('scanone_1214952578%03d.bmp',
  height = 12, width = 16, res = 72, pointsize = 20)
plot(out, use.cM = TRUE)
dev.off()
```

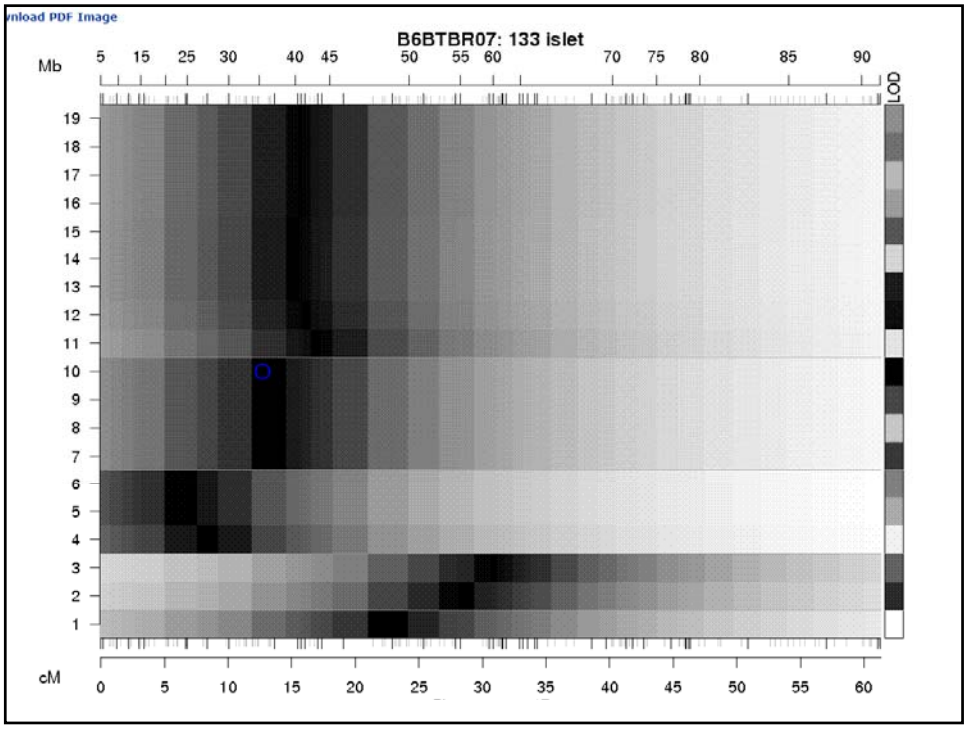

B6BTBR07: 133 islet
LOD
1 2 3 4 5 6 7 9 10 12 13 15 16 17 18
Chromosome
B6BTBR07: 133 islet
Count of 1.5 LOD Support Intervals (blue = peak
15
10
5
0
1 2 3 4 5 6 7 9 10 12 13 15 16 17 18
Chromosome


ownload PDF Image
B6BTBR07: 133 islet
Mb
5 15 25 30 40 45 50 55 60 70 75 80 85 90
LOD
cM
0 5 10 15 20 25 30 35 40 45 50 55 60
B6BTBR07: 133 islet
Mb
5 15 25 30 40 45 50 55 60 70 75 80 85 90
Count of 1.5 LOD Support Intervals (blue = peaks
15
10
5
0
cM
0 10 20 30 40 50 60
Map position (cM)


B6BTBR07: 133 islet
Mb
5 15 25 30 40 45 50 55 60 70 75 80 85 90
LOD
16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
cM
0 5 10 15 20 25 30 35 40 45 50 55 60


Download PDF Image
B6BTBR07: 133 islet
Mb
5 15 25 30 40 45 50 55 65 70 75 80 85 90
LOD
14 12 10 8 6 4 2 0
cM
0 10 20 30 40 50 60
Map position (cM)


ownload PDF Image
B6BTBR07: 133 islet
LOD
Phenotype Index
100
80
60
40
20
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
Chromosome


nload PDF Image
B6BTBR07: 133 islet
Mb
5 15 25 30 40 45 50 55 60 70 75 80 85 90
LOD
19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
cM
0 5 10 15 20 25 30 35 40 45 50 55 60

Inferring Causal Phenotype Networks

Elias Chaibub Neto & Brian S. Yandell
UW-Madison
June 2009

outline

- QTL-driven directed graphs
  - Assume QTLs known, network unknown
  - Infer links (edges) between pairs of phenotypes (nodes)
    - Based on partial correlation
  - Infer causal direction for edges
  - Chaibub et al. (2008 *Genetics*)
  - Software R/qdg available on CRAN
- Causal graphical models in systems genetics
  - QTLs unknown, network unknown
  - Infer both genetic architecture (QTLs) and pathways (networks)
  - Chaibub et al. (2009 *Ann Appl Statist* tent accept)
  - Software R/QTLnet in preparation for CRAN

QTL-driven directed graphs

- See edited slides by Elias Chaibub Neto
  - BIOCOMP 2008 talk
  - Chaibub Neto, Ferrara, Attie, Yandell (2008) Inferring causal phenotype networks from segregating populations. *Genetics 179*: 1089-1100.
  - Ferrara et al. Attie (2008) Genetic networks of liver metabolism revealed by integration of metabolic and transcriptomic profiling. *PLoS Genet 4*: e1000034.

causal graphical models in systems genetics

- Chaibub Neto, Keller, Attie , Yandell (2009) Causal Graphical Models in Systems Genetics: a unified framework for joint inference of causal network and genetic architecture for correlated phenotypes. *Ann Appl Statist* (tent. accept)
- Related references
  - Schadt et al. Lusis (2005 *Nat Genet*); Li et al. Churchill (2006 *Genetics*); Chen Emmert-Streib Storey(2007 *Genome Bio*); Liu de la Fuente Hoeschele (2008 *Genetics*); Winrow et al. Turek (2009 *PLoS ONE*)
- Jointly infer unknowns of interest
  - genetic architecture
  - causal network

Basic idea of QTLnet

- Genetic architecture given causal network
  - Trait y depends on parents pa(y) in network
  - QTL for y found conditional on pa(y)
    - Parents pa(y) are interacting covariates for QTL scan
- Causal network given genetic architecture
  - Build (adjust) causal network given QTL

MCMC for QTLnet

- Propose new causal network with simple changes to current network
  - Change edge direction
  - Add or drop edge
- Find any new genetic architectures (QTLs)
  - Update phenotypes whose parents pa(y) change in new network
- Compute likelihood for new network and QTL
- Accept or reject new network and QTL
  - Usual Metropolis-Hastings idea

Future work

- Incorporate latent variables
  - Aten et al. Horvath (2008 *BMC Sys Biol*)
- Allow for prior information about network
  - Werhli and Husmeier (2007 *SAGMB*); Dittrich et al. Müller (2008 *Bioinfo*); Zhu et al. Schadt (2008 *Nat Genet*); Lee et al. Koller (2009 *PLoS Genet*); Thomas et al. Portier (2009 *Genome Bio*); Wu et al. Lin (2009 *Bioinfo*)
- Improve algorithm efficiency
  - Ramp up to 1000s of phenotypes
- Extend to outbred crosses, humans

Inferring Causal Phenotype Networks from Segregating Populations

Elias Chaibub Neto
chaibub@stat.wisc.edu

Statistics Department, University of Wisconsin - Madison

July 15, 2008

Overview

- Introduction
- Description of the approach
  - PC algorithm.
  - QDG algorithm.
- Remarks
- Performance on simulations.
- Real data example.
- Future work.

Introduction

- Our objective is to learn metabolic pathways from data.
- We represent these pathways by directed networks composed by transcripts, metabolites and clinical traits.
- These phenotypes are quantitative in nature, and can be analyzed using quantitative genetics techniques.

Introduction

- In particular, we use Quantitative Trait Loci (QTL) mapping methods to identify genomic regions affecting the phenotypes.

- Since variations in the genotypes (QTLs) cause variations in the phenotypes, but not the other way around, we can unambiguously determine the causal direction

$$\text{QTL} \Rightarrow \text{phenotype}$$

- Knowing that a QTL causally affects a phenotype will allow us to infer causal direction between phenotypes.

Introduction

- We assume that a set of QTLs associated with the phenotypes has been previously determined.
- We assume linear relationships between phenotypes and between phenotypes and QTLs.

Introduction

Our procedure is composed of two parts:

1. First we infer the skeleton of the causal model (phenotype network) using the PC-algorithm.

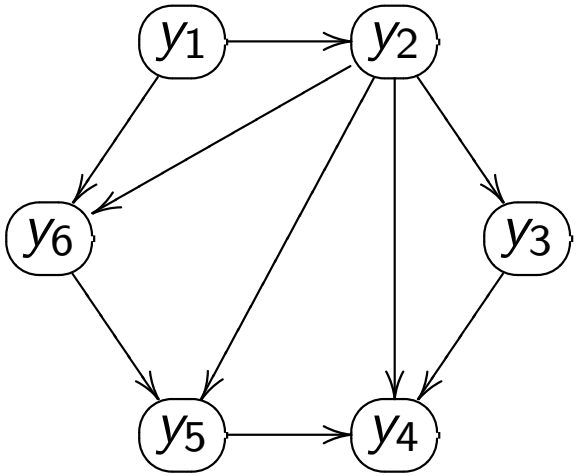
2. Orient the edges in the skeleton using the QDG algorithm.

PC algorithm

- Causal discovery algorithm developed by Spirtes et al 1993.
- It is composed of two parts:
  1. Infers the skeleton of the causal model.
  2. Partially orient the graph (orient some but not all edges).
- We are only interested in the first part (the "PC skeleton algorithm"). We do **not** use the PC algorithm to edge orientation (we use the QDG algorithm instead).

Step 1 (PC skeleton algorithm)

Suppose the true network describing the causal relationships between six transcripts is

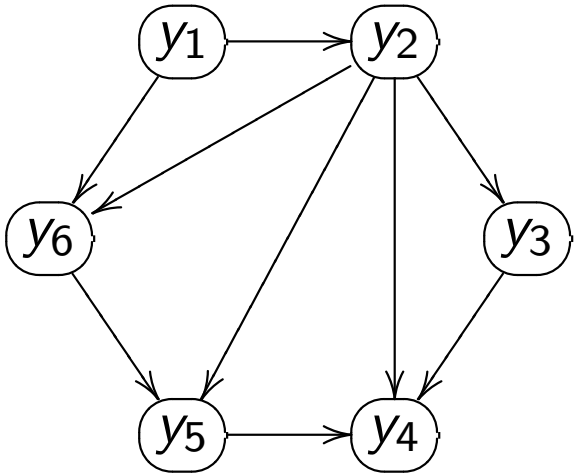


The PC-algorithm starts with the complete undirected graph

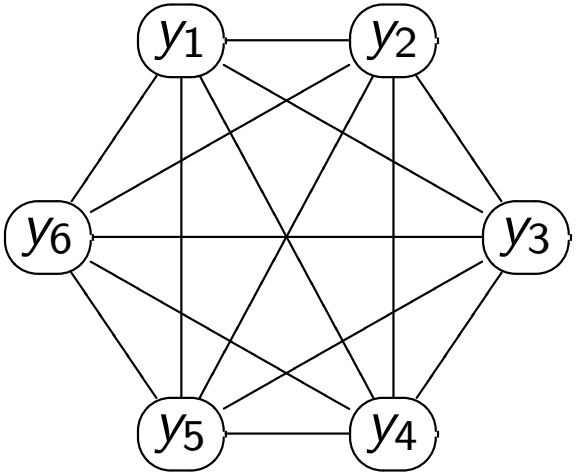
and progressively eliminates edges based on conditional independence tests.

Step 1 (PC skeleton algorithm)

Suppose the true network describing the causal relationships between six transcripts is



The PC-algorithm starts with the complete undirected graph



and progressively eliminates edges based on conditional independence tests.

Step 1 (PC skeleton algorithm)

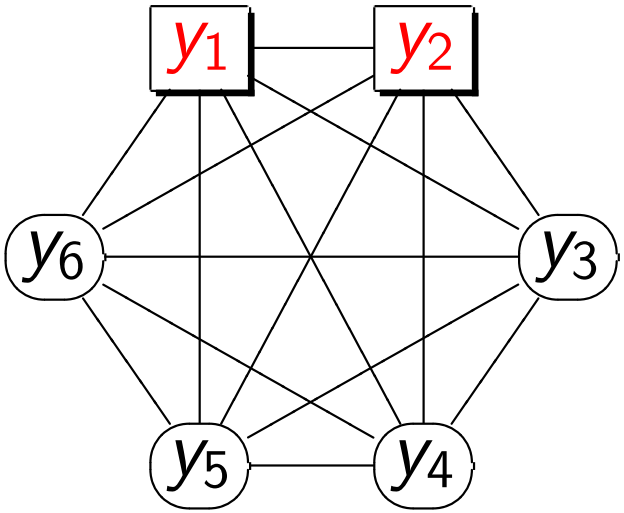
The algorithm performs several rounds of conditional independence tests of increasing order.

It starts with all zero order tests, then performs all first order, second order ...

- Notation: $\perp\!\!\!\perp \equiv$ independence. We read $i \perp\!\!\!\perp j \mid k$ as *i is conditionally independent from j given k*.
- Remark: in the Gaussian case zero partial correlation implies conditional independence, thus

$$i \perp\!\!\!\perp j \mid k \quad \Leftrightarrow \quad cor(i, j \mid k) = 0 \quad \Rightarrow \quad \text{drop } (i, j) \text{ edge}$$

Example (order 0)

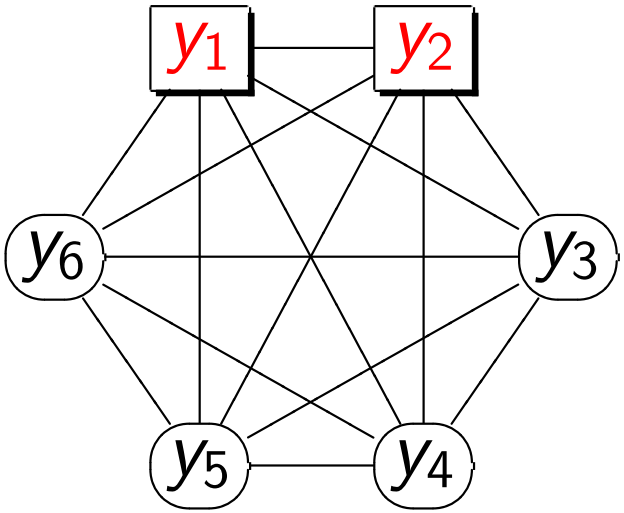


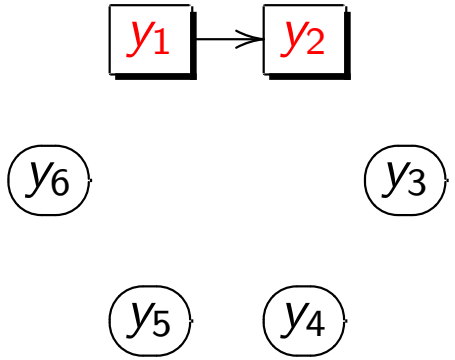
$$1 \perp\!\!\!\perp 2$$

vs

$$1 \not\perp\!\!\!\perp 2$$

Example (order 0)





direct effect of y_1 on y_2

$1 \perp\!\!\!\perp 2$

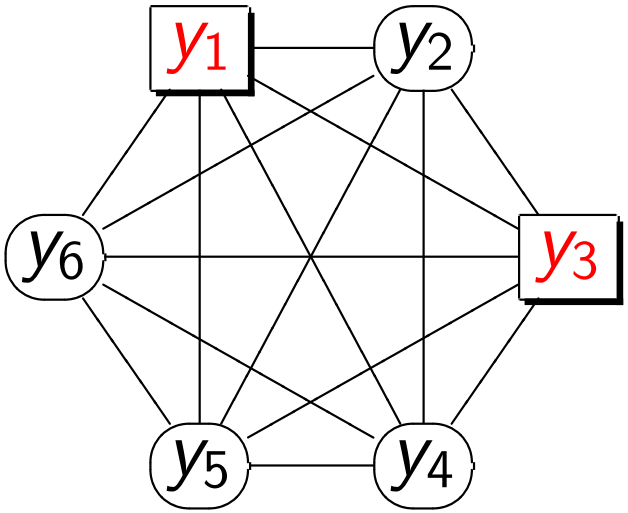
vs

$1 \not\perp\!\!\!\perp 2$

$1 \not\perp\!\!\!\perp 2$

keep edge and move to next one

Example (order 0)

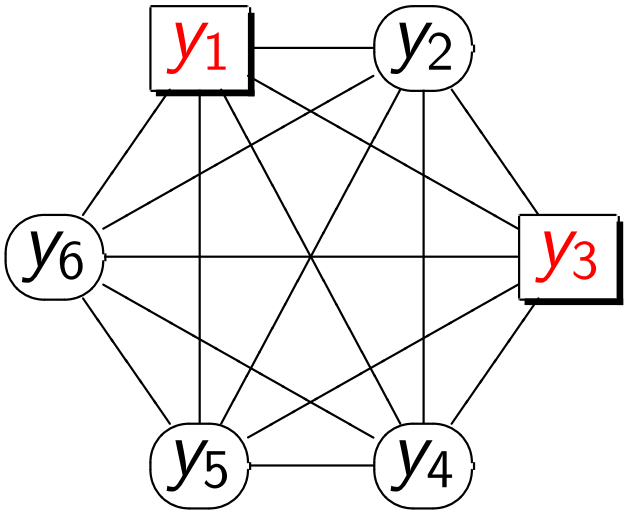


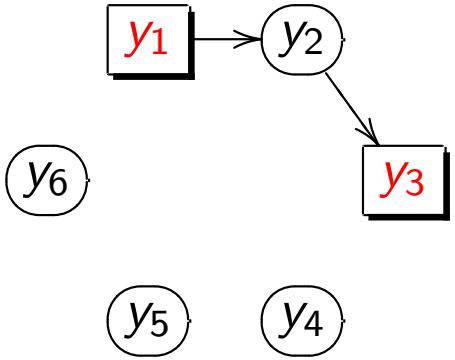
$$1 \perp\!\!\!\perp 3$$

vs

$$1 \not\perp\!\!\!\perp 3$$

Example (order 0)





indirect effect of y_1 on y_3

$1 \perp\!\!\!\perp 3$

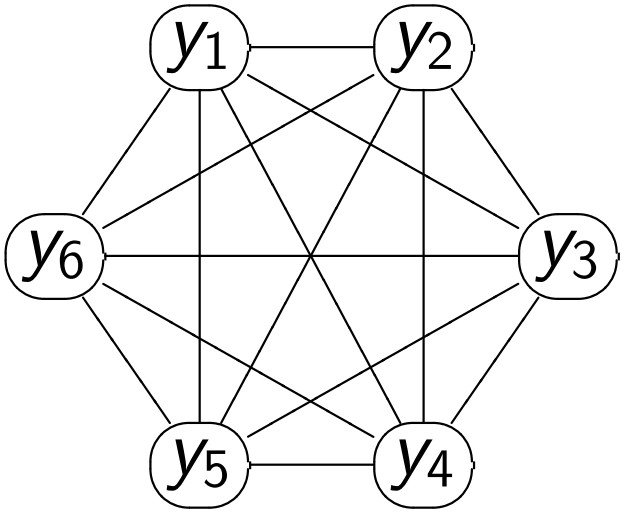
vs

$1 \not\perp\!\!\!\perp 3$

$1 \not\perp\!\!\!\perp 3$

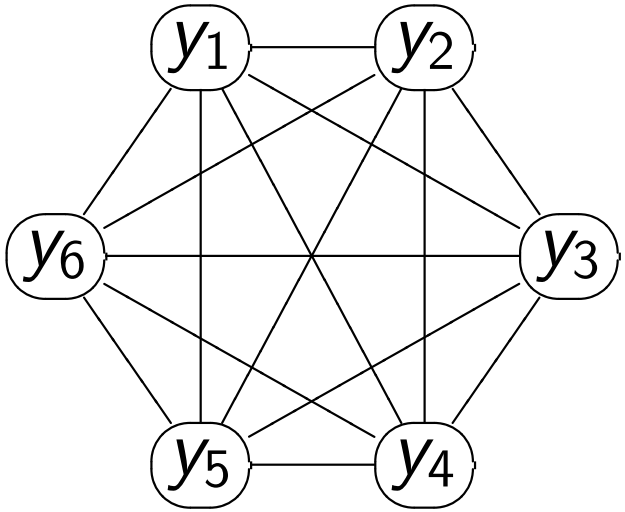
keep edge and move to next one

Example (order 0)



After all zero order conditional independence tests.

Example (order 0)



After all zero order conditional independence tests.

The algorithm then moves to first order conditional independence tests.

Example (order 1)

For any edge (i, j) the algorithm tests whether

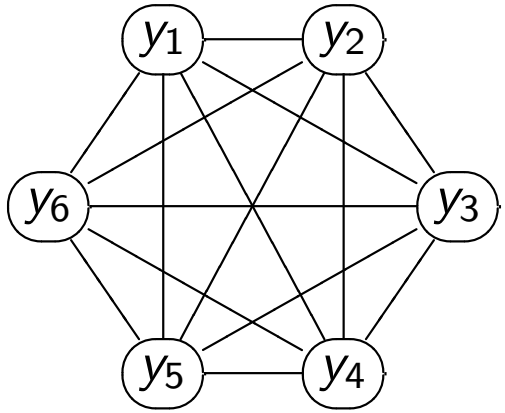
$$i \perp\!\!\!\perp j \mid k$$

for all

$$k \in A(i) \setminus j$$

where $A(i)$ represent the set of nodes adjacent to node i.

For example,



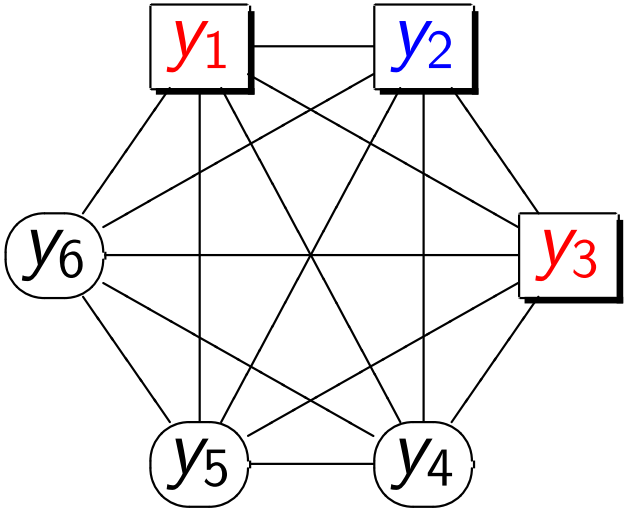
$$A(1) \setminus 2 = \{3, 4, 5, 6\}$$

and the algorithm tests whether

$$1 \perp\!\!\!\perp 2 \mid 3 \qquad 1 \perp\!\!\!\perp 2 \mid 4$$

$$1 \perp\!\!\!\perp 2 \mid 5 \qquad 1 \perp\!\!\!\perp 2 \mid 6$$

Example (order 1)



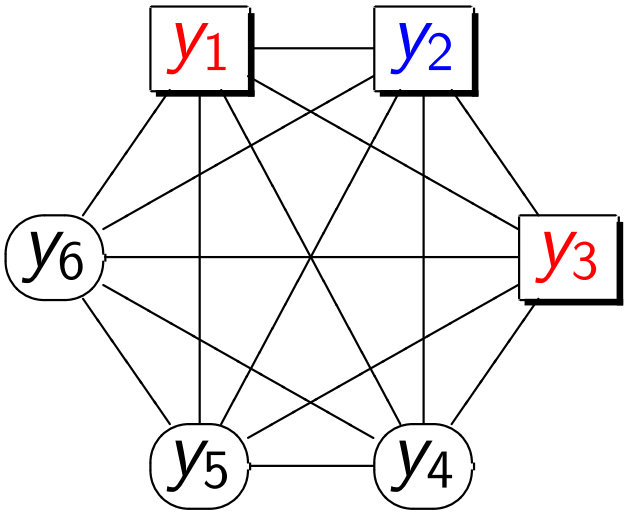
$$A(1) \setminus 2 = \{2, 4, 5, 6\}$$

$$1 \perp\!\!\!\perp 3 \mid 2$$

vs

$$1 \not\perp\!\!\!\perp 3 \mid 2$$

Example (order 1)

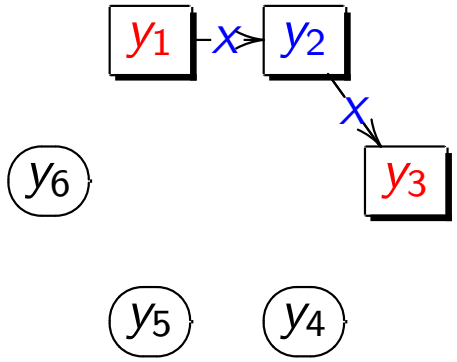


$A(1) \setminus 2 = \{2, 4, 5, 6\}$

$1 \perp\!\!\!\perp 3 \mid 2$

vs

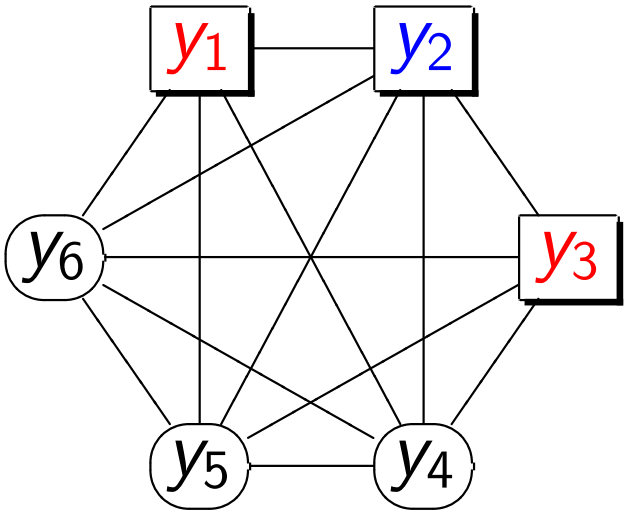
$1 \not\perp\!\!\!\perp 3 \mid 2$



y_2 d-separates y_1 from y_3

$1 \perp\!\!\!\perp 3 \mid 2$

Example (order 1)

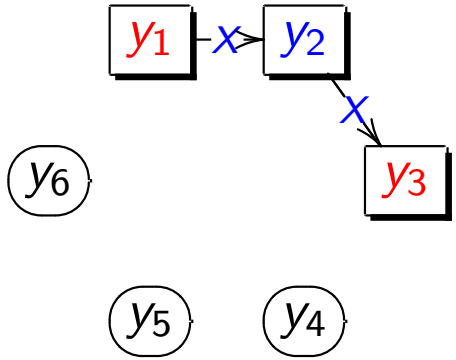


$A(1) \setminus 2 = \{2, 4, 5, 6\}$

$1 \perp\!\!\!\perp 3 \mid 2$

vs

$1 \not\perp\!\!\!\perp 3 \mid 2$



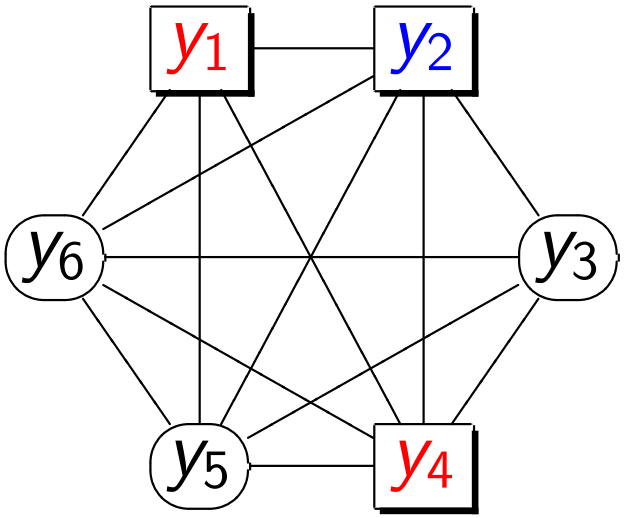
y_2 d-separates y_1 from y_3

$1 \perp\!\!\!\perp 3 \mid 2$

drop edge

move to next edge

Example (order 1)



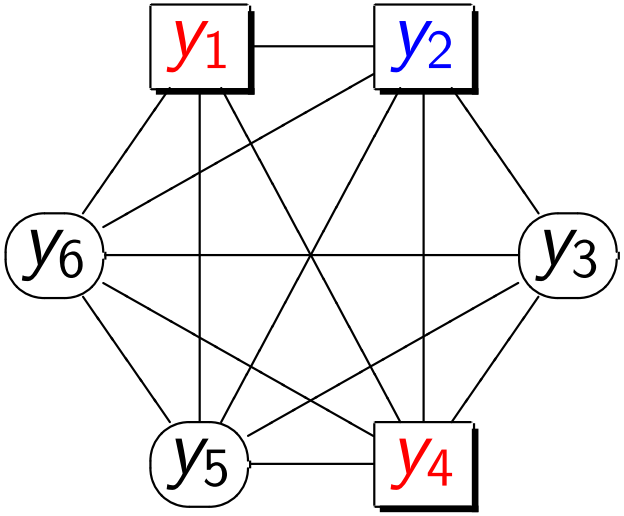
$$A(1) \setminus 4 = \{2, 5, 6\}$$

$$1 \perp\!\!\!\perp 4 \mid 2$$

vs

$$1 \not\perp\!\!\!\perp 4 \mid 2$$

Example (order 1)

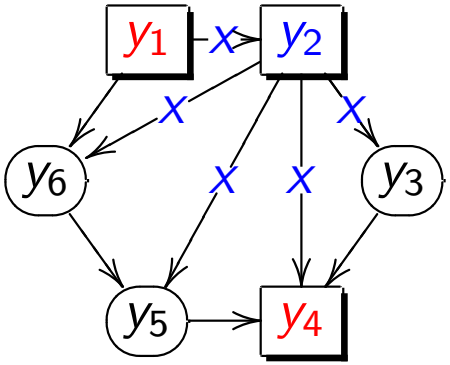


$A(1) \setminus 4 = \{2, 5, 6\}$

$1 \perp\!\!\!\perp 4 \mid 2$

vs

$1 \not\perp\!\!\!\perp 4 \mid 2$

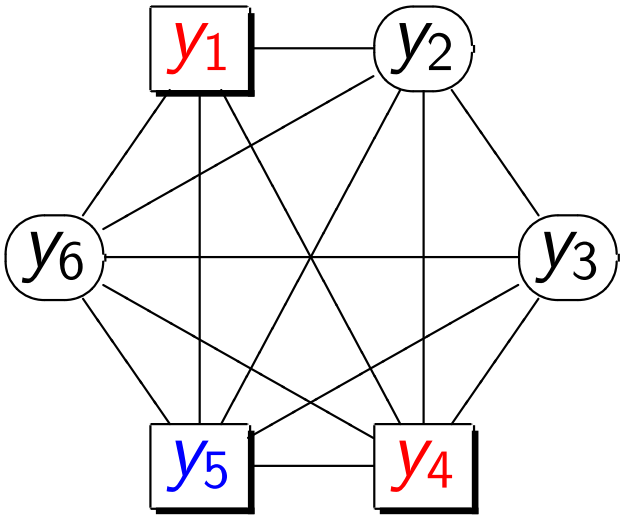


$1 \not\perp\!\!\!\perp 4 \mid 2$

keep edge

move to next conditioning set

Example (order 1)



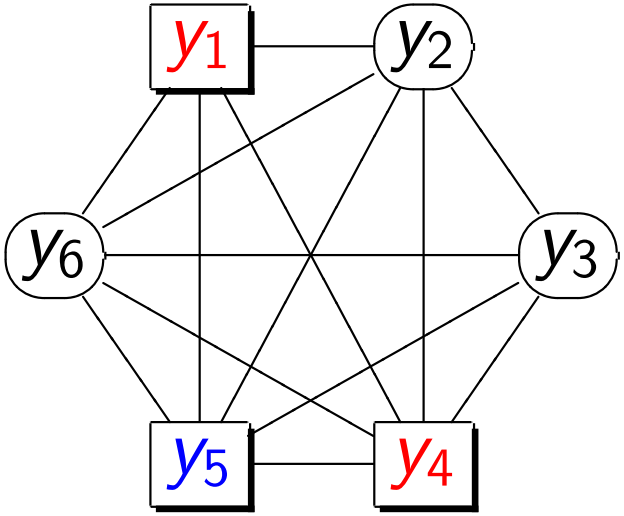
$$A(1) \setminus 4 = \{2, 5, 6\}$$

$$1 \perp\!\!\!\perp 4 \mid 5$$

vs

$$1 \not\perp\!\!\!\perp 4 \mid 5$$

Example (order 1)

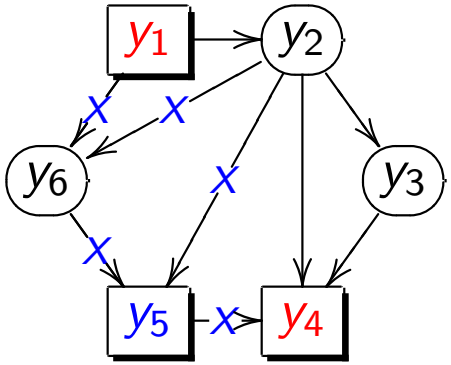


$A(1) \setminus 4 = \{2, 5, 6\}$

$1 \perp\!\!\!\perp 4 \mid 5$

vs

$1 \not\perp\!\!\!\perp 4 \mid 5$

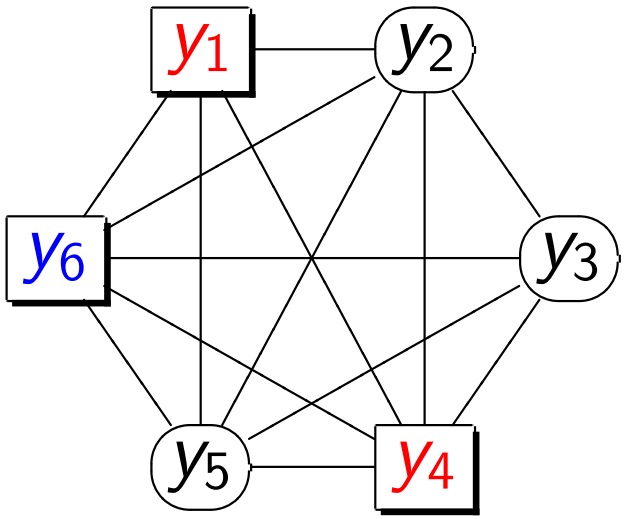


$1 \not\perp\!\!\!\perp 4 \mid 5$

keep edge

move to next conditioning set

Example (order 1)



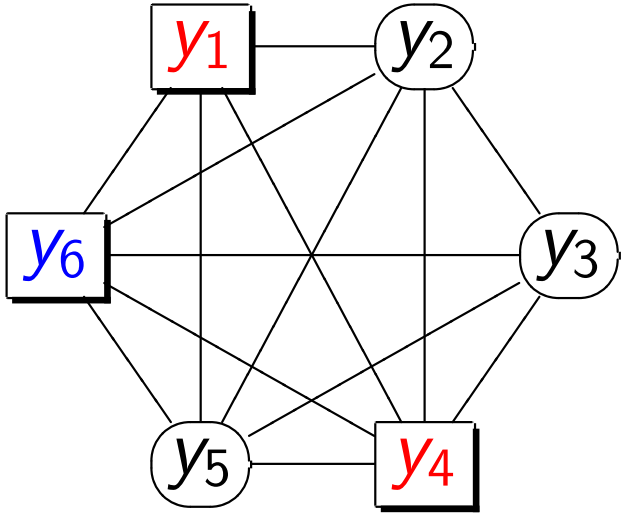
$$A(1) \setminus 4 = \{2, 5, 6\}$$

$$1 \perp\!\!\!\perp 4 \mid 6$$

vs

$$1 \not\perp\!\!\!\perp 4 \mid 6$$

Example (order 1)

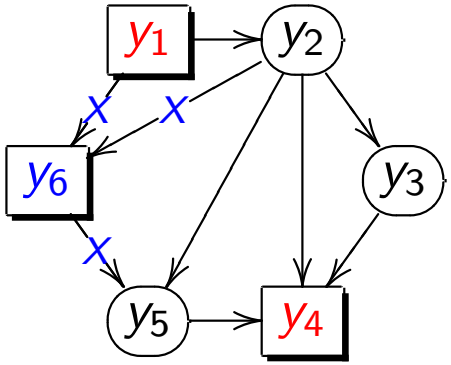


$A(1) \setminus 4 = \{2, 5, 6\}$

$1 \perp\!\!\!\perp 4 \mid 6$

vs

$1 \not\perp\!\!\!\perp 4 \mid 6$

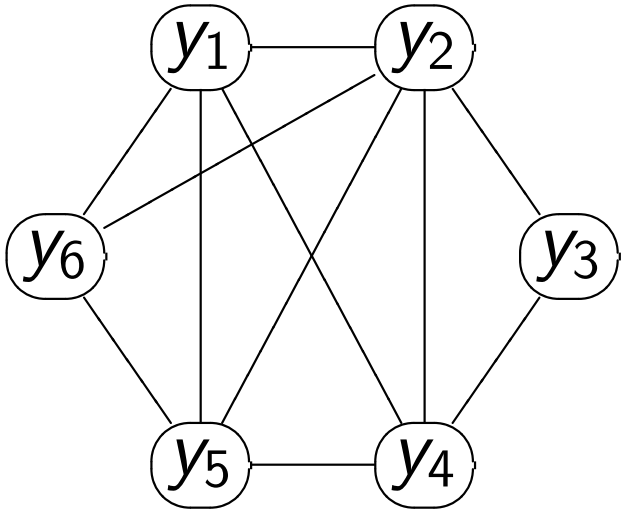


$1 \not\perp\!\!\!\perp 4 \mid 6$

keep edge

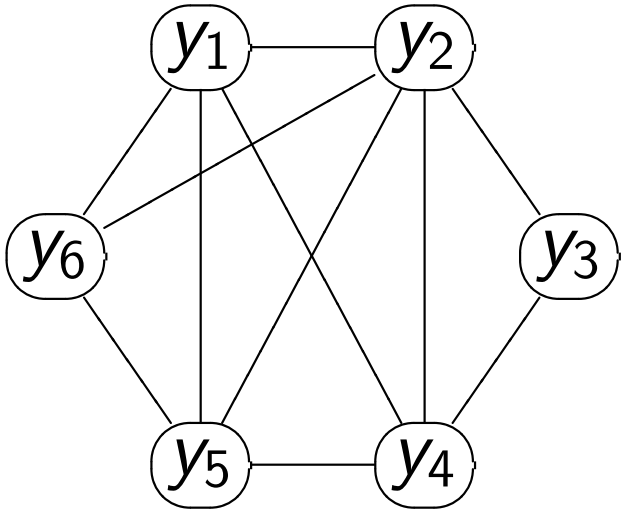
move to next edge

Example (order 1)



After all first order conditional independence tests.

Example (order 1)



After all first order conditional independence tests.

The algorithm then moves to second order conditional independence tests.

Example (order 2)

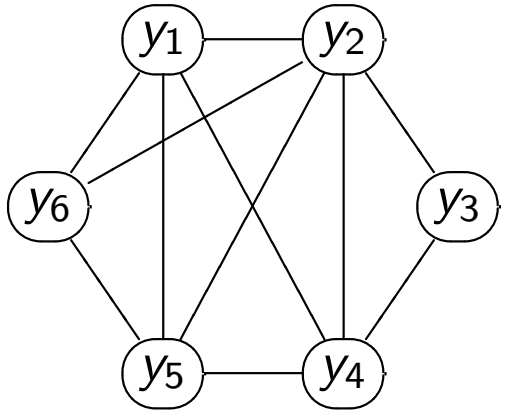
For any edge (i, j) the algorithm tests whether

$$i \perp\!\!\!\perp j \mid k, l.$$

for all

$$(k, l) \in A(i) \setminus j$$

For example,



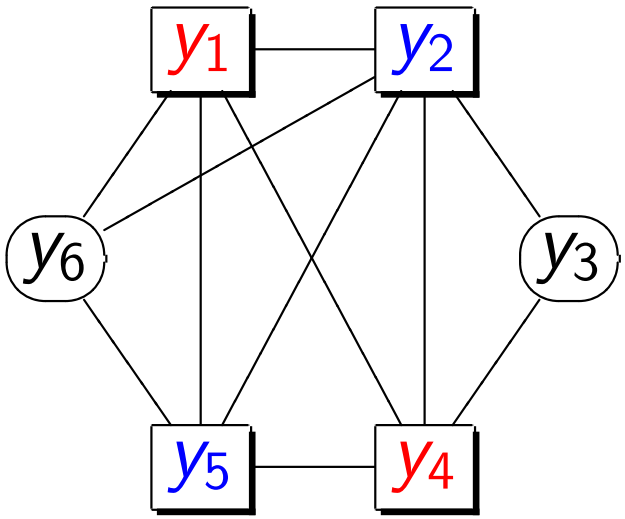
$$A(1) \setminus 2 = \{4, 5, 6\}$$

and the algorithm tests whether

$$1 \perp\!\!\!\perp 2 \mid 4, 5 \qquad 1 \perp\!\!\!\perp 2 \mid 4, 6$$

$$1 \perp\!\!\!\perp 2 \mid 5, 6$$

Example (order 2)



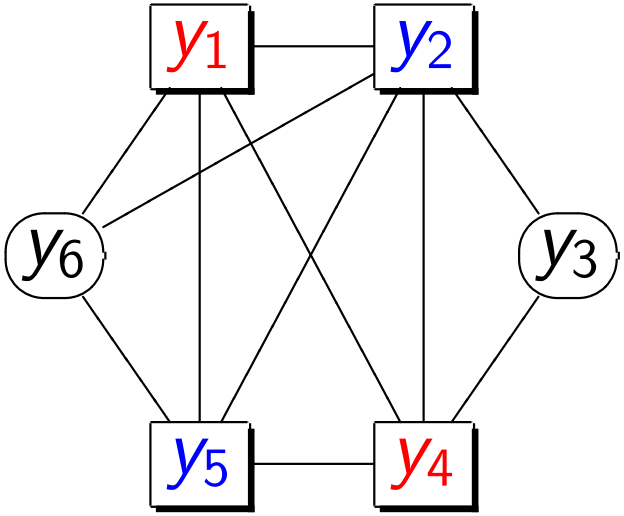
$$A(1) \setminus 4 = \{2, 5, 6\}$$

$$1 \perp\!\!\!\perp 4 \mid 2, 5$$

vs

$$1 \not\perp\!\!\!\perp 4 \mid 2, 5$$

Example (order 2)

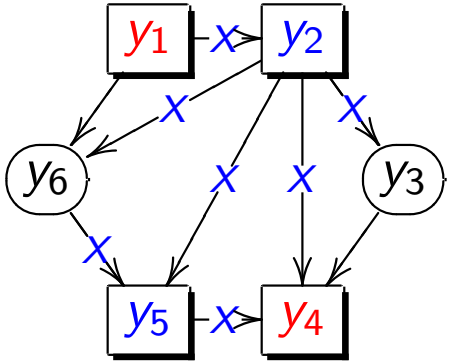


$A(1) \setminus 4 = \{2, 5, 6\}$

$1 \perp\!\!\!\perp 4 \mid 2, 5$

vs

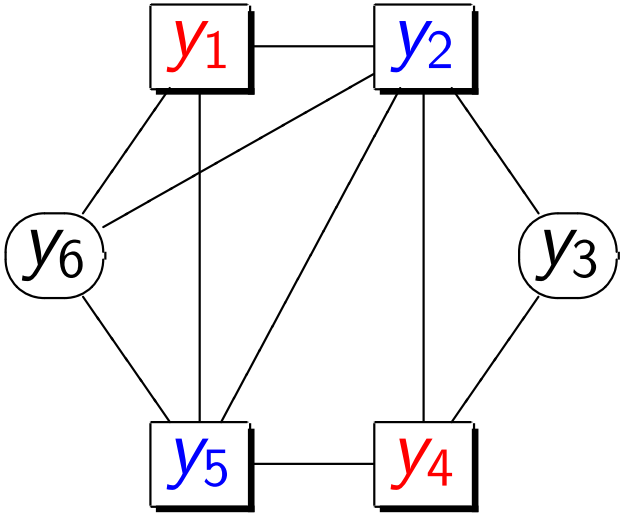
$1 \not\perp\!\!\!\perp 4 \mid 2, 5$



(y_2, y_5) d-separate y_1 from y_4

$1 \perp\!\!\!\perp 4 \mid 2, 5$

Example (order 2)

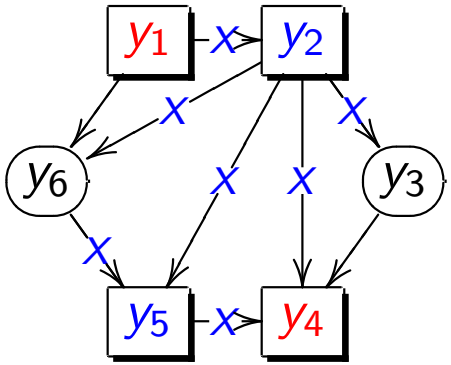


$A(1) \setminus 4 = \{2, 5, 6\}$

$1 \perp\!\!\!\perp 4 \mid 2, 5$

vs

$1 \not\perp\!\!\!\perp 4 \mid 2, 5$



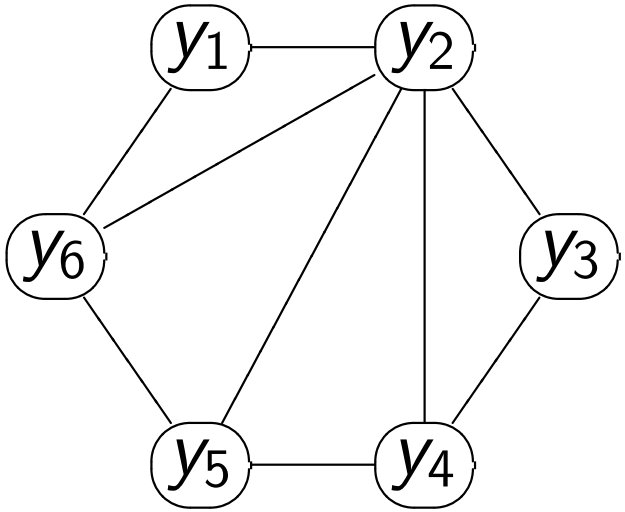
(y_2, y_5) d-separate y_1 from y_4

$1 \perp\!\!\!\perp 4 \mid 2, 5$

drop edge

move to next edge

Example (order 2)



After all second order conditional independence tests.

The algorithm than moves to third order, fourth order ...

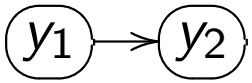
It stops when for each pair (i, j) the cardinality of

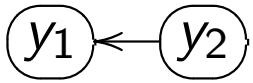
$$A(i) \setminus j$$

is smaller then the order of the algorithm.

Edge orientation

Consider two traits y_1 and y_2. Our problem is to decide between models:

M_1 : 

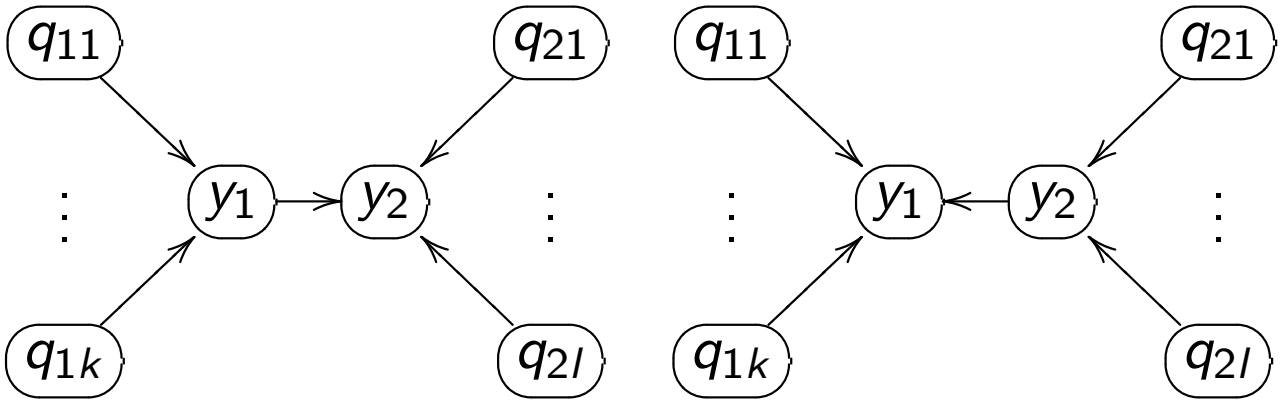
$M2$: 

Problem: the above models are likelihood equivalent,

$$f(y_1)f(y_2 \mid y_1) = f(y_1, y_2) = f(y_2)f(y_1 \mid y_2) \ .$$

Edge orientation

However, models



are *not* likelihood equivalent because

$$f(\mathbf{q}_1)f(y_1 \mid \mathbf{q}_1)f(y_2 \mid y_1, \mathbf{q}_2)f(\mathbf{q}_2)$$
$$\neq$$
$$f(\mathbf{q}_2)f(y_2 \mid \mathbf{q}_2)f(y_1 \mid y_2, \mathbf{q}_1)f(\mathbf{q}_1)$$

Edge orientation

We perform model selection using a direction LOD score

$$LOD = \log_{10} \left\{ \frac{\prod_{i=1}^{n} f(y_{1i} \mid \mathbf{q}_{1i}) f(y_{2i} \mid y_{1i}, \mathbf{q}_{2i})}{\prod_{i=1}^{n} f(y_{2i} \mid \mathbf{q}_{2i}) f(y_{1i} \mid y_{2i}, \mathbf{q}_{1i})} \right\}$$

where $f()$ represents the predictive density, that is, the sampling model with parameters replaced by the corresponding maximum likelihood estimates.

QDG algorithm

QDG stands for QTL-directed dependency graph.

The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

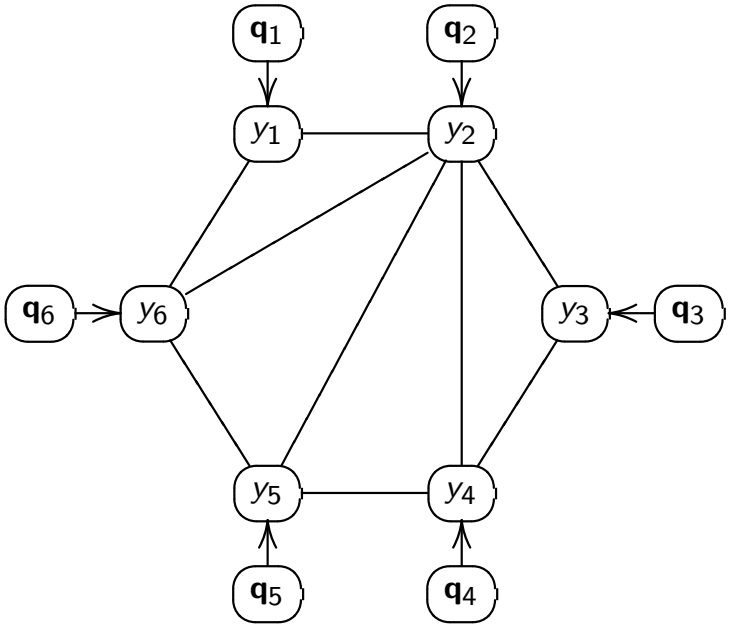
QDG algorithm

QDG stands for QTL-directed dependency graph.
The QDG algorithm is composed of 7 steps:

1. Get the causal skeleton (with the PC skeleton algorithm).
2. Use QTLs to orient the edges in the skeleton.
3. Choose a random ordering of edges, and
4. Recompute orientations incorporating causal phenotypes in the models (update the causal model according to changes in directions).
5. Repeat 4 iteratively until no more edges change direction (the resulting graph is one solution).
6. Repeat steps 3, 4, and 5 many times and store all different solutions.
7. Score all solutions and select the graph with best score.

Step 2

Now suppose that for each transcript we have a set of e-QTLs



Given the QTLs we can distinguish causal direction:

$\mathbf{q}_1 \rightarrow y_1 \rightarrow y_2 \leftarrow \mathbf{q}_2$

$\mathbf{q}_1 \rightarrow y_1 \leftarrow y_2 \leftarrow \mathbf{q}_2$

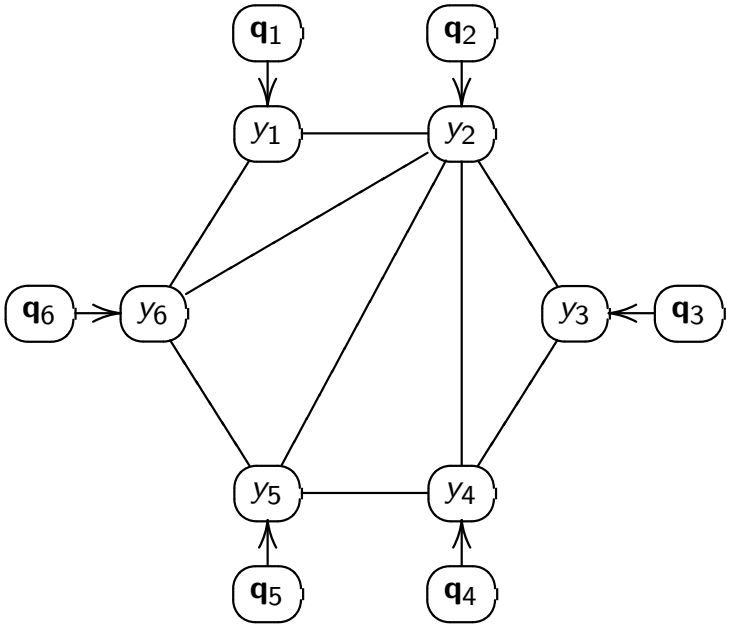
$\vdots$

$\mathbf{q}_5 \rightarrow y_5 \rightarrow y_6 \leftarrow \mathbf{q}_6$

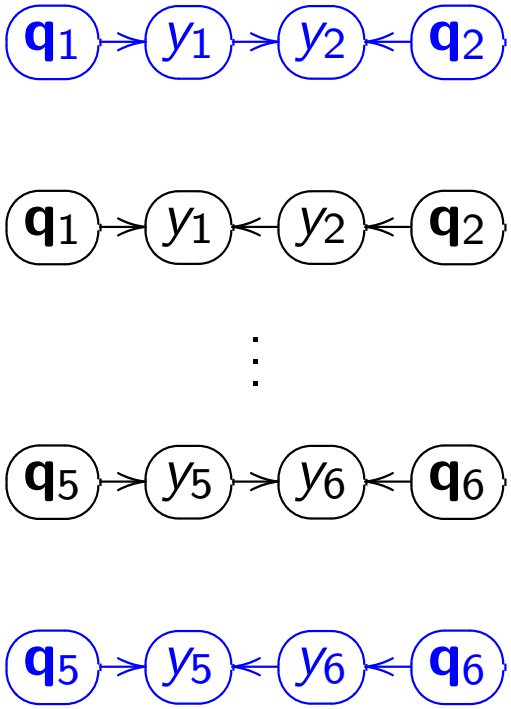
$\mathbf{q}_5 \rightarrow y_5 \leftarrow y_6 \leftarrow \mathbf{q}_6$

Step 2

Now suppose that for each transcript we have a set of e-QTLs

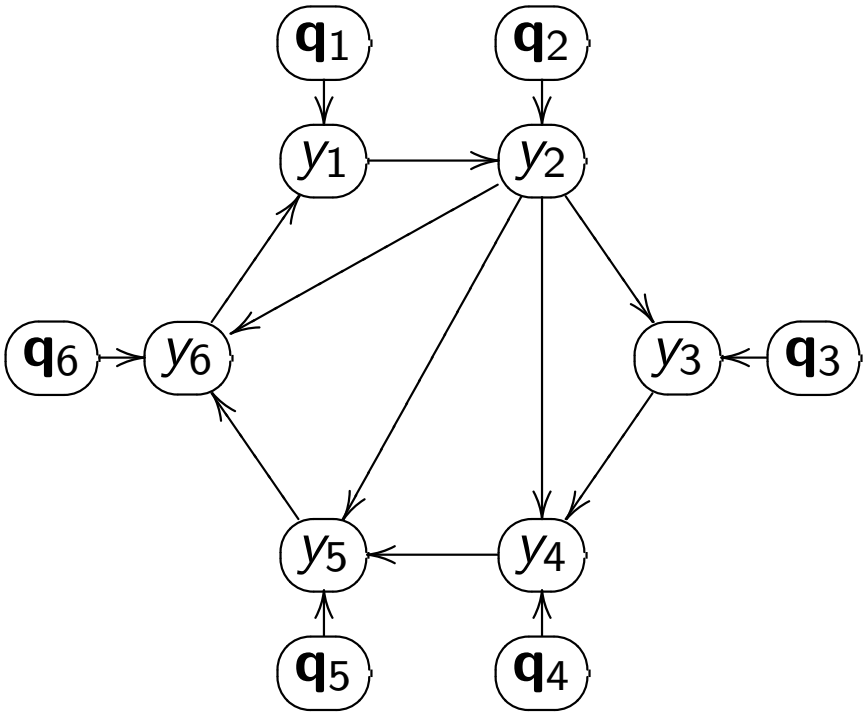


Given the QTLs we can distinguish causal direction:



Steps 2 and 3

First estimate of the causal model (DG_0)



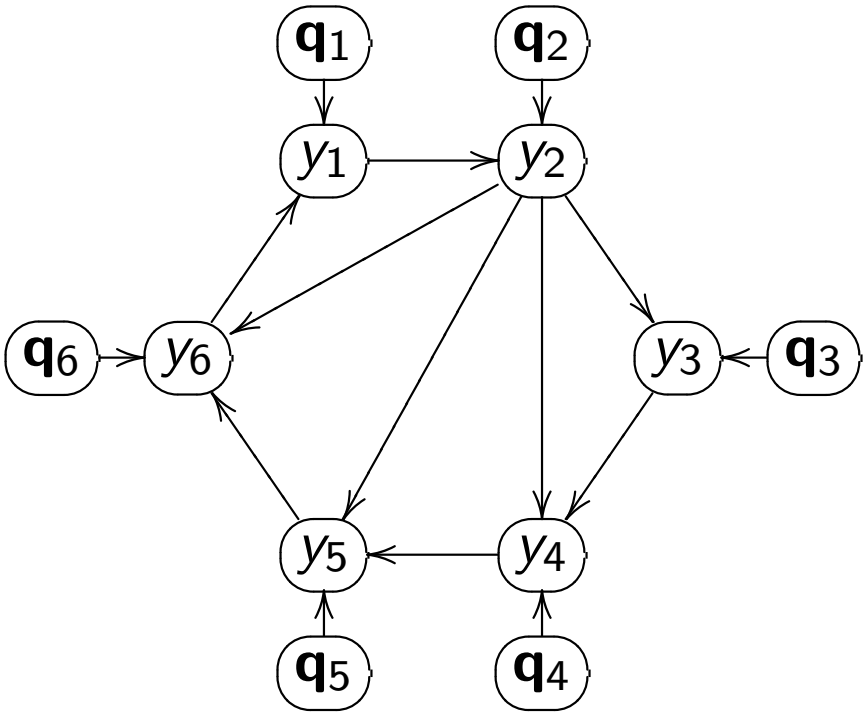
(using only QTLs to infer causal direction)

In step 3 we randomly choose an ordering of all edges in DG_0. Say,

In step 4 we recompute the directions including other transcripts as covariates in the models (following the above ordering).

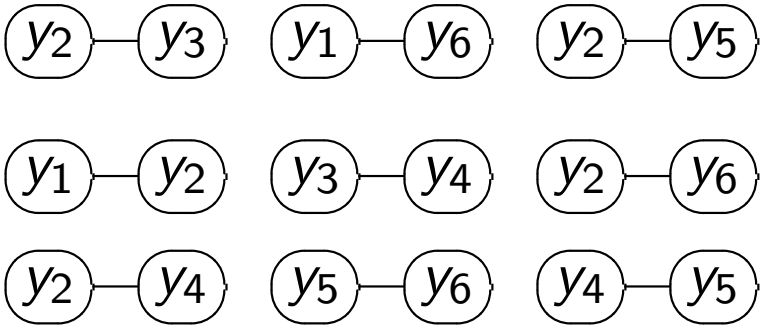
Steps 2 and 3

First estimate of the causal model (DG_0)



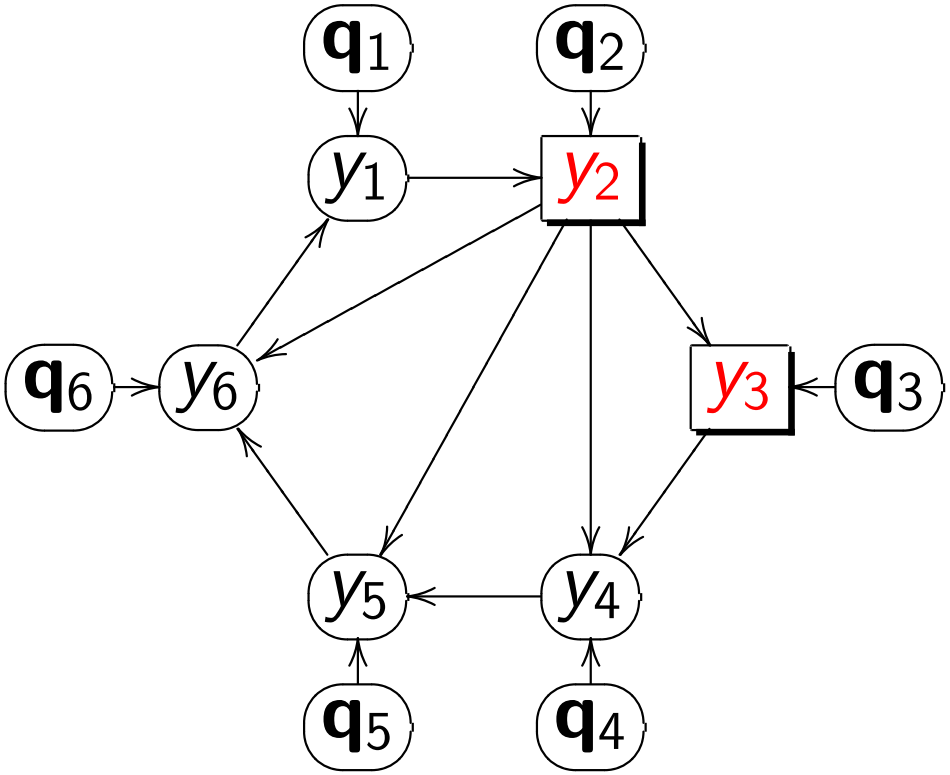
(using only QTLs to infer causal direction)

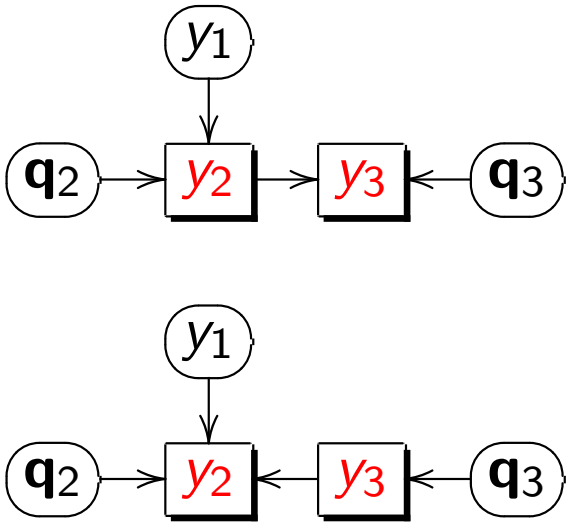
In step 3 we randomly choose an ordering of all edges in DG_0. Say,



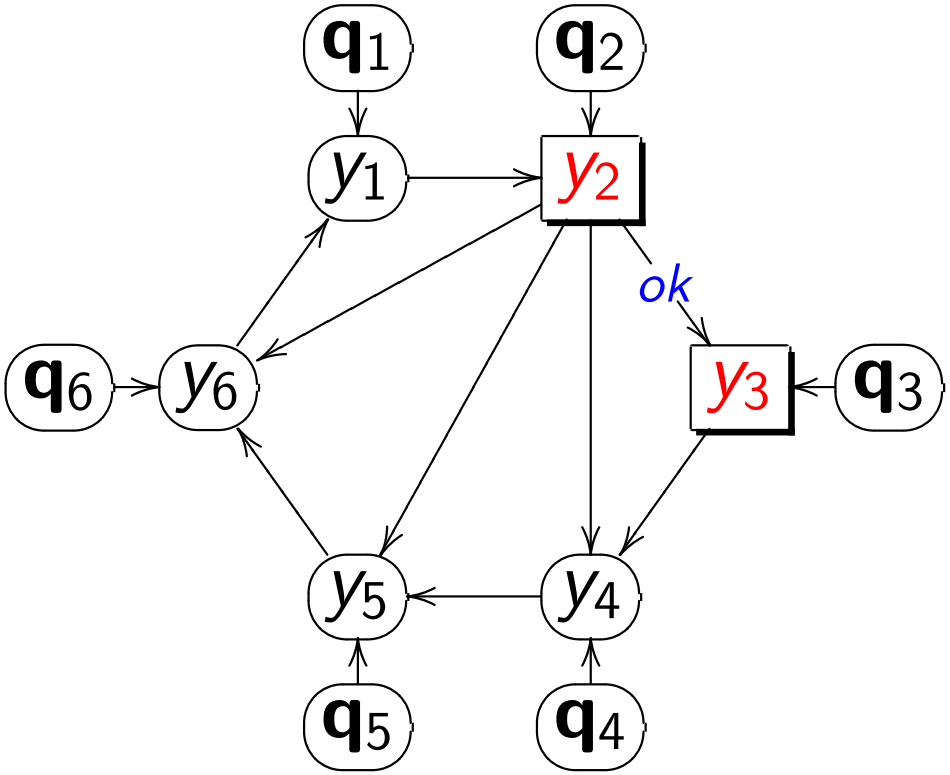
In step 4 we recompute the directions including other transcripts as covariates in the models (following the above ordering).

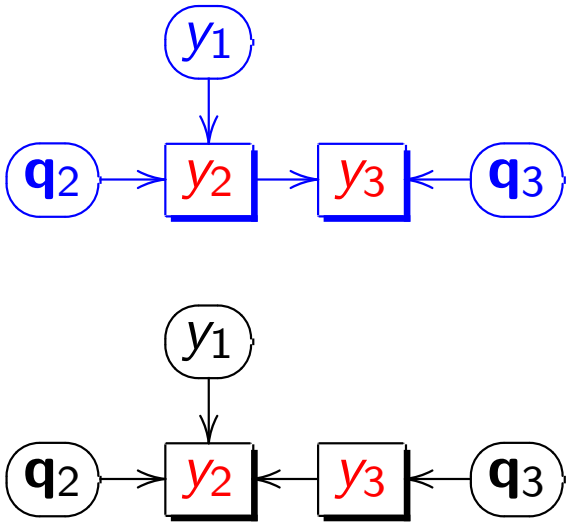
Steps 4 and 5 (first iteration)



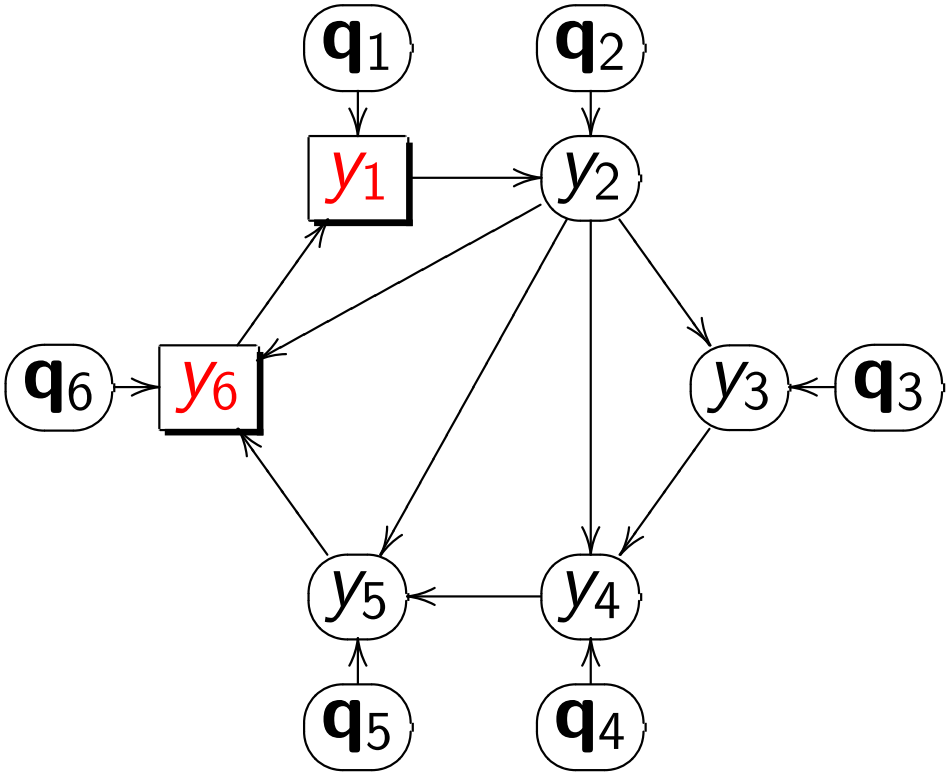


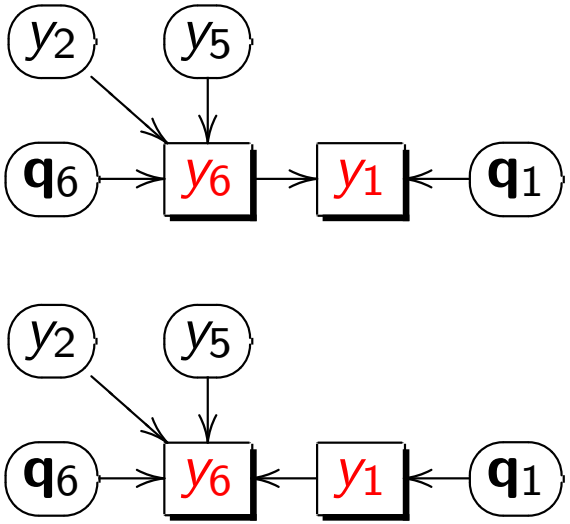
Steps 4 and 5 (first iteration)



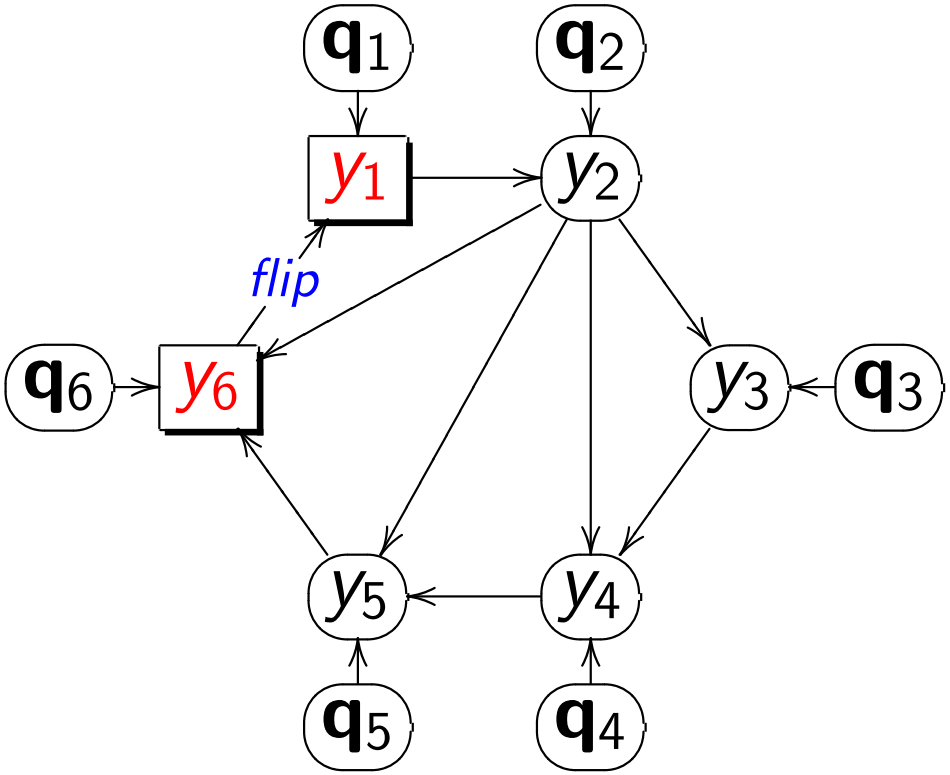


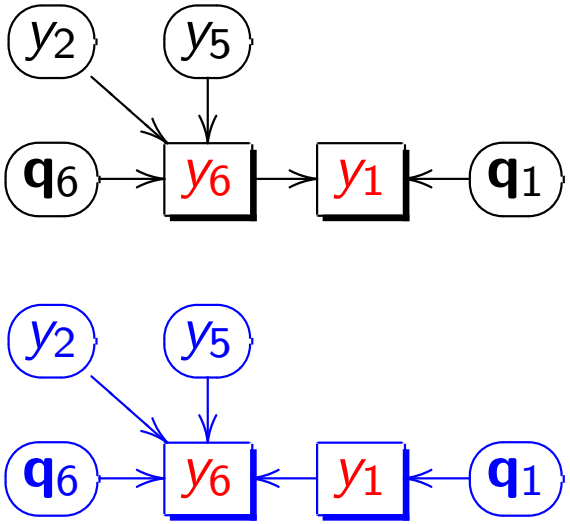
Steps 4 and 5 (first iteration)



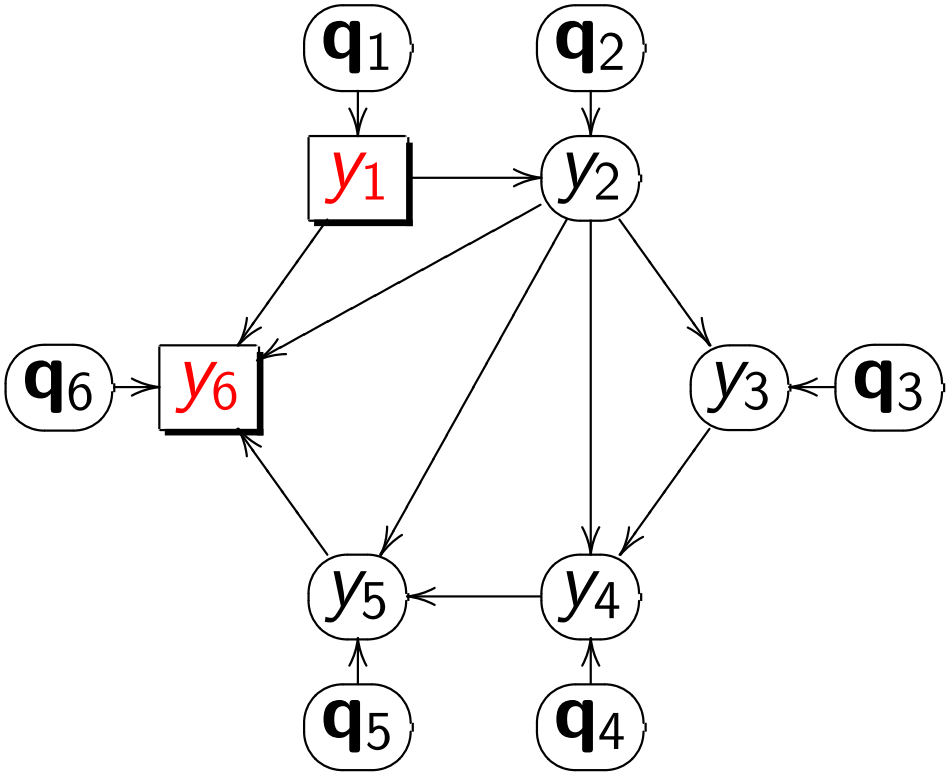


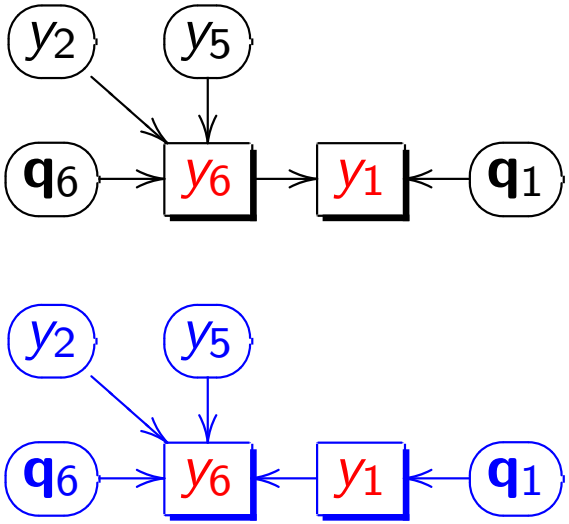
Steps 4 and 5 (first iteration)



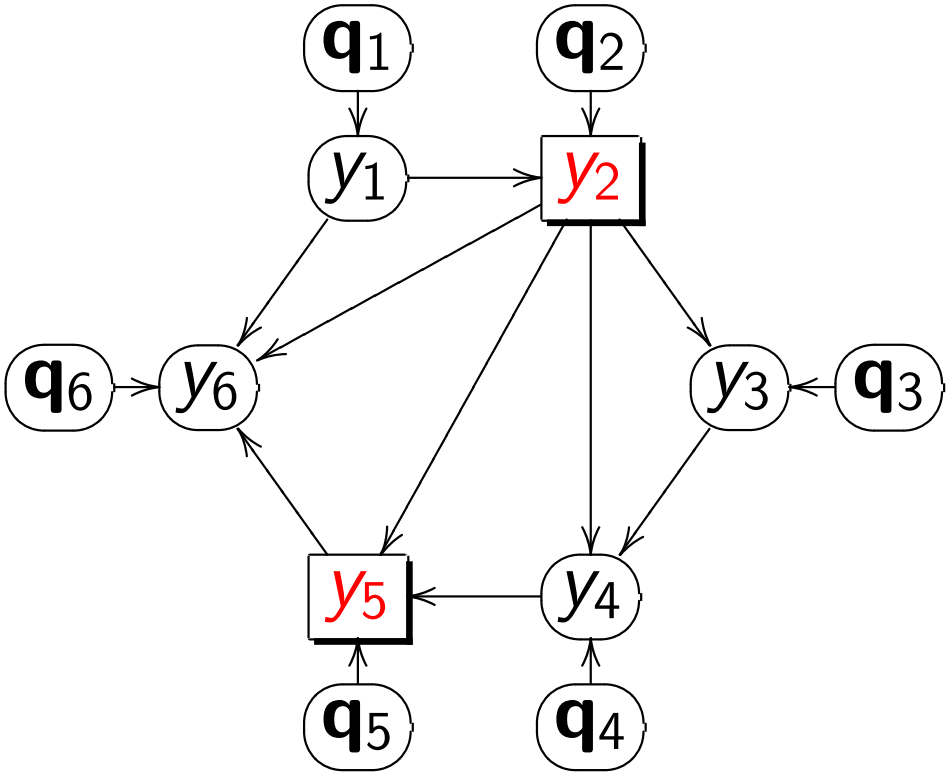


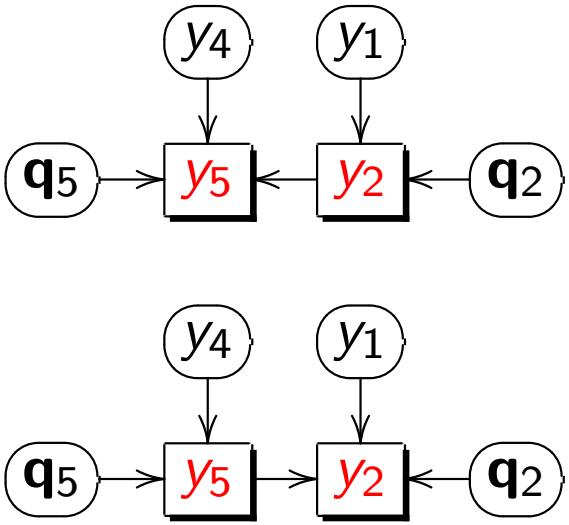
Steps 4 and 5 (first iteration)



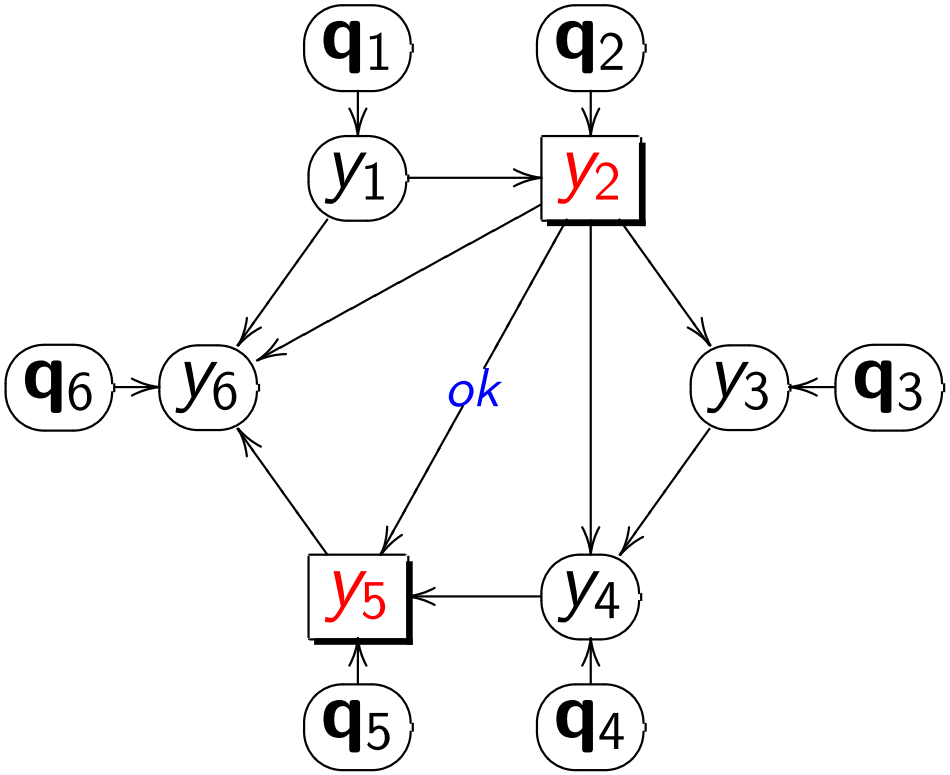


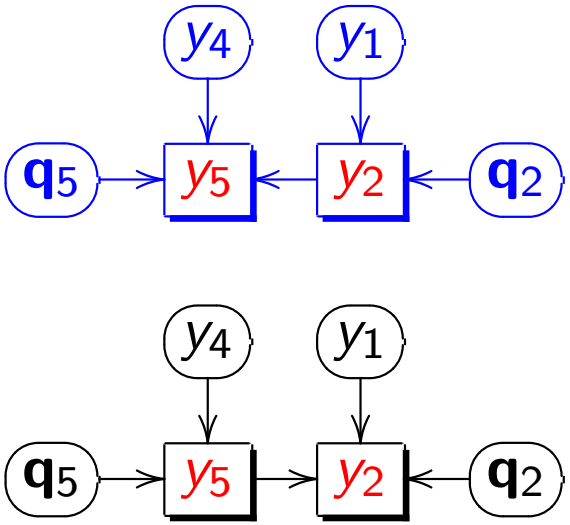
Steps 4 and 5 (first iteration)





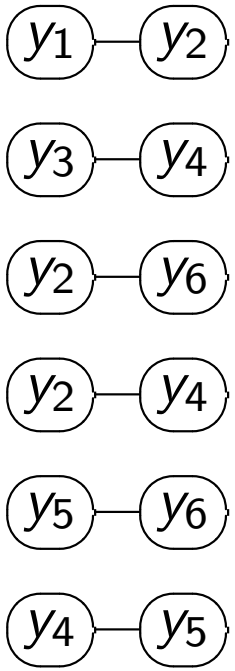
Steps 4 and 5 (first iteration)





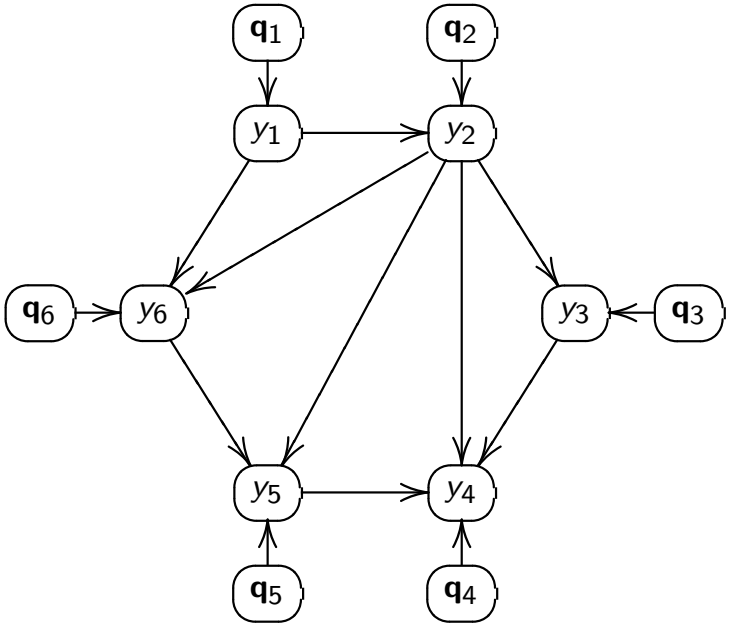
Steps 4 and 5 (first iteration)

And so on until the algorithm recheck the directions for all remaining ordered edges.



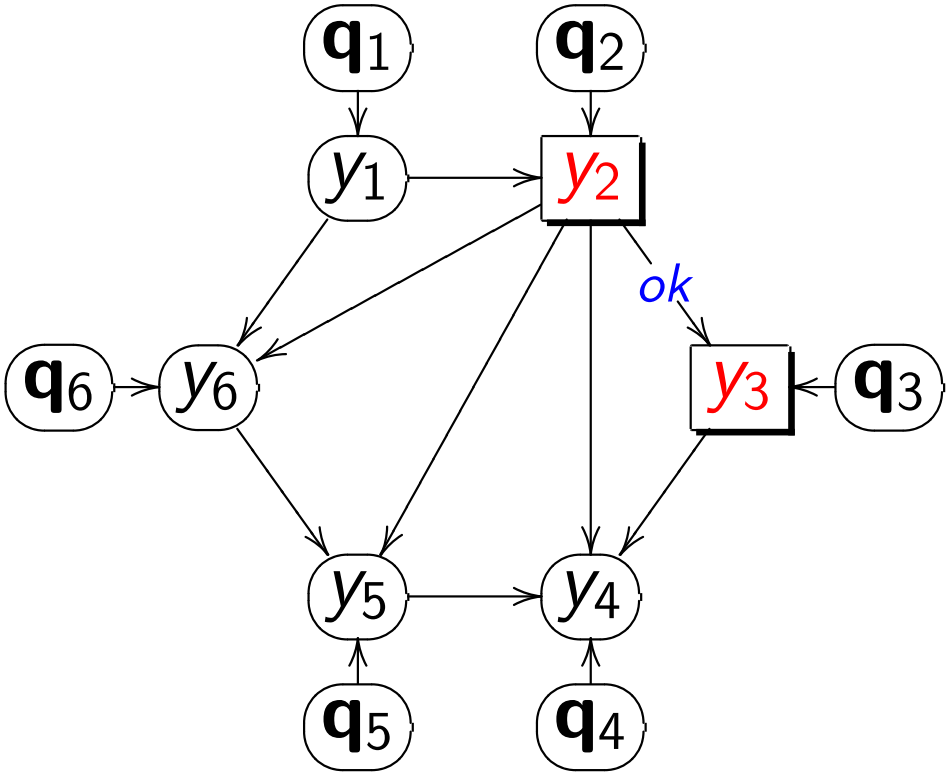
Steps 4 and 5 (first iteration)

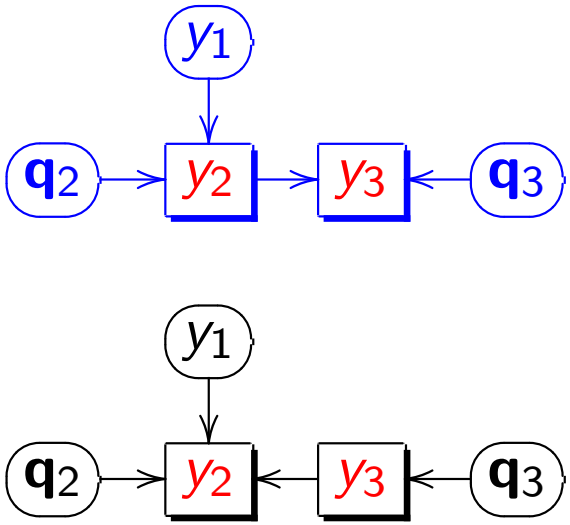
Suppose the updated causal model after the first iteration (DG_1) is



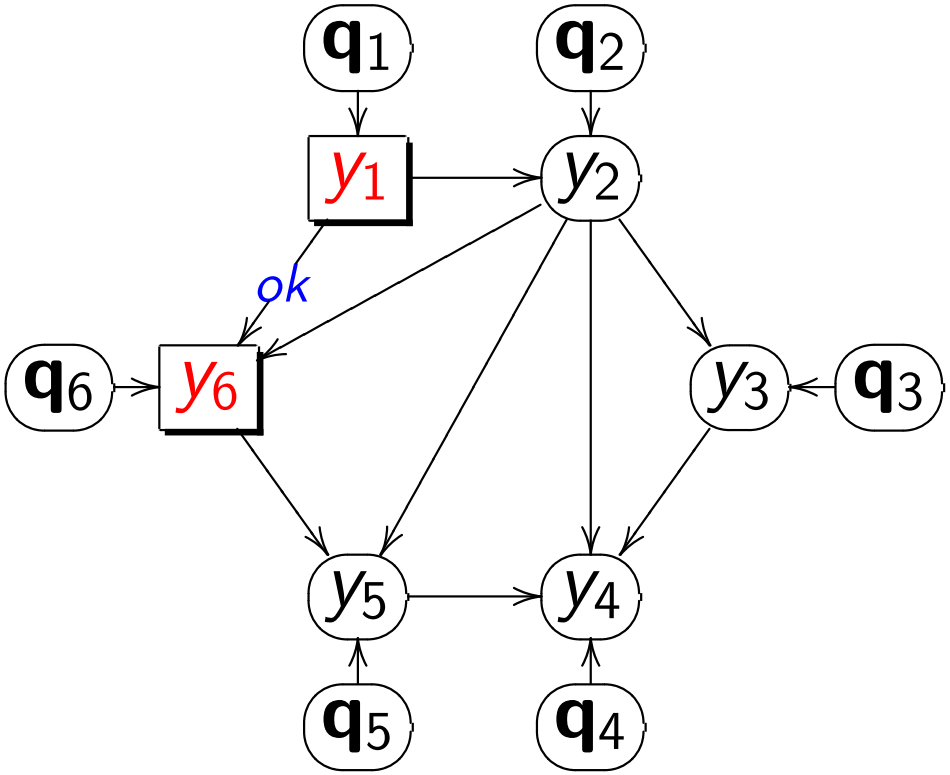
Since some arrows changed direction ($DG_1 \neq DG_0$), the algorithm goes for another round of re-computations.

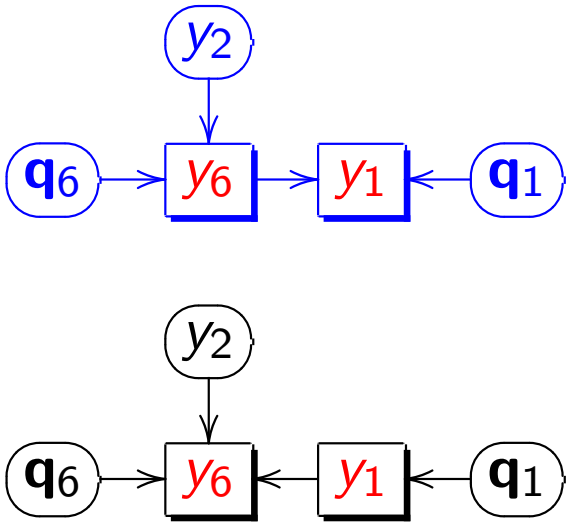
Steps 4 and 5 (second iteration)



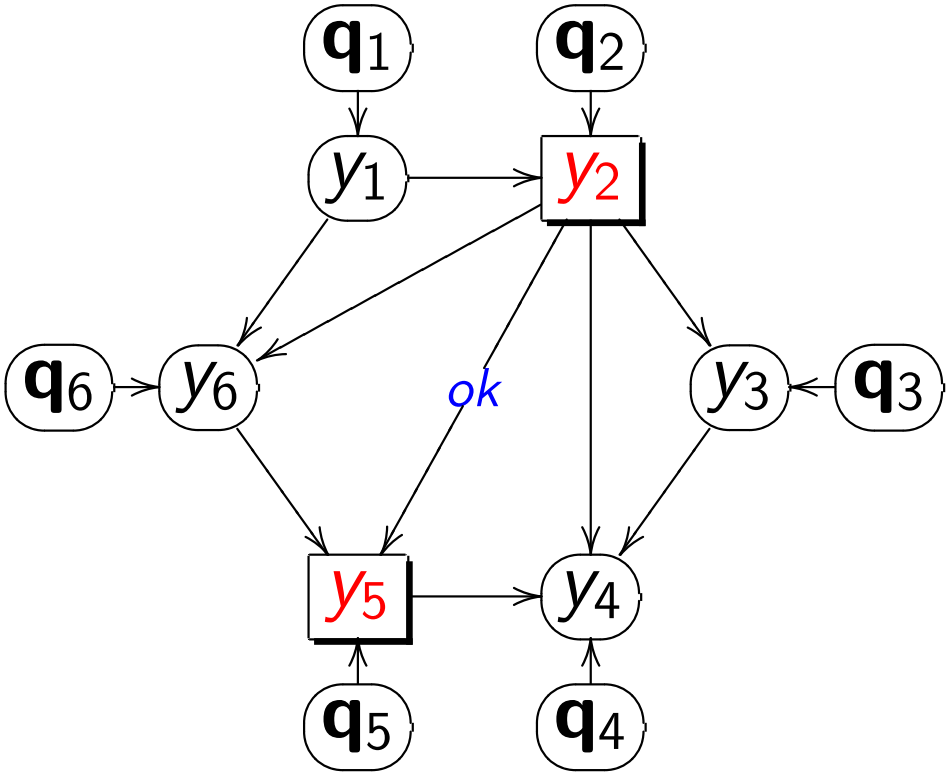


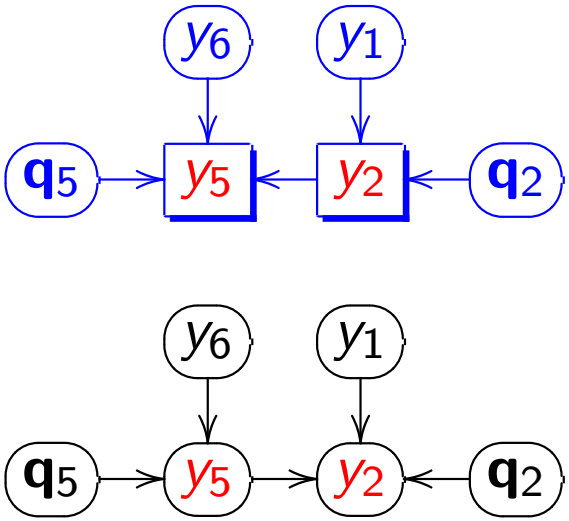
Steps 4 and 5 (second iteration)





Steps 4 and 5 (second iteration)





Steps 4 and 5 (second iteration)

And so on . . .

If no further arrows change direction, the algorithm converged to a solution.

Steps 6 and 7

Different random orderings (step 3) can result in different solutions.

- Step 6: repeat Steps 3 to 5 many times and store all different solutions.
- Step 7: score all solutions and select the graph with best score (maximized log-likelihood or BIC).

Steps 6 and 7

Different random orderings (step 3) can result in different solutions.

- Step 6: repeat Steps 3 to 5 many times and store all different solutions.
- Step 7: score all solutions and select the graph with best score (maximized log-likelihood or BIC).

Steps 6 and 7

Different random orderings (step 3) can result in different solutions.

- Step 6: repeat Steps 3 to 5 many times and store all different solutions.
- Step 7: score all solutions and select the graph with best score (maximized log-likelihood or BIC).

Sparsity assumption

The PC skeleton algorithm and QDG algorithm perform well in sparse graphs.

Directing edges without QTLs

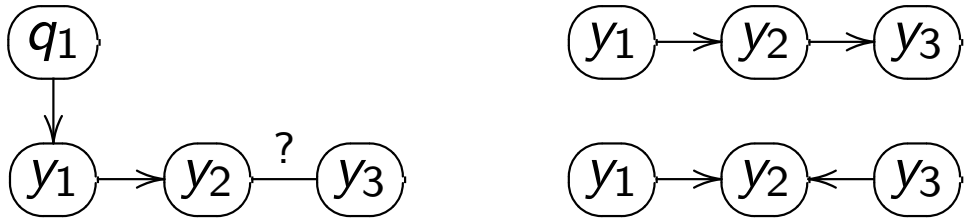
- In general we need to have at least one QTL per pair of phenotypes to infer causal direction.
- In some situations, however, we may be able to infer causal direction for a pair of phenotypes without QTLs. Eg.

since $f(y_1)\,f(y_2 \mid y_1)\,f(y_3 \mid y_2) \neq f(y_1)\,f(y_2 \mid y_1, y_3)\,f(y_3)$.

- So both QTLs and phenotypes play important roles in the orientation process.

Directing edges without QTLs

- In general we need to have at least one QTL per pair of phenotypes to infer causal direction.
- In some situations, however, we may be able to infer causal direction for a pair of phenotypes without QTLs. Eg.

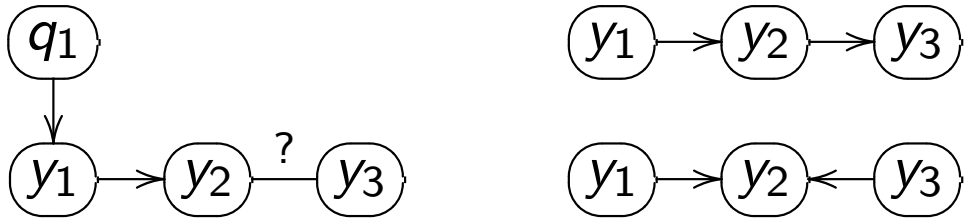


since $f(y_1)\, f(y_2 \mid y_1)\, f(y_3 \mid y_2) \neq f(y_1)\, f(y_2 \mid y_1, y_3)\, f(y_3)$.

- So both QTLs and phenotypes play important roles in the orientation process.

Directing edges without QTLs

- In general we need to have at least one QTL per pair of phenotypes to infer causal direction.
- In some situations, however, we may be able to infer causal direction for a pair of phenotypes without QTLs. Eg.

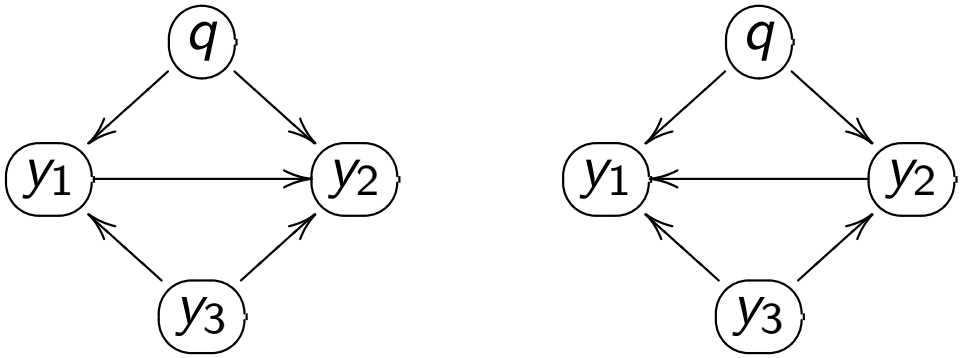


since $f(y_1)\, f(y_2 \mid y_1)\, f(y_3 \mid y_2) \neq f(y_1)\, f(y_2 \mid y_1, y_3)\, f(y_3)$.

- So both QTLs and phenotypes play important roles in the orientation process.

Unresolvable situation

- We cannot infer direction when the phenotypes have exactly same set of QTLs and causal phenotypes



since

$$f(y_1 \mid y_3, q)\, f(y_2 \mid y_1, y_3, q) \;=\; f(y_1 \mid y_2, y_3, q)\, f(y_2 \mid y_3, q)$$

Reducing graph space

The QDG algorithm drastically reduces the number of graphs that need to be scored.

1. The maximum number of graphs is 2^k models, where k is the number of edges in the skeleton.

2. The number of solutions of the QDG algorithm is generally much smaller than 2^k.

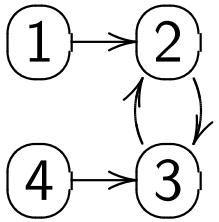
Cyclic networks

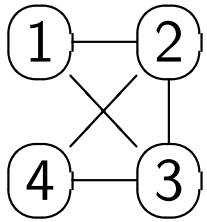
- Cycles are a common feature of biological networks (homeostatic mechanisms).
- The PC skeleton algorithm assumes an acyclic causal graph, and cycles may lead to spurious edges. E.g.

$1 \not\perp 3 \quad 1 \not\perp 3 \mid 2 \quad 1 \not\perp 3 \mid 2,4$
$2 \not\perp 4 \quad 2 \not\perp 4 \mid 3 \quad 2 \not\perp 4 \mid 1,3$

Cyclic networks

- Cycles are a common feature of biological networks (homeostatic mechanisms).
- The PC skeleton algorithm assumes an acyclic causal graph, and cycles may lead to spurious edges. E.g.



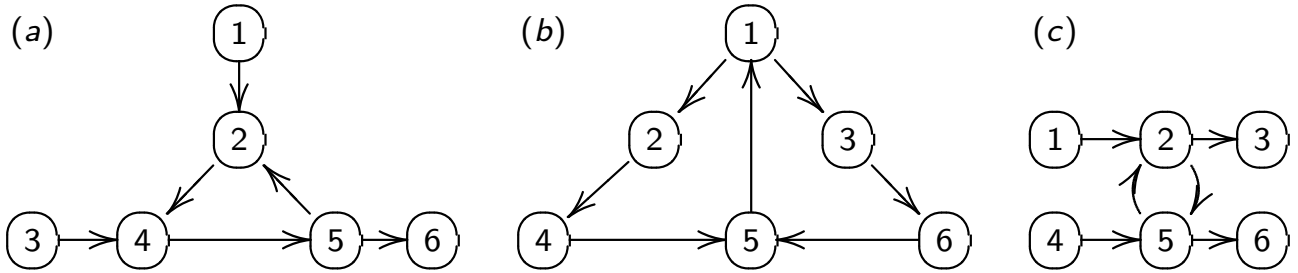


$1 \not\perp 3$ $\quad 1 \not\perp 3 \mid 2$ $\quad 1 \not\perp 3 \mid 2, 4$

$2 \not\perp 4$ $\quad 2 \not\perp 4 \mid 3$ $\quad 2 \not\perp 4 \mid 1, 3$

Cyclic networks

- Our simulations showed good performance with toy cyclic graphs, though.

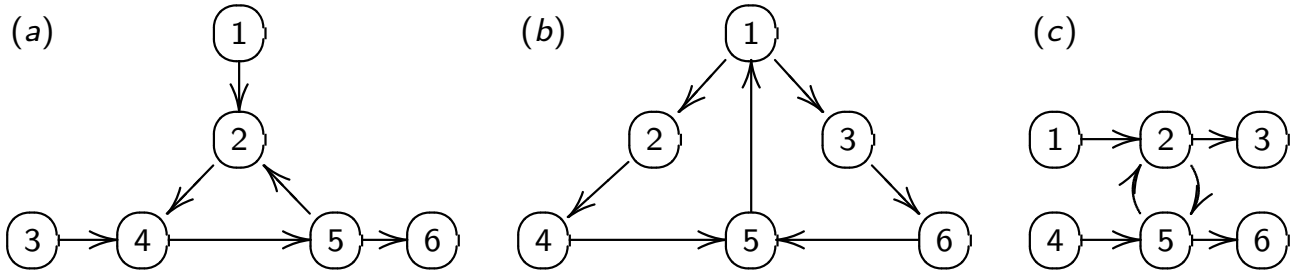


- The spurious edges in graph (c) were detected at low rates.

- QDG approach cannot detect reciprocal interactions. In graph (c) it orients the edge (2)—(5) in the direction with higher strength.

Cyclic networks

- Our simulations showed good performance with toy cyclic graphs, though.

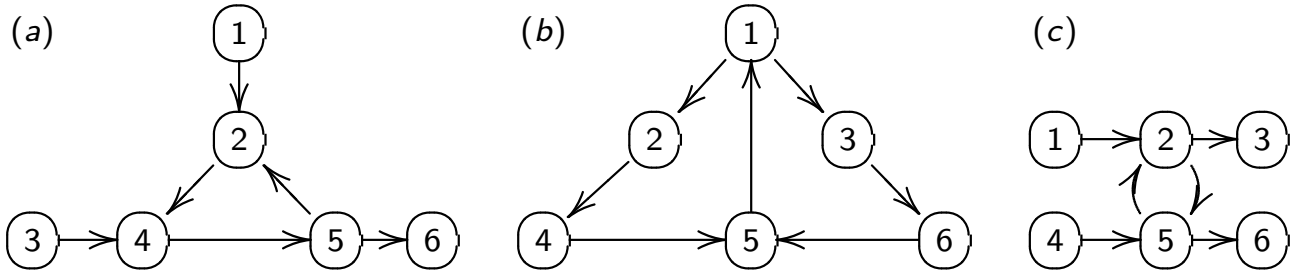


- The spurious edges in graph (c) were detected at low rates.

- QDG approach cannot detect reciprocal interactions. In graph (c) it orients the edge (2)—(5) in the direction with higher strength.

Cyclic networks

- Our simulations showed good performance with toy cyclic graphs, though.

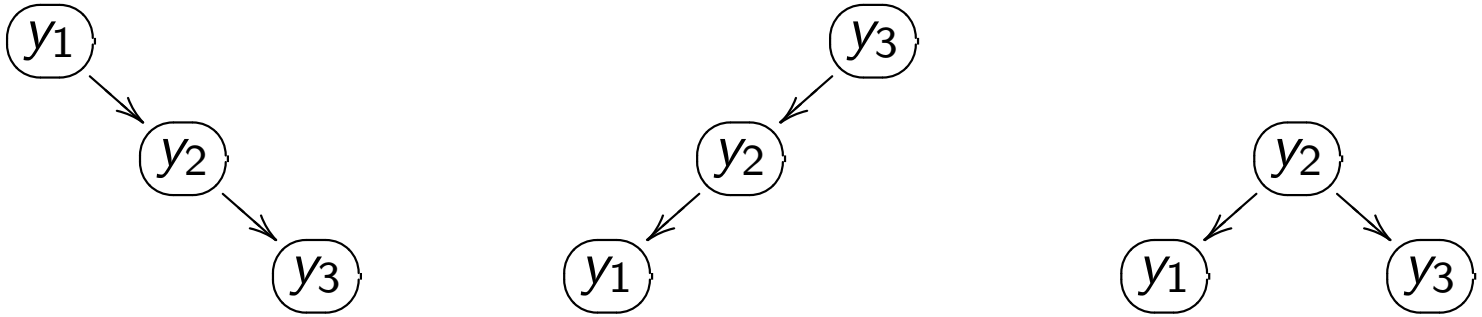


- The spurious edges in graph (c) were detected at low rates.

- QDG approach cannot detect reciprocal interactions. In graph (c) it orients the edge (2)—(5) in the direction with higher strength.

Unique graph instead of an equivalence class

Two DAGs are Markov equivalent iff they have the same skeleton and the same set of v-structures. For example

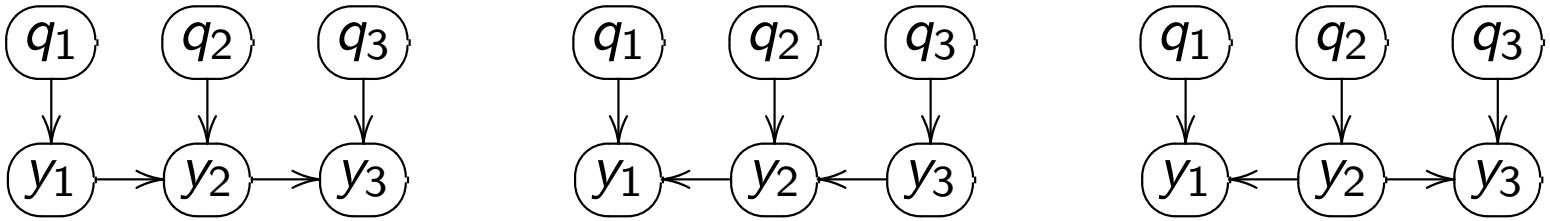


The three graphs have the same skeleton, 

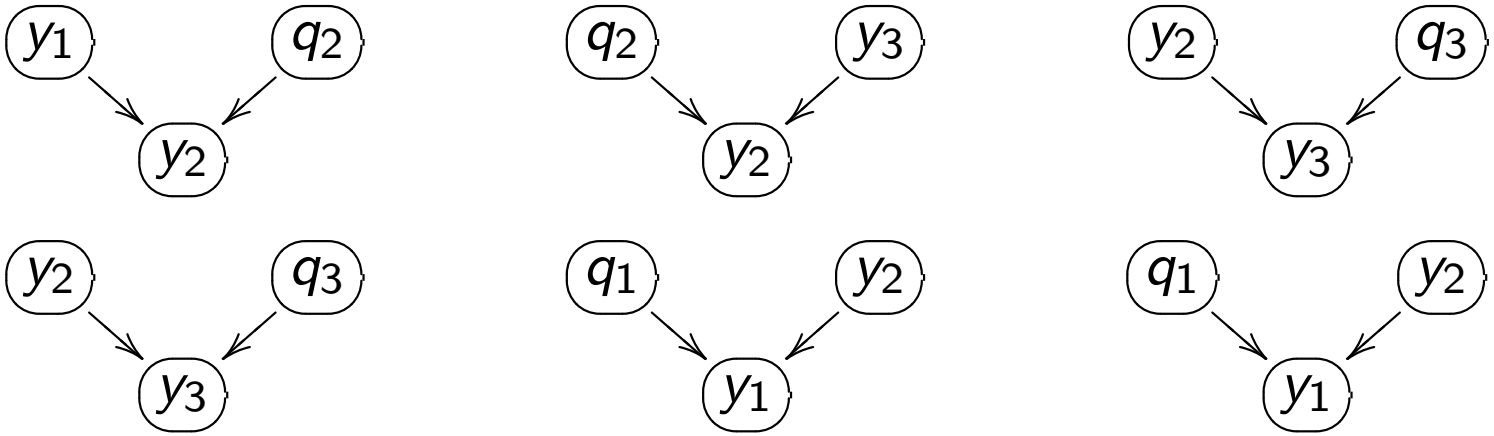
, and the same set of v-structures (none).

The graphs will also be likelihood equivalent if we assume a linear regression with Gaussian errors.

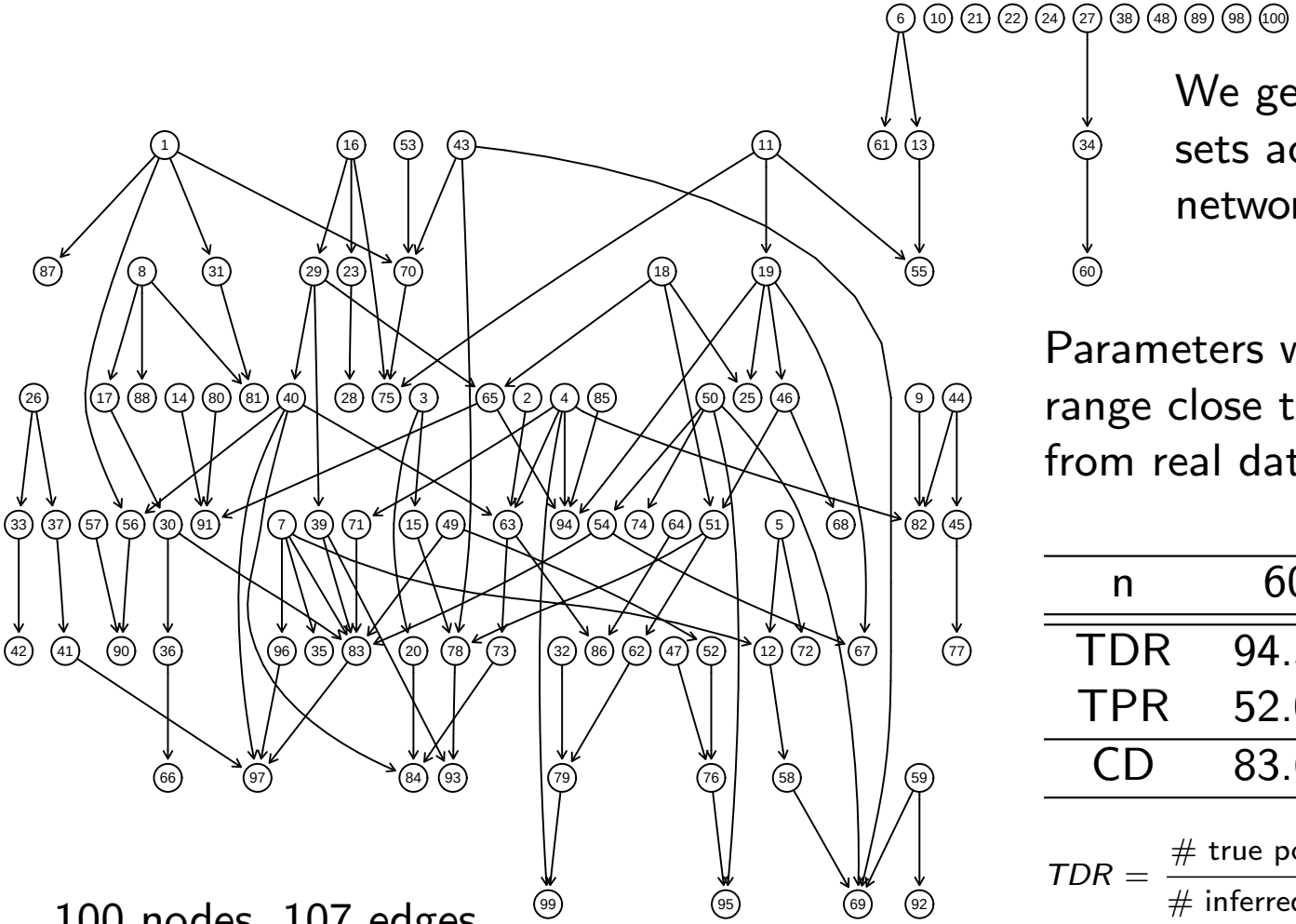
Unique graph instead of an equivalence class



Same skeleton, but different sets of v-structures



Simulations



100 nodes, 107 edges

2 or 3 QTLs per phenotype (not shown)

We generated 100 data sets according to this network.

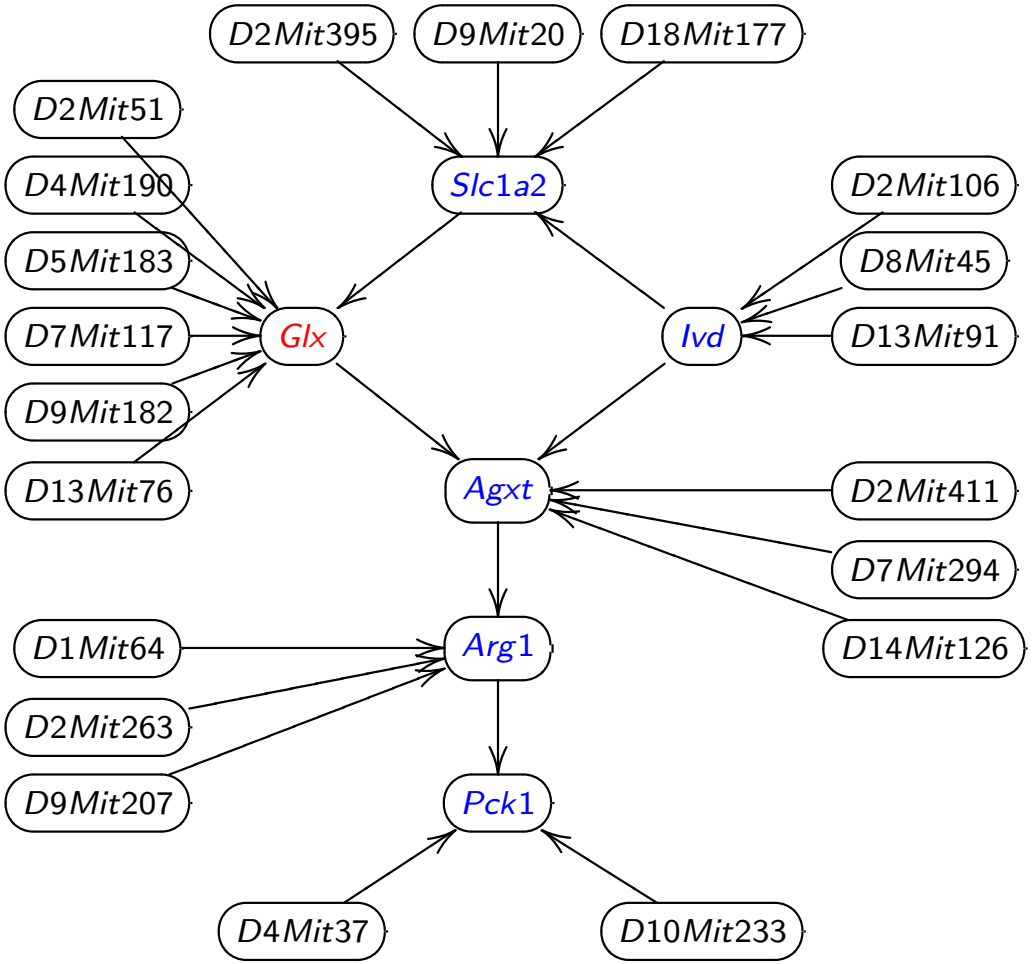
Parameters were chosen in a range close to values estimated from real data.

n	60	300	500
TDR	94.53	95.18	91.22
TPR	52.07	87.33	93.64
CD	83.65	98.58	99.63

$$TDR = \frac{\#\text{ true positives}}{\#\text{ inferred edges}}, \quad TPR = \frac{\#\text{ true positives}}{\#\text{ true edges}}$$

CD: correct direction

Real data example



- We constructed a network from metabolites and transcripts involved in liver metabolism.
- We validated this network with *in vitro experiments* (Ferrara et al 2008). Four out of six predictions were confirmed.

Software and references

The *qdg* R package is available at CRAN.

References:

- Chaibub Neto et al 2008. Inferring causal phenotype networks from segregating populations. Genetics 179: 1089-1100.
- Ferrara et al 2008. Genetic networks of liver metabolism revealed by integration of metabolic and transcriptomic profiling. PLoS Genetics 4: e1000034.
- Spirtes et al 1993. Causation, prediction and search. MIT press.

Acknowledgements

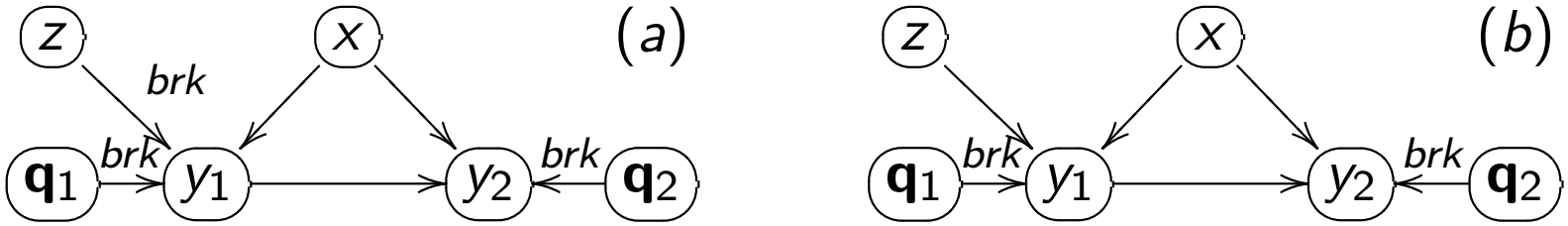
Co-authors:

- Alan D. Attie
- Brian S. Yandell
- Christine T. Ferrara

Funding:

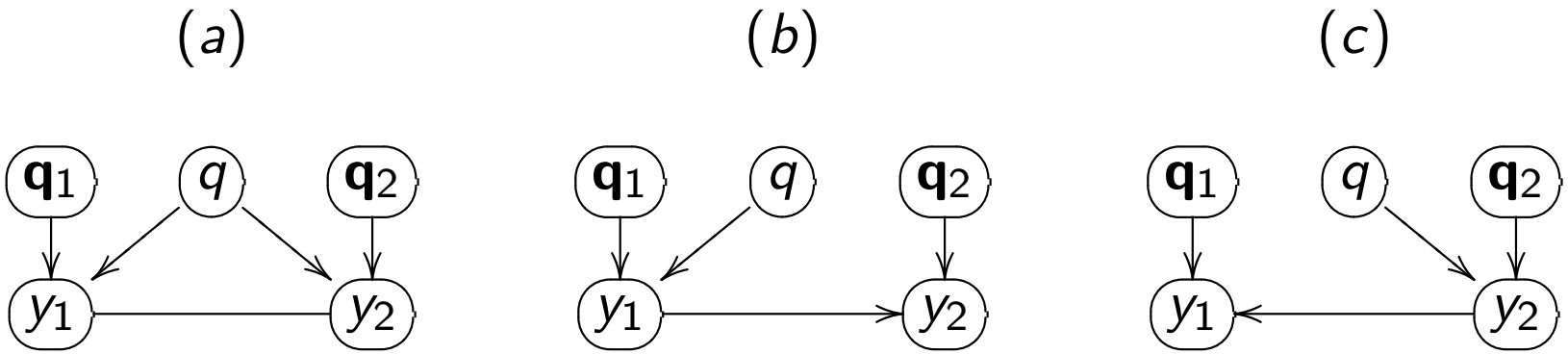
- CNPq Brazil
- NIH grants DK66369, DK58037 and DK06639

Permutation p-values



- To break the connections (brk) that affect direction of an edge, we permute the corresponding pair of nodes (and their common covariates) as a block.
- In panel (a) we permute (y_1, y_2, x) as a block breaking the connections with z, $\mathbf{q}_1$ and $\mathbf{q}_2$;
- In panel (b) we incorrectly keep z in the permutation block.

Direct versus indirect effects of a common QTL



- A strong QTL directly affecting an upstream trait may also be (incorrectly) detected as a QTL for a downstream phenotype.
- To resolve this situation we apply a generalization of Schadt et al. 2005 allowing for multiple QTLs.
- Model (a) supports both traits being directly affected by the common QTL q. Model (b) implies that q directly affects y_1 but should not be included as a QTL of phenotype y_2. Model (c) supports the reverse situation.