

Untitled

Marcos godoy e Wellingson Araujo

16/07/2019

Mapemento de QTL

O objetivo desse trabalho foi ter contato com tres metodologias de mapeamento de QTL podendo assim ter uma base mais solida sobre os metodos que ja foram e que ainda hoje sÃ£o utilizados para este fim.

Procederemos da seguinte maneira, a principio sera abordado o metodo de Inteval mapping (mapeamento por intervalo), Composite Interval mapping (mapeamento por intervalo composto) e Multiple interval mapping (mapemaento por multiplos intervalos)

A caracteristica que foi escolhida para esse trabalho Ã© a numero 8 - viabilidade do polÃ©m

Inicialmente chamamos o pacote que sera utilizado - Nesse caso sera o RQTL

```
library(qtl)
```

@ Entrada dos dados

Nesse passo chamaos o arquivo tipo raw que possui tanto as caracteristicas fenotipicas quanto os dados de marcadores e txt que apresenta a saida do mapa de ligacao que foi construido anteriormenete, tambem realizado como atividade da disciplina.

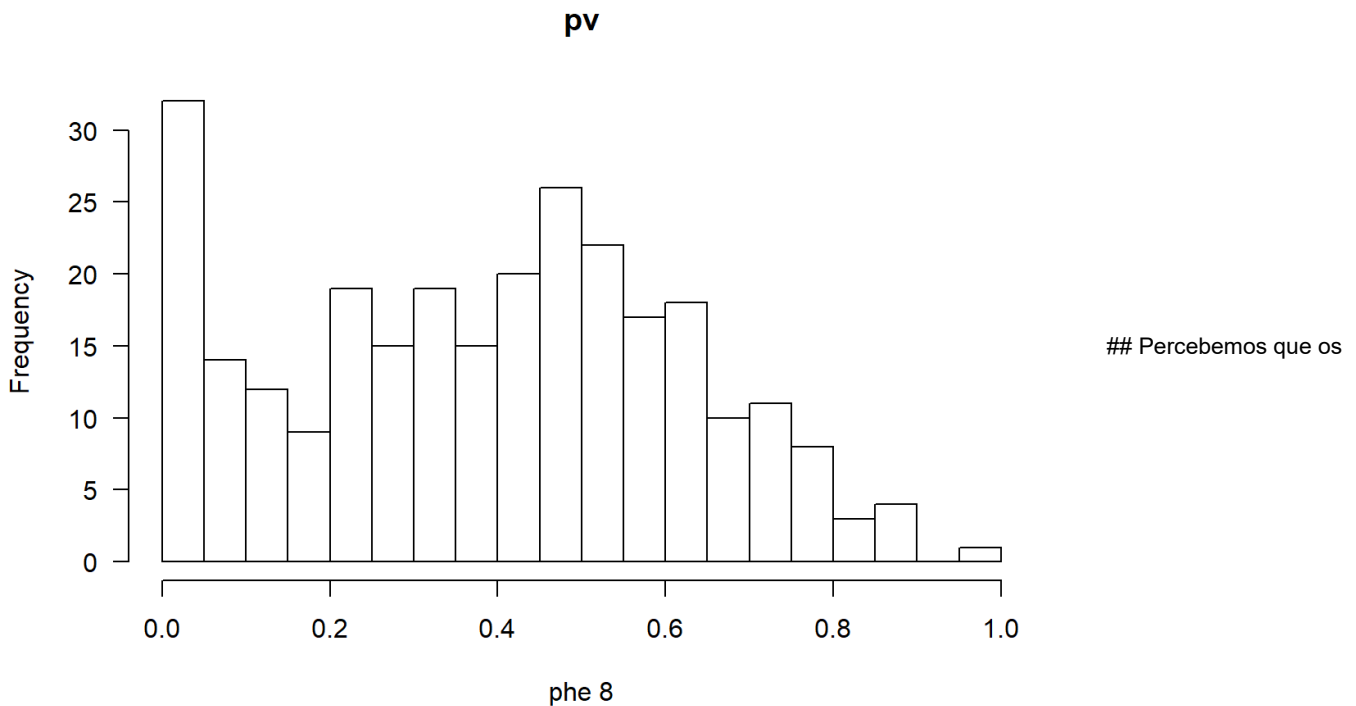
```
mapamimulus <-read.cross(format = "mm", file = "m_feb06.raw" , mapfile = "mimulus.map")
```

```
## --Read the following data:
## Type of cross:          f2
## Number of individuals: 287
## Number of markers:     418
## Number of phenotypes:  16
## --Cross type: f2
```

PRIMEIRO METODO - MAPEAMENTO POR INTERVALO

Plotando os dados e possível verificar a distribuição que os dados da característica quantitativa escolhida segue.

```
plotPheno(mapamimulus,8)
```



dados segue uma distribuição normal

A função abaixo `CALC.GENOPROB` foi utilizada para estimar a probabilidade de qtl dado o genótipo dos marcadores flaqueadores do intervalo. O erro aqui foi fixado em 0,001, esse valor pode ser considerado bem razoável para esse tipo de análise.

```
qtl<- calc.genoprob(mapamimulus, step = 1, error.prob = 0.001)
```

Para a análise dos dados podem ser mapeados QTL segundo algumas metodologias.

Primeiramente foi utilizado o algoritmo EM - um algoritmo completo, muito utilizado e velho conhecido nosso na disciplina, entretanto demanda um esforço computacional grande comparado a outras abordagens aqui também fizemos uma análise utilizando um método de aproximação muito razoável que tem base na regressão linear o qual é denominado de Haley Knott

```
em <- scanone(qtl, method = "em", pheno.col = 8) # mapeando via EM
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 12 individuals with missing phenotypes.
```

```
hk <- scanone(qtl, method = "hk", pheno.col = 8) # mapeando via Haley-Knott
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 12 individuals with missing phenotypes.
```

```
summary(em)
```

	chr <fctr>	pos <dbl>	lod <dbl>
BC194C	1	181.37246	1.8492553
c2.loc47	2	47.00000	1.8184249
c3.loc196	3	196.00000	1.5965570
CC387	4	97.09738	0.8837442
CA258C	5	56.82428	1.3924252
c6.loc56	6	56.00000	1.9732071
MgSTS586	7	38.30073	1.3364204
c8.loc178	8	178.00000	4.0668772
MgSTS31	9	72.45737	2.0634686
c10.loc18	10	18.00000	2.6759450
1-10 of 14 rows			Previous 1 2 Next

```
summary(hk)
```

	chr <fctr>	pos <dbl>	lod <dbl>
c1.loc181	1	181.00000	1.8340104
c2.loc51	2	51.00000	2.0286079
c3.loc196	3	196.00000	1.5702221
CC387	4	97.09738	0.7108106
CA258C	5	56.82428	1.3987421
c6.loc56	6	56.00000	1.9424727
c7.loc39	7	39.00000	1.3276499
c8.loc177	8	177.00000	4.1988149
MgSTS31	9	72.45737	2.0590606
c10.loc18	10	18.00000	2.6520892
1-10 of 14 rows			Previous 1 2 Next

Os resultados mostram os valores dos maiores lods em cada cromossomo e sua posicao

Como podemos perceber temos grande semelhancas entre as saidas dos dois metodos, por isso os dois

podem ser utilizados.

Estimando limiar via permutacoes

O limiar da analise pode ser estimado por meio de permutacoes, essa metodologia tem grande utilizacao devido as vantagens que ela possui como um ajuste aos dados, por ser calcula com base nestes, ou seja, a depender dos dados o limiar pode mudar, diferente de quando utilizamos um limiar fixo.

Aqui foram simuladas 3000 permutações, esse numero é criterio do pesquisador que estiver realizado a analise, nesse caso escolhemos 3 mil, um numero razoavel é acima de mil permutações.

```
lod <- scanone(qtl, method = "hk", n.perm = 3000, pheno.col = 8)
```

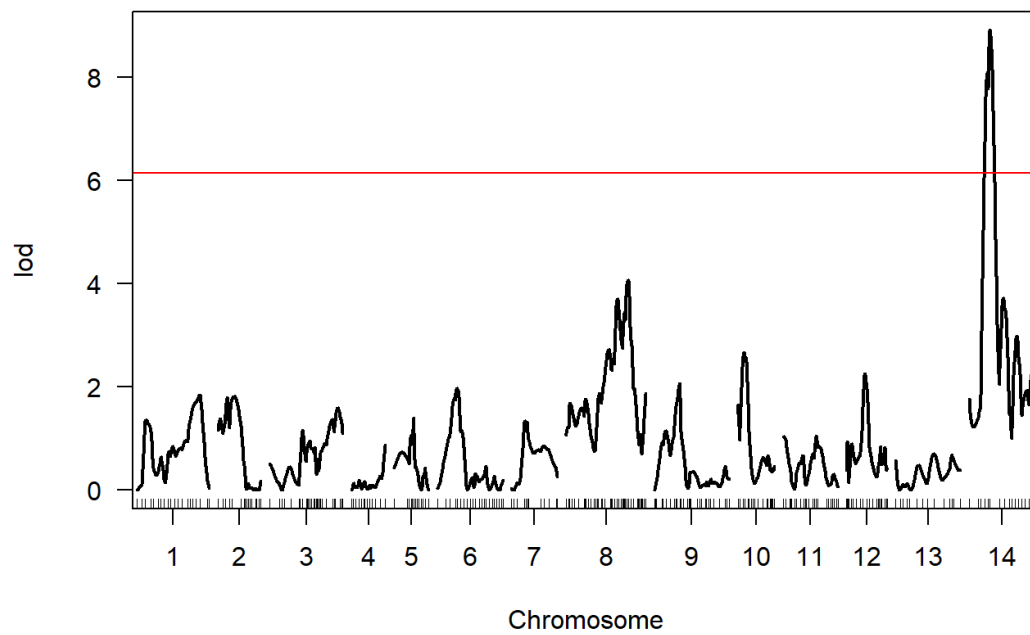
```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 12 individuals with missing phenotypes.
```

```
## Doing permutation in batch mode ...
```

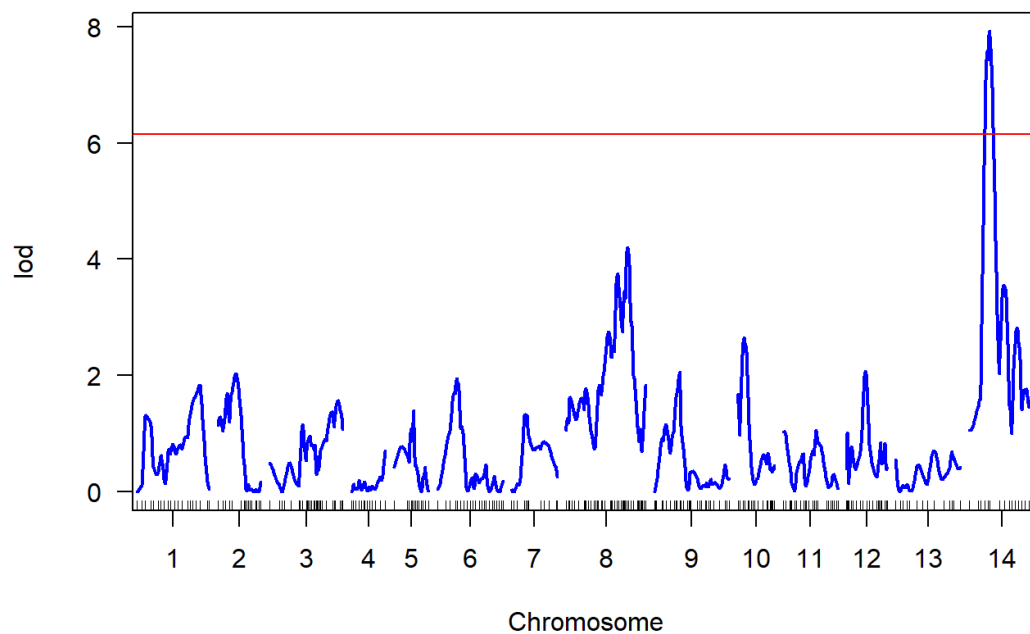
```
lod1 <- summary(lod, alpha = 0,05)  
lod1 <- as.numeric(lod1)
```

Utilizando as funcoes seguintes nos conseguiremos ter acesso a plotagem dos resultados , onde é possivel ver onde existem possiveis Qtl dando uma melhor compreensão da caracteristica.

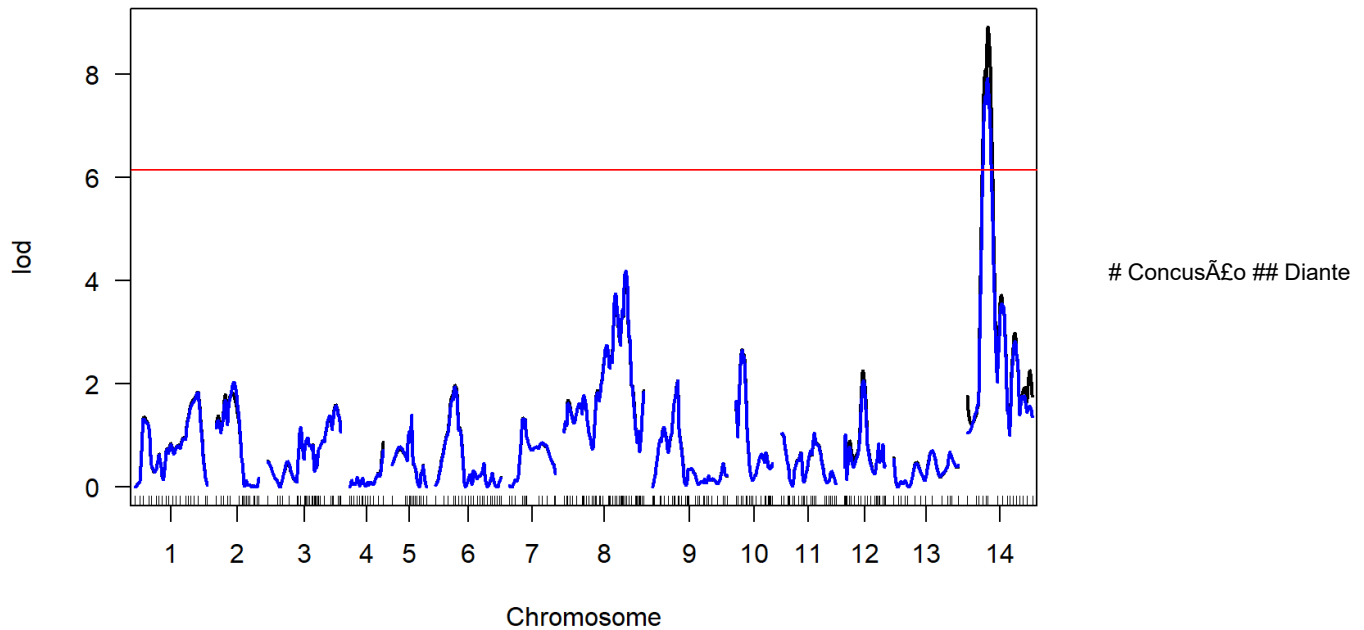
```
plot(em, col = c ("black"))  
abline(h = lod1, col = "red")
```



```
plot(hk, col = c ("blue"))  
abline(h = lod1, col = "red")
```



```
plot(em,hk, col = c ("black", "blue"))  
abline(h = lod1, col = "red")
```



dos graficos percebos que ambos os metodos detectaram um possivel QTL no cromossomo 14 proximo a posiÃ§Ã£o entre 58 e 59.8, as quais podem ser observadas no sumary

O segundo metodo Ã© o mapeamento por intervalo composto, esse metodo Ã© superior ao metodo anteriormente utilizado, pois o mapeamento por intervalo nÃ£o leva em consideracao a possibilidade de outros Qtl ao longo do genoma.

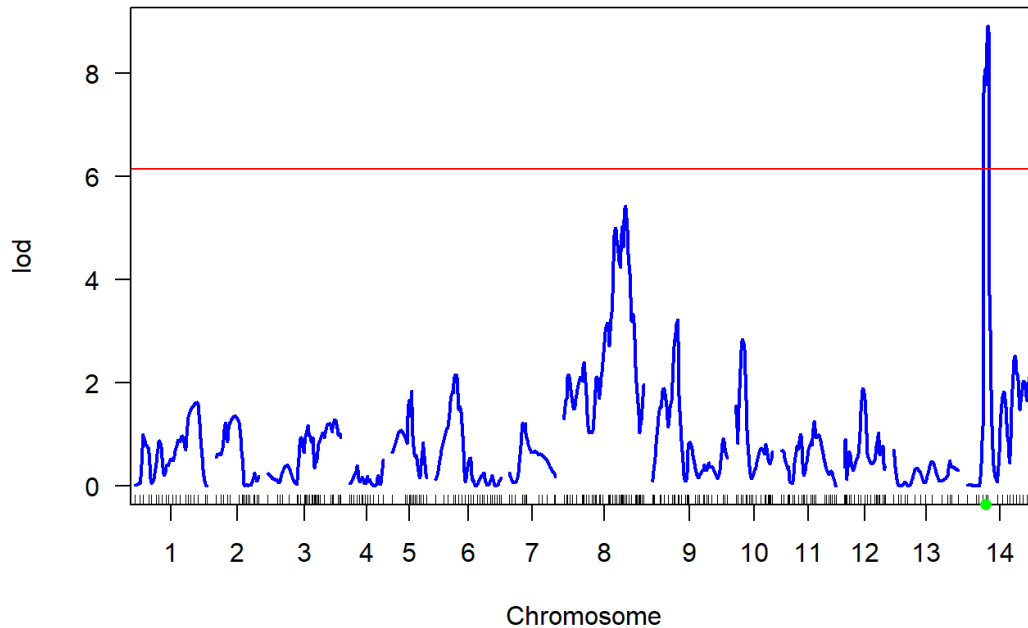
No Mapeamento por Intervalo Composto Ã© incluido covariaveis ao longo do genoma com a finalidade dessas covariaveis nÃ£o interferirem no Qtl que esta sendo mapeado, por isso aqui foram testados modelos com 1, 2 e 3 covariaveis, esse teste Ã© importante para se definir qual seria o melhor modelo.

Nesse metodo utilizamos uma janela de mapeamento que no nosso exemplo escolhemos o tamanho de 15 cM , porem isso tambem depende do pesquisador, podem ser utilizados valores encontrados na literatura ou outra forma de informaÃ§Ã£o geralmente usa-se entre 10 e 15 cM. aqui optamos por 15, essa janela Ã© importante para que nÃ£o controlemos covatores que estejam muito proximas ao Qtl que esta sendo mapeado.

O numero de permutacoes para a obtencao do limiar tambem foi de 3 mil.

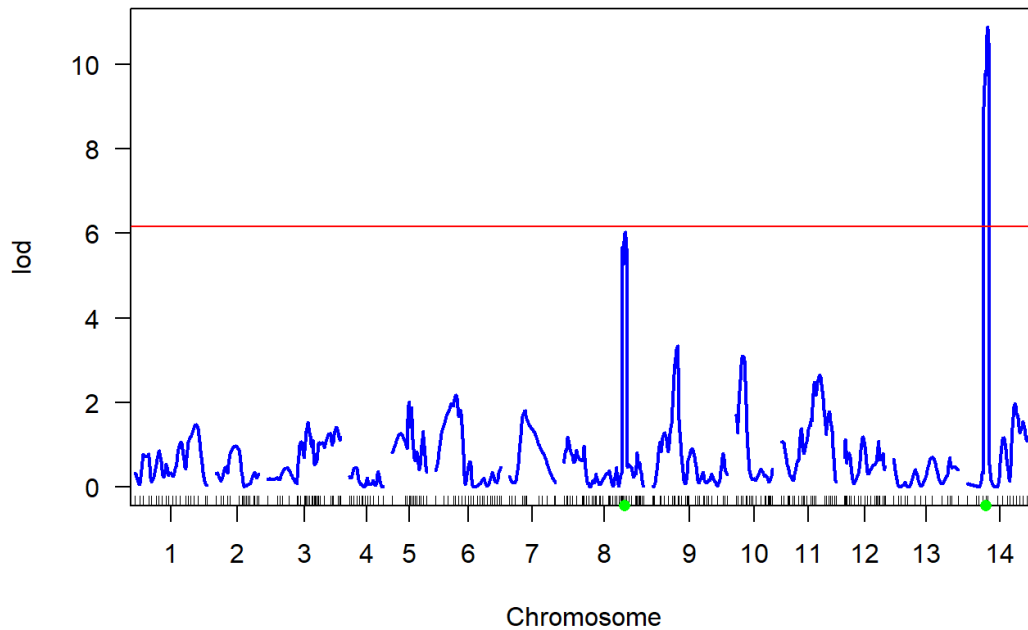
```
# Modelo CIM usando 1 covariavel
```

```
CIM1 <- cim(qtl, n.marcovar = 1, window = 15, pheno.col = 8)
plot(CIM1, col = c("blue"))
abline(h = lod1, col = "red")
add.cim.covar(CIM1, col = "green")
```



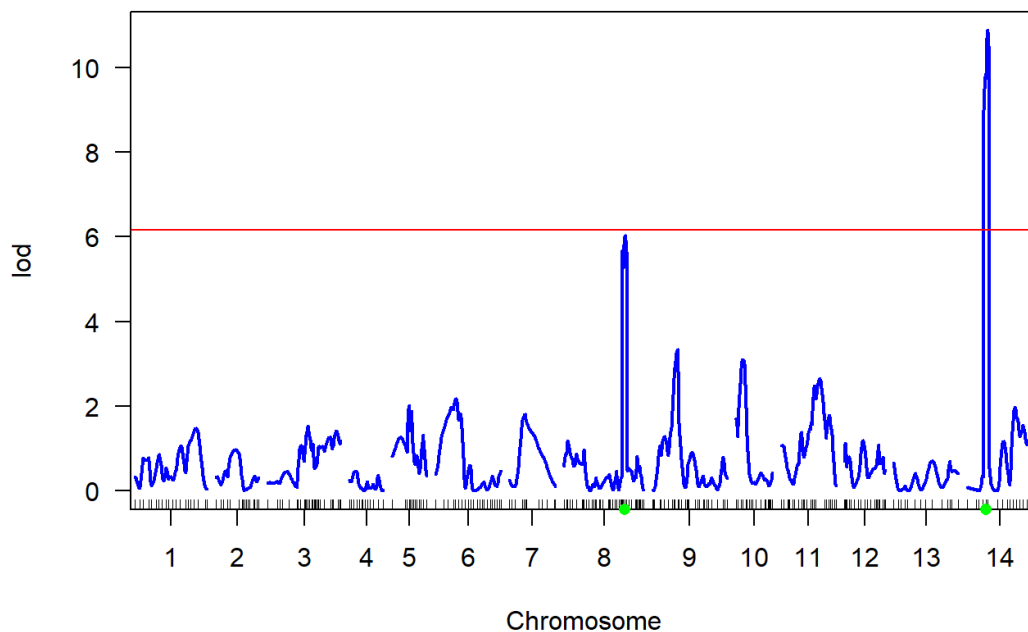
```
# modelo CIM usando 2 covariaveis
```

```
CIM2 <- cim(qtl, n.marcovar = 2, window = 15, pheno.col = 8)
plot(CIM2, col = c("blue"))
abline(h = lod1, col = "red")
add.cim.covar(CIM2, col = "green")
```



```
# modelo CIM usando 2 covariaveis
```

```
CIM3 <- cim(qtl, n.marcovar = 3, window = 15, pheno.col = 8)
plot(CIM2, col = c("blue"))
abline(h = lod1, col = "red")
add.cim.covar(CIM2, col = "green")
```



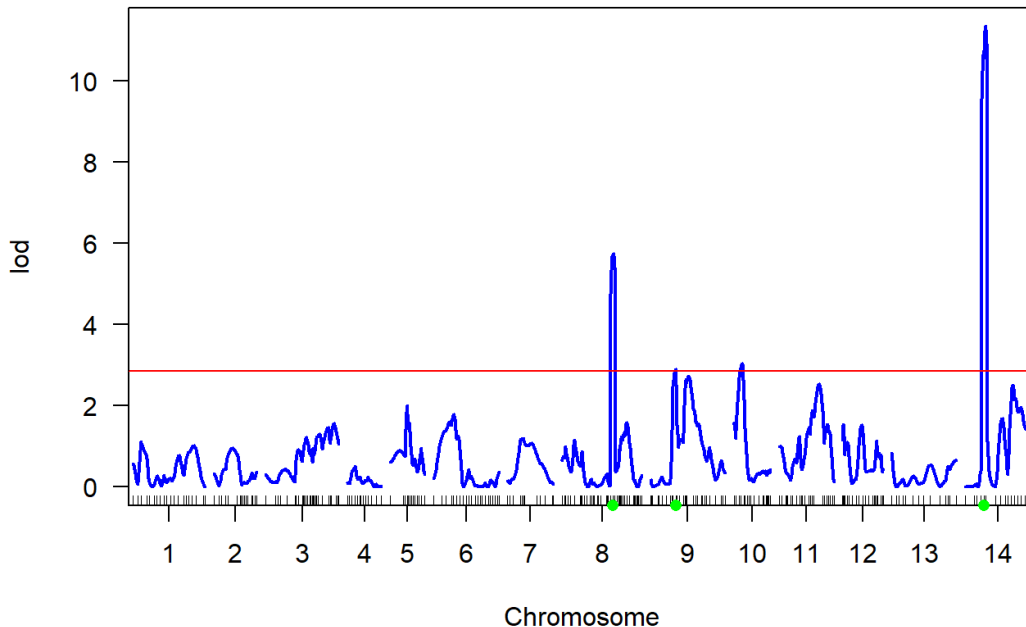
Percebemos que o

melhor modelo testado foi o com 3 covariaveis entao procedemos as analises com esse modelo, É importante resaltar que aqui tambem foi mapeada a mesma caracteristica quantitativa do metodo anterior.

Limiar usando permutacoes e 3 covariaveis - Assim como o metodo anterior, o limiar foi definido para continuar as analises e estabelecer a significancia do metodo estatistico.


```
limiar<- cim(qtl, n.marcovar = 3, window = 15, n.perm = 3.000, pheno.col = 8)
lim <- summary(limiar, alpha = 0.05)

# utilizando o modelo com 3 cofatores
CIM3 <- cim(qtl, n.marcovar = 3, window = 15, pheno.col = 8)
plot(CIM3, col = c ("blue"))
add.cim.covar(CIM3, col= "green") # marcas verdes s?o as os cofatores
abline(h = lim , col = "red")
```



```
qtl14 <- lodint( CIM3, chr= 14,drop = 1)
qtl8 <- lodint( CIM3, chr= 8,drop = 1)
qtl10 <- lodint( CIM3, chr= 10,drop = 1)
conjuntoqtls <- rbind(qtl14,qtl8,qtl10)
summary(conjuntoqtls)
```

	chr <fctr>	pos <dbl>	lod <dbl>
c14.loc58	14	58	11.354980
c8.loc148	8	148	5.747244
c10.loc24	10	24	3.037199
3 rows			

Diante do grafico podemos perceber grandes evidencias de QTL nos cromossomos 8 e 14

Em vermelho esta a linha do limiar da nalise e os picos que etao acima dela sÃ£o significativos

Em verde podemos perceber onde estÃ£o os tres cofatores, ou seja, aqueles que podem ter influencia no QTL um dos pontos esta justamente num pico que aparenta ser significativo no

cromossomo 14.

MIM

Nesse metodo sÃ£o considerados multiplos qtl e multiplo intervalos nas analises. Aqui as variacoes nao sao controladas como cofatores, pois as variacoes ja estÃ£o includas no modelo de multiplos intervalos.

obs: AQUI TAMBEM USAREMOS A MESMA CARACTERISTICA DOS METODOS ACIMA

Probabilidades condicionais e simulacao dos genotipos.

Aqui sao estimados as probabilidades dos genotpos dos qtl com base nos marcadores flaqueadores

```
prob<- calc.genoprob(mapamimulus, step = 1, error.prob=0.001)
```

Abaixo a simulacao dos genotipos Ã© feita baseando-se nos dados genotipicos

```
mimulus.sim <- sim.geno(prob, n.draws=5, step=1, off.end=0, error.prob=0.001, map.function=c("kosambi"))
```

Buscando QTLs

A funcao scantwo usa um modelo de dois QtL, o qual faz uma analise no genoma todo em busca das possiveis interaÃ§ÃÃµes

```
out_mim<- scantwo(mimulus.sim, pheno.col=8, method="hk", verbose=T)
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 12 individuals with missing phenotypes.
```

```
## --Running scanone
## --Running scantwo
## (1,1)
## (1,2)
## (1,3)
## (1,4)
## (1,5)
## (1,6)
## (1,7)
## (1,8)
## (1,9)
## (1,10)
## (1,11)
## (1,12)
## (1,13)
## (1,14)
## (2,2)
## (2,3)
## (2,4)
## (2,5)
## (2,6)
## (2,7)
## (2,8)
## (2,9)
## (2,10)
## (2,11)
## (2,12)
## (2,13)
## (2,14)
## (3,3)
## (3,4)
## (3,5)
## (3,6)
## (3,7)
## (3,8)
## (3,9)
## (3,10)
## (3,11)
## (3,12)
## (3,13)
## (3,14)
## (4,4)
## (4,5)
## (4,6)
## (4,7)
## (4,8)
## (4,9)
## (4,10)
## (4,11)
## (4,12)
## (4,13)
## (4,14)
## (5,5)
## (5,6)
## (5,7)
## (5,8)
## (5,9)
## (5,10)
## (5,11)
## (5,12)
## (5,13)
## (5,14)
## (6,6)
## (6,7)
## (6,8)
## (6,9)
## (6,10)
## (6,11)
## (6,12)
## (6,13)
```

```
## (6,14)
## (7,7)
## (7,8)
## (7,9)
## (7,10)
## (7,11)
## (7,12)
## (7,13)
## (7,14)
## (8,8)
## (8,9)
## (8,10)
## (8,11)
## (8,12)
## (8,13)
## (8,14)
## (9,9)
## (9,10)
## (9,11)
## (9,12)
## (9,13)
## (9,14)
## (10,10)
## (10,11)
## (10,12)
## (10,13)
## (10,14)
## (11,11)
## (11,12)
## (11,13)
## (11,14)
## (12,12)
## (12,13)
## (12,14)
## (13,13)
## (13,14)
## (14,14)
```

Calculando limiar via permutacao - aqui temos mais uma vez o caulo do limiar, mas queremos alerta que um bom numero de permuta~oes est~a acima de mil. todavia a demanda computacional para realizar esse numero de permuta~oes ~e muito grande, como o objetivo ~e entender o metodo usaremos apenas 20 permuta~oes, ressaltamos que o ideal seria usar acima de mil.

```
probmim <- calc.genoprob(mapamimulus, step=1, map.function = "kosambi")
limiarMIM<- scantwo(probmim,pheno.col=8, n.perm= 20,model="normal",method="hk")
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 12 individuals with missing phenotypes.
```

```
## Doing permutation in batch mode ...
```

```
limMim <- summary(limiarMIM, alpha=0.05)

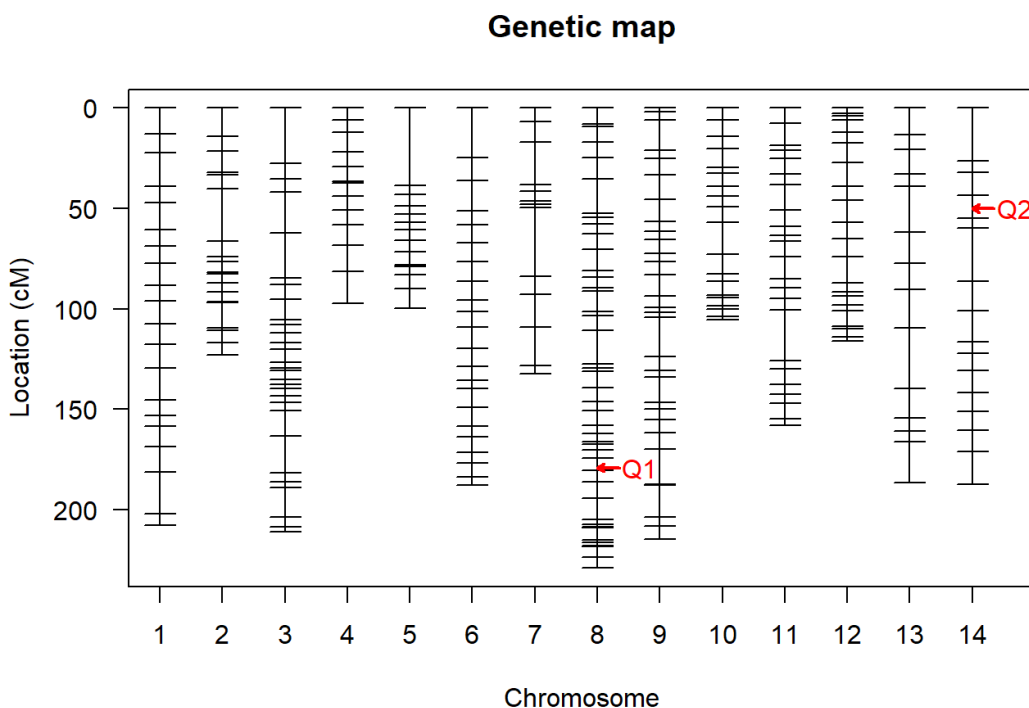
library(knitr)
table<- data.frame(summary(out_mim,perms=limiarMIM, alpha=0.05, pvalues = T))
knitr::kable(table)
```

chr1	chr2	pos1f	pos2f	lod.full	pval	lod.fv1	pval.1	lod.int	pval.2	pos1a	pos2a	lod.add	pval.3	lod.av1	pval.4
8	14	179	50	20.44976	0	12.52885	0	6.306069	0	176	58	14.14369	0	6.222782	0

observando os p-values dos modelos com QTLs que passaram o limiar , lod full é referente ao modelo completo e o lod referente ao modelo aditiv, além das posições dos qtls nos modelos

Aqui sÃ£o apresentados os cromossomos e as possiveis posiÃ§Ãoes dos qtl. aqui foram encontrados os cromossomos 8 e 14

```
qtl_mim <- makeqtl(mimulus.sim, chr=c(8,14), pos=c(179,50), what="prob")
plot(qtl_mim)
```



A funcao fitqtl calcula o efeitos dos qtls

```
model1<- fitqtl(mimulus.sim, pheno.col=8, qtl=qtl_mim, method="hk", get.ests=TRUE)
```

```
## Warning in fitqtlengine(pheno = pheno, qtl = qtl, covar = covar, formula = formula, : Dropping 12 individuals with missing phenotypes.
```

Verificando efeitos aditivos e de dominancia de cada qtl identificado

```
summary(model1)
```

```
##
##      fitqtl summary
##
## Method: Haley-Knott regression
## Model:  normal phenotype
## Number of observations : 275
##
## Full model result
## -----
## Model formula: y ~ Q1 + Q2
##
##      df      SS      MS      LOD      %var Pvalue(Chi2)  Pvalue(F)
## Model   4  2.996446 0.74911141 13.12881 19.73656 2.321476e-12 3.560485e-12
## Error 270 12.185764 0.04513246
## Total 274 15.182209
##
##
## Drop one QTL at a time ANOVA table:
## -----
##      df Type III SS      LOD      %var F value Pvalue(Chi2) Pvalue(F)
## 8@179.0  2      1.215 5.677  8.006   13.47          0 2.67e-06 ***
## 14@50.0  2      1.974 8.964 12.999   21.86          0 1.58e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Estimated effects:
## -----
##      est      SE      t
## Intercept 0.38447 0.01319 29.143
## 8@179.0a -0.09468 0.01959 -4.834
## 8@179.0d -0.06741 0.02746 -2.455
## 14@50.0a 0.12537 0.01941 6.460
## 14@50.0d 0.08162 0.03086 2.645
```

As penalizações são calculadas com base nas permutações feitas anteriormente as quais definiram o limiar

```
penalties <- calc.penalties(limiarMIM, alpha = 0.05)
```

Melhor modelo - aqui o modelo é ajustado com base nas penalidades calculadas acima

```
model_mim <- stepwiseqtl(mimulus.sim,pheno.col=8, penalties = penalties,
                        max.qtl=5, method="hk", verbose=T, refine.locations = TRUE)
```

```
## -Initial scan
## initial lod: 7.920911
## ** new best ** (pL0D increased by 4.4978)
##   no.qtl = 1   pL0D = 4.497833   formula: y ~ Q1
## -Step 1
## ---Scanning for additive qtl
##     plod = 7.297537
## ---Scanning for QTL interacting with Q1
##     plod = 9.444601
## ---Refining positions
## --- Moved a bit
##   no.qtl = 2   pL0D = 9.980235   formula: y ~ Q1 + Q2 + Q1:Q2
## ** new best ** (pL0D increased by 5.4824)
## -Step 2
## ---Scanning for additive qtl
##     plod = 10.02645
## ---Scanning for QTL interacting with Q1
##     plod = 4.881477
## ---Scanning for QTL interacting with Q2
##     plod = 5.523434
## ---Look for additional interactions
## ---Refining positions
## --- Moved a bit
##   no.qtl = 3   pL0D = 10.24829   formula: y ~ Q1 + Q2 + Q1:Q2 + Q3
## ** new best ** (pL0D increased by 0.2681)
## -Step 3
## ---Scanning for additive qtl
##     plod = 10.57445
## ---Scanning for QTL interacting with Q1
##     plod = 5.318099
## ---Scanning for QTL interacting with Q2
##     plod = 6.468153
## ---Scanning for QTL interacting with Q3
##     plod = 8.983807
## ---Look for additional interactions
##     plod = 4.808537
## ---Refining positions
## --- Moved a bit
##   no.qtl = 4   pL0D = 10.6956   formula: y ~ Q1 + Q2 + Q1:Q2 + Q3 + Q4
## ** new best ** (pL0D increased by 0.4473)
## -Step 4
## ---Scanning for additive qtl
##     plod = 10.0479
## ---Scanning for QTL interacting with Q1
##     plod = 5.60894
## ---Scanning for QTL interacting with Q2
##     plod = 5.533598
## ---Scanning for QTL interacting with Q3
##     plod = 8.890099
## ---Scanning for QTL interacting with Q4
##     plod = 8.825631
## ---Look for additional interactions
##     plod = 6.606409
## ---Refining positions
##   no.qtl = 5   pL0D = 10.0479   formula: y ~ Q1 + Q2 + Q1:Q2 + Q3 + Q4 + Q5
## -Starting backward deletion
## ---Dropping Q5
##   no.qtl = 4   pL0D = 10.6956   formula: y ~ Q1 + Q2 + Q3 + Q4 + Q1:Q2
## ---Refining positions
## ---Dropping Q4
##   no.qtl = 3   pL0D = 10.16459   formula: y ~ Q1 + Q2 + Q3 + Q1:Q2
## ---Refining positions
## --- Moved a bit
## ---Dropping Q3
##   no.qtl = 2   pL0D = 9.820934   formula: y ~ Q1 + Q2 + Q1:Q2
## ---Refining positions
## --- Moved a bit
## ---Dropping Q1:Q2
##   no.qtl = 2   pL0D = 6.28265   formula: y ~ Q1 + Q2
## ---Refining positions
```

```
## --- Moved a bit
## ---Dropping Q2
## no.qtl = 1 pLOD = 4.497833 formula: y ~ Q1
## ---Refining positions
## ---One last pass through refineqtl
```

A funcao refineQTL faz a busca dos qtls agora baseando-se na metodologia de multiplos qtl

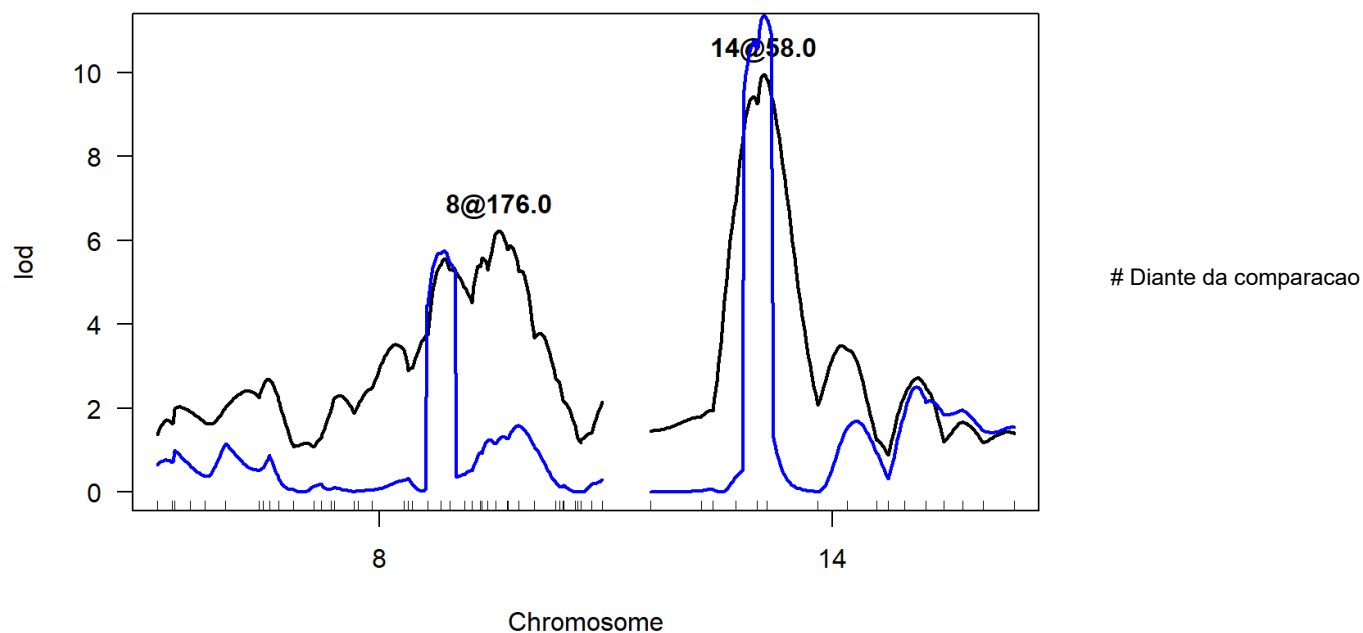
```
ref <- refineqtl(mimulus.sim, qtl=qtl_mim, method="hk", pheno.col = 8)
```

```
## pos: 179 50
## Iteration 1
## Q1 pos: 179 -> 176
## LOD increase: 0.22
## Q2 pos: 50 -> 58
## LOD increase: 0.795
## all pos: 179 50 -> 176 58
## LOD increase at this iteration: 1.015
## Iteration 2
## Q2 pos: 58 -> 58
## LOD increase: 0
## Q1 pos: 176 -> 176
## LOD increase: 0
## all pos: 176 58 -> 176 58
## LOD increase at this iteration: 0
## overall pos: 179 50 -> 176 58
## LOD increase overall: 1.015
```

```
ref
```

```
## QTL object containing genotype probabilities.
##
## name chr pos n.gen
## Q1 8@176.0 8 176 3
## Q2 14@58.0 14 58 3
```

```
plotLodProfile(ref)
plot(CIM3, col="blue", chr=c(8,14), ylim = c(0,10), add=TRUE)
```

dos dois metodos podemos perceber que a caractreistica pode ser explicada por 2 QTL nos cromossomos 8 e 14, todavia a nalise feita com o metodo de mapeamento por muitos intervalos, a que Ã© representada pela cor preta, possui picos melhor definidos e mais acertivos, diantes dos resultados desse metodo pode-se proceder com demais estudos das regioes gÃªnicas associadas a caracteristica. # Falando em questÃµes estatisticas o MIM tambÃ©m Ã© uma metodologia mais assertiva, pois o modelo tem um maior controle sobre os parametros e tambem sobre as possiveis interferencias que levariam a um mapeamento erroneo da caracteristica quantitativa, alÃ©m disso a escolha do melhor modelo para o estudo da caracteristica tambÃ©m Ã© uma grande vantagem.