

Exercício 1 da Aula 10 - Mapeamento de QTLs IV (Mapeamento por Intervalo Composto; F1 segregante)

Alunos: Felipe Maniero Nazato, Diane Simon Rozzetto e Marisângela Rodrigues Santos

15/06/2015

Contents

1 Considere o conjunto de dados de Mimulus, já usado em aulas anteriores. Usando o mapa que você construiu anteriormente com o OneMap, faça mapeamento por intervalo e por intervalo composto para localizar QTL's para uma das características presentes no conjunto de dados (escolha a que julgar mais interessante).	1
1.1 Mapa genético MIMULUS	1
1.2 Entrada dos dados no R/qlt	2
1.3 Nível de significância ("Threshold")	6
1.4 Mapeamento por intervalo (IM)	7
1.4.1 Resultado do IM	10
1.5 Mapeamento por intervalo composto (CIM)	11
1.5.1 Resultado do CIM	16
1.6 Comparação dos 2 métodos (IM vs CIM)	17

1 Considere o conjunto de dados de Mimulus, já usado em aulas anteriores. Usando o mapa que você construiu anteriormente com o OneMap, faça mapeamento por intervalo e por intervalo composto para localizar QTL's para uma das características presentes no conjunto de dados (escolha a que julgar mais interessante).	
<ul style="list-style-type: none">• Determine o limiar para rejeitar H_0 e apresente os resultados de forma gráfica e em tabelas de forma semelhante aos artigos publicados na área;• Use o software R/qlt, verificando no tutorial do OneMap como exportar o mapa construído para o formato do R/qlt;• Use o Fórum para resolver suas dúvidas;• O exercício pode ser feito em grupo, pela mesma equipe que fez o mapa genético;• Apresente os resultados em um arquivo html, da mesma forma que para o mapa de Mimulus.	

1.1 Mapa genético MIMULUS

Em um exercício anterior, foi obtido o mapa genético para o conjunto de dados do MIMULUS.

O mapa do MIMULUS é composto por 14 grupos de ligação e pode ser observado na Figura abaixo:


```
library("qtl")
```

```
## Warning: package 'qtl' was built under R version 3.1.3
```

- Carregar os arquivos contendo o conjunto completo dos dados (**m_feb06.raw**) e o arquivo contendo o mapa genético (**mimulus_mapa_final.txt**):

OBS₁: O arquivo original do mimulus está salvo em **.raw** (contém dados genotípicos e fenotípicos)

OBS₂: Seguindo o tutorial do R/qtl foi possível identificar que o programa reconhece arquivos **.txt** exportados do OneMap. O mapa genético do conjunto de dados MIMULUS foi obtido pelo programa OneMap e exportado através da função **write.map(mapa, "mimulus_mapa_final")**. A saída foi salva em um arquivo **.txt** contendo o mapa final do MIMULUS (arquivo **mimulus_mapa_final.txt**).

OBS₃: para abrir os arquivos no R/qtl, foi utilizado a função **read.cross** e o formato MapMaker ("**mm**").

```
dados_mimulus = read.cross("mm", file="m_feb06.raw", mapfile = "mimulus_mapa_final.txt")
```

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

```
## Warning in summary.cross(cross): Some markers at the same position on chr
## 2,6,9,11,14; use jittermap().
```

```
## --Cross type: f2
```

- Como haviam marcadores sobrepostos uma mesma posição no cromossomo, foi utilizado a função **jittermap** (segundo o tutorial do R/qtl) para o ajuste.

```
mimulus = jittermap(dados_mimulus)
```

- Após o carregamento, é possível observar um resumo dos dados do MIMULUS

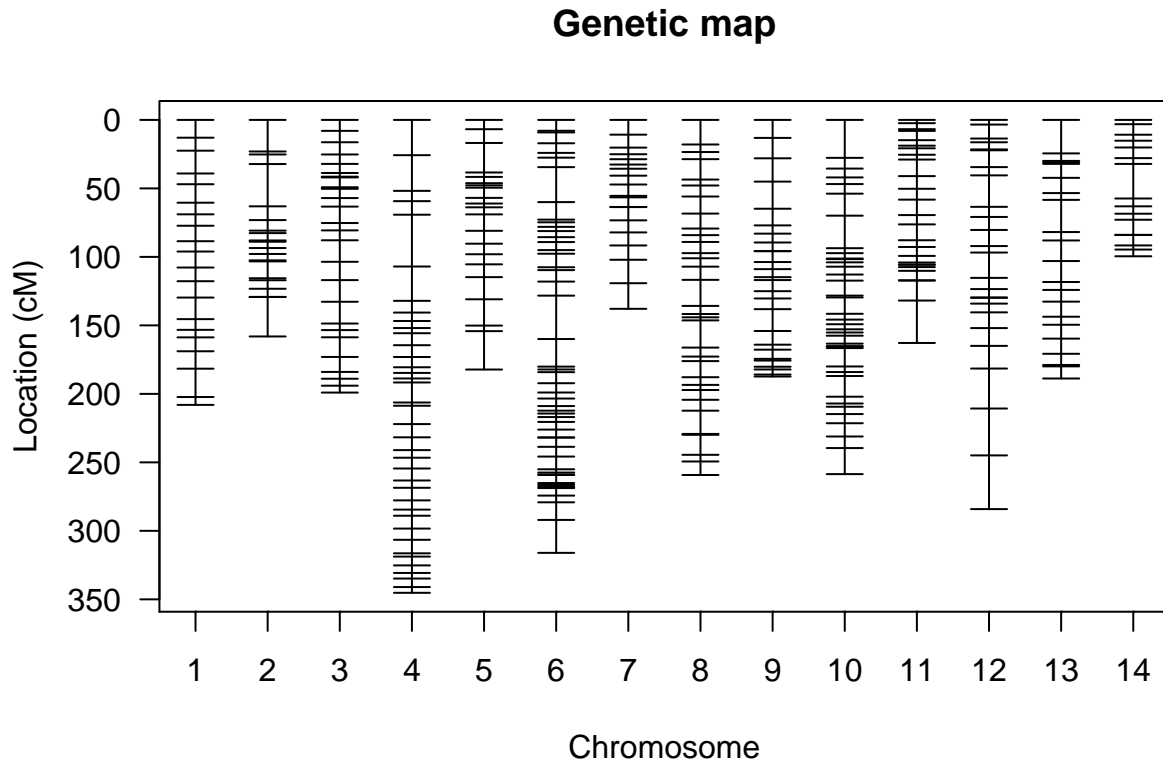
```
#Resumo dos dados
summary(mimulus)
```

```
##      F2 intercross
##
##      No. individuals:    287
##
##      No. phenotypes:    16
##      Percent phenotyped: 96.2 93.4 96.2 93 95.8 87.8 96.2 95.8 96.2 96.2
##                        90.2 96.2 95.8 96.2 95.8 96.2
##
##      No. chromosomes:   14
##      Autosomes:         1 2 3 4 5 6 7 8 9 10 11 12 13 14
##
##      Total markers:     377
##      No. markers:       20 20 26 38 21 49 19 32 26 38 26 25 21 16
##      Percent genotyped: 89.6
##      Genotypes (%):     AA:15.4 AB:24.3 BB:23.0 not BB:20.5 not AA:16.8
```

De acordo com o resumo dos dados, pode-se verificar que na população estudada (população F_2) existem 287 indivíduos, 377 marcadores (418 no total, no entanto o mapa final foi construído com 377 marcadores) e 16 características fenotípicas.

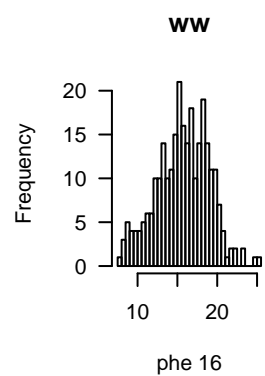
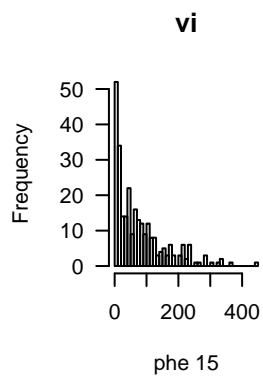
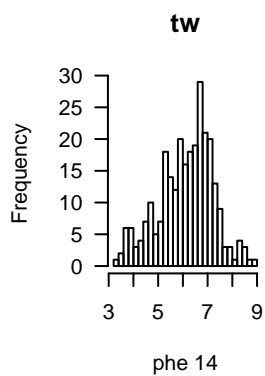
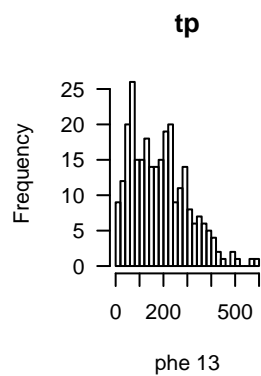
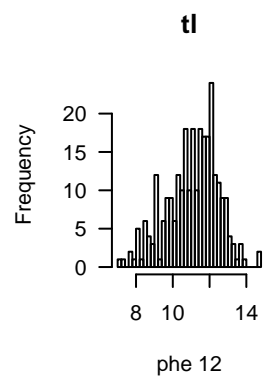
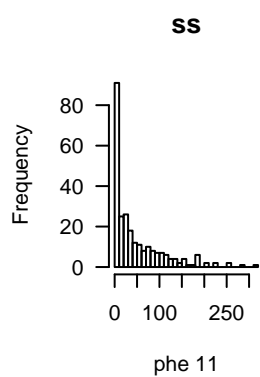
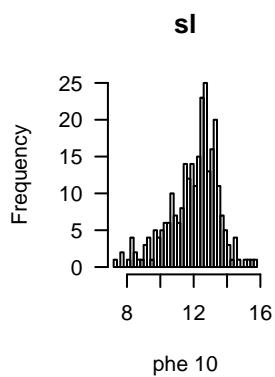
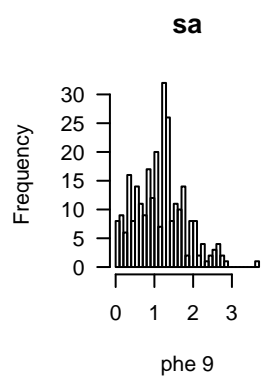
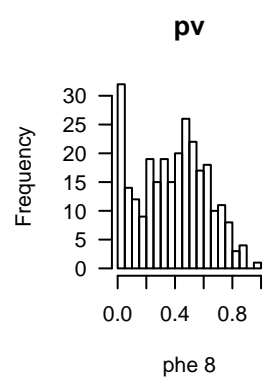
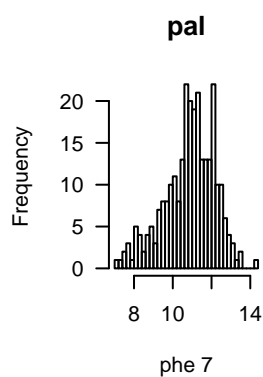
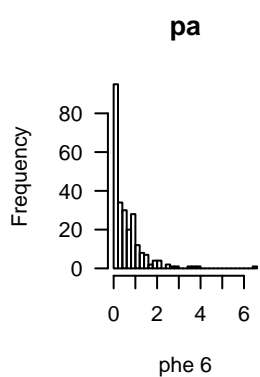
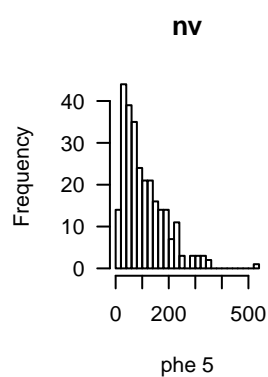
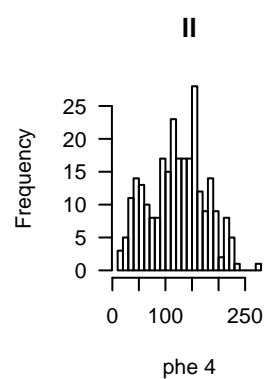
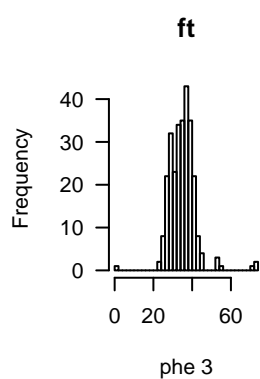
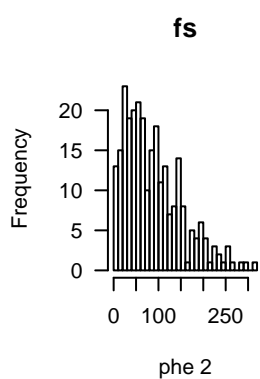
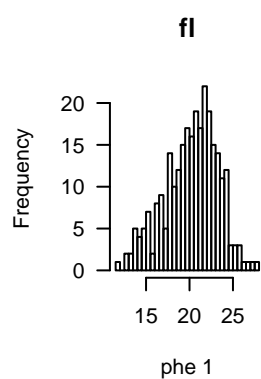
- Mapa genético (versão R/qlt)

```
#Mapa genético Mimulus
plot.map(mimulus)
```



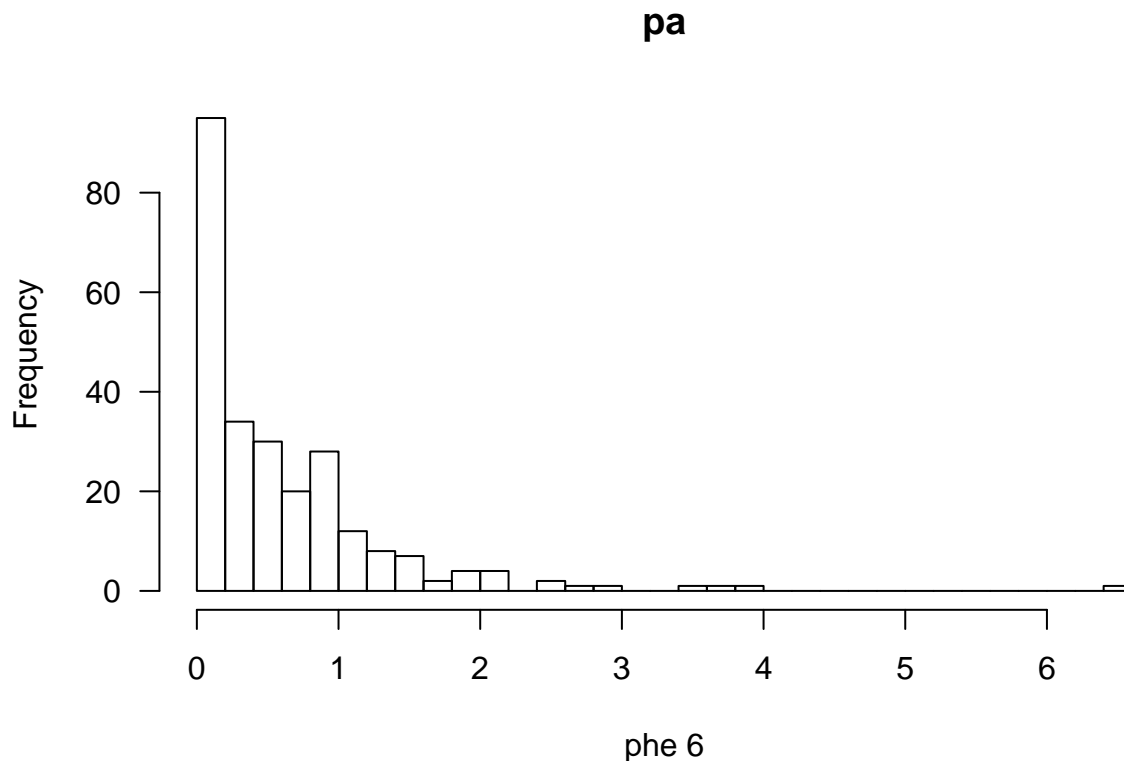
- Para observar cada uma das características fenotípicas, foram plotados os seguintes gráficos:

```
par(mfrow=c(2,4))
for(i in 1:16)
  plot.pheno(mimulus, pheno.col=i)
```



- Iremos trabalhar com a característica fenotípica **pa** (phe 6)

```
plot.pheno(mimulus, pheno.col= 6)
```



1.3 Nível de significância (“Threshold”)

Primeiramente foi calculado o nível de significância (LOD) para declarar como significativo o efeito dos QTLs (“Threshold”).

Para isso, criou-se a distribuição de LODs sob H_0 utilizando os seguintes comandos:

OBS_1 : foi utilizado a aproximação de Haley-Knott (“hk”) no cálculo das permutações;

OBS_2 : foram realizadas 1000 permutações.

```
#Significância - "uso de permutações"
mimulus_gp = calc.genoprob(mimulus, step = 1, error.prob=0.0001)
sig_perm = scanone(mimulus_gp, method = "hk", pheno.col=6, tol=1e-4, n.perm=1000)
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 35 individuals w
```

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

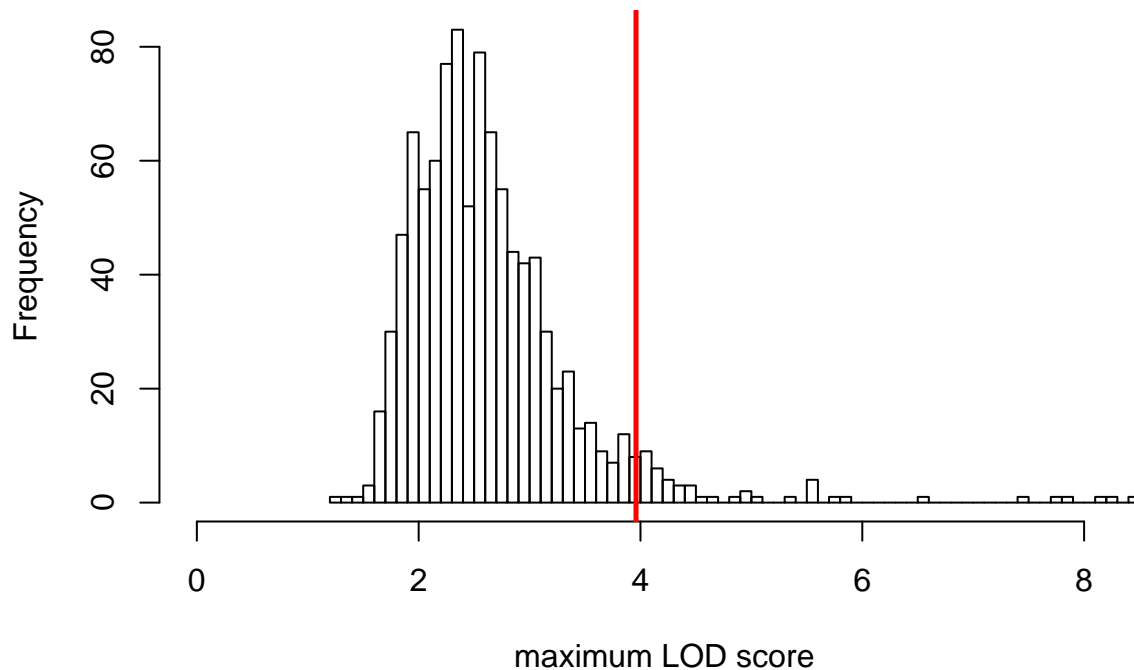
Foram obtidos os valores do threshold (LOD) para 5% e 1% de significância:

```
summary(sig_perm, alpha=c(0.05, 0.01))
```

```
## LOD thresholds (1000 permutations)
##      lod
## 5% 3.96
## 1% 5.55
```

E o gráfico da distribuição da estatística (LOD) sob H_0 :

```
plot(sig_perm)
abline(v=3.96, col="red", lwd=2.5)
```



1.4 Mapeamento por intervalo (IM)

O mapeamento por intervalo foi realizado utilizando os seguintes comandos:

OBS_1 : foi utilizado o algoritmo “EM”

OBS_2 : foi utilizado um **threshold de 3,96** (calculado por permutação).

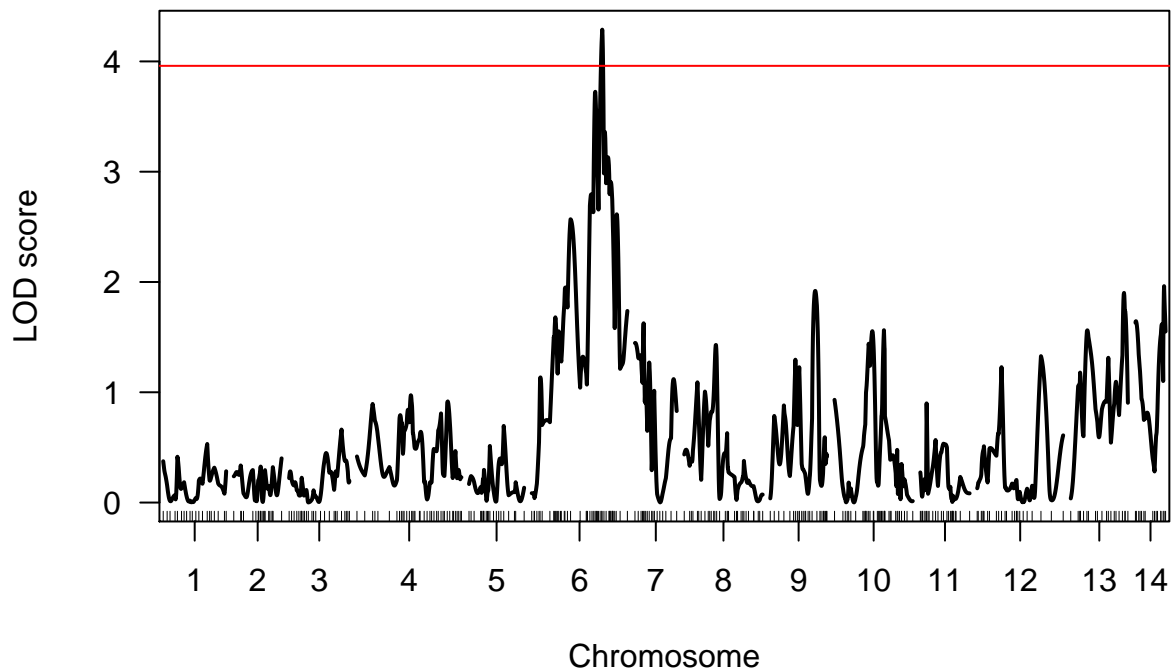
```
#IM (mapeamento intervalo - método "EM")
mimulus_gp = calc.genoprob(mimulus, step = 1, error.prob=0.0001)
mimulus_em = scanone(mimulus_gp, method = "em", pheno.col=6, tol=1e-6)
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 35 individuals w
```

```
summary(mimulus_em)
```

```
##      chr  pos  lod
## AA173C    1 145.4 0.531
## CA279     2 158.1 0.401
## MgSTS425   3 173.1 0.662
## c4.loc178  4 178.0 0.973
## AA163     5 114.8 0.695
## c6.loc233  6 233.0 4.288
## MgSTS574a  7  28.7 1.626
## c8.loc105  8 105.0 1.429
## c9.loc149  9 149.0 1.919
## MgSTS27   10 163.3 1.563
## BA449     11  20.7 0.899
## BC219     12 210.7 1.327
## c13.loc176 13 176.0 1.901
## MgSTS263  14  94.6 1.962
```

```
plot(mimulus_em, ylab="LOD score")
abline(h = 3.96, col="red") #5% significância
```



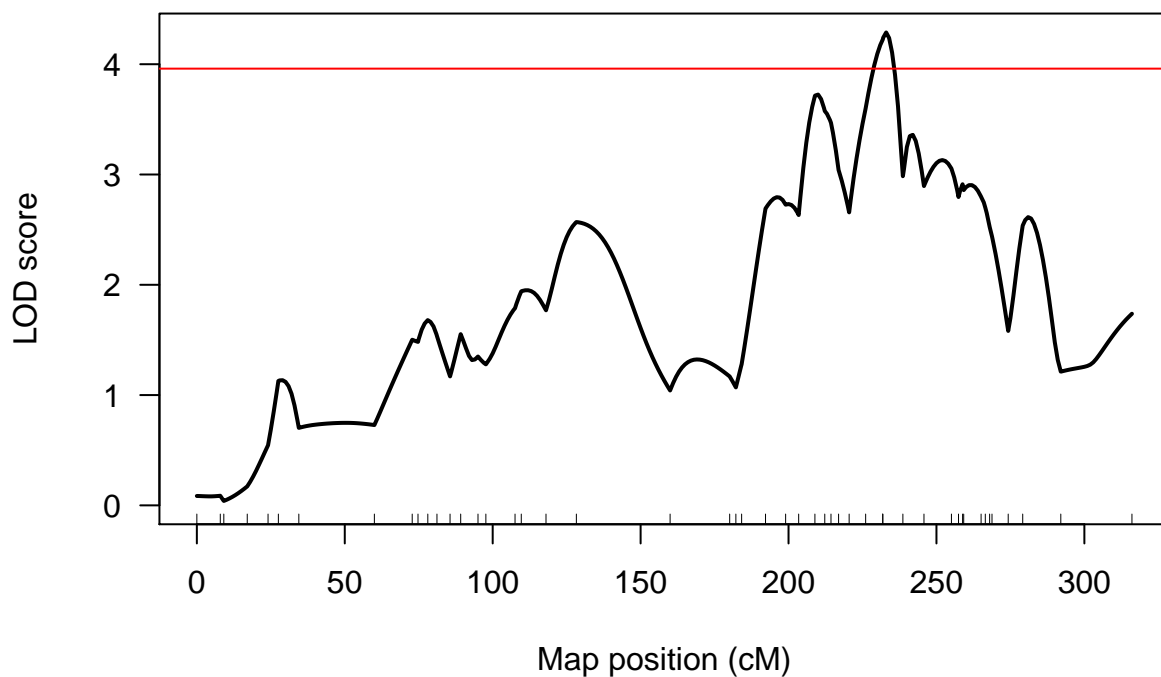
De acordo com o gráfico é observado que, para a característica fenotípica **pa**, há evidências de um QTL no grupo de ligação 6 (cromossomo 6), acima do threshold (LOD = 3,96).

Pelos comando abaixo é possível observar em detalhes o grupo de ligação 6:

```
#Cromossomo 6 somente
mimulus_em2 = scanone(mimulus_gp, method = "em", pheno.col=6, tol=1e-6, chr = c(6))
```

```
## Warning in checkcovar(cross, pheno.col, addcovar, intcovar, perm.strata, : Dropping 35 individuals w
```

```
plot(mimulus_em2, ylab="LOD score")
abline(h = 3.96, col="red") #5%
```



```
summary(mimulus_em2)
```

```
##           chr pos lod
## c6.loc233    6 233 4.29
```

O pico do gráfico evidencia um QTL na posição **233 cM**, acima do threshold (LOD = 3,96).

Para encontrar os efeitos do QTL em questão, foi utilizado os seguintes comandos:

```
#Efeito do QTL
mimulus_qtl_3 = makeqtl(mimulus_gp, chr=c(6), pos=c(233), what="prob")
qtl_3 = fitqtl(mimulus_gp, qtl = mimulus_qtl_3,
               method="hk", pheno.col=6, get.ests=TRUE)
```

```
## Warning in fitqtlengine(pheno = pheno, qtl = qtl, covar = covar, formula = formula, : Dropping 35 in
```

```
summary(qtl_3)
```

```
##
##      fitqtl summary
##
## Method: Haley-Knott regression
## Model:  normal phenotype
## Number of observations : 252
##
## Full model result
## -----
## Model formula: y ~ Q1
##
##      df      SS      MS      LOD      %var Pvalue(Chi2)      Pvalue(F)
## Model    2  10.61799  5.3089962  4.122062  7.256128  7.549843e-05  8.452985e-05
## Error  249  135.71338  0.5450337
## Total  251  146.33138
##
##
## Estimated effects:
## -----
##      est      SE      t
## Intercept  6.243e-01  4.690e-02  13.313
## 6@233.0a  -3.057e-01  7.008e-02  -4.362
## 6@233.0d  -7.777e-05  9.934e-02  -0.001
```

```
n=282
LOD1=4.122062
r2=1-10^(((2)/n)*LOD1)
r2 #Valor do Rsquared
```

```
## [1] 0.06509923
```

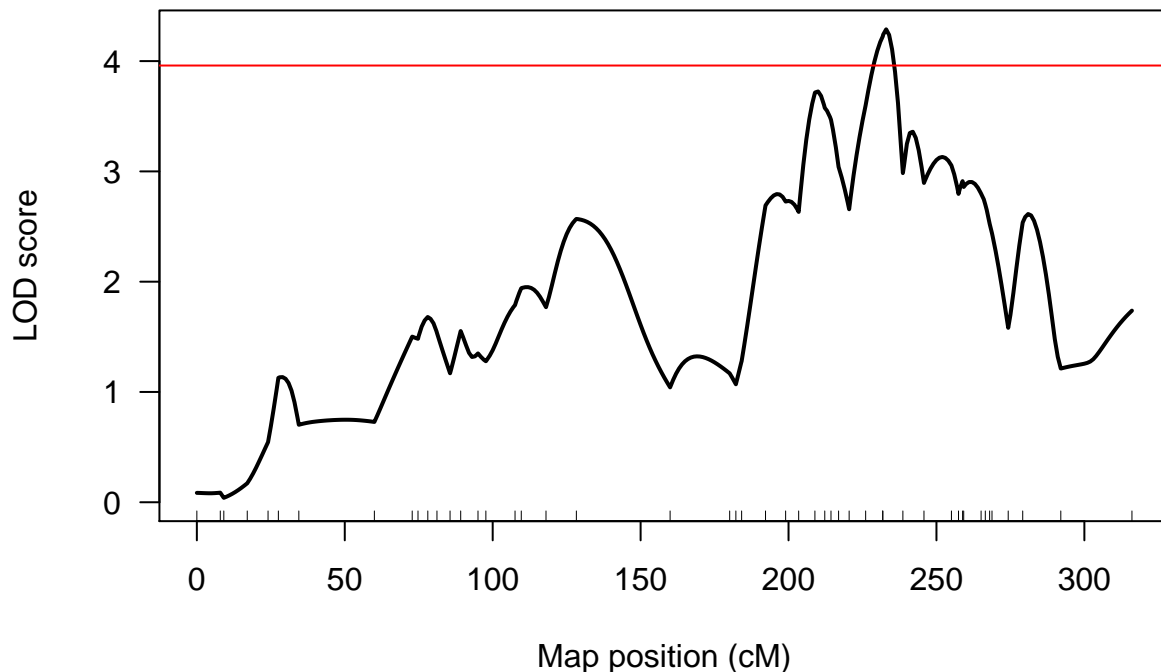
1.4.1 Resultado do IM

O resultado do mapeamento por intervalo (IM) foi resumido na Tabela abaixo (contendo o nome do QTL, o cromossomo que o contém, a posição no cromossomo, o valor do intercepto da regressão, os valores dos efeito do QTL (aditivo e de dominância), o LOD e o R^2):

QTL	Cromossomo	Posição (cM)	Intercepto	Ef. aditivo	Ef. de dominância	LOD	R^2	p-valor
Q1	6	233	0,6243	-0,305	-0,000078	4,122	0,065	0,000084

E também na Figura:

```
plot(mimulus_em, ylab="LOD score",chr=c(6))
abline(h = 3.96, col="red") #5%
```



1.5 Mapeamento por intervalo composto (CIM)

Primeiramente foi realizado o teste de **stepwise (regressão múltipla)** para escolha do modelo que melhor se ajusta aos dados, ou seja, para verificar o número de cofatores que irão entrar no modelo de mapeamento por intervalo composto (CIM).

OBS₁: Testou-se um máximo de 5 QTLs que poderiam entrar no modelo como cofatores (testou-se 5 cofatores a princípio).

```
#Step Wise - escolha do modelo
mimulus_gp = calc.genoprob(mimulus, step = 1, error.prob=0.0001)
mimulus_stepwise = stepwiseqtl(mimulus_gp, chr = c(1:14), pheno.col=6,
                               max.qtl = 5, method = "hk")
```

```
## -Initial scan
## initial lod: 4.244194
## ** new best ** (pLOD increased by 0.7242)
## no.qtl = 1 pLOD = 0.7241939 formula: y ~ Q1
## -Step 1
## ---Scanning for additive qtl
## plod = -0.5819009
## ---Scanning for QTL interacting with Q1
## plod = 0.4357183
## ---Refining positions
## --- Moved a bit
```

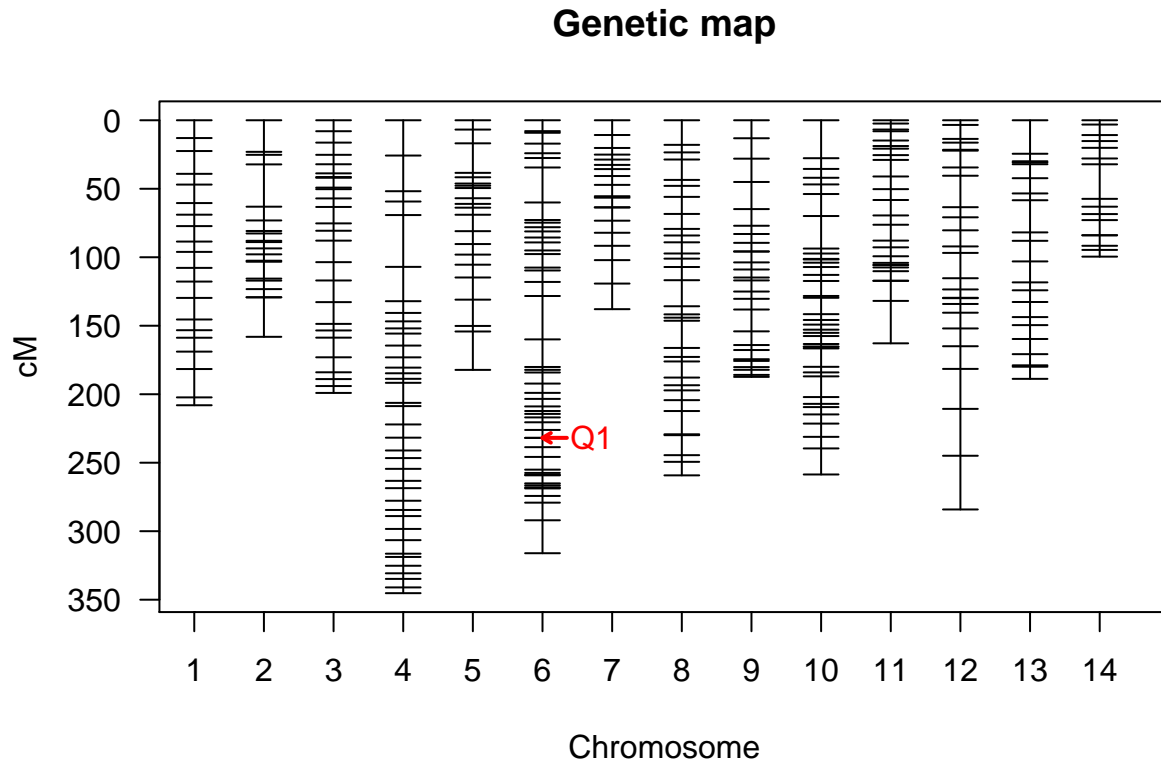
```

##      no.qtl = 2    pL0D = 0.6418344    formula: y ~ Q1 + Q2 + Q1:Q2
## -Step 2
## ---Scanning for additive qtl
##      plod = 0.001723726
## ---Scanning for QTL interacting with Q1
##      plod = -2.271356
## ---Scanning for QTL interacting with Q2
##      plod = -2.364854
## ---Look for additional interactions
## ---Refining positions
## --- Moved a bit
##      no.qtl = 3    pL0D = 0.215045    formula: y ~ Q1 + Q2 + Q1:Q2 + Q3
## -Step 3
## ---Scanning for additive qtl
##      plod = -1.657973
## ---Scanning for QTL interacting with Q1
##      plod = -3.373935
## ---Scanning for QTL interacting with Q2
##      plod = -2.307195
## ---Scanning for QTL interacting with Q3
##      plod = -0.7009392
## ---Look for additional interactions
##      plod = -2.050625
## ---Refining positions
## --- Moved a bit
##      no.qtl = 4    pL0D = -0.07738179    formula: y ~ Q1 + Q2 + Q1:Q2 + Q3 + Q4 + Q3:Q4
## -Step 4
## ---Scanning for additive qtl
##      plod = -0.9362347
## ---Scanning for QTL interacting with Q1
##      plod = -3.457227
## ---Scanning for QTL interacting with Q2
##      plod = -2.083689
## ---Scanning for QTL interacting with Q3
##      plod = -3.109916
## ---Scanning for QTL interacting with Q4
##      plod = -4.171124
## ---Look for additional interactions
##      plod = -3.812095
## ---Refining positions
## --- Moved a bit
##      no.qtl = 5    pL0D = -0.8237022    formula: y ~ Q1 + Q2 + Q1:Q2 + Q3 + Q4 + Q3:Q4 + Q5
## -Starting backward deletion
## ---Dropping Q5
##      no.qtl = 4    pL0D = -0.09101883    formula: y ~ Q1 + Q2 + Q3 + Q4 + Q1:Q2 + Q3:Q4
## ---Refining positions
## --- Moved a bit
## ---Dropping Q4
##      no.qtl = 3    pL0D = 0.1254446    formula: y ~ Q1 + Q2 + Q3 + Q1:Q2
## ---Refining positions
## --- Moved a bit
## ---Dropping Q3
##      no.qtl = 2    pL0D = 0.5161596    formula: y ~ Q1 + Q2 + Q1:Q2
## ---Refining positions

```

```
## --- Moved a bit
## ---Dropping Q2
##   no.qtl = 1   pLOD = 0.6791916   formula: y ~ Q1
## ---Refining positions
## --- Moved a bit
## ---One last pass through refineqtl
```

```
plot(mimulus_stepwise, ylab="cM")
```



```
summary(mimulus_stepwise)
```

```
##   QTL object containing genotype probabilities.
##
##      name chr    pos n.gen
## Q1 6@231.9   6 231.85     3
##
##   Formula: y ~ Q1
##
##   pLOD: 0.724
```

Pelo método de stepwise, é definido que $Y = Q1$, ou seja, apenas 1 cofator entra no modelo de mapeamento por intervalo composto.

Dessa forma o mapeamento por intervalo composto pode ser realizado de acordo com os comando abaixo:

OBS_1 : foi utilizado o algoritmo “EM”;

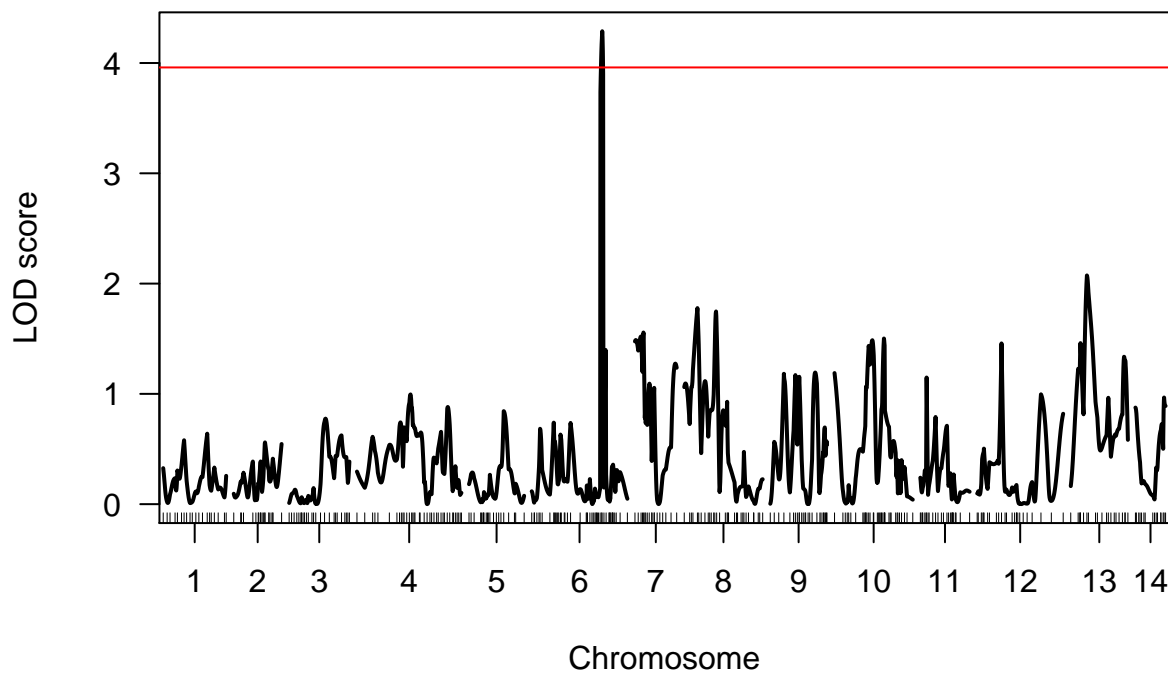
OBS_2 : foi utilizado **1 cofator** (pela regressão múltipla de stepwise foi identificado um QTL de efeito maior sobre a característica em questão);

OBS_3 : foi utilizado uma **window size = 10 cM**;

OBS_4 : foi utilizado um **threshold = 3,96** (cálculo por permutação)

```
# Mapeamento - intervalo composto (CIM)
mimulus_cim = cim(mimulus_gp, pheno.col=6, n.marcovar=1,
                  window=10, method = "em", error.prob=0.0001, map.function = "kosambi")
plot(mimulus_cim, ylab="LOD score")

abline(h = 3.96, col="red") #5%
```



```
summary(mimulus_cim)
```

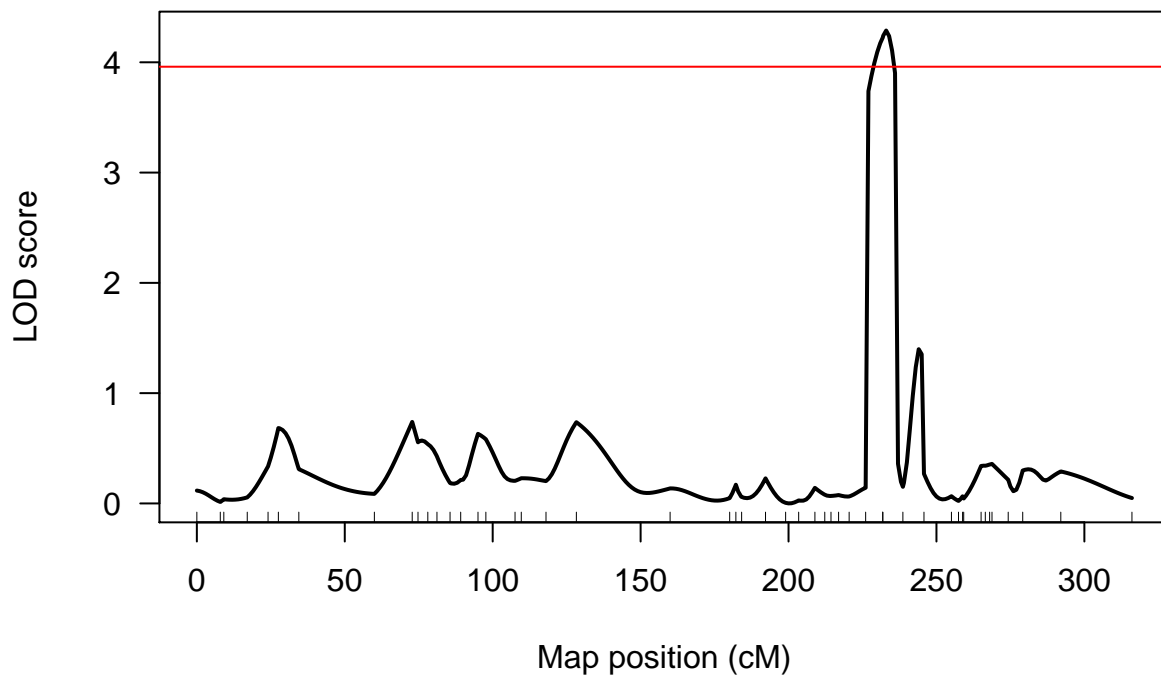
##	chr	pos	lod
## AA173C	1	145.4	0.639
## MgSTS589	2	103.3	0.560
## c3.loc120	3	120.0	0.776
## c4.loc177	4	177.0	0.995
## AA163	5	114.8	0.843
## c6.loc233	6	233.0	4.288
## c7.loc28	7	28.0	1.556
## MgSTS621	8	43.5	1.778
## c9.loc148	9	148.0	1.193

```
## MgSTS27      10 163.3 1.502
## BA449        11  20.7 1.148
## AAT364       12  80.3 1.458
## MgSTS599     13  53.4 2.074
## MgSTS263     14  94.6 0.967
```

No gráfico é observado que, para a característica fenotípica **pa**, há evidências de um QTL, com LOD acima do threshold (LOD = 3,96), no grupo de ligação 6 (cromossomo 6) na posição **231 cM**.

Pelos comandos abaixo é possível observar em detalhes o grupo de ligação 6:

```
plot(mimulus_cim, ylab="LOD score", chr=c(6))
abline(h = 3.96, col="red") #5%
```



Para encontrar os efeitos do QTL em questão, foi utilizado os seguintes comandos:

```
#Efeito do QTL
mimulus_qtl_cim = makeqtl(mimulus_gp, chr=c(6), pos=c(231), what="prob")
qtl_cim_6 = fitqtl(mimulus_gp, qtl = mimulus_qtl_cim,
                  method="hk", pheno.col=6, get.ests=TRUE)
```

```
## Warning in fitqtlengine(pheno = pheno, qtl = qtl, covar = covar, formula = formula, : Dropping 35 in
```

```
summary(qtl_cim_6)
```

```
##
##      fitqtl summary
##
## Method: Haley-Knott regression
## Model:  normal phenotype
## Number of observations : 252
##
## Full model result
## -----
## Model formula: y ~ Q1
##
##          df          SS          MS          LOD          %var Pvalue(Chi2)    Pvalue(F)
## Model      2  10.91116  5.4555806  4.240399  7.456474  5.749118e-05  6.457764e-05
## Error  249  135.42021  0.5438563
## Total  251  146.33138
##
##
## Estimated effects:
## -----
##          est          SE          t
## Intercept  6.238e-01  4.681e-02  13.326
## 6@231.0a  -3.080e-01  6.935e-02  -4.441
## 6@231.0d   3.555e-05  9.779e-02   0.000
```

```
n=282
LOD1=4.240399
r21=1-10^((( -2)/n)*LOD1)
r21
```

```
## [1] 0.06690417
```

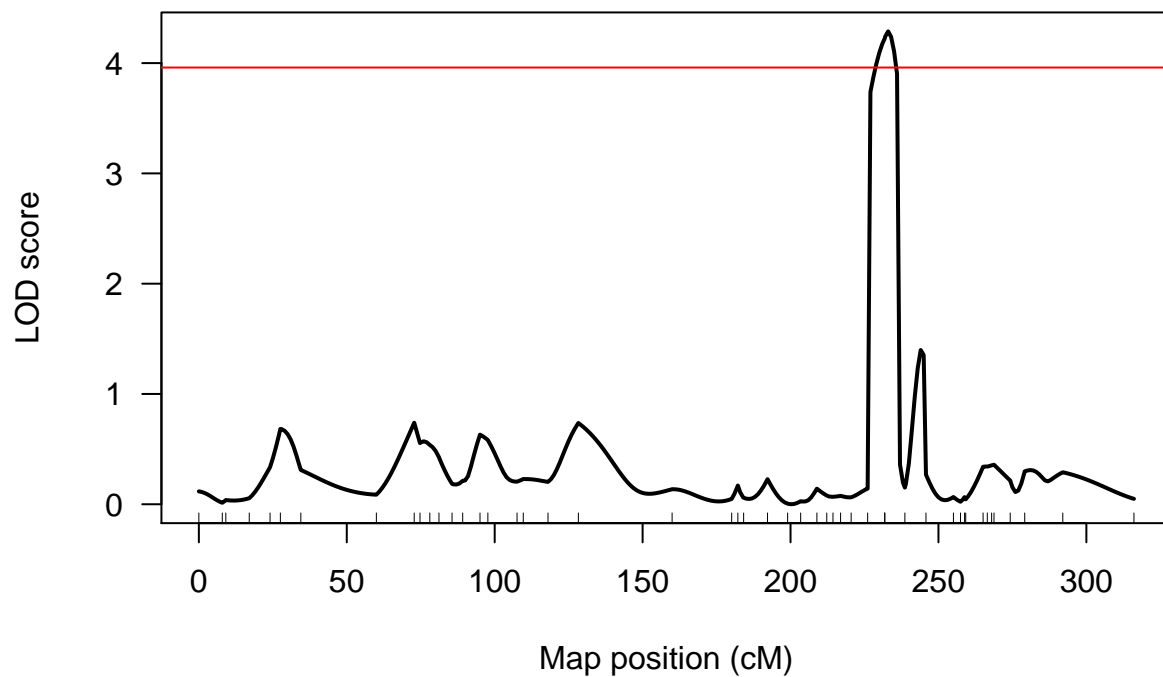
1.5.1 Resultado do CIM

O resultado do mapeamento por intervalo composto (CIM) foi resumido na Tabela abaixo (contendo o nome do QTL, o cromossomo que o contém, a posição no cromossomo, o valor do intercepto da regressão, os valores dos efeitos do QTL (aditivo e de dominância), o LOD e o R^2):

QTL	Cromossomo	Posição (cM)	Intercepto	Ef. aditivo	Ef. de dominância	LOD	R^2	p-valor
Q1	6	231	0,6238	-0,308	0,0000355	4,2404	0,067	0,0000646

E também no gráfico:

```
plot(mimulus_cim, ylab="LOD score",chr=c(6))
abline(h = 3.96, col="red") #5%
```

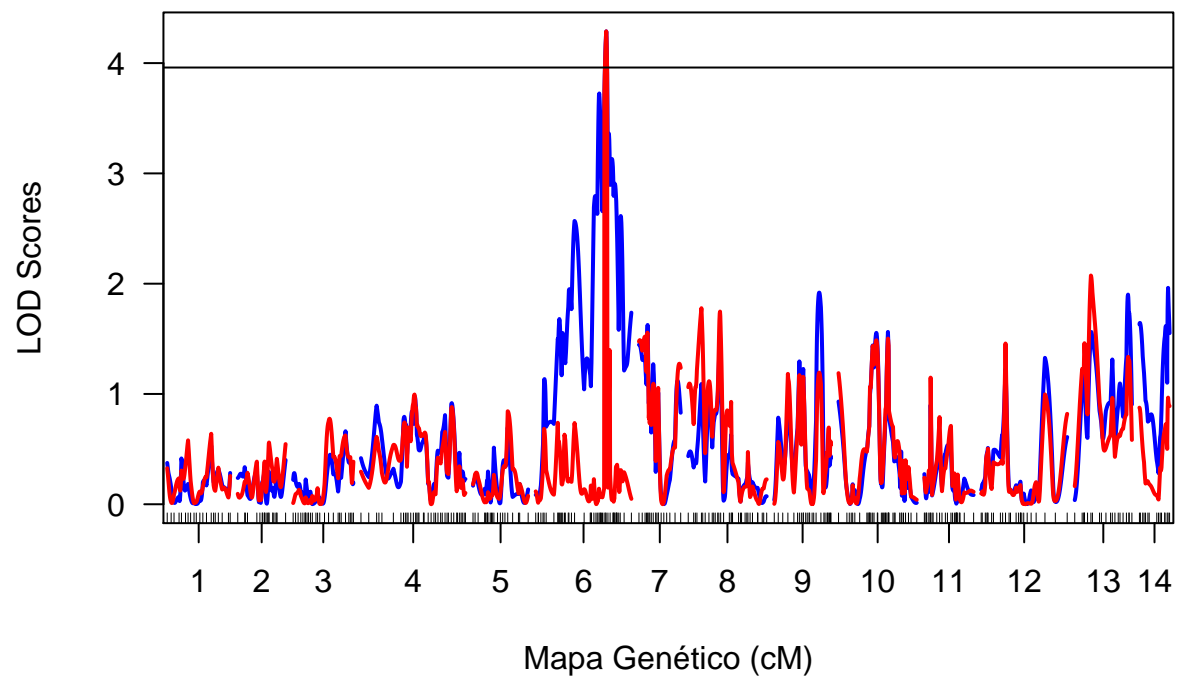



1.6 Comparação dos 2 métodos (IM vs CIM)

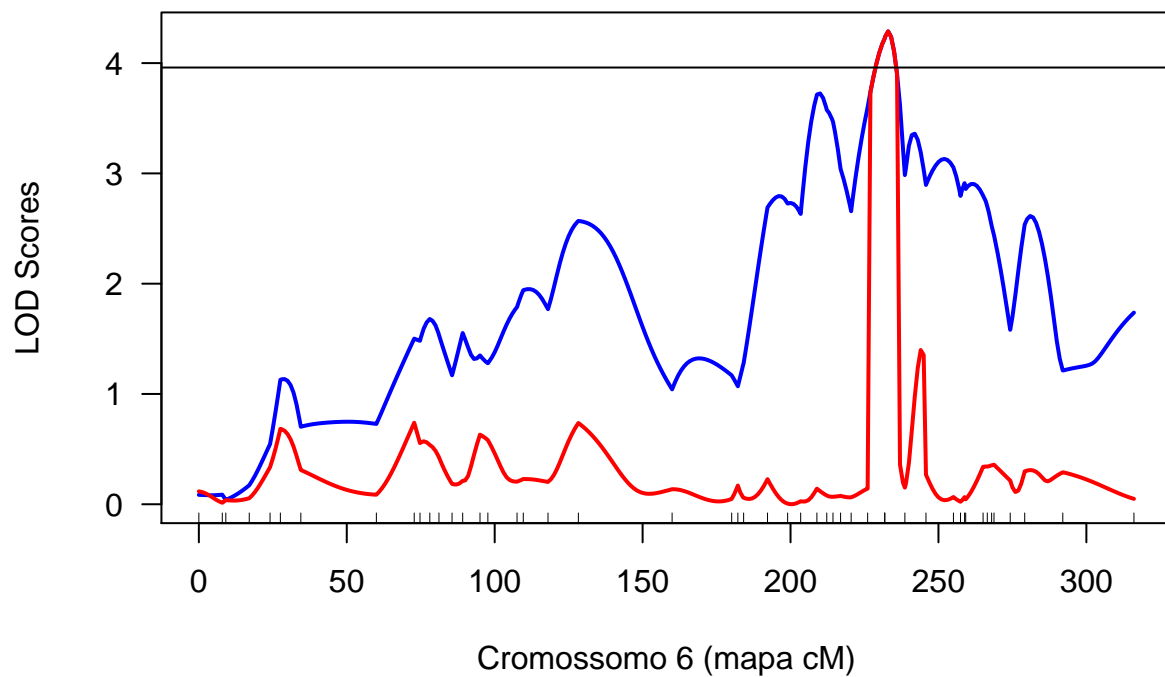
Os dois métodos (IM e CIM) foram comparados pela seguinte análise gráfica:

- O gráfico em azul representa o resultado do IM para a característica **pa**
- O gráfico em vermelho representa o CIM para a característica **pa**

```
plot(mimulus_em,mimulus_cim, col=c("blue","red"), ylab="LOD Scores", xlab="Mapa Genético (cM)")
abline(h = 3.96) #5%
```



```
plot(mimulus_em,mimulus_cim, col=c("blue","red"),
     chr=c(6), ylab="LOD Scores", xlab="Cromossomo 6 (mapa cM)")
abline(h = 3.96) #5%
```



Pelos dois métodos foi encontrado um QTL no cromossomo 6 relacionado com a característica fenotípica **pa**. Observando os resultados gráficos dos dois métodos é evidente a maior precisão do CIM (pico do gráfico com maior precisão da posição do QTL), pois o uso do cofator nesse método permitiu explicar parte da variação genética e minimizar erros, como o próprio efeito do QTL na região adjacente.