

Case study I

In this chapter and the next, we present two case studies to illustrate the QTL mapping process in its entirety. We bring together tools from previous chapters and demonstrate their combined use to solve two moderately difficult problems. Both case studies have features that require special handling. In this sense they are not typical. On the other hand, almost every dataset has quirks that require an alert analyst to recognize them and respond accordingly. Our case studies illustrate the investigative process of QTL data analysis and improvisation using R/qtl.

In this chapter, we will consider the data of Orgogozo *et al.* (2006), included in the R/qtlbook package as the data set `ovar`. This is from a cross between two *Drosophila* species: *D. simulans* was crossed to *D. sechellia*, and the F₁ hybrid was crossed back to *D. simulans*. The phenotype of interest was ovariole number in females, a measure of fitness.

An initial cross produced 402 progeny; 383 had complete phenotype data. Initial genotyping focused on 94 individuals with extreme phenotype, though all individuals were genotyped for five morphological markers.

To increase the resolution of a major QTL identified on chromosome 3, an additional set of approximately 7000 flies were generated, and the 1050 individuals showing a recombination event between two morphological markers, *st* (bright red eyes) and *e* (dark brown body), were phenotyped and genotyped; 1038 individuals had complete phenotype data. The aim was to oversample recombinants in the QTL region, thereby increasing the fine mapping resolution.

There are genotype data for 24 markers on 3 chromosomes. (The fourth chromosome had one marker, but it showed no effect and was excluded from further consideration.)

We begin with data diagnostics that show us the special features of these data: the selective genotyping and phenotyping strategies that were employed. We will then analyze the initial cross, followed by a combined analysis of both crosses. Finally, we discuss the strengths and limitations of the conclusions.

10.1 Diagnostics

Before jumping into QTL mapping analyses, it is useful to perform exploratory analyses of the phenotype and genotype data. This will familiarize us with the data and help diagnose potential artifacts. It is not uncommon to devote half of the intellectual effort of a project on data diagnostics. The use of the various diagnostic tools described in Chap. 3 should become routine.

We must first load the R/qlt and R/qltbook packages, and then the data set, `ovar`.

```
> library(qlt)
> library(qltbook)
> data(ovar)
```

We first get a quick overview of the data.

```
> summary(ovar)
```

Backcross	
No. individuals:	1452
No. phenotypes:	4
Percent phenotyped:	97.9 99 98.1 100
No. chromosomes:	3
Autosomes:	1 2 3
Total markers:	24
No. markers:	2 9 13
Percent genotyped:	29.5
Genotypes (%):	II:50.1 IE:49.9

There are a total of 1452 backcross individuals, with four phenotypes and genotypes at 24 markers. More than two-thirds of the genotype data are missing. The alleles I and E correspond to *D. simulans* and *D. sechellia*, respectively.

We make a summary plot as follows; see Fig. 10.1.

```
> plot(ovar)
```

The genetic map has some large gaps, with markers separated by as much as 40 cM. The phenotypes `on1` and `on2` are the ovariole counts for the two gonads. The phenotype `onm` is the average of the two counts. For several individuals, the ovariole count for one of the two gonads was missing, and so `onm` is missing. The fourth phenotype, `cross`, indicates which individuals were from the initial cross and which were from the second, selectively phenotyped cross.

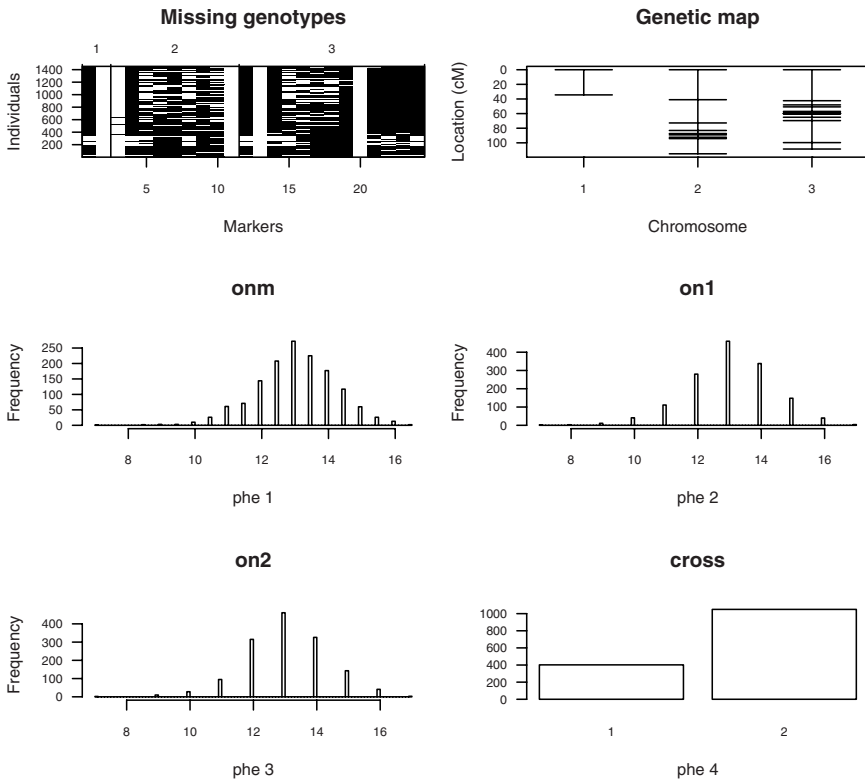


Figure 10.1. The summary plot of the `ovar` data provided by the `plot.cross` function, including the pattern of missing genotype data (upper left; black pixels indicate missing data), the genetic map of the typed markers (upper right), histograms of the three phenotypes, `onm` (average ovariole count), and `on1` and `on2` (gonad-specific ovariole counts), and a bar plot of the `cross` phenotype, indicating the numbers of individuals from the initial cross and from the second, selectively phenotyped cross.

Let us plot the two gonad-specific ovariole counts against each other. We use the function `jitter` to add a small amount of random noise to the counts so that we can distinguish multiple individuals with the same ovariole counts.

```
> plot(jitter(on2) ~ jitter(on1), data=ovar$pheno,
+       xlab="on1", ylab="on2", cex=0.6,
+       xlim=c(6.66, 17.53), ylim=c(6.66, 17.53))
```

We use `cex` to make the points smaller, and `xlim` and `ylim` to force the x - and y -axis limits to be the same.

Figure 10.2 indicates clear but not overly strong association between the two counts. We may use `cor` to calculate the sample correlation between the two counts. The argument `use="complete"` is required due to the missing

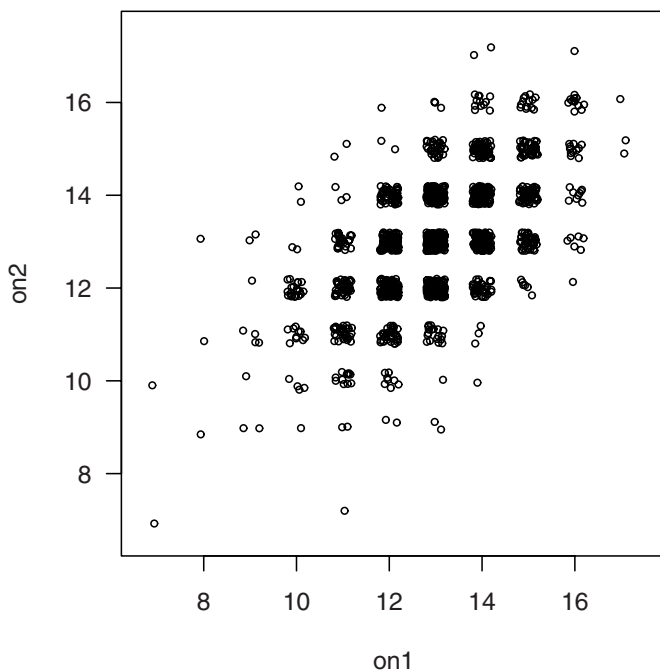


Figure 10.2. Scatterplot of `on1` and `on2`, the gonad-specific ovariole counts in the `ovar` data, with points randomly jittered so that overlapping points may be distinguished.

data; the correlation is then calculated using only those individuals with complete data.

```
> cor(pull.pheno(ovar, "on1"), pull.pheno(ovar, "on2"),
+     use="complete")
[1] 0.582
```

The phenotype `onm` should be the average of `on1` and `on2`. It is a good idea to check this.

```
> max(abs(pull.pheno(ovar, "onm") -
+       (pull.pheno(ovar, "on1")+pull.pheno(ovar, "on2"))/2),
+     na.rm=TRUE)
[1] 0
```

The phenotype `onm` should be missing if either `on1` or `on2` is missing. We should check this, too.

```
> table(apply(is.na(pull.pheno(ovar, 1:3)), 1, paste,
+            collapse=":"))
```

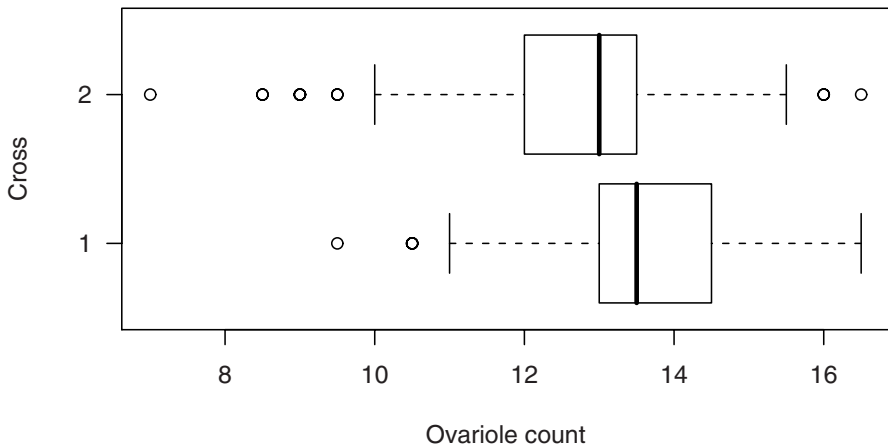


Figure 10.3. Box plot of `onm` for the two crosses forming the `ovar` data.

FALSE:FALSE:FALSE	FALSE:TRUE:FALSE	TRUE:FALSE:TRUE
1419	2	19
TRUE:TRUE:FALSE	TRUE:TRUE:TRUE	
3	9	

The three TRUE/FALSE values indicate whether the `onm`, `on1` and `on2` phenotypes, respectively, are missing or not. There are 1419 individuals with no missing phenotype, 19 individuals missing `onm` and `on2`, 3 individuals missing `onm` and `on1`, and 9 individuals missing all three phenotypes. There are also two individuals for which `onm` is not missing but `on1` is missing.

```
> ovar$pheno[!is.na(ovar$pheno$onm) & is.na(ovar$pheno$on1),]
      onm on1 on2 cross
1300  13  NA  13     2
1325  12  NA  12     2
```

We will fix these.

```
> ovar$pheno$onm[is.na(ovar$pheno$on1)] <- NA
```

A boxplot of the `onm` phenotypes in the two crosses will indicate whether there are systematic differences.

```
> boxplot(onm ~ cross, data=ovar$pheno, horizontal=TRUE,
+         xlab="Ovariole count", ylab="Cross")
```

As seen in Fig. 10.3, the ovariole counts were quite a bit smaller in the second cross. A *t* test will indicate whether this could reasonably be ascribed to chance variation (though the figure is sufficiently convincing).

```
> t.test(onm ~ cross, data=ovar$pheno)
```

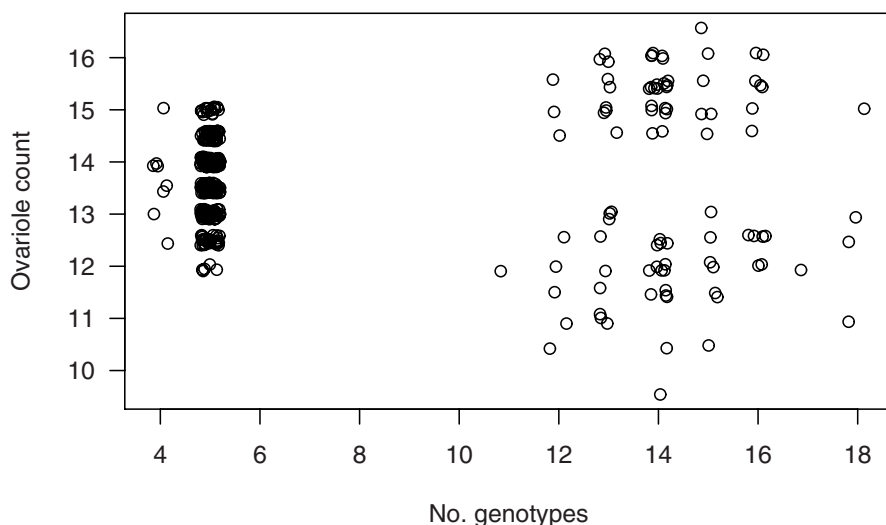


Figure 10.4. Plot of ovariole count against number of typed marker genotypes for the initial cross in the `ovar` data. On the right are points corresponding to the 94 genotyped individuals; on the left are the remaining individuals, for which only five morphological markers were scored.

Welch Two Sample t-test

```
data:  onm by cross
t = 11.75, df = 710.2, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.6590 0.9235
sample estimates:
mean in group 1 mean in group 2
    13.64         12.85
```

Let us now study the genotype data in the first cross. We first use `subset.cross` to pull out the individuals from the first cross.

```
> ovar1 <- subset(ovar, ind=(pull.pheno(ovar, "cross")==1))
```

Note the selective genotyping.

```
> plot(jitter(ntyped(ovar1)), jitter(pull.pheno(ovar1, "onm")),
+       xlab="No. genotypes", ylab="Ovariole count")
```

As seen in Fig. 10.4, the individuals with the most genotype data have high (≥ 14.5) or low (≤ 13) phenotype, but there were no strict cutoffs: there is some overlap between the more and less highly genotyped groups.

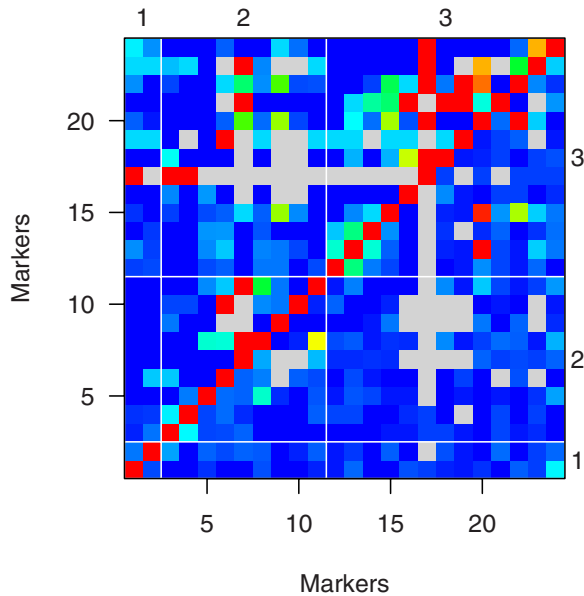


Figure 10.5. Plot of estimated recombination fractions (upper left) and LOD scores for a test of $r = 1/2$ (lower right) for all pairs of markers in the initial cross in the `ovar` data. Red indicates linkage, blue indicates no linkage, and gray indicates missing values (for marker pairs that were not both typed in any one individual).

The extensive missing genotype data cause some odd patterns in the estimated recombination fractions between markers, but it is nevertheless useful to plot these (see Sec. 3.4.1).

```
> ovar1 <- est.rf(ovar1)
> plot.rf(ovar1)
```

In Fig. 10.5, the LOD scores (in the lower right triangle) indicate no evidence for linkage between markers on different chromosomes.

The map associated with the data was established based on Macdonald and Goldstein (1999). It is a good idea to reestimate the intermarker distances, using the observed data, keeping marker order fixed, and plot the estimated map against that which came with the data.

```
> newmap <- est.map(ovar1, error.prob=0.001, verbose=FALSE)
> plot.map(ovar1, newmap)
```

There is some evidence for map expansion (see Fig. 10.6), though this may be partly due to the choice of map function. By default, `est.map` uses the Haldane map function to convert estimated recombination fractions to genetic distances. The choice of map function has a greater effect on larger marker intervals.

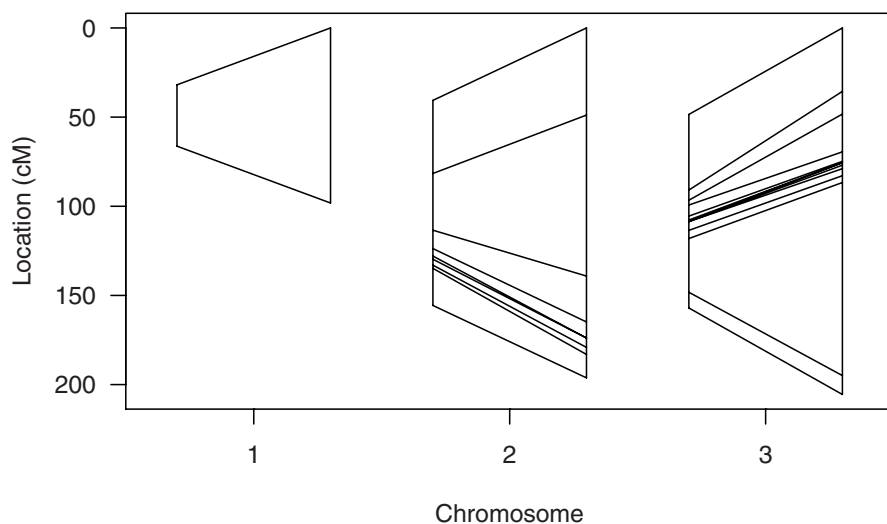


Figure 10.6. The genetic map in the `ovar` data plotted against the map estimated from the individuals in the initial cross. For each chromosome, the line on the left is the map provided with the data, and the line on the right is the map estimated using the Haldane mapping function; line segments connect the positions for each marker.

If we use the Kosambi map function (rather than the Haldane map function), the chromosomes are quite a bit smaller.

```
> newmap.k <- est.map(ovar1, err=0.001, map.function="kosambi")
> summary(newmap)
```

	n.mar	length	ave.spacing	max.spacing
1	2	98.2	98.2	98.2
2	9	196.2	24.5	90.3
3	13	205.5	17.1	108.3
overall	24	500.0	23.8	108.3

```
> summary(newmap.k)
```

	n.mar	length	ave.spacing	max.spacing
1	2	64.6	64.6	64.6
2	9	147.1	18.4	60.3
3	13	153.8	12.8	70.0
overall	24	365.5	17.4	70.0

We will stick with the map that came with the data, and we will use the Kosambi map function in our QTL analyses. There is not a compelling reason to replace the original map, and since crossover interference is known to exist in *Drosophila*, it seems prudent to use the Kosambi map function.

Finally, let us consider the chromosome 3 genotypes for the individuals in the second cross (which were selected to be recombinant between the markers *st* and *e*). The function `geno.crosstab` can be used to create a table of two-locus genotypes.

```
> ovar2 <- subset(ovar, ind=(pull.pheno(ovar,"cross")==2))
> geno.crosstab(ovar2, "st", "e")

      e
st    -  II  IE
-     0   1   0
II    0   0 566
IE    0 481   2
```

There are two individuals that are not recombinant (these might be errors) and one individual that has missing genotype for the *st* marker.

Consider the same cross-tabulation for the first cross.

```
> geno.crosstab(ovar1, "st", "e")

      e
st    -  II  IE
-     0   1   0
II    0 146  72
IE    0  47 136
```

There were 30% recombinants in the first cross.

Let us plot the chromosome 3 genotypes for a random set of 15 individuals from the second cross.

```
> topplot <- sort(sample(nind(ovar2), 15))
> plot.geno(ovar2, chr=3, ind=topplot)
```

As seen in Fig. 10.7, these individuals were genotyped just within the region of the selected recombination event.

10.2 Initial cross

We now turn to QTL mapping. We first focus on the initial cross of 402 individuals (of which 383 have complete phenotype data). In the last section, we created a cross object, `ovar1`, with just those individuals. To avoid repeated warning messages, we omit individuals for which the `onm` phenotype is missing.

```
> ovar1 <- subset(ovar1, ind=!is.na(pull.pheno(ovar1,"onm")))
```

We will use multiple imputation for the QTL mapping, because of the selective genotyping and our desire to fit multiple-QTL models (see Sec. 4.2.4). The extensive missing genotype data suggests that we should perform a large number of imputations; we will use 512. (We like powers of 2.) We begin with a call to `sim.geno`, to do the imputations.

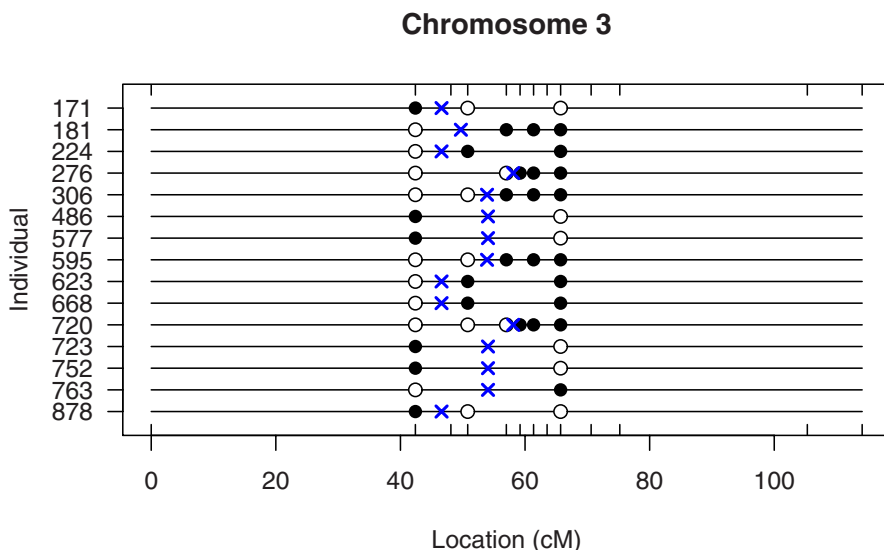


Figure 10.7. Chromosome 3 genotypes for a random set of individuals from the second *ovar* cross, showing the selective genotyping of recombinants. Blue $\times$'s indicate recombination events.

```
> ovar1 <- sim.geno(ovar1, n.draws=512, step=2, err=0.001,
+                   map.function="kosambi")
```

We now use `scanone` to perform a single-QTL genome scan by the multiple imputation approach.

```
> out1 <- scanone(ovar1, method="imp")
```

A quick summary and plot of the genome scan results (Fig. 10.8) indicates strong evidence for a QTL on chr 3, and some evidence for an additional QTL on chr 2.

```
> summary(out1)

      chr  pos    lod
per     1   4.5  0.213
SRPK    2  87.3  2.228
cpo     3  82.4 14.010
```

```
> plot(out1, ylab="LOD score")
```

The evidence for the QTL on chr 3 (with a LOD score of 14.01) is clear. It is worthwhile considering the estimated effect of this QTL. We can make a plot with `effectplot`, or we could get a numerical estimate of the effect with `fitqtl`. Let us do both.

First, we use `effectplot` to plot the estimated phenotype averages for each of the two QTL groups (see Sec. 4.6).

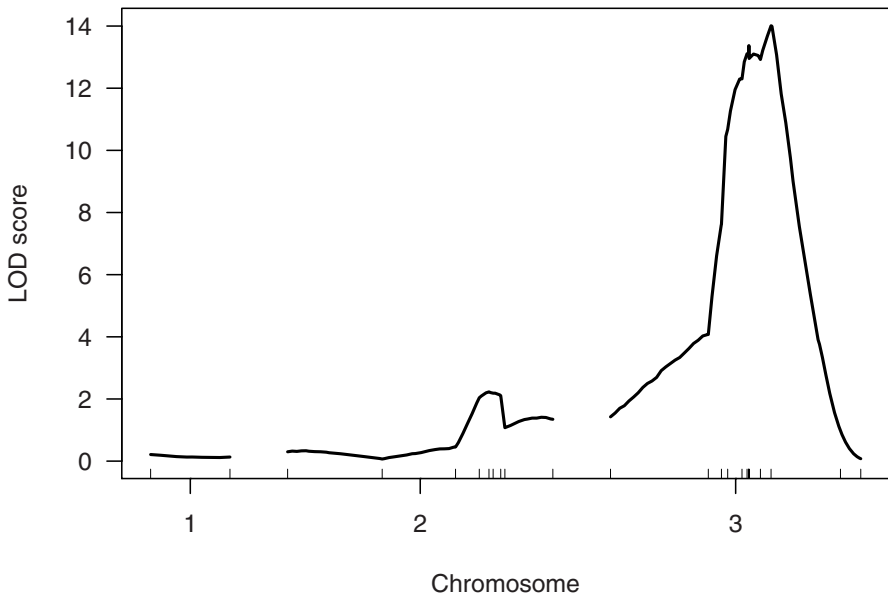


Figure 10.8. LOD curves from a genome scan by multiple imputation for the initial cross in the `ovar` data.

```
> effectplot(ovar1, mname1="cpo")
```

The plot, in Fig. 10.9, indicates that the *D. simulans* allele (I) is associated with one additional ovariole per gonad.

To use `fitqtl` to get a numerical estimate of the QTL effect, we first use `makeqtl` to create a QTL object and then call `fitqtl` with `dropone=FALSE`, as we can skip the drop-one-QTL analysis, and `get.ests=TRUE`, to get the estimates (see Sec. 9.3.1).

```
> qtl <- makeqtl(ovar1, chr=3, pos=82.4)
> summary(fitqtl(ovar1, qtl=qtl, dropone=FALSE, get.ests=TRUE))
```

Full model result

Model formula: $y \sim Q1$

	df	SS	MS	LOD	%var	Pvalue(Chi2)	Pvalue(F)
Model	1	73.3	73.296	14.01	15.50	9.992e-16	1.110e-15
Error	381	399.5	1.049				
Total	382	472.8					

Estimated effects:

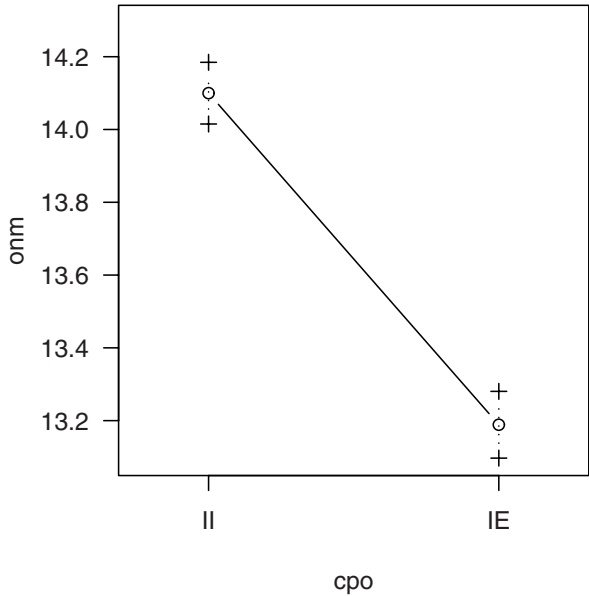


Figure 10.9. Estimated phenotype averages for the two groups defined by genotypes at marker *cpo*, for the initial cross in the *ovar1* data. Error bars are ± 1 SE.

	est	SE	t
Intercept	13.65411	0.05323	256.499
3082.4	-0.89491	0.10602	-8.441

The estimated effect of the chr 3 QTL is -0.89 ± 0.11 .

Given the large effect of the chr 3 QTL, let us repeat the genome scan, controlling for this locus. If we had complete genotype data at the inferred QTL, we could include it as an additive covariate in `scanone`. But in the current situation, with considerable missing data at the inferred QTL, we use `addqtl` to scan for an additional QTL using multiple imputation (see Sec. 9.3.4).

```
> out1.c3 <- addqtl(ovar1, qtl=qtl)
```

Again, we make a quick summary and plot (Fig. 10.10).

```
> summary(out1.c3)
```

	chr	pos	lod
c1.loc32	1	36.5	0.173
c2.loc90	2	90.0	3.531
slo	3	121.3	2.543

```
> plot(out1.c3, ylab="LOD score")
```

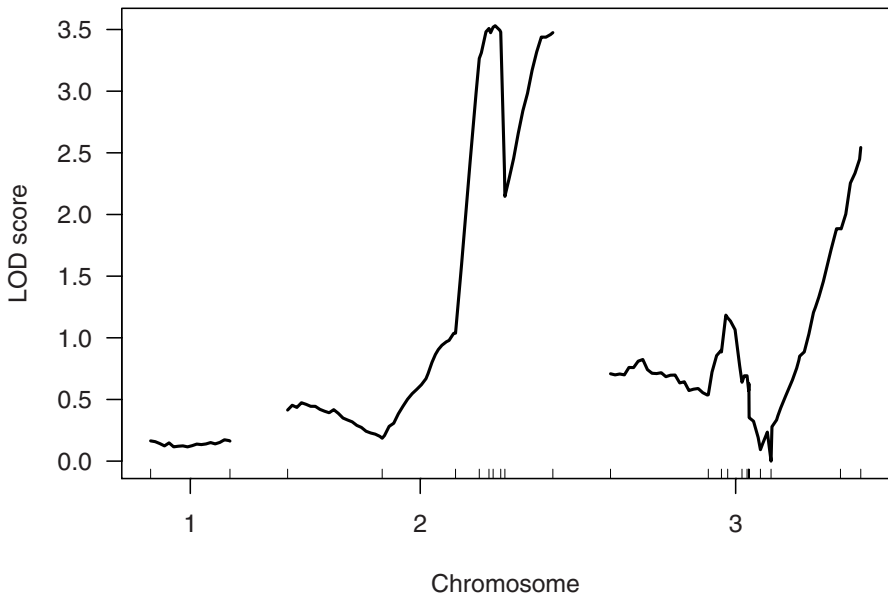


Figure 10.10. LOD curves from a genome scan by multiple imputation for the initial cross in the *ovar* data, adjusting for an inferred QTL at the marker *cpo*.

We have improved evidence for a QTL on chr 2 (the LOD increased from 2.23 to 3.53), and there is some evidence for an additional QTL at the distal end of chr 3.

Let us perform a two-dimensional, two-QTL scan on chr 3, to assess the evidence for a second QTL on the chromosome. We will include the inferred QTL on chr 2 as an additive covariate. This requires that we first create a new QTL object (containing just the chr 2 QTL) and run `addpair` (see Sec. 9.3.5).

```
> qt12 <- makeqtl(ovar1, 2, 90)
> out1.ap <- addpair(ovar1, qt1=qt12, chr=3, verbose=FALSE)
```

We can use `summary.scantwo` to pick off the largest peak for both a full model (with two interacting QTL) and an additive model (see Sec. 8.1).

```
> summary(out1.ap)
```

	pos1f	pos2f	lod.full	lod.fv1	lod.int	pos1a	pos2a
c3:c3	82.8	121	17.6	2.30	0.143	82.8	121
	lod.add	lod.av1					
c3:c3			17.5	2.16			

The evidence for a second QTL on chr 3 remains, but it is not strong, and actually the evidence is weaker once the QTL on chr 2 is considered. (The LOD for two additive QTL on chr 3, versus a single QTL near the *cpo* marker, is 2.16 when the chr 2 locus is included in the model and is 2.54 when the chr 2

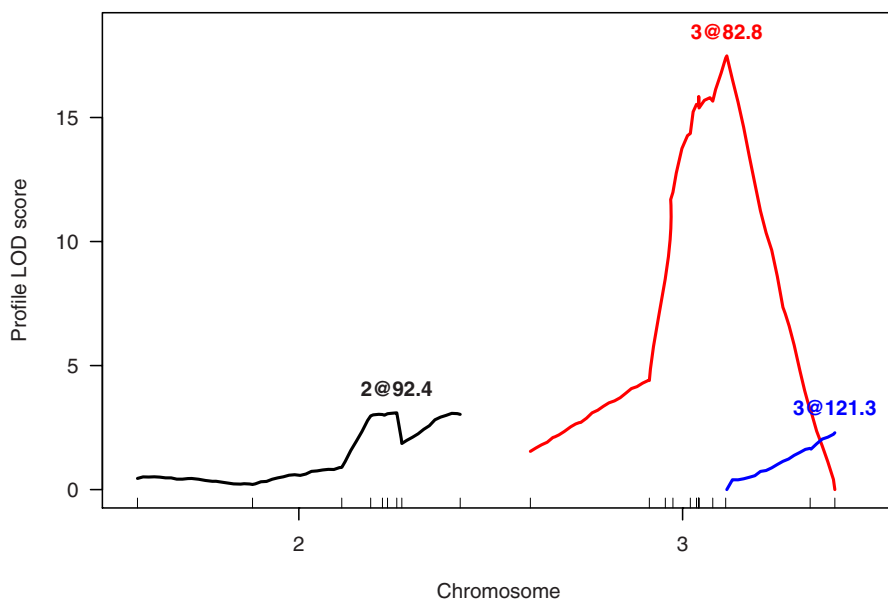


Figure 10.11. Profile LOD curves for a three-QTL model, for the initial cross in the *ovar* data.

locus is not included.) There is no evidence for an interaction between the loci on chr 3.

We now bring all three putative QTL into one model, and use `refineqtl` to improve our estimates of the QTL positions (see Sec. 9.3.2).

```
> qtl <- makeqtl(ovar1, c(2, 3, 3), c(90, 82.8, 121))
> rqtl <- refineqtl(ovar1, qtl=qtl, verbose=FALSE)
```

The QTL moved slightly:

```
> rqtl
```

	name	chr	pos	n.gen
Q1	2@92.4	2	92.4	2
Q2	3@82.8	3	82.8	2
Q3	3@121.3	3	121.3	2

We may use `plotLodProfile` to plot LOD profiles for the three QTL (see Sec. 9.3.2). The results appear in Fig. 10.11.

```
> plotLodProfile(rqtl, col=c("black", "red", "blue"),
+               ylab="Profile LOD score")
```

We are particularly interested in deriving a confidence interval for the location of the proximal QTL on chr 3, as the selective phenotyping in the second cross in these data was performed with the aim of more precisely

mapping this locus. As described in Sec. 9.3.2, we may use `lodint` or `bayesint` to derive an approximate confidence interval for QTL location, based on the LOD profile calculated by `refineqtl`. These intervals need to be viewed with some caution, however, as they fail to account for the uncertainty in the location of the other QTL, and the performance of these intervals in the context of a multiple-QTL model is not well understood.

In the use of `lodint` and `bayesint`, we refer to the QTL by their numeric index within the QTL object. The proximal QTL on chr 3 is the second locus within `rqtl`, and so we use `qtl.index=2`.

```
> lodint(rqtl, qtl.index=2)

      chr  pos   lod
nos      3 77.8 15.66
c3.loc70  3 82.8 17.48
c3.loc74  3 86.8 15.62

> bayesint(rqtl, qtl.index=2)

      chr  pos   lod
c3.loc66  3 78.8 16.13
c3.loc70  3 82.8 17.48
c3.loc72  3 84.8 16.53
```

These intervals are reasonably small. (The 1.5-LOD support interval is 9.0 cM long, and the approximate 95% Bayesian credible interval is 6.0 cM long.)

It is surprising that the intervals start at 77.8 cM and so do not cover the region used to select recombinants in the second cross: selected individuals showed a recombination event between markers *st* (at 55.2 cM) and *e* (at 72.9 cM). If our confidence intervals are accurate, the selective phenotyping should not be expected to narrow the location of the QTL, as the selected recombination events will be flanking the QTL, rather than covering it. The region used to select recombinants was chosen based on an initial analysis with a simpler model and with the QTL Cartographer software, which indicated a QTL peak to the left of the *e* marker; see Fig. 2B in Orgogozo *et al.* (2006). Moreover, the *e* marker was the most distal morphological marker available in the *D. simulans* strain that was used for these crosses.

We have neglected to be precise about the strength of evidence for the three inferred QTL. The proximal QTL on chr 3 (with LOD score ~ 14) is clear, but in considering the other two QTL, we must take this large-effect QTL into account. Let us apply the multiple-QTL model comparison criterion described in Sec. 9.1.4.

We must first derive appropriate penalties for the penalized LOD score criterion, using a permutation test with a two-dimensional, two-QTL genome scan. The computational effort is forbidding, but it can be made feasible by the parallel use of multiple computers (see Sec. 8.1, page 223). The permutation

test here took about four days of computer time, but we split it across 16 processors, so it took about six hours in real time.

Due to the selective genotyping in this initial cross (only 94 individuals chosen by their relatively extreme phenotypes were typed at most markers), it is best to perform a stratified permutation test (see Sec. 4.4.3).

```
> strat <- (nmissing(ovar1) < 15)
> operm <- scantwo(ovar1, method="imp", n.perm=1000,
+                 perm.strata=strat)
```

The significance thresholds can be calculated as follows.

```
> summary(operm)

onm (1000 permutations)
      full  fv1  int  add  av1  one
5%    3.83  2.87  2.09  3.01  1.77  1.70
10%   3.46  2.47  1.83  2.61  1.49  1.45
```

Most importantly, we may calculate the penalties as follows. Note the use of `print` to simultaneously print the results and assign them to the object `pen`.

```
> print(pen <- calc.penalties(operm))

main heavy light
1.698 2.085 1.176
```

The significance thresholds and penalties are much smaller than others we have seen in this book, but keep in mind that the null distributions of the various LOD scores depend on not just the type of LOD score (e.g., the LOD score from a single-QTL genome scan versus the “full” LOD, comparing a model with two interacting loci to the null model, in a two-dimensional, two-QTL genome scan) and the type of cross (backcross versus intercross), but also on the genetic length of the genome. *Drosophila* has a much smaller genetic map than the mouse (which has been the focus of all of our previous examples), and so the significance thresholds and penalties are much smaller. Here, the penalty on main effects is just 1.70, and so all three of our QTL would be selected.

Let us complete our analysis of the initial cross in the `ovar` data by applying the stepwise model search algorithm described in Sec. 9.1.3, using the function `stepwiseqtl`. We can start forward selection at our current model, defined by `rqtl`.

```
> stepout1 <- stepwiseqtl(ovar1, penalties=pen, qtl=rqtl,
+                        max.qtl=8, verbose=FALSE)

> stepout1
```

	name	chr	pos	n.gen
Q1	2@92.4	2	92.4	2
Q2	2@94.2	2	94.2	2
Q3	2@112.0	2	112.0	2
Q4	3@82.8	3	82.8	2
Q5	3@121.3	3	121.3	2

Formula: $y \sim Q1 + Q2 + Q3 + Q4 + Q5 + Q1:Q2$
 pLOD: 14.82

It is a bit of a surprise to see three QTL on chr 2, and the two tightly linked epistatic loci on chr 2 are rather suspicious. We often have difficulty with tightly linked QTL, particularly if they are allowed to interact. Let's look at a couple of plots of the phenotypes against the two-locus genotypes, using both `effectplot` and `plot.pxd`, to see what's going on. We use `find.marker` to find the marker closest to each QTL, for use in `plot.pxd`. In `effectplot`, we may refer directly to pseudomarkers (that is, the positions on the grid between markers), by their chromosome and cM position.

```
> mar <- find.marker(ovar1, 2, c(92.4, 94))
> par(mfrow=c(1,2))
> effectplot(ovar1, mname1="2@92.4", mname2="2@94")
> plot.pxd(ovar1, marker=mar)
```

The right panel in Fig. 10.12 suggests that the phenotypes for just a few individuals are leading to the large LOD score. (There are just three individuals with genotypes IE at marker *Amy-d* and II at marker *grh*. No individual has observed genotypes II at marker *Amy-d* and IE at marker *grh*; the points in the plot come from a single imputation, using data from surrounding markers.) Thus, we should discount the evidence for the two tightly linked QTL.

Let's omit the QTL at 94.2 cM (using `dropfromqtl`) and run `refineqtl` to refine the locations of the other QTL.

```
> qtl <- dropfromqtl(stepout1, 2)
> qtl <- refineqtl(ovar1, qtl=qtl, verbose=FALSE)
```

We can print the object to see if the QTL moved.

```
> qtl
```

	name	chr	pos	n.gen
Q1	2@92.4	2	92.4	2
Q2	2@94.0	2	94.0	2
Q3	3@82.8	3	82.8	2
Q4	3@121.3	3	121.3	2

The distal locus on chr 2 moved back to the 94 cM position, which we don't trust! We should probably stick with the three-QTL model, with just one QTL on chr 2.

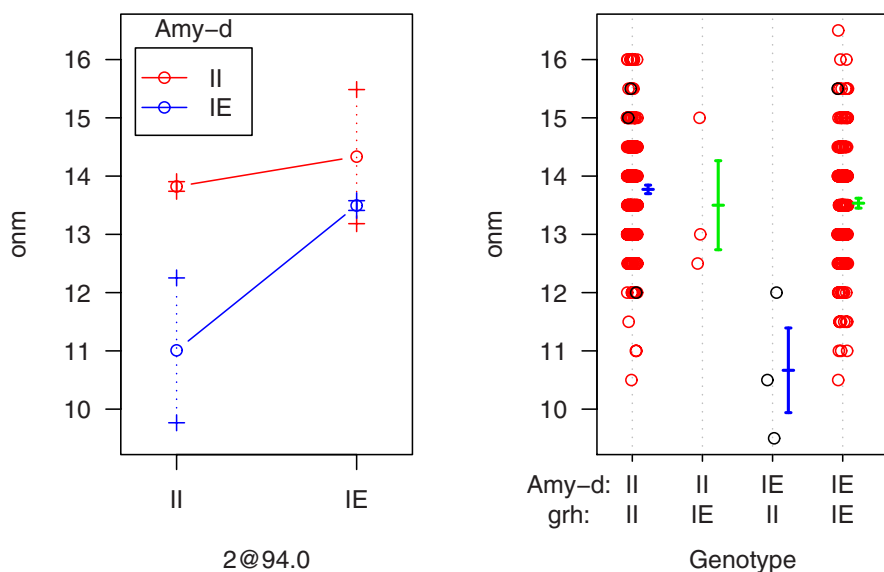


Figure 10.12. Plot of the *onm* phenotype against genotype at the two tightly linked loci on chr 2, for the initial cross in the *ovar* data. Red dots in the right panel are for imputed genotypes. Error bars are ± 1 SE.

10.3 Combined data

We now turn to an analysis of the combined data, including the second cross of 1050 individuals (of which 1038 have complete phenotype data) that were selected to be recombinant between the morphological markers, *st* (bright red eyes) and *e* (dark brown body).

Imputed genotypes (from `sim.geno`) take up a great deal of memory, and so it might be best to either remove them from the `ovar1` object (which we were working with in the last section) using `clean.cross`, or just remove the `ovar1` object from our workspace (using `rm`). One may use `object.size` to estimate the amount of memory (in bytes) that an object is taking up. Another useful tool is the function `gc`, which tells R to perform “garbage collection” to clean up memory and also gives information about R’s current memory usage.

```
> object.size(ovar1)
```

```
[1] 237010992
```

Dividing by 1024^2 gives the result in megabytes (Mb).

```
> object.size(ovar1)/1024^2
```

```
[1] 226.0
```

We now use `clean` to remove all of the intermediate calculations from the object, and check the new size, in bytes.

```
> ovar1 <- clean(ovar1)
> object.size(ovar1)
```

```
[1] 103304
```

Dividing by 1024 gives the result in kilobytes (kb).

```
> object.size(ovar1)/1024
```

```
[1] 100.9
```

And so, by removing the intermediate calculations, the space used by `ovar1` went from 226 Mb to 101 kb.

We turn now to the full data, `ovar`. Let us first remove individuals missing the `onm` phenotype.

```
> ovar <- subset(ovar, ind=!is.na(pull.pheno(ovar, "onm")))
```

We use `sim.geno` to perform the multiple imputations. We will use the same settings as we had used for `ovar1` in the previous section.

```
> ovar <- sim.geno(ovar, n.draws=512, step=2, err=0.001,
+                 map.function="kosambi")
```

Because of the systematic difference in ovariole counts between the two crosses (see Fig. 10.3 on page 287), it would be best to include a cross indicator as an additive covariate in our analyses. In doing so, we assume that the effects of QTL are the same in the two crosses, but that there is an overall shift in the mean phenotype. The covariate must be numeric, but the coding of the two crosses (e.g., 0/1 or 1/2) has no influence on the LOD scores or estimated QTL effects that we calculate.

```
> cross <- pull.pheno(ovar, "cross")
> outc <- scanone(ovar, method="imp", addcovar=cross)
```

We will perform the same genome scan with just the individuals from the second cross, for comparison purposes. We do not need to rerun `sim.geno`, as the imputations will be retained in the output of `subset.cross`.

```
> out2 <- scanone(subset(ovar, ind=(cross==2)), method="imp")
```

We now plot the results for the combined data and for the two individual crosses.

```
> plot(outc, out1, out2, ylab="LOD score")
```

In the LOD curves in Fig. 10.13, we see that the results for the combined data (the black curves) are similar to those for the initial cross (the blue curves), though the LOD scores are much larger and the location of the QTL on chr 3 appears to be shifted to the left slightly. On chr 3, the LOD curve for

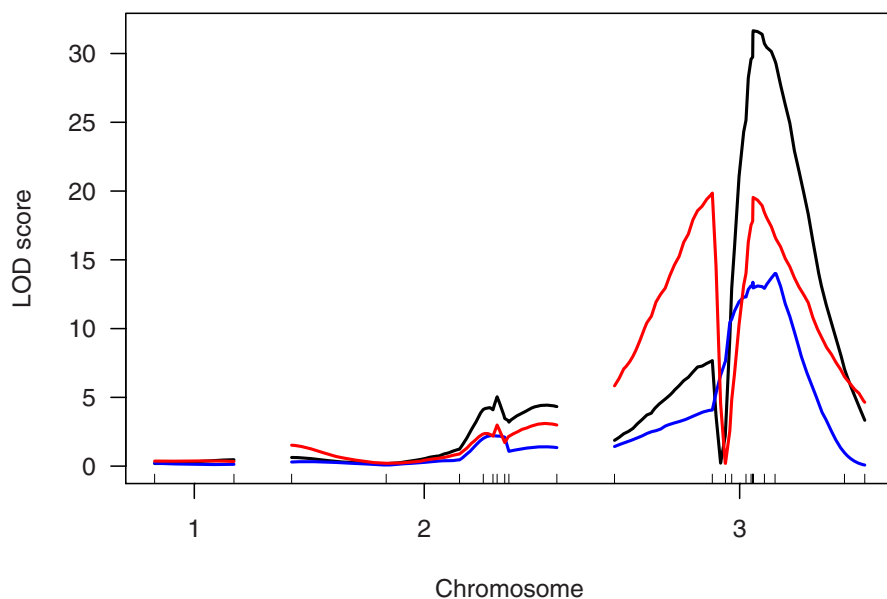


Figure 10.13. LOD curves from a genome scan by multiple imputation for the *ovar* data. The black curves are for the combined data, the blue curves are for the initial cross, and the red curves are for the second cross.

the second cross (the red curve) has an odd double peak. As the individuals were selected to be recombinant in the region between markers *st* and *e*, they have exactly opposite genotypes on either side of that region (and recall that they were not genotyped outside the region), and so a QTL to one side will show a mirror image on the other side (with the apparent QTL on the other side having an effect of the opposite sign).

We note the location of the maximum LOD score from the single-QTL genome scan, for the combined data.

```
> max(outc)
```

```
chr pos lod
e   3 72.9 31.7
```

The QTL has moved to the left, from 82.4 cM (as inferred with the initial cross) to 72.9 cM (as inferred from the combined data).

Let us again adjust for this major locus, and scan for a second QTL. We will again consider the combined data as well as the second cross, on its own, and we will compare the results to those from the initial cross. Note that for `addqtl`, the covariates need to form a matrix or data frame, and so in the first line we convert our `cross` covariate to a data frame.

```
> cross <- data.frame(cross=cross)
> qt1c <- makeqtl(ovar, chr=3, pos=72.9)
```

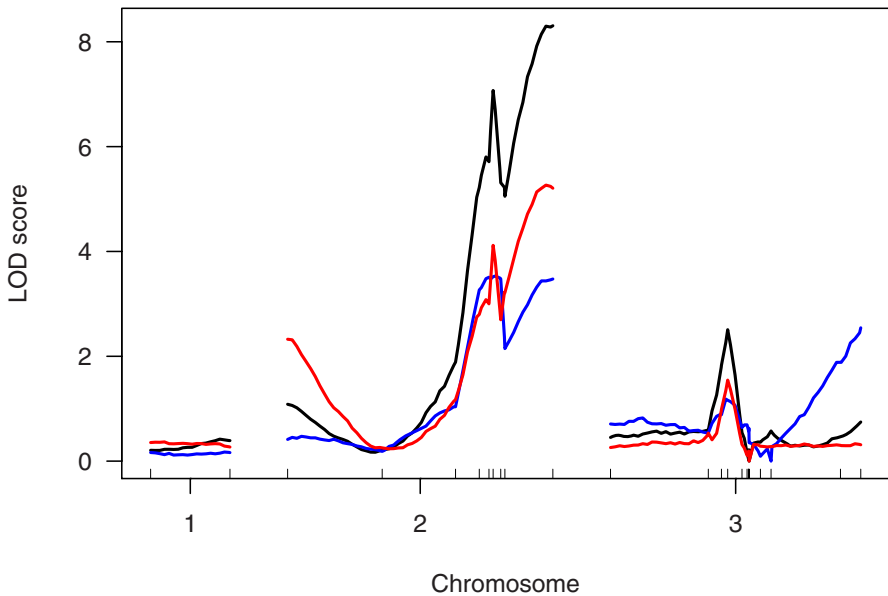


Figure 10.14. LOD curves from a genome scan by multiple imputation for the *ovar* data, adjusting for an inferred QTL on chr 3. The black curves are for the combined data, the blue curves are for the initial cross, and the red curves are for the second cross. Note that the position of the inferred QTL (on which we are conditioning) is different for the initial cross versus for the combined data and for the second cross.

```
> outc.c3 <- addqtl(ovar, qtl=qtlc, covar=cross)
> qt12 <- makeqtl(subset(ovar, ind=(cross==2)), chr=3, pos=72.9)
> out2.c3 <- addqtl(subset(ovar, ind=(cross==2)), qtl=qt12)
```

A plot of the LOD curves, controlling for the major locus on chr 3, is obtained as follows; see Fig. 10.14.

```
> plot(outc.c3, out1.c3, out2.c3, ylab="LOD score")
```

The results on chr 3 are quite different for the combined data; we now have clear evidence for a second QTL on chr 3. There remains strong evidence for a QTL on chr 2 (and possibly there are two QTL on chr 2).

Before proceeding further, let's run the necessary permutation tests that will help us to make sense of the statistical significance of our results. We will perform a permutation test with a two-dimensional, two QTL scan. It will be best to perform a stratified permutation test, as we had done for the permutation test with the data from initial cross in the last section. Here we will want three strata: the individuals from the initial cross that were selected for the initial genotyping, the other individuals from the initial cross, and the individuals from the second cross.

The `perm.strata` argument in `scantwo` should be a vector of indices that indicate which individuals are in which stratum. We can create such a vector as follows.

```
> strat <- pull.pheno(ovar, "cross")
> strat[strat==1 & nmissing(ovar)<15] <- 3
> table(strat)
```

```
strat
  1    2    3
289 1036  94
```

The particular numeric codes that we assign to the three strata can be anything, and so we do whatever is most convenient.

We now are ready to perform the permutation test. We should again emphasize that the computation time is long, and so it is best to split the permutations into batches using multiple computers running in parallel. (These computations took 30 days of CPU time.)

```
> opermc <- scantwo(ovar, method="imp", addcovar=cross,
+                  n.perm=n.perm, perm.strata=strat)
```

We obtain the estimated significance thresholds and penalties for our multiple-QTL model comparison criterion as follows.

```
> summary(opermc)

onm (1000 permutations)
      full fv1 int  add  av1  one
5%   3.01 1.99 1.53 2.46 1.090 1.63
10%  2.72 1.79 1.33 2.06 0.896 1.33

> print(penc <- calc.penalties(opermc))

      main heavy light
1.6285 1.5346 0.3603
```

The significance thresholds and penalties (particularly the interaction penalties) are quite a bit smaller than we had obtained based on the initial cross alone (see page 298).

Returning to the QTL analyses, we have reasonable evidence for at least one QTL on chr 2 and likely two QTL on chr 3. A simple approach to explore these models would be to perform two-dimensional, two-QTL scans on each chromosome, controlling for the major locus on the other chromosome. Let's begin with chromosome 3. We first create a QTL object containing the chr 2 locus, and then use `addpair` to scan for a pair of QTL on chr 3.

```
> qtl.c2 <- makeqtl(ovar, 2, 115)
> out.ap.c3 <- addpair(ovar, qtl=qtl.c2, chr=3, covar=cross,
+                   verbose=FALSE)
```

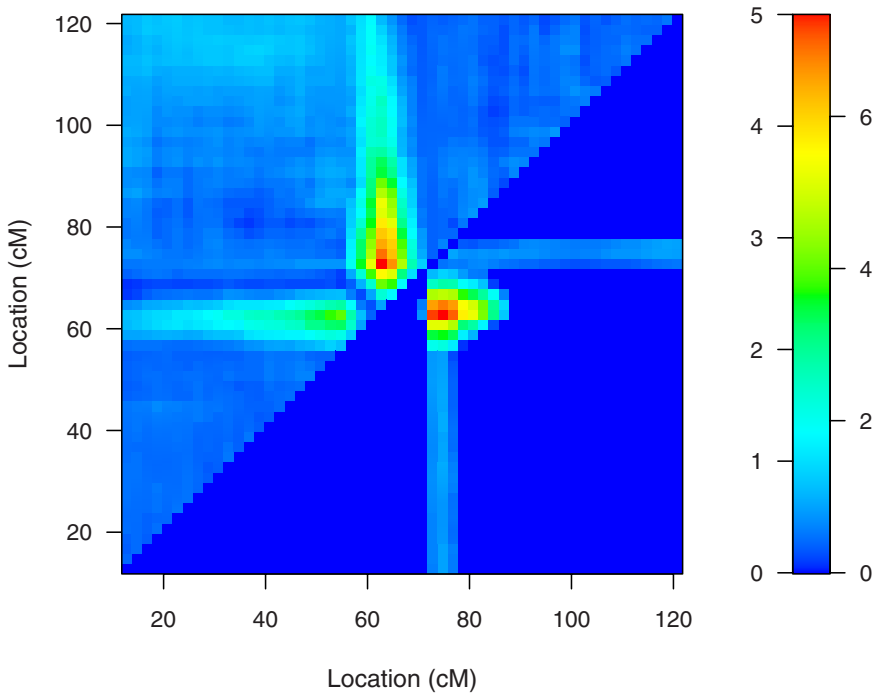


Figure 10.15. LOD scores from a two-dimensional, two-QTL scan on chr 3, controlling for a locus on chr 2, for the *ovar* data. LOD_i is displayed in the upper left triangle; LOD_{fv1} is displayed in the lower right triangle. In the color scale on the right, numbers to the left and right correspond to LOD_i and LOD_{fv1} , respectively.

The summary of the results indicates strong evidence for two interacting QTL on chr 3.

```
> summary(out.ap.c3)
```

	pos1f	pos2f	lod.full	lod.fv1	lod.int	pos1a	pos2a
c3:c3	62.8	74.8	42.9	7.33	4.68	62.8	74.8
	lod.add		lod.av1				
c3:c3	38.3	2.65					

A plot of the LOD scores from the two-dimensional scan is useful for assessing the precision of localization of the two QTL. We will plot the interaction LOD scores in the upper left triangle (the default); in the lower right triangle, let us plot LOD scores comparing the full model (with two interacting loci) to the best single-QTL model.

```
> plot(out.ap.c3, lower="cond-int")
```

As seen in Fig. 10.15, the locations of the QTL are reasonably well defined.

We use a similar two-dimensional, two-QTL scan on chr 2, controlling for the major locus on chr 3.

```
> out.ap.c2 <- addpair(ovar, qtl=qtlc, chr=2, covar=cross,
+                       verbose=FALSE)
```

In summarizing the output of the two-dimensional, two-QTL scan on chr 2, we will include the permutation results to get approximate p -values. The permutation results do not formally apply to the present situation, as they concern the global null hypothesis (of no QTL), while the scan was conditional on the major locus on chr 3. However, they provide useful landmarks for evaluating the evidence.

```
> summary(out.ap.c2, perms=opermc, pval=TRUE)

      pos1f pos2f lod.full pval lod.fv1  pval lod.int pval
c2:c2    80   114    9.13    0    1.24 0.467  0.227    1
      pos1a pos2a lod.add pval lod.av1  pval
c2:c2      4   114    8.9    0    1.01 0.064
```

We see some evidence for a second QTL on chr 2, with p -value = 6%; surprisingly, the two inferred QTL are on opposite ends of the chromosome, rather than at the two distal peaks seen in Fig. 10.14. There is no evidence for an interaction between the putative QTL on chr 2.

We plot the LOD scores to get a sense of the precision of localization of the two QTL. We will look at the LOD scores comparing two-locus models to the best single-locus model.

```
> plot(out.ap.c2, lower="cond-int", upper="cond-add")
```

The upper left triangle in Fig. 10.16, with LOD scores comparing models with two additive QTL to the best single-QTL model, are most interesting. While the location of the distal QTL (at around 114 cM) is quite well defined, the location of the proximal QTL is not at all well defined. It is inferred to be at the proximal tip of the chromosome, but large LOD scores are also seen at around 80–90 cM.

Let us bring the four QTL together into a single model and run `refineqtl` to refine the QTL positions.

```
> qtl <- makeqtl(ovar, c(2, 2, 3, 3), c(4, 114, 62.8, 74.8))
> rqtl <- refineqtl(ovar, qtl=qtl, covar=cross, verbose=FALSE,
+                  formula=y~cross+Q1+Q2+Q3+Q4+Q3:Q4)
```

The proximal QTL on chr 3 moved slightly.

```
> rqtl

      name chr  pos n.gen
Q1   2@4.0   2   4.0     2
Q2  2@114.0   2 114.0     2
```

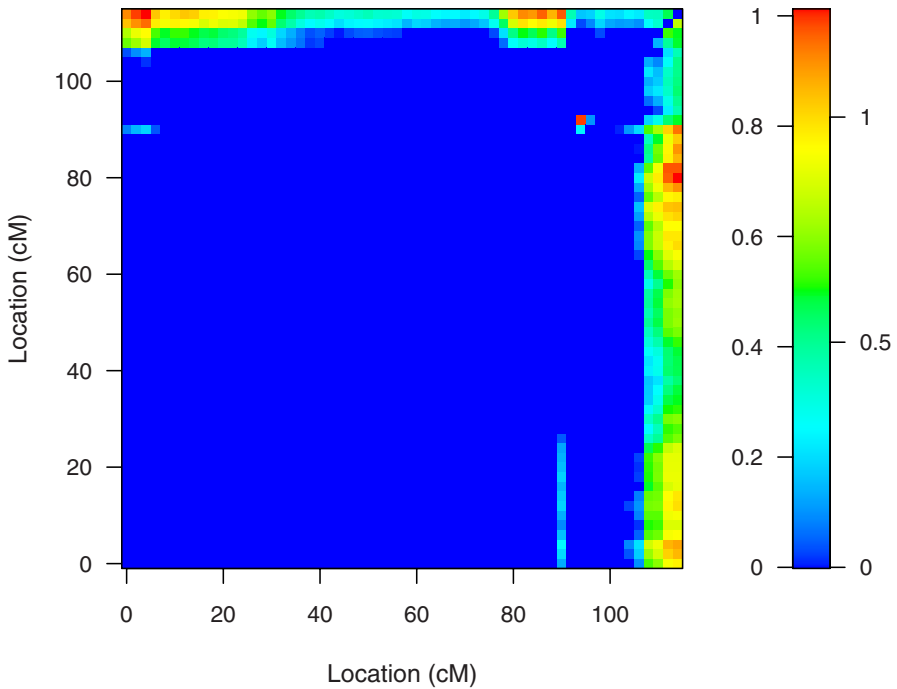


Figure 10.16. LOD scores from a two-dimensional, two-QTL scan on chr 2, controlling for a locus on chr 3, for the **ovar** data. LOD_{av1} is displayed in the upper left triangle; LOD_{fv1} is displayed in the lower right triangle. In the color scale on the right, numbers to the left and right correspond to LOD_{av1} and LOD_{fv1} , respectively.

```
Q3  3@63.6    3  63.6    2
Q4  3@74.8    3  74.8    2
```

We use `fitqtl` to perform a drop-one-QTL analysis with this four-QTL model. Note the use of `pvalues=FALSE` to omit the two columns of p -values.

```
> summary(fitqtl(ovar, qtl=rqtl, covar=cross,
+             formula=y~cross+Q1+Q2+Q3+Q4+Q3:Q4),
+             pvalues=FALSE)
```

Full model result

Model formula: y ~ cross + Q1 + Q2 + Q3 + Q4 + Q3:Q4

	df	SS	MS	LOD	%var
Model	6	450.8	75.139	76.63	22.02
Error	1412	1596.7	1.131		
Total	1418	2047.5			

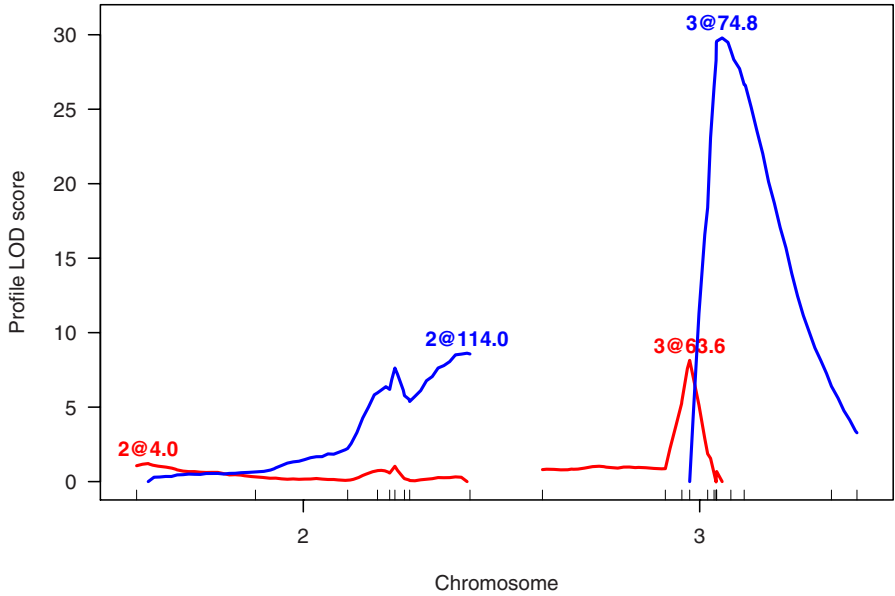


Figure 10.17. Profile LOD curves for a four-QTL model, for the full *ovar* data; the two QTL on chr 3 were allowed to interact.

Drop one QTL at a time ANOVA table:

	df	Type III SS	LOD	%var	F value
cross	1	193.8113	35.3001	9.4656	171.391
2@4.0	1	6.3004	1.2135	0.3077	5.572
2@114.0	1	45.3005	8.6203	2.2124	40.060
3@63.6	2	42.7445	8.1403	2.0876	18.900
3@74.8	2	162.0437	29.7841	7.9141	71.649
3@63.6:3@74.8	1	27.0815	5.1823	1.3226	23.949

The evidence for the proximal QTL on chr 2 looks weak, but there is strong evidence for the others.

We may plot LOD profiles, to get another view of the localization of the QTL.

```
> plotLodProfile(rqtl, col=c("red","blue","red","blue"),
+               ylab="Profile LOD score")
```

In Fig. 10.17, we again see that the location of the proximal QTL on chr 2 is poorly defined. Perhaps we are being overly generous in calling this a QTL.

We might consider adding additional terms to our QTL model. First, let us use `addint` to explore the possibility of additional interactions (see Sec. 9.3.3). We use `pvalues=FALSE` to omit the two columns of *p*-values.

```
> addint(ovar, qtl=rqtl, covar=cross,
+        formula=y~cross+Q1+Q2+Q3+Q4+Q3:Q4,
+        pvalues=FALSE)
```

Model formula: $y \sim \text{cross} + Q1 + Q2 + Q3 + Q4 + Q3:Q4$

Add one pairwise interaction at a time table:

```
-----
              df Type III SS      LOD      %var F value
cross:2@4.0    1      9.18153 1.77696 0.44842   8.161
cross:2@114.0  1      0.54432 0.10506 0.02658   0.481
cross:3@63.6   1      0.49906 0.09632 0.02437   0.441
cross:3@74.8   1      0.26691 0.05151 0.01304   0.236
2@4.0:2@114.0  1      0.31004 0.05984 0.01514   0.274
2@4.0:3@63.6   1      2.91815 0.56366 0.14252  2.583
2@4.0:3@74.8   1      0.34807 0.06718 0.01700   0.308
2@114.0:3@63.6 1      1.86901 0.36089 0.09128   1.654
2@114.0:3@74.8 1      3.31378 0.64016 0.16184  2.934
```

Note that QTL $\times$ covariate interactions are considered as well as QTL $\times$ QTL interactions. There appears to be some evidence for a difference in the effect of the proximal chr 2 locus in the two crosses; otherwise, we see no evidence for additional interactions.

Let us also use `addqtl` once more, to scan for an additional QTL.

```
> onemore <- addqtl(ovar, qtl=rqtl, covar=cross,
+                  formula=y~cross+Q1+Q2+Q3+Q4+Q3:Q4)
```

The summary of the results indicates the possibility of yet another QTL on chr 2; the conditional LOD score is the same as what we have for the proximal QTL on chr 2.

```
> summary(onemore)
```

```
      chr  pos  lod
c1.loc30   1 34.5 0.423
acc004516   2 89.1 1.164
slo        3 121.3 0.598
```

Let us add this additional QTL to the model, reorder the QTL according to their positions in the genome, and run `refineqtl` to refine the QTL positions.

```
> qtl2 <- addtoqtl(ovar, rqtl, 2, 89.1)
> qtl2 <- reorderqtl(qtl2)
> rqtl2 <- refineqtl(ovar, qtl=qtl2, covar=cross, verbose=FALSE,
+                   formula=y~cross+Q1+Q2+Q3+Q4+Q5+Q4:Q5)
```

The distal QTL on chr 2 shifted by 1 cM.

```
> rqtl2
```

	name	chr	pos	n.gen
Q1	2@4.0	2	4.0	2
Q2	2@89.1	2	89.1	2
Q3	2@115.0	2	115.0	2
Q4	3@63.6	3	63.6	2
Q5	3@74.8	3	74.8	2

We perform the drop-one-QTL analysis again, to assess the evidence for each QTL in the context of this five-QTL model.

```
> summary(fitqtl(ovar, qtl=rqtl2, covar=cross,
+               formula=y~cross+Q1+Q2+Q3+Q4+Q5+Q4:Q5),
+         pvalues=FALSE)
```

Full model result

Model formula: y ~ cross + Q1 + Q2 + Q3 + Q4 + Q5 + Q4:Q5

	df	SS	MS	LOD	%var
Model	7	457.2	65.314	77.86	22.33
Error	1411	1590.3	1.127		
Total	1418	2047.5			

Drop one QTL at a time ANOVA table:

	df	Type III SS	LOD	%var	F value
cross	1	194.6205	35.5733	9.5051	172.673
2@4.0	1	6.8879	1.3317	0.3364	6.111
2@89.1	1	6.6325	1.2824	0.3239	5.885
2@115.0	1	11.5318	2.2262	0.5632	10.231
3@63.6	2	41.8306	8.0000	2.0430	18.557
3@74.8	2	160.6383	29.6505	7.8454	71.261
3@63.6:3@74.8	1	26.5197	5.0959	1.2952	23.529

The evidence for each of the QTL on chr 2 is weak but non-negligible. With the QTL on chr 2 at 89 cM in the model, there is a big drop in the residual evidence for the most distal chr 2 locus.

Finally, let us apply our automated stepwise model selection procedures with `stepwiseqtl`.

```
> stepout1 <- stepwiseqtl(ovar, covar=cross, pen=pen,
+                         max.qtl=8, verbose=FALSE)
```

```
> stepout1
```

	name	chr	pos	n.gen
Q1	2@89.1	2	89.1	2
Q2	2@92.4	2	92.4	2

Q3	2@94.2	2	94.2	2
Q4	2@110.0	2	110.0	2
Q5	3@63.6	3	63.6	2
Q6	3@76.8	3	76.8	2

```
Formula: y ~ cross + Q1 + Q2 + Q3 + Q4 + Q5 + Q6 + Q5:Q6 +
          Q2:Q3
```

```
pL0D: 43.96
```

The results are much like what we obtained above, though without the most proximal chr 2 locus, and with the addition of that untrustworthy pair of tightly linked epistatic loci on chr 2 seen in the analysis of the initial cross (see page 299).

Let us run `stepwiseqtl` again, this time using exclusively heavy interaction penalties.

```
> stepout2 <- stepwiseqtl(ovar, covar=cross, pen=pen[1:2],
+                          max.qtl=8, verbose=FALSE)
```

The results, with the exclusive use of heavy penalties, are identical to what we obtained above, with the mixture of heavy and light penalties.

```
> stepout2
```

	name	chr	pos	n.gen
Q1	2@89.1	2	89.1	2
Q2	2@92.4	2	92.4	2
Q3	2@94.2	2	94.2	2
Q4	2@110.0	2	110.0	2
Q5	3@63.6	3	63.6	2
Q6	3@76.8	3	76.8	2

```
Formula: y ~ cross + Q1 + Q2 + Q3 + Q4 + Q5 + Q6 + Q5:Q6 +
          Q2:Q3
```

```
pL0D: 41.61
```

10.4 Discussion

Our principal aim in this case study was to illustrate many of the techniques for exploring multiple-QTL models. We have focused on the analysis techniques rather than the biological interpretation of the results; for the latter, see the original article describing these data (Orgogozo *et al.*, 2006).

Our primary tools include single- and two-dimensional scans for QTL. We repeat these, controlling for major loci, to identify additional QTL. A variety of paths may be taken. Hopefully, they all lead to similar conclusions.

The assessment of evidence for individual loci and epistatic interactions, in the context of a multiple-QTL model, can be difficult. We use the results of permutation tests as our guide, even though they formally apply only in a restricted context (a single- or two-QTL model). The independent segregation of chromosomes is extremely useful, in this regard, as it ensures that the null distribution of a maximized LOD score is approximately the same, whether or not one has controlled for the presence of additional QTL on other chromosomes.

There is extensive missing genotype information in the data of Orgogozo *et al.* (2006), due to selective genotyping in the initial cross and selective phenotyping (and incomplete genotyping) in the second cross. This is a nuisance when it comes to the QTL analysis; with the multiple imputation approach, a large number of imputations are required and so computation times can be great. The limited genotype information can make it difficult to separate multiple linked loci, and it exacerbates the common problem that is seen with tightly linked loci. One should be cautious about the fit of large QTL models in the presence of appreciable missing genotype information.

The selective phenotyping strategy used in the second cross can be extremely valuable for the fine-mapping of a QTL, but we were a bit disconcerted to see that our analysis of the initial cross placed the QTL outside of the interval that was used to select recombinants. With the combined data, we did infer the presence of a QTL within this selected region, and the estimated location of the major QTL moved closer to the region, but we must admit lingering concern that the extensively missing genotype information results in some bias in the estimated locations of the QTL.

QTL analysis is a complex model-building exercise. Our fully automated system for model selection may be useful to many scientists, but the exploratory tools are important supplements, and in either case, the final conclusions can be subject to considerable uncertainty.