Construction of genetic maps using Map Maker

WEB address of Mapmaker
http://www-genome.wi.mit.edu/genome_software/

WEB address of the MapMaker manual
http://www-genome.wi.mit.edu/genome_software/other/mapmaker.html

Before start…

Log in
Username: WMmaps
Password: *****

Create a directory called MAP in the C: drive (**C:\MAP**)
Use Windows Explores to copy from the Z: drive the following directories into C:\MAP

- Data (for your Mapmaker and Mapmaker QTL exercises)
- Documentation for students (lecture, notes, papers, etc)
- QTL (for your Cartographer exercises)

1. PREPARING DATA FOR MAPMAKER VERSION 3.0

We will analyze together different exercise matrices and the matrix representing the map of chromosome 1AL constructed from recombinant substitution lines of the chromosome 1A of bread wheat and chromosome $1A^m$ of *Triticum monococcum.* The name of this file is PH.txt. It includes the HMW-glutenin locus we mapped by hand in the previous class at the end (not in its correct position). All other markers are in their best order.

As **exercise** you will construct the map of chromosome $7A^m$ of *Triticum monococcum* based on an F_2 population. The name of this file is MAP7A.txt it includes data from *Xpsr129* and *XLMG-glutenin,* at the end in an incorrect position.

MAPMAKER Version 3.0 can analyze data derived from progeny of several types of crosses, including mapping populations derived from a single meiosis like F_2 backcross, double haploids and RSLs and F2 intercross. Data files are flat ASCII text files. MAPMAKER is generally insensitive to extra spaces, uppercase-lowercase distinctions, and (after the top two lines) blank lines.

The very first line of your raw data file should read like:

data type xxxx

where xxxx is one of the allowed data types, in our examples either:

f2 intercross (for file MAP7A.txt)
f2 backcross (for file PH.txt)
ri self (for file ri_seld corresponding to exercise 4 last lab)

The second line of the raw file should contain a list of three numbers, separated by spaces, such as:

46 36 0

The first of these values indicates the number of progeny for which data are included in the file (in this case, 46). The second indicates the number of genetic loci for which data are supplied (36). The third indicates the number of quantitative traits in the data set (0 in all our examples).

By default, the codes used for F_2 backcross (BC1, RSLs and double haploids) data are:

'A'	Homozygote for the recurrent parent genotype.
'H'	Heterozygote (or other genotype in RSLs or double haploids).
'-'	Missing data for the individual at this locus.

For F_2 intercross data, the default codes are indicated in Table 1. Also by default, MAPMAKER will match genotype characters in a case-insensitive manner (that is 'a' and 'A' indicate the same genotypes).

Table 1. Symbols for genotypic classes for molecular data.

Genotypic class	Genotype	Symbol
Parent A homozygote	AA	A
Parent B homozygote	BB	B
Heterozygote	AB	H
Dominant marker, parent A	AA or AB	D
Dominant marker, parent B	BB or AB	C
Missing data		- (hyphen)

After the first two lines, the raw file should then present the genetic locus data, in a simple format. For each locus, you list (1) the name of the locus, preceded by an asterisk ("*"); (2) one or more spaces (or tabs etc.); and (3) the genotypic data for all individuals, in order. For example:

```
*locus1      BA-HHHAAABBB-HHAA
```

would provide data for a locus named "locus1" with individual #1 having the B genotype, individual #2 having the A genotype, and so forth. Data for each new locus should begin on a new line (with blank lines allowed); although the genetic data for any one locus may be "broken" by any number of spaces, tabs, and line breaks. This means that, among other things, tab-delimited-text files (such as those often exported by spreadsheet programs) will work well, for example:

```
*L2   B   A   -   H   H   H   A   A   A   B   B   B   -   H
```

There is a system-dependent maximum line length, although it is fairly large (at least 1,000 characters, where a tab counts as one character.

Locus names should be kept to at most 8 characters, and must be limited to alphabetic and numeric characters, along with the underscore character ('_') and periods ('.'). No other characters are allowed (although any dashes in locus names ('-") will be converted to underscores). ***Locus names must start with alphabetic character*** (so that they are not confused with locus numbers in MAPMAKER sequences). Finally, note that **comments** may be inserted on any line starting with a number sign character (**"#"**).

An example of a complete raw file follows:

```
data type f2 intercross
20 5 2
# Joe's tiny data set, 10/21 version.

*locus 1  BBBHH-AAABBBHHH-AABA
*locus 2  AB-ABHABHAB-ABHABHBH
*locus 3  ABBAHHHBHABHABHBBHH-
#Locus 3  may be mis-scored in individual 12!
*locus4   ABHABAAAHAB-ABHABHHB
*locus 5  ABHABHAA-ABHABHAHHHB
```

Moving into the correct directory:

1) Start DOS by clicking START -> run -> type cmd -> enter
2) Type CD C:\Mapmaker (**C**hange **D**irectory)
3) Type **SYSTEM**: to go to DOS operating system
4) Type C**:** to move to the right disk
5) To change directory use the CD command. Type **cd MAP\data** to go to the data subdirectory in your MAP directory where the working files re located.
6) Type **EXIT** to return to MAPMAKER.

Comparison between RF calculated by hand and MAPMAKER

Mapping populations derived from one meiotic event

Once your data are in the raw file format, to process them into a form usable by MAPMAKER Version 3.0 you must use the "**PREPARE DATA**" command. The PREPARE DATA command loads the information from your raw data files into MAPMAKER. Your raw file remains unaltered and should be saved as a backup copy of your data. To prepare a raw file, simply start up MAPMAKER, and type the command:

1) **PREPARE DATA BC.txt** (this corresponds to the calculation of RF we did in a backcross on page 10 first notes.
2) **SEQUENCE 1 2**
3) **UNITS RF**
4) **MAP**

Compare the result with our hand calculation: 12 recombinants in 25 individuals= 12/25= 0.48.

- **F$_2$ mapping populations**

1) **PREPARE DATA F2.txt** (this correspond to the hand calculation of RF we did for the F$_2$ data on page 57.
2) **SEQUENCE 1 2**
3) **UNITS RF**
4) **MAP**

Compare the result with our hand calculation: 3 recombinants in 24 plants (48 chromosomes) = 3/48= 0.0625.

• Finding Linkage Groups by Two-Point Linkage

The first step in the analysis is a classical “two-point”, or pairwise, linkage analysis of the data set for *identifying* linkage groups of markers in preliminary analysis. We will use the file MIXED.txt that has two mixed linkage groups (identified by A and B names).

1) **PREPARE DATA MIXED.txt**
2) **SEQUENCE all**
3) **GROUP**
4) **LIST LOCI**

The **SEQUENCE ALL** command tells MAPMAKER to include all loci. **LIST LOCI** will list the name of the different loci.

Note that for two-point analysis, the order in which the loci are listed is unimportant. We then type MAPMAKER’s “**GROUP**” command, instructing the program to divide the markers in the sequence into linkage groups. To determine whether any two markers are linked, MAPMAKER calculates the maximum-likelihood distance and corresponding LOD score between the two markers: If the LOD score is greater than some threshold, and if the distance is less than some other threshold, then the markers will be considered *linked*. ***By default, the LOD threshold is 3.0, and the RF threshold is 0.50***.

For the purpose of finding linkage groups, MAPMAKER considers linkage *transitive*. That is, if marker A is linked to marker B, and if B is linked to C, then A, B, and C will be included in the same linkage group. Finally, MAPMAKER divides the data set into linkage groups, whit names “group1",“group2", etc. In the file Mixed.txt loci belong to two different linkage groups and there are no unlinked markers.

• Constructing maps

Once you have a file with a single linkage group, more powerful techniques can be used to order markers within a group. The next step is the construction of a map for the RSL population (= backcross)

```
5)  PREPARE DATA PH.txt
6)  PHOTO PH.out
```

The **PHOTO** command will create an output file (PH.out) with all the results from the analysis. If you specify a directory for the file name, the prepared and output files will be placed in that directory. You can the open an edit it with any word processor.

```
7)  SEQUENCE all
```

To tell MAPMAKER which loci we wish to consider in our two-point analysis we use the **SEQUENCE** command. It can be also used to indicate MAPMAKER to consider only certain loci sequence locus1 locus2 locus3...

If the loci names are between set-braces, {1 2 3 4 …}, this indicates MAPMAKER that the order of the markers contained within them is unknown. Since almost all of MAPMAKER's analysis functions use the "current sequence" to indicate which loci they should consider, you will find that ***the "sequence" command must be entered before performing almost any analysis function***. The sequence of loci in use remains unchanged until you again type the "sequence" command to change it.

Exploring Map Orders by Hand

To determine the most likely order of markers within a linkage group, we could imagine using the following simple procedure: For each possible order of the group, we calculate the maximum-likelihood map (e.g. the distances between all markers given the data), and the corresponding map's likelihood. We then compare these likelihoods and choose the most likely order as the answer. This type of exhaustive analysis may be performed using MAPMAKER's "**COMPARE**" command.

In practice however, this sort of "exhaustive" analysis is not practical for even medium sized groups: a group of N markers has N!/2 possible orders, a number which becomes unwieldy (for most computers) when N gets to be between 6 and 10. (In practice, one need to order subsets of the linkage group and then overlap those subsets, mapping any remaining markers relative to those already mapped, a process we will illustrate later.)

We will illustrate the use of the fully exhaustive analysis using 5 markers of the PH.TXT file. To do this, we first change MAPMAKER's sequence to {1 3 5 7 8}. Here, the set-braces indicate that the order of the markers contained within them is unknown, and thus that all possible orders need to be considered.

```
8)  SEQUENCE {1  3 5 7 8}
9)  COMPARE
```

The "COMPARE" command, instructs MAPMAKER to compute the maximum likelihood map for each specified order of markers, and to report the orders sorted by the likelihoods of their maps. Note that while MAPMAKER examines all possible orders, only the 20 most likely ones are reported (by default).

For each of these 20 orders, MAPMAKER displays the *log-likelihood* of that order relative to the best likelihood found. Thus the best order:

1 3 5 7 8   and 1 3 5 8 7

is indicated as having a *relatively log-likelihood* of 0.0 ad agrees with the order we have previously established by visual analysis of the crossing over points. In this case the second best order is not significantly less likely than the best. This is because the order between 7 and 8 is uncertain (based on a single double corssover).

If the relative log-likelihood of the second order is, for example, -3.0, this indicates that the best order of this group is supported by an odds ration of roughly 1000:1 (10 to the 3rd power to one), over any other order.

• Displaying a Genetic Map

The "COMPARE" command however only reports the relative log-likelihoods, and afterwards forgets the map distances. To actually display the genetic distances we must instead use the **"MAP"** command. Like "compare", the "map" command instructs MAPMAKER to calculate the maximum likelihood map of each order specified by the current sequence. If the current sequence specifies more than one order (for example, the sequence "{1 2 3 5 7}" specifies 60 orders) then the maps for all specified orders will be calculated and displayed.

Because we found one order of this group to be much more likely than any other, we probably only care to see the map distances for this single order. First, we set MAPMAKER's sequence, putting the markers in their best order and doing away with the set brackets. Then type MAP.

```
10) SEQUENCE 1 3 5 7 8
11) MAP
```

Distances between neighboring markers are displayed. Note however, that these distances may be considerably different than the "two-point" distances between those markers. This is because MAPMAKER's so-called *multipoint analysis* facility can take into account much more information, such as flanking marker genotypes and some amount of missing data. This is precisely the reason that we use multipoint analysis rather than two-point analysis to order markers. More data is taken into account and there is a smaller chance of making a mistake.

Mapping a larger group

Exhaustive analyses of large linkage groups are not practical. Instead, to find a map order of a larger group, we need to find a subset of markers on which we can perform an exhaustive "compare" analysis. Generally, this is true for sets of markers which have (i) as little missing data as possible, and (ii) do not have many closely spaced markers. We will start using the previous subset: 1 3 5 7 8. A starting group could have been automatically selected using MAPMAKER's **"SUGGEST SUBSET"** command, documented in the reference section.

To determine the map position of the remaining markers in this map, we will use the following procedure: Starting with the known order of 5 markers, we will place the others, one at a time using the **"TRY"** command. In its output, MAPMAKER displays relative log-likelihood of each position for the inserted markers.

```
12) TRY 2  9
```

In this case, we see that marker 2 strongly prefers to be in-between markers 1 and 3 and marker 9 after marker 8. The "try" command not only tries to place markers in each interval in the framework, but also tries to place each marker infinitely far away (that is, forced 50% recombination between it and the framework). The relative log-likelihoods for this position are indicated following the "INF" entry in the MAPMAKER output. The relative log-likelihoods indicate the odds supporting linkage between one locus and a framework of loci when the locus is placed in its most likely position.

Determine the position of other markers using the TRY command.

Automatically finding map orders

As an alternative to the manual mapping commands presented earlier (such as "try" and "compare"), MAPMAKER has more automated functions. First, we will use the "SEQUENCE all" command to select all the loci on the chromosome. Next, we will use the **"THREE POINT"** command to pre-compute the likelihoods of all three-point crosses for this chromosome. We do *not* have to do this step to proceed, although three-point analysis provides a powerful way to speed up the steps we perform below. Three-point analysis simply excludes the majority of the very unlikely orders from consideration, allowing MAPMAKER to spend time carefully examining only those orders reasonably consistent with the observed data. When you type the "three-point" command, MAPMAKER first finds every linked triple of markers in the current sequence. For each triple, MAPMAKER computes the most likely map distances and likelihoods for all 3 possible orders. For each order, MAPMAKER displays the 'relative log-likelihood' of that order as compared to the most likely (or best) order of the triple. As before, the most likely order of the three has a relative log-likelihood of 0.0, while the others have negative relative log-likelihoods.

```
13) SEQUENCE all
14) THREE POINT
15)  ORDER 3.0  50  5  2.0
```

MAPMAKER will make use of these data as follows: any three-point order will be considered *excluded* if its relative log-likelihood is worse than the best by some threshold (by default, the threshold is 4.0). Any multiple locus order which contains one or more excluded three-point sub-orders will itself be considered excluded, and only non-excluded multipoint orders will be evaluated by full multipoint analysis. If the “THREE POINT” step is not executed before **“ORDER”** MAPMAKER uses full-multipoint analysis to evaluate all possible orders. This definitely would be slower, but presumably would produce identical answers. The
MAPMAKER's "ORDER" command will find a linear order of the markers on chromosome 1. The arguments are:

ORDER <minimum LOD> <maximum distance> <start size> <threshold> <number of tries>

The default threshold is 3.0 but a lower one is used here because of the low number of individuals included in the study. Briefly, this command performs the following analyses:

(1) it tries to find a small subset of loci (by default, 5 loci), for which a single order is found to be much more likely than any other using a "compare" style analysis;
(2) remaining markers which can be mapped to a unique position are added to this order one at a time;
(3) any markers which cannot be mapped to a *unique* position in the order are mapped into multiple intervals.

Verifying a Map Order

MAPMAKER uses a semi-random starting point and addition order. The "order" command can be run repeatedly to verify the consistency of the results. MAPMAKER's error detection algorithms can be also used to limit the possible ill-effects of small data errors. Moreover, MAPMAKER provides a variety of simple ways of testing the results found by the "order" command.

One powerful command for accomplishing this test is called "**RIPPLE**". Essentially, given a known (or assumed) map order, "ripple" instructs MAPMAKER to permute the order of neighboring markers, and to compare the likelihoods of the resulting maps. Any order, which has the log-likelihood within some threshold amount of the assumed order's likelihood, will be displayed as a viable alternative. Like the "ORDER" command, "RIPPLE” knows how to use three-point analysis to speed its search, although in the end it uses the power of multipoint analysis with *all* flanking markers to finally compare likelihoods of the consistent orders.

First use MAPMAKER's "SEQUENCE" command to select the final order. Next, type the "RIPPLE" command. By default, this command will permute 5 neighboring loci at a time and flag all alternative orders within a log-likelihood of 2.0 (that is, within 100:1 or better odds) of that of our known order.

```
16) RIPPLE
```

Automatic Error Detection

A method for dealing with the possibility of genotyping error in data sets is incorporated into MAPMAKER (Genomics 14: 604-610). It calculates *a posteriori* (e.g. in light of all available raw data) the probability that each individual genotype is right or wrong. These numbers are presented as a "LOD of error", and represent on a log-scale the strength of the evidence that a marker is mistyped. For typical data sets, double-checking all genotypes with a LOD-error of about 1.0 or greater (usually a small fraction of the data set) will correct the vast majority of the errors. Note that MAPMAKER does not calculate LOD-error values for markers at the end of an order (simply because, without flanking markers, there is minimal power to tell recombination from mistyping).

Turn the "ERROR DETECTION" option "ON", and then re-display the map shown on the previous pages.

```
17) ERROR DETECTION on
18) MAP
```

Exercises

1. Construct the best map order for the MAP7A.txt file using MAPMAKER

2. Use the Error detection on option to find the three data points in the matrix that are more likely to contain errors and that need to be checked (indicate the name of the loci and the number of individual for each case).

3. An error was introduced in the **error.txt** file. Find it using Mapmaker

4. Construct maps for the two chromosomes included in file '**Final.txt**'