

# R/QTL Part 2

## QC, Non-F<sub>2</sub> Populations

Tom Kono  
2017-03-14

<http://z.umn.edu/d0crqtl2>

# First!

# Plant Breeding Symposium

- <http://plantsciencesymposium.umn.edu/>
- 24 March, 2017, with a workshop on 23 March.
- Karl Broman, Briana Gross, Allison Miller, John Burke, John Doebley, among others...

# Recall: QTL Mapping

- Identify genomic regions underlying variation in quantitative traits
- Usually involves structured populations with known genotype and phenotype
- R/QTL package by Karl Broman is one of the best, both in methodology and documentation

# Get the Materials

```
$ git clone https://github.com/MorrellLAB/DoesNaughtCompute.git  
$ cd DoesNaughtCompute/R_QTL_Part2/
```

In R:

```
> install.packages("qtl")
```

If you'd like to re-generate the sample dataset:

```
> install.packages("devtools")  
> library(devtools)  
> install_github("kbroman/simcross")
```

# Sample Dataset

- $F_2$  and  $F_3$  families, with no phenotypic data
- Genotyped with some markers on two chromosomes
- Some variation in marker density along chromosome
- Some markers have high missing data
- Some genotyping errors

# F<sub>2</sub>: Quick Review

- Use the `read.cross()` function to load the data

```
> pop <- read.cross(  
  format="csv",  
  file="Data/Simulated_Genotypes_F2.csv")
```

- Summarize the population

```
> summary(pop)
```

- Summarize the markers

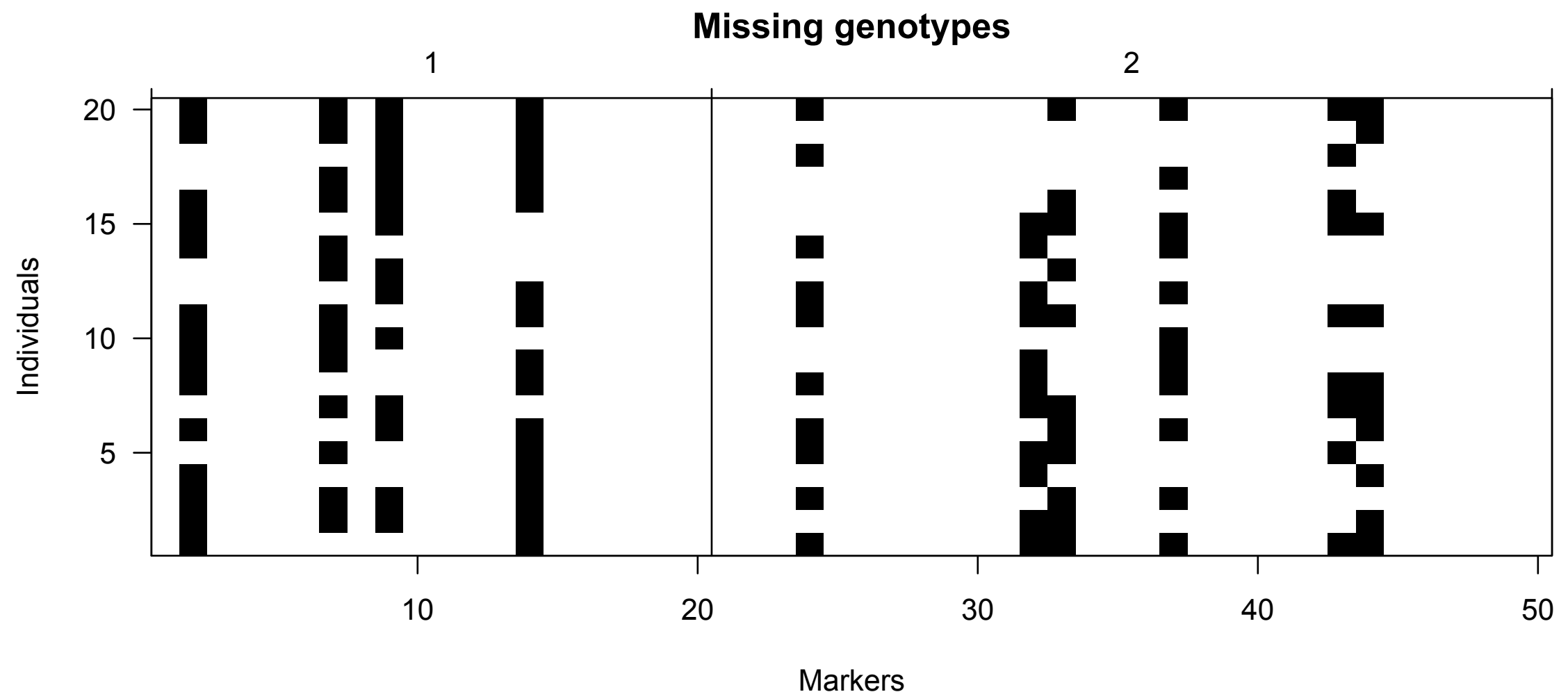
```
> summary.map(pop)
```

# F<sub>2</sub>: Quick Review

- How many individuals are in the F<sub>2</sub> population?
- How many markers are genotyped? How many on each chromosome?

# F<sub>2</sub>: Missing Data

- First QC step: visualize missing genotypes  
`> plotMissing (pop)`



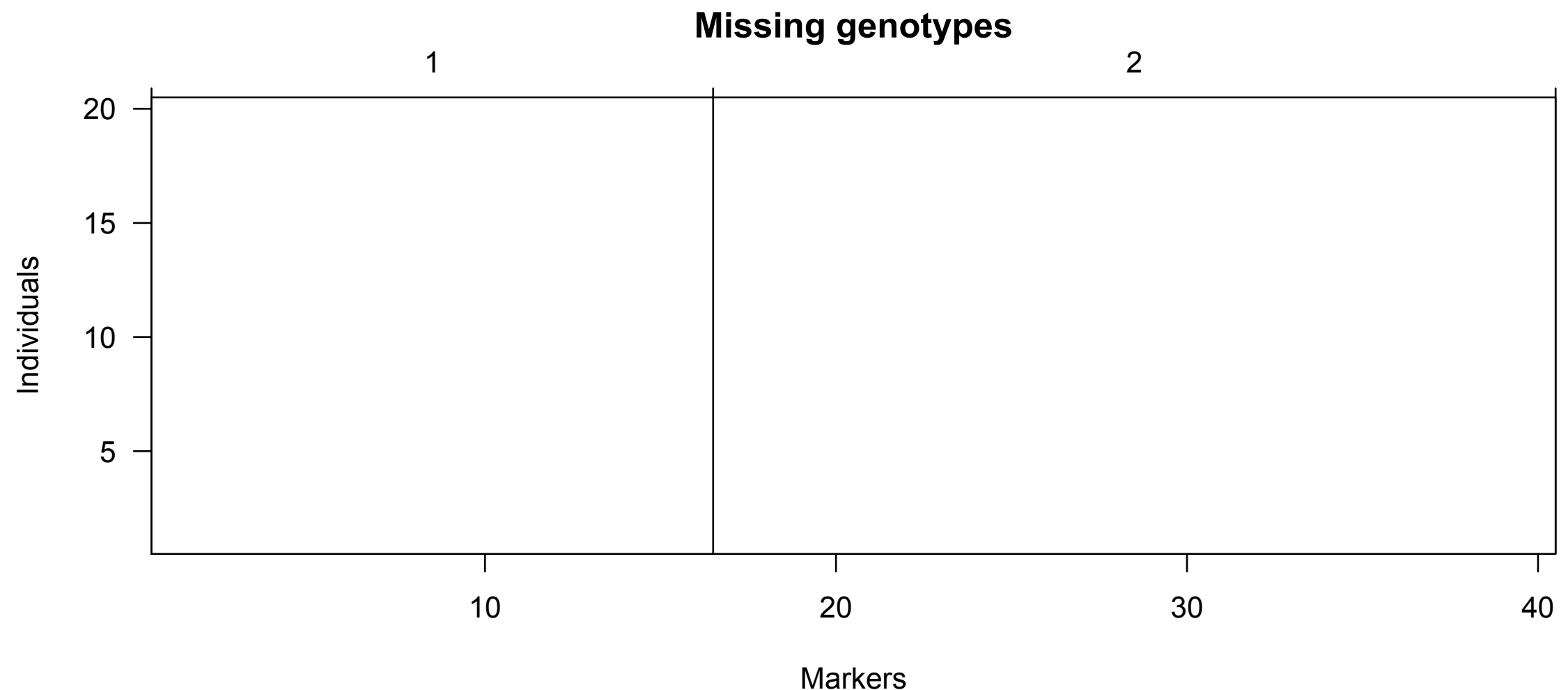


# F<sub>2</sub>: Missing Data

- Let's remove markers with lots of missing data:

```
> todrop <- markernames(pop)[nmissing(pop, what="mar") > 5]  
> pop <- drop.markers(pop, todrop)  
> plotMissing(pop)
```

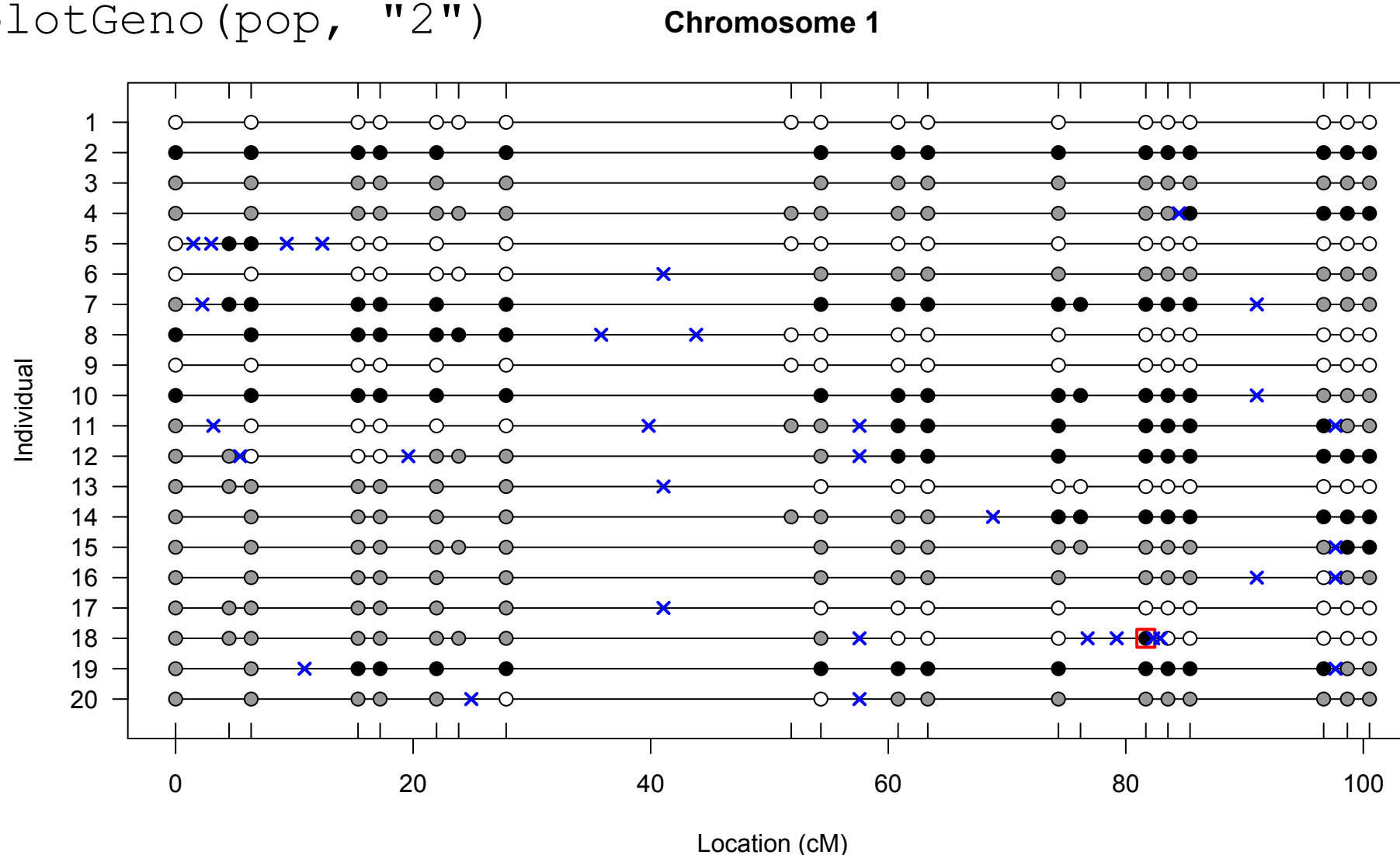
- How many markers remain?



# F<sub>2</sub>: Genotyping Errors

- Search for genotyping errors using the method of Lincoln and Lander (1992)

```
> pop <- calc.errorlod(pop, error.prob=0.01)
> plotGeno(pop, "1")
> plotGeno(pop, "2")
```



# F<sub>2</sub>: Genotyping Errors

- Unfortunately, R/QTL does not have a utility to fix genotyping errors
- Must go back to raw data
- If not, maybe set to missing and impute?

# BC<sub>s</sub>F<sub>t</sub>: Read the Data

- R/QTL has limited support for more advanced populations: backcross and selfing
- The `read.cross()` function takes additional args:  

```
> pop <- read.cross(  
  format="csv",  
  file="Data/Simulated_Genotypes.csv",  
  BC.gen=0,  
  F.gen=3)
```
- Plot the summaries, like in the F<sub>2</sub> case. Different?

# BC<sub>s</sub>F<sub>t</sub>: Features

- Adjusts map distances and genotype probabilities for repeated backcrossing, selfing, or combinations of the two
- Similarly, these calculations trickle down to error detection and QTL scans
- Always read the docs!  
<http://www.rqtl.org/tutorials/bcsft.pdf>

# More Complicated Designs?

- **Nested Association Mapping populations: 'NAM' package. See next week's session!**

<https://cran.r-project.org/web/packages/NAM/NAM.pdf>

- Multiparent Advanced Intercross populations:  
'mpMap' package

<https://cran.r-project.org/web/packages/mpMap/mpMap.pdf>

Or, 'mpMap2'

<https://github.com/rohan-shah/mpMap2>

# A Note About the Data

- All the data used here was made up!
- Simulated with 'simcross' - also from Karl Broman  
<http://kbroman.org/simcross/>
- See 'simulate\_genotypes.R' for script used to generate the data