Maintaining, supporting, and sustaining scientific software

Biostatistics & Medical Informatics, UW–Madison
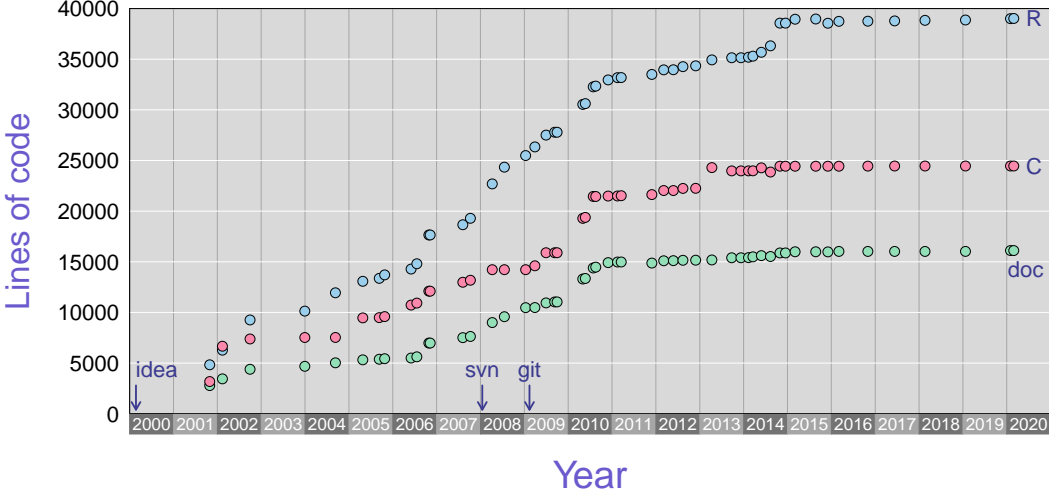
kbroman.org
github.com/kbroman
@kwbroman
Course web: kbroman.org/AdvData

20 years of R/qtl

Why?

Good things

Good things

- ► some of the code
- ► basics of the user interface
- ► diagnostics and data visualization
- ► quite comprehensive
- ► quite flexible

Bad things

Input file

	A	B	C	D	E	F	G	H	I
1	liver	spleen	sex	pgm	D1Mit18	D1Mit80	D1Mit17	D2Mit379	D2Mit75
2					1	1	1	2	2
3					27.3	51.4	110.4	38.3	48.1
4	61.92	153.16	m	1	BB	SB	SB	SB	SB
5	88.33	178.58	m	1	–	–	–	BB	BB
6	58	131.91	m	1	BB	SB	SB	SB	SB
7	78.06	126.13	m	1	SB	SB	BB	SS	SS
8	65.31	181.05	m	1	–	–	–	SB	SB
9	59.26	191.54	m	1	–	–	–	SS	SS
10	59.47	154.88	m	1	BB	BB	BB	SB	SB
11	65.63	184.12	m	1	–	–	–	SB	SB
12	38.64	133.05	m	1	SB	BB	SB	SB	SB
13	60.94	275.63	m	1	–	–	–	SB	BB
14	51.48	395.25	m	1	–	–	–	SB	BB
15	47.12	260.45	m	1	BB	SB	SB	BB	BB

Stupidest code ever

```
n <- ncol(data)
temp <- rep(FALSE,n)
for(i in 1:n) {
  temp[i] <- all(data[2,1:i]=="")
  if(!temp[i]) break
}
if(!any(temp)) stop("...")
n.phe <- max((1:n)[temp])
```

Open source means
everyone can see my stupid mistakes

Open source means
everyone can see my stupid mistakes

Version control means
everyone can see every stupid mistake I've ever made

More typically bad code

The `scantwo()` function is 1446 lines long.

The related C code is 20% of the C code in R/qtl.

Baroque data structures

```
attr(mycross$geno[["X"]]$probs, "map")
```

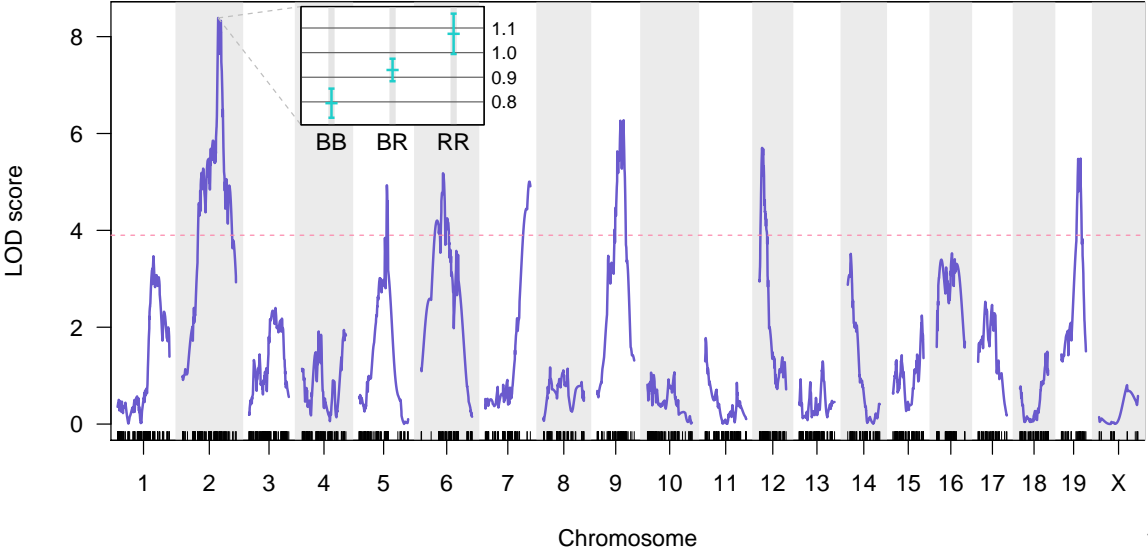# Documentation

User support

"I tried X and it didn't work."

"Could you look at the attached 25-page Word document containing code and output and tell me if I'm doing something wrong?"
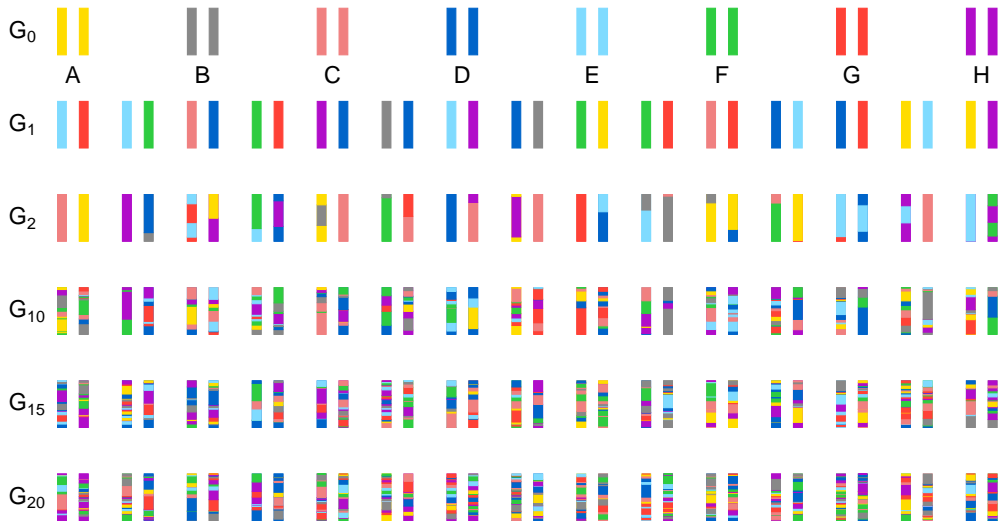
Incorporating others' code

Version control

Tests

QTL mapping

Heterogeneous stock

Challenge: scale of results

genotypes

phenotypes

Challenge: scale of results

genotypes   phenotypes

results

Challenge: organizing, automating

genotypes   phenotypes

Challenge: organizing, automating

genotypes

phenotypes

Challenge: organizing, automating

genotypes

phenotypes

Challenge: organizing, automating

genotypes

phenotypes

Challenge: organizing, automating

genotypes

phenotypes

Challenge: organizing, automating

genotypes

phenotypes

Challenge: organizing, automating

R/qtl2

- ► High-density genotypes

- ► High-dimensional phenotypes

- ► Multi-parent populations

- ► Linear mixed models

kbroman.org/qtl2

R/qtl2: Let's not make the same mistakes

- ▶ C++ and Rcpp
- ▶ Roxygen2 for documentation
- ▶ Unit tests
- ▶ A single "switch" for cross type

R/qtl2: Let's not make the same mistakes

- ▶ C++ and Rcpp
- ▶ Roxygen2 for documentation
- ▶ Unit tests
- ▶ A single "switch" for cross type
- ▶ Yet another data input format
- ▶ Flatter data structures, but still complex

Sustainable academic software

Acknowledgments

Danny Arends
Gary Churchill
Nick Furlotte
Dan Gatti
Ritsert Jansen
Pjotr Prins
Śaunak Sen
Petr Simecek
Artem Tarasov
Hao Wu
Brian Yandell

Robert Corty
Timothée Flutre
Lars Ronnegard
Rohan Shah
Laura Shannon
Quoc Tran
Aaron Wolen

NIH/NIGMS