Steps toward reproducible research

Karl Broman

Biostatistics & Medical Informatics
Univ. Wisconsin–Madison

kbroman.org
github.com/kbroman
@kwbroman
Slides: bit.ly/jax2017-05

Karl -- this is very interesting, however you used an old version of the data (n=143 rather than n=226).

I'm really sorry you did all that work on the incomplete dataset.

Bruce

The results in Table 1 don’t seem to correspond to those in Figure 2.

In what order do I run these scripts?

Where did we get this data file?

Why did I omit those samples?

How did I make that figure?

“Your script is now giving an error.”

“The attached is similar to the code we used.”

Reproducible

vs.

Replicable

Reproducible

vs.

Correct

Steps toward reproducible research

kbroman.org/steps2rr

1. Organize your data & code

File organization and naming
are powerful weapons against chaos.

– Jenny Bryan

1. Organize your data & code

Your closest collaborator is you six months ago,
but you don’t reply to emails.

(paraphrasing Mark Holder)

1. Organize your data & code

```
RawData/                    Notes/
DerivedData/                Refs/

Python/                     ReadMe.txt
R/                          ToDo.txt
Ruby/                       Makefile
```

Chaos

```
AimeeNullSims/     Deuterium/              Ping/
AimeeResults/      ExtractData4Gary/       Ping2/
AnnotationFiles/   FromAimee/              Ping3/
Brian/             GoldStandard/           Ping4/
Chr6_extrageno/    HumanGWAS/              Play/
Chr6_segdis/       Insulin/                Prdm9/
ChrisPlaisier/     Int2_for_Mark/          RBM_PlasmaUrine_2012-03-08/
Code4Aimee/        Islet_2011-05/          Slco1a6/
CompAnnot/         MappingProbes/          StudyLineupMethods/
CondScans/         MultiProbes/            kidney_chr6.R
D2O_2012-02-14/    NewMap/                 pck2_sucla2.R
D2O_cellcycle/     Notes/                  penalties.txt
D2Ocorr/           NullSims/               transeQTL4Lude/
Data4Aimee/        NullSims_2009-09-10/
Data4Tram/         PepIns_2012-02-09/
```

2. Everything with a script

If you do something once,
you'll do it 1000 times.

3. Automate the process (GNU Make)

```
R/analysis.html: R/analysis.Rmd Data/cleandata.csv
    cd R;R -e "rmarkdown::render('analysis.Rmd')"

Data/cleandata.csv: R/prepData.R RawData/rawdata.csv
    cd R;R CMD BATCH prepData.R

RawData/rawdata.csv: Python/xls2csv.py RawData/rawdata.xls
    Python/xls2csv.py RawData/rawdata.xls > RawData/rawdata.csv
```

3. Automate the process (GNU Make)

```
R/analysis.html: R/analysis.Rmd Data/cleandata.csv
    cd R;R -e "rmarkdown::render('analysis.Rmd')"

Data/cleandata.csv: R/prepData.R RawData/rawdata.csv
    cd R;R CMD BATCH prepData.R

RawData/rawdata.csv: Python/xls2csv.py RawData/rawdata.xls
    Python/xls2csv.py RawData/rawdata.xls > RawData/rawdata.csv
```

3. Automate the process (GNU Make)

```
R/analysis.html: R/analysis.Rmd Data/cleandata.csv
    cd R;R -e "rmarkdown::render('analysis.Rmd')"

Data/cleandata.csv: R/prepData.R RawData/rawdata.csv
    cd R;R CMD BATCH prepData.R

RawData/rawdata.csv: Python/xls2csv.py RawData/rawdata.xls
    Python/xls2csv.py RawData/rawdata.xls > RawData/rawdata.csv
```

3. Automate the process (GNU Make)

```
R/analysis.html: R/analysis.Rmd Data/cleandata.csv
    cd R;R -e "rmarkdown::render('analysis.Rmd')"

Data/cleandata.csv: R/prepData.R RawData/rawdata.csv
    cd R;R CMD BATCH prepData.R

RawData/rawdata.csv: Python/xls2csv.py RawData/rawdata.xls
    Python/xls2csv.py RawData/rawdata.xls > RawData/rawdata.csv
```

4. Turn scripts into reproducible reports

Gough project diagnostics

Karl Broman, 3 March 2014

Combine genotypes and phenotypes

I've combined the initial genotypes (using the re-clustered genotypes for plates 14-16) with the well-behaved portion of the re-run genotypes. I'm focusing on 36813 markers that are informative (though, as we'll see, there are still a lot of badly behaved and basically non-informative markers that need to be removed). I've combined data on replicate samples, to give one set of genotype calls for each sample.

There are 1497 genotyped mice and 1464 phenotyped mice. All of the mice in the phenotype data have genotypes, but there are 33 genotyped mice with no phenotypes, including 3 Gough mice and 30 F2 progeny.

4. Turn scripts into reproducible reports

Gough project diagnostics

Karl Broman, 3 March 2014

Comb

I've comb
the well-b
informat
informat
give one

There are
data have
mice and

```
I've combined the initial genotypes (using the re-clustered genotypes
for plates 14-16) with the well-behaved portion of the re-run
genotypes. I'm focusing on `r totmar(g)` markers that are informative
(though, as we'll see, there are still a lot of badly behaved and
basically non-informative markers that need to be removed).
I've combined data on replicate samples, to give one set of genotype
calls for each sample.

There are `r nind(g)` genotyped mice and `r nrow(phe)` phenotyped
mice. All of the mice in the phenotype data have genotypes, but there
are `r sum(is.na(match(gid, pid)))` genotyped mice with no phenotypes,
including `r sum(g$pheno$gen[which(is.na(match(gid, pid)))]==0)`
Gough mice and `r sum(g$pheno$gen[which(is.na(match(gid, pid)))]==2)`
F2 progeny.
```

5. Turn repeated code into functions

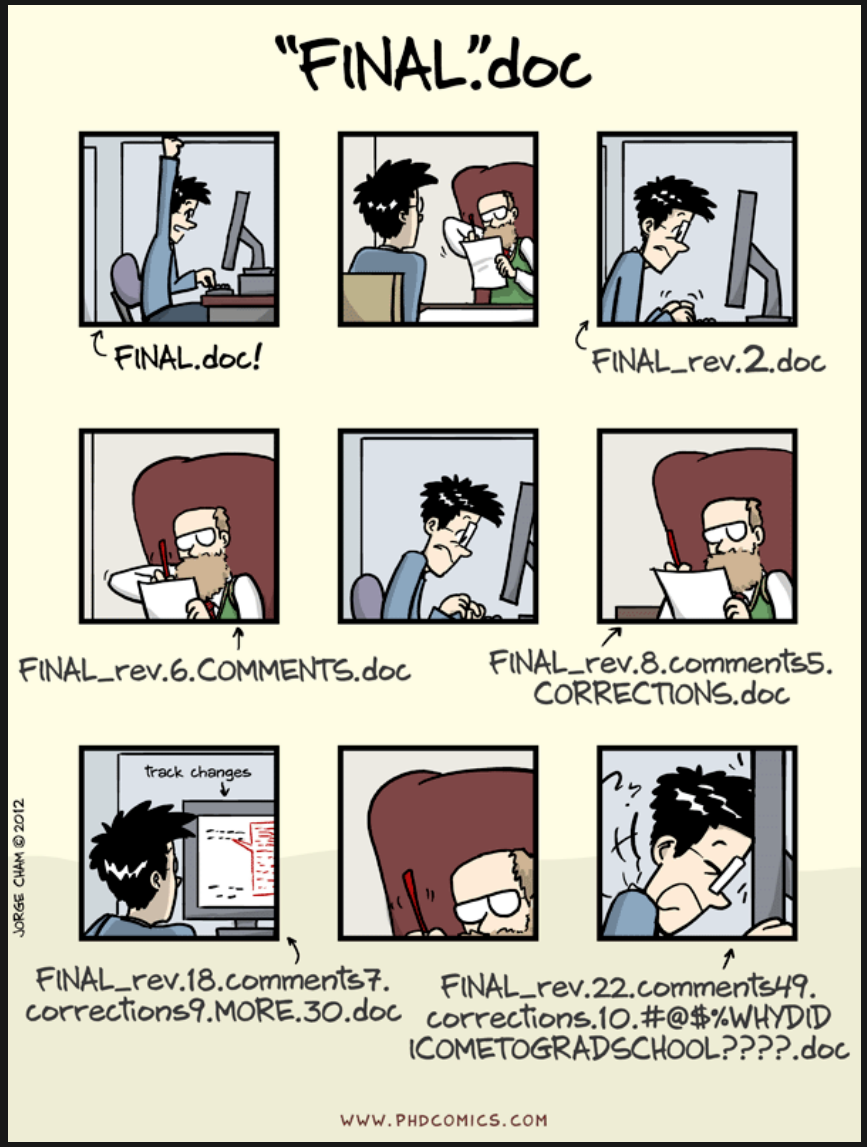
```
# Python
def read_genotypes (filename):
    "Read matrix of genotype data"
```

```
# R
plot_genotypes <-
function(genotypes, ...)
{
}
```

6. Create a package/module

Don’t repeat yourself

7. Use version control (git/GitHub)



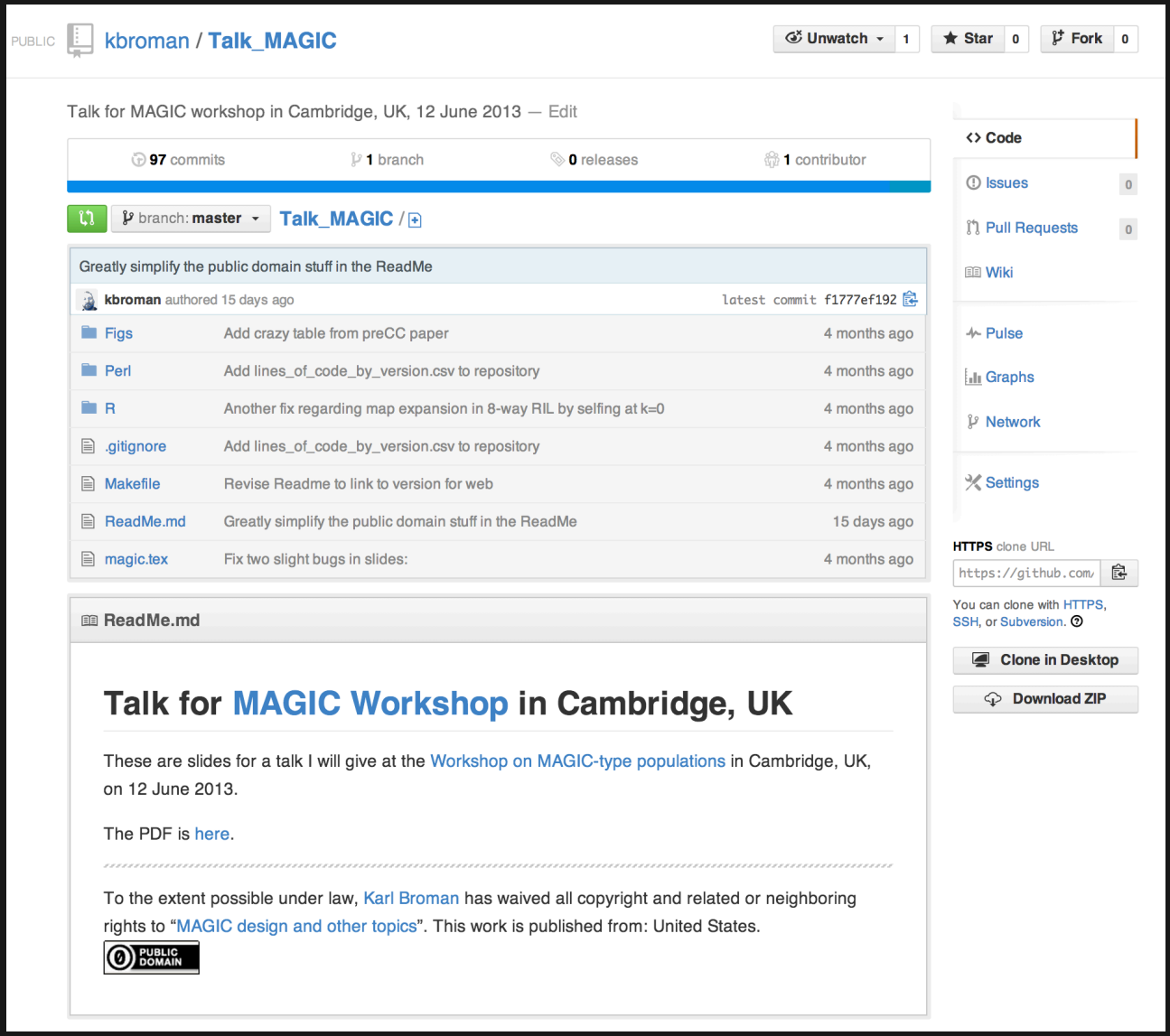
No “final” in file names

```
Deprecated/                              hypo_prcomp.RData
ReadMe.txt                               islet_int1_final.RData
adipose_int1_final.RData                 islet_int2_final.RData
adipose_int2_final.RData                 islet_mlratio_final.RData
adipose_mlratio_final.RData              islet_mlratio_nqrank_final.RData
adipose_mlratio_nqrank_final.RData       islet_prcomp.RData
adipose_prcomp.RData                     kidney_int1_final.RData
aligned_geno_with_pmap.RData             kidney_int2_final.RData
batches_final.RData                      kidney_mlratio_final.RData
batches_raw_final.RData                  kidney_mlratio_nqrank_final.RData
cpl_final.RData                          kidney_prcomp.RData
d2o_final.RData                          lipomics_final_rev2.RData
gastroc_int1_final.RData                 liverTG_final.RData
gastroc_int2_final.RData                 liver_int1_final.RData
gastroc_mlratio_final.RData              liver_int2_final.RData
gastroc_mlratio_nqrank_final.RData       liver_mlratio_final.RData
gastroc_prcomp.RData                     liver_mlratio_nqrank_final.RData
hypo_int1_final.RData                    liver_prcomp.RData
hypo_int2_final.RData                    mirna_final.RData
hypo_mlratio_final.RData                 necropsy_final_rev2.RData
hypo_mlratio_final_old.RData             plasmaurine_final_rev.RData
hypo_mlratio_nqrank_final.RData          pmark.RData
hypo_mlratio_nqrank_final_old.RData      rbm_final.RData
hypo_omit.RData
```

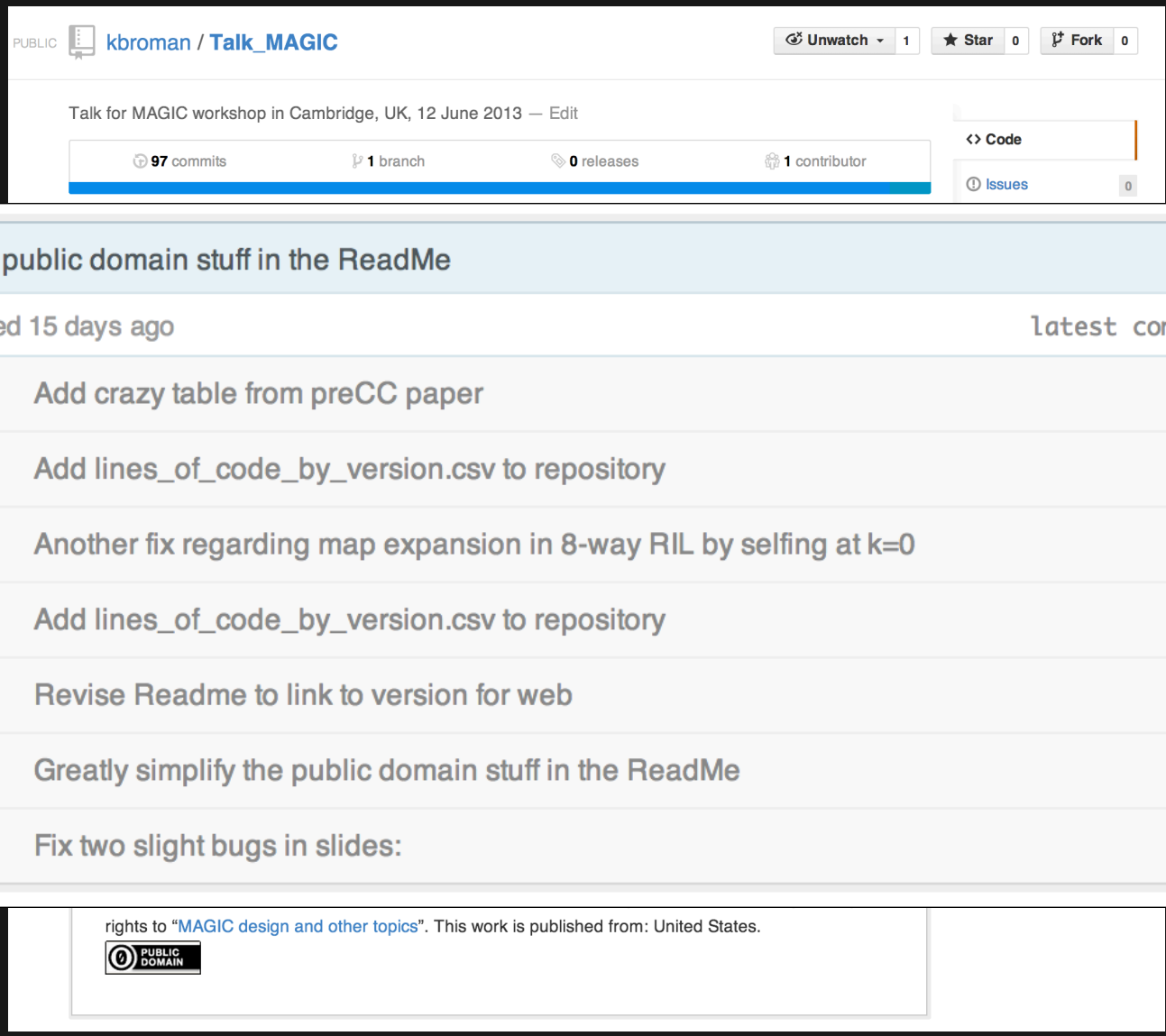
No “final” in file names

```
Deprecated/                                hypo_prcomp.RData
ReadMe.txt                                 islet_int1_final.RData
adipose_int1_final.RData                   islet_int2_final.RData
adipose_int2_final.RData                   islet_mlratio_final.RData
adipose_mlratio_final.RData                islet_mlratio_nqrank_final.RData
adipose_mlratio_nqrank_final.RData         islet_prcomp.RData
adipose_prcomp.RData                       kidney_int1_final.RData
aligned_geno_with_pmap.RData               kidney_int2_final.RData
batches_final.RData                        kidney_mlratio_final.RData
batches_raw_final.RData                    kidney_mlratio_nqrank_final.RData
cpl_final.RData                            kidney_prcomp.RData
d2o_final.RData                            lipomics_final_rev2.RData
gastroc_int1_final.RData                   liverTG_final.RData
gastroc_int2_final.RData                   liver_int1_final.RData
gastroc_mlratio_final.RData                liver_int2_final.RData
gastroc_mlratio_nqrank_final.RData         liver_mlratio_final.RData
gastroc_prcomp.RData                       liver_mlratio_nqrank_final.RData
hypo_int1_final.RData                      liver_prcomp.RData
hypo_int2_final.RData                      mirna_final.RData
hypo_mlratio_final.RData                   necropsy_final_rev2.RData
hypo_mlratio_final_old.RData               plasmaurine_final_rev.RData
hypo_mlratio_nqrank_final.RData            pmark.RData
hypo_mlratio_nqrank_final_old.RData        rbm_final.RData
hypo_omit.RData
```

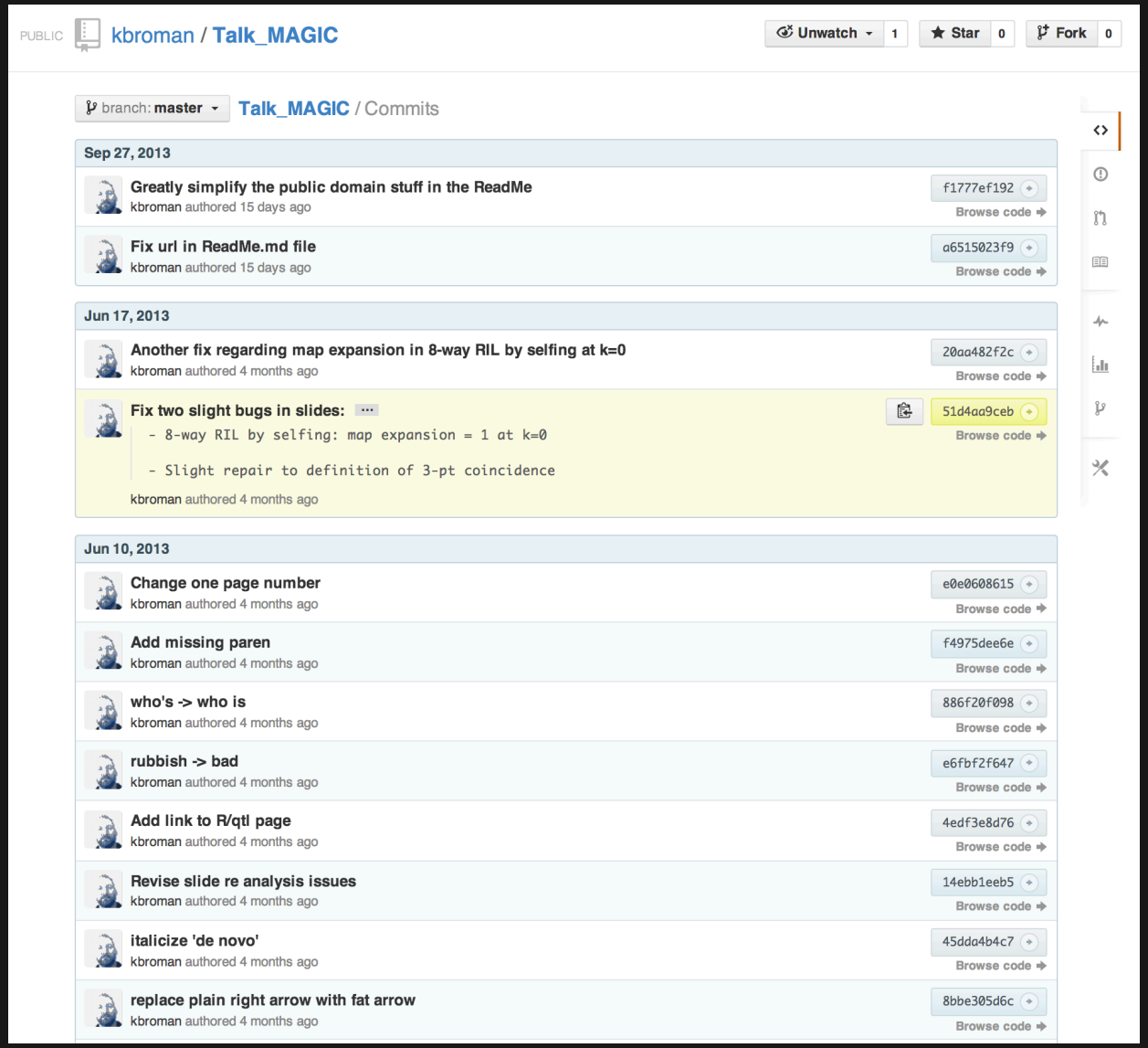
7. Use version control (git/GitHub)



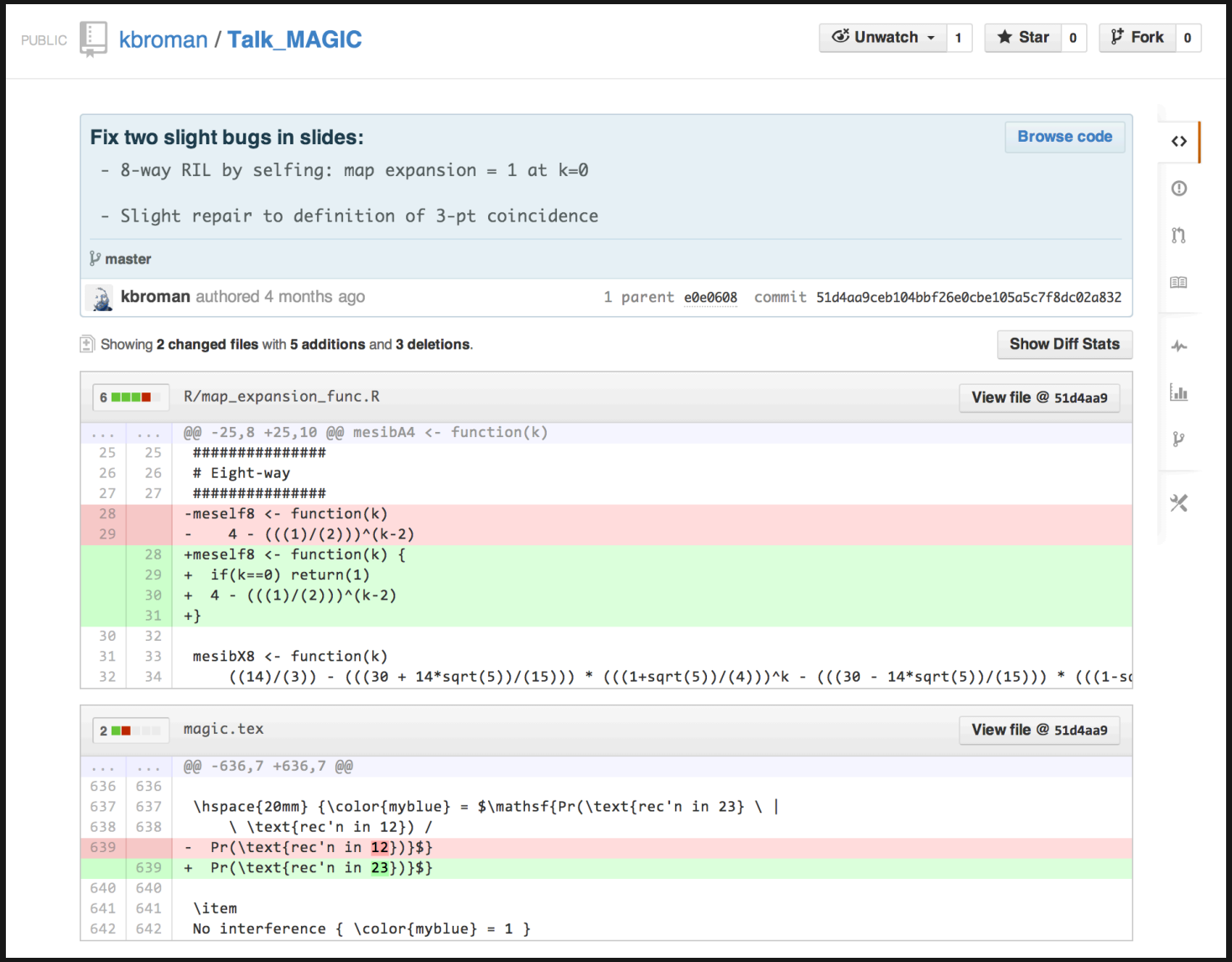
7. Use version control (git/GitHub)



7. Use version control (git/GitHub)



7. Use version control (git/GitHub)



PUBLIC kbroman / Talk_MAGIC

Unwatch 1 | Star 0 | Fork 0

Fix two slight bugs in slides:

Browse code

```
- 8-way RIL by selfing: map expansion = 1 at k=0

- Slight repair to definition of 3-pt coincidence
```

master

kbroman authored 4 months ago 1 parent e0e0608 commit 51d4aa9ceb104bbf26e0cbe105a5c7f8dc02a832

Showing **2 changed files** with **5 additions** and **3 deletions**. Show Diff Stats

6 R/map_expansion_func.R View file @ 51d4aa9

```
@@ -25,8 +25,10 @@ mesibA4 <- function(k)
 ################
 # Eight-way
 ################
-meself8 <- function(k)
--    4 - (((1)/(2)))^(k-2)
+meself8 <- function(k) {
+  if(k==0) return(1)
+  4 - (((1)/(2)))^(k-2)
+}

 mesibX8 <- function(k)
     ((14)/(3)) - (((30 + 14*sqrt(5))/(15))) * (((1+sqrt(5))/(4)))^k - (((30 - 14*sqrt(5))/(15))) * (((1-s
```

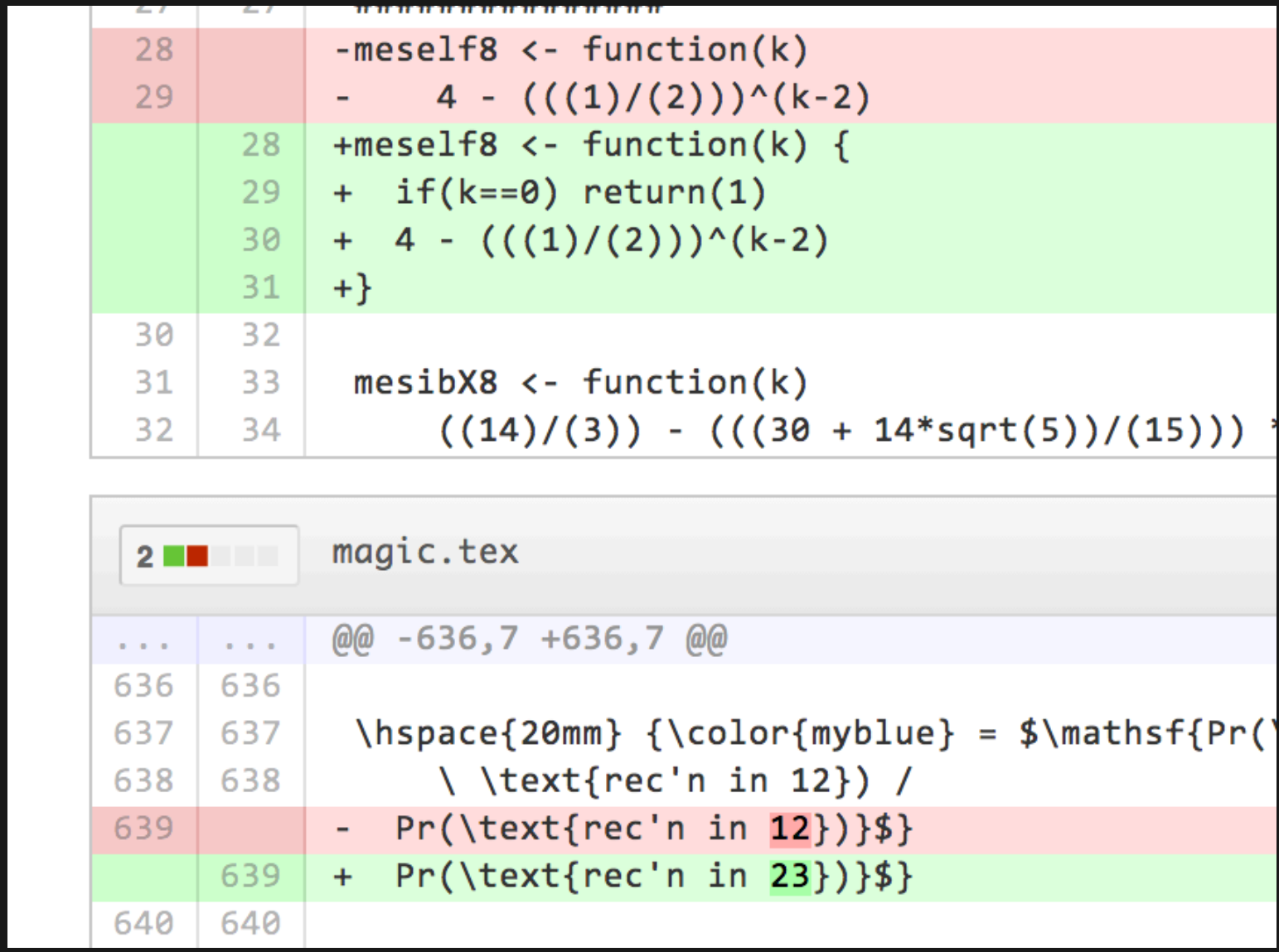
2 magic.tex View file @ 51d4aa9

```
@@ -636,7 +636,7 @@

 \hspace{20mm} {\color{myblue} = $\mathsf{Pr(\text{rec'n in 23} \ |
     \ \text{rec'n in 12}) /
-  Pr(\text{rec'n in 12})}$}
+  Pr(\text{rec'n in 23})}$}

 \item
 No interference { \color{myblue} = 1 }
```

7. Use version control (git/GitHub)



```
28      -meself8 <- function(k)
29      -    4 - (((1)/(2)))^(k-2)
    28  +meself8 <- function(k) {
    29  +  if(k==0) return(1)
    30  +  4 - (((1)/(2)))^(k-2)
    31  +}
30  32
31  33   mesibX8 <- function(k)
32  34       ((14)/(3)) - (((30 + 14*sqrt(5))/(15)))
```

2 magic.tex

```
...  ...  @@ -636,7 +636,7 @@
636  636
637  637   \hspace{20mm} {\color{myblue} = $\mathsf{Pr(
638  638       \ \text{rec'n in 12}) /
639      -  Pr(\text{rec'n in 12})}$}
    639  +  Pr(\text{rec'n in 23})}$}
640  640
```

8. License your software

Pick a license, any license

– Jeff Atwood

Other considerations

- Testing
  - are you getting the right answers?
- Software versions
  - will your stuff work when dependencies change?
- Large-scale computations
  - computation time + dependence on cluster environment
- Collaborations
  - coordinating who does what and where things live
- Distribution
  - where and how to distribute data and code?

The most important tool is the mindset,
when starting, that the end product
will be reproducible.

– Keith Baggerly

Summary

1. Organize your data & code
2. Everything with a script
3. Automate the process (GNU Make)
4. Turn scripts into reproducible reports
5. Turn repeated code into functions
6. Create a package/module
7. Use version control (git/GitHub)
8. Pick a license, any license

Slides: bit.ly/jax2017-05

kbroman.org

github.com/kbroman

@kwbroman