Steps toward reproducible research

Karl Broman

Biostatistics & Medical Informatics
Univ. Wisconsin–Madison

kbroman.org
github.com/kbroman
@kwbroman

Slides: bit.ly/StPaul-0

Preparations

Install R, RStudio, and some R packages.

`bit.ly/StPaul-prep`

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. Arrange data to ease analysis

Write programs for people

Organize data for computers

Activity 1

Arrange data to ease analysis

`bit.ly/StPaul-1`

Activity 1: original file

	A	B	C	D	E	F	G
1							
2	1min						
3			Normal			Mutant	
4		10−05−16	10−12−16	10−19−16	10−05−16	10−12−16	10−19−16
5	B6	146.6	138.6	155.6	166	179.3	186.9
6	BTBR	245.7	240	243.1	177.8	171.6	188.1
7							
8	5min						
9			Normal			Mutant	
10		10−05−16	10−12−16	10−19−16	10−05−16	10−12−16	10−19−16
11	B6	333.6	353.6	408.8	450.6	474.4	423.8
12	BTBR	514.4	610.6	597.9	412.1	447.4	446.5

bit.ly/StPaul-1

Activity 1: tidy file

	A	B	C	D	E
1	strain	genotype	treatment_time	date	response
2	B6	Normal	1min	2016−10−05	146.6
3	B6	Normal	1min	2016−10−12	138.6
4	B6	Normal	1min	2016−10−19	155.6
5	B6	Mutant	1min	2016−10−05	166
6	B6	Mutant	1min	2016−10−12	179.3
7	B6	Mutant	1min	2016−10−19	186.9
8	BTBR	Normal	1min	2016−10−05	245.7
9	BTBR	Normal	1min	2016−10−12	240
10	BTBR	Normal	1min	2016−10−19	243.1

bit.ly/StPaul-1-tidy

Organizing data in spreadsheets

- Make it a rectangle (rows = observations, cols=variables)
- Use a single header row; avoid spaces.
- Be consistent.
- Use care about dates. (3 separate columns?)
- Put just one thing in a cell.
- Fill in all cells.
- No calculations in the raw data files.
- Don't use font color or highlighting as data.
- Make backups.
- Use data validation to avoid data entry mistakes.
- Save the data in plain text files.

kbroman.org/dataorg

2. Organize your data & code

File organization and naming
are powerful weapons against chaos.

– Jenny Bryan

2. Organize your data & code

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

(paraphrasing Mark Holder)

Activity 2

Organizing and naming of files for a project

`bit.ly/StPaul-2`

Activity 2

```
Raw phenotype data

    CPL_Rosetta_Lipids_FINAL.xlsx
    Complete F2 Liver TG Set.xlsx
    D2O_Summary_of_All_F2_Samples_MF_30July2009.xlsx
    FINAL_RBM_DATA_102989_26Sep2007.xlsx
    Mapped_Urine_Plasma_Data_to_Statgen.xlsx
    Necropsy_Tracking_Report_rk61412.xlsx
    Necropsy_Tracking_Report_rk_052912_atb.xlsx
    Necropsy_Tracking_Report_rk_2011-04-26.xlsx
    Original_Necropsy_Tracking_Report_rk.xlsx
    RBM_Tube_Number_Key.xlsx

Raw genotype data

    Final Fit1_Filtered_Assay_Allele_Signals_and_Genotypes_18Sep.txt

Converted data

    clinpheno.csv
    detailed_genotypes.csv
    genotypes4rqtl.csv
    genotypes_karl.csv
```

```
R scripts to organize data

    check_necropsy_files.R
    check_necropsy_files_2012-06-02.R
    combine_pheno.R
    combine_pheno2.R
    combine_pheno3.R
    compareData.R
    func.R
    prepData.R

Analysis

    fig1.png
    fig2.png
    fig3.png
    fig4.png
    fig5.png
    fig6.png
    fig7.png
    fig8.png
    scanone_clinphe.Rmd
    scanone_clinphe.html
```

bit.ly/StPaul-2b

2. 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/
```

3. Everything with a script

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

4. 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
```

kbroman.org/minimal_make

4. 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
```

kbroman.org/minimal_make

4. 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
```

kbroman.org/minimal_make

4. 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
```

kbroman.org/minimal_make

5. 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.

5. Turn scripts into reproducible reports

Gough project diagnostics

Karl Broman, 3 March 2014

Comb

I've comb
the well-
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.
```

Activity 3

Create an R Markdown report within RStudio

bit.ly/StPaul-3

6. Turn repeated code into functions

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

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

7. Create a package/module

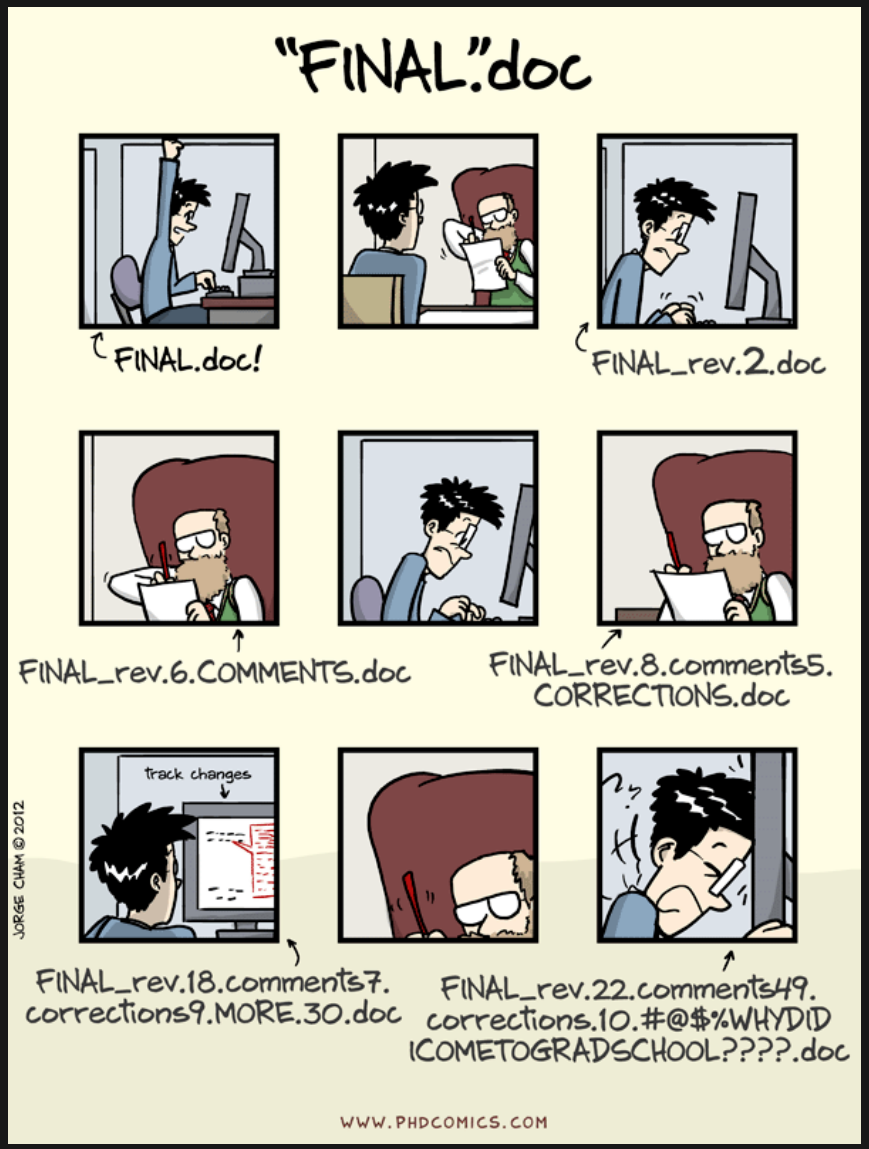
Don’t repeat yourself

7. Create a package/module

Don’t repeat yourself

kbroman.org/pkg_primer

8. Use version control (git/GitHub)



No “final” in file names

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

No “final” in file names

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

8. 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)
25   25    ###############
26   26    # Eight-way
27   27    ###############
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))) * (((1+sqrt(5))/(4)))^k - (((30 - 14*sqrt(5))/(15))) * (((1-s
```

2 magic.tex View file @ 51d4aa9

```
...  ...  @@ -636,7 +636,7 @@
636  636
637  637   \hspace{20mm} {\color{myblue} = $\mathsf{Pr(\text{rec'n in 23} \ |
638  638       \ \text{rec'n in 12}) /
639        -  Pr(\text{rec'n in 12})}$}
     639  +  Pr(\text{rec'n in 23})}$}
640  640
641  641   \item
642  642   No interference { \color{myblue} = 1 }
```

kbroman.org/github_tutorial

9. License your software

Pick a license, any license

– Jeff Atwood

Summary

1. Arrange data to ease analysis
2. Organize your data and code
3. Everything with a script
4. Automate the process
5. Turn scripts into reproducible reports
6. Turn repeated code into functions
7. Create a package/module
8. Use version control
9. License your software

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

– Keith Baggerly

Slides: bit.ly/StPaul-0

kbroman.org

github.com/kbroman

@kwbroman