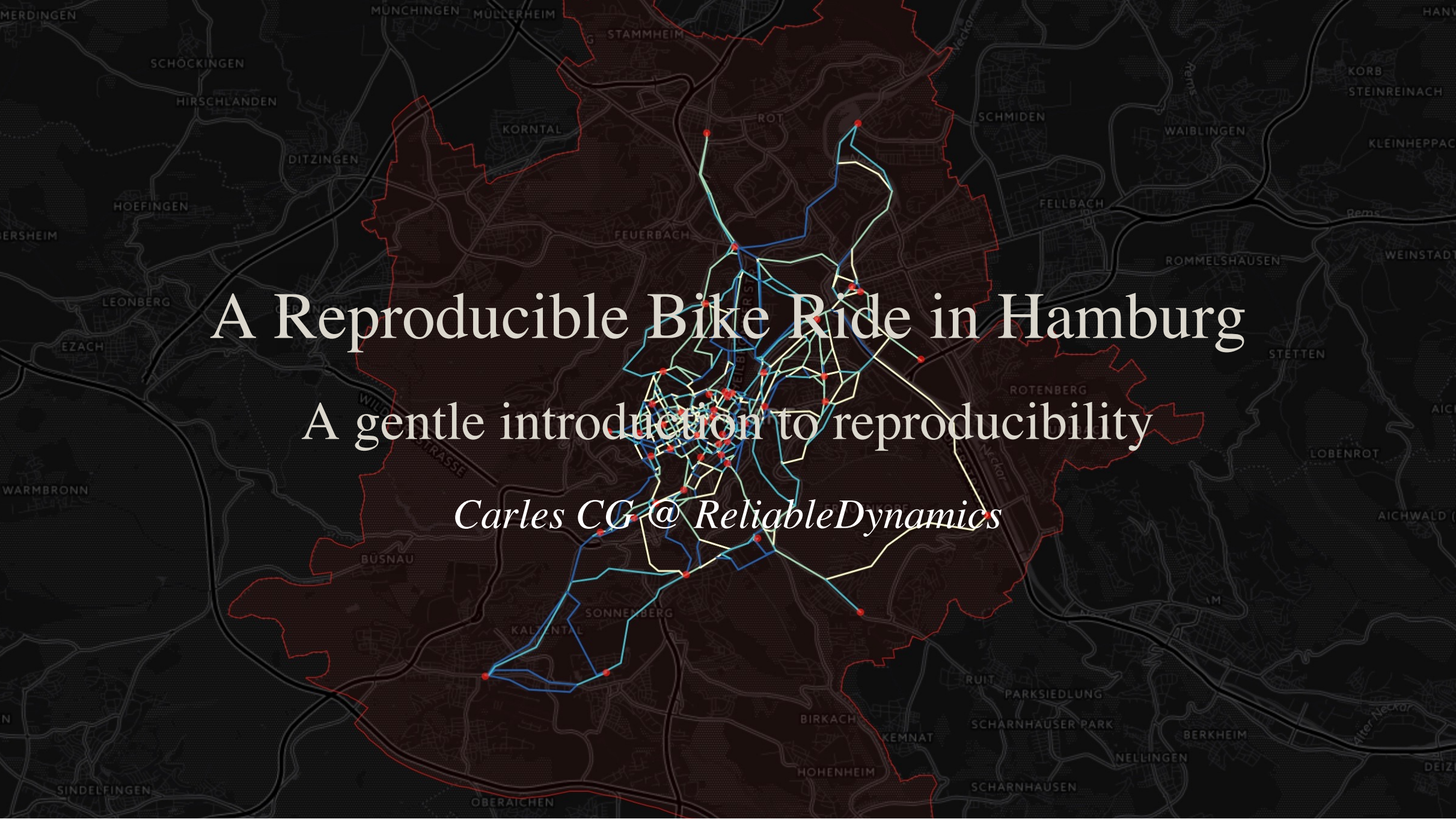


A Reproducible Bike Ride in Hamburg

A gentle introduction to reproducibility

Carles CG @ ReliableDynamics

Reproducibility + Application in a Case

Slides available at http://bit.ly/bike_rideR

BIKE SHARING USAGE IN HAMBURG

THE DATA

The map shows the bike sharing usage of StadtRAD, the bike sharing system in Hamburg - Germany. The data is available on the open data platform from Deutsche Bahn, the public railway company in Germany. The last new StadtRAD station was put into operation in May 2016, that is why a have chosen to display the usage of June 2016. The brighter the lines, the more bikes have been cycled along that street.

THE PROCESSING

From data processing and spatial analysis to visualization the whole project was done in R. I have used the leaflet and shiny package to display the data interactively. The bikes themselves don't have GPS, so the routes are estimated on a fastest route basis using the awesome CycleStreets API. The biggest challenge has been the aggregation of overlapping routes, found the osmline function from the stplanr package very helpful. It converts a series of overlaping lines and aggregates their values for overlapping segments.

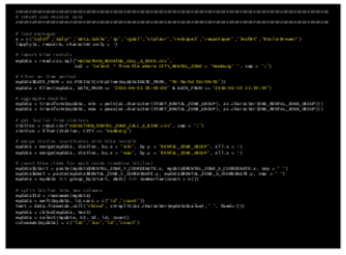
The raw data file from Deutsche Bahn is quite huge so I struggled to import the data into R to be able to process it. In the end the readLines.sqlite function from the sqldf package did the job. This way I did not need to import the whole file and just could filter out the bike rides for Hamburg.

The code could easily be used to map other spatial data, for example the car sharing data from car2go which is available via their API. This might be a future project.

THE MAP

As a cycling ethusiast and Hamburg native I have been riding the streets of Hamburg for a long time now. Over the years I found my favorite cycle routes throughout the city but also know the tight and problematic corners of Hamburg, where missing or overcrowded cycling paths bring you too close to other bikers, cars or pedestrians. It is amusing to see that the data set can proof some of my hypothesis about the current state of the bicycle infrastructure in Hamburg and even bring up new questions I have not even thought about before.

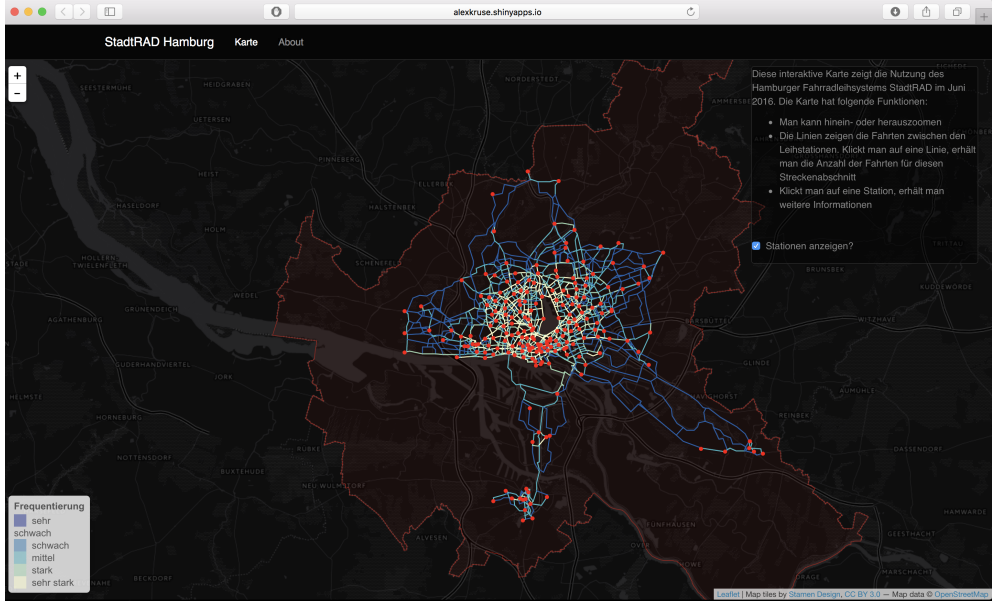
When you look at the map you can see a widely spread bike sharing network over big parts of the city but also notice some enclosing processes where missing stations disconnect bike riders from the high frequented and well connected city center. As the Elbe river seperates Hamburg in a northern and southern part it seems like bike sharing became a well accepted means of transportation to keep both parts of the city connected.



github/kruse-alex @krusealex2013

Alex Kruse

kruse-alex



Feeling lucky?

1. Look at Github and found the repo

   kruse-alex / **bike_sharing**

2. Found the script file!

   Branch: **master** **bike_sharing** / **stadtrad_processing.R**

3. Download original data & play with it

PERSPECTIVE

Reproducible Research in Computational Science

Roger D. Peng

Computational science has led to exciting new developments, but the nature of the work has exposed limitations in our ability to evaluate published findings. Reproducibility has the potential to serve as a minimum standard for judging scientific claims when full independent replication of a study is not possible.

PLOS | COMPUTATIONAL BIOLOGY

PERSPECTIVE

Good enough practices in scientific computing

Greg Wilson[1☯]*, Jennifer Bryan[2☯], Karen Cranston[3☯], Justin Kitzes[4☯], Lex Nederbragt[5☯], Tracy K. Teal[6☯]

1 Software Carpentry Foundation, Austin, Texas, United States of America, **2** RStudio and Department of Statistics, University of British Columbia, Vancouver, British Columbia, Canada, **3** Department of Biology, Duke University, Durham, North Carolina, United States of America, **4** Energy and Resources Group, University of California, Berkeley, Berkeley, California, United States of America, **5** Centre for Ecological and Evolutionary Synthesis, University of Oslo, Oslo, Norway, **6** Data Carpentry, Davis, California, United States of America

☯ These authors contributed equally to this work.
* gvwilson@software-carpentry.org

OPEN ACCESS Freely available online

PLOS | BIOLOGY

Community Page

Best Practices for Scientific Computing

Greg Wilson[1]*, D. A. Aruliah[2], C. Titus Brown[3], Neil P. Chue Hong[4], Matt Davis[5], Richard T. Guy[6¤], Steven H. D. Haddock[7], Kathryn D. Huff[8], Ian M. Mitchell[9], Mark D. Plumbley[10], Ben Waugh[11], Ethan P. White[12], Paul Wilson[13]

1 Mozilla Foundation, Toronto, Ontario, Canada, **2** University of Ontario Institute of Technology, Oshawa, Ontario, Canada, **3** Michigan State University, East Lansing, Michigan, United States of America, **4** Software Sustainability Institute, Edinburgh, United Kingdom, **5** Space Telescope Science Institute, Baltimore, Maryland, United States of America, **6** University of Toronto, Toronto, Ontario, Canada, **7** Monterey Bay Aquarium Research Institute, Moss Landing, California, United States of America, **8** University of California Berkeley, Berkeley, California, United States of America, **9** University of British Columbia, Vancouver, British Columbia, Canada, **10** Queen Mary University of London, London, United Kingdom, **11** University College London, London, United Kingdom, **12** Utah State University, Logan, Utah, United States of America, **13** University of Wisconsin, Madison, Wisconsin, United States of America

About me

Reliable Dynamics

Carles CG
Data Scientist & freelanceR

carles@reliabledynamics.com
@carles_

Reproducibility + Application in a Case

Extended... but how?

A series of extensions were made.

Extended... but how?

A series of extensions were made.

- Archive raw, pre/post process and final data

Extended... but how?

A series of extensions were made.

- Archive raw, pre/post process and final data
- Rewrite some parts of the code with `tidyverse` principles

Extended... but how?

A series of extensions were made.

- Archive raw, pre/post process and final data
- Rewrite some parts of the code with `tidyverse` principles
- Benchmark reading and munging code

Extended... but how?

A series of extensions were made.

- Archive raw, pre/post process and final data
- Rewrite some parts of the code with `tidyverse` principles
- Benchmark reading and munging code
- Extend the analysis

Extended... but how?

A series of extensions were made.

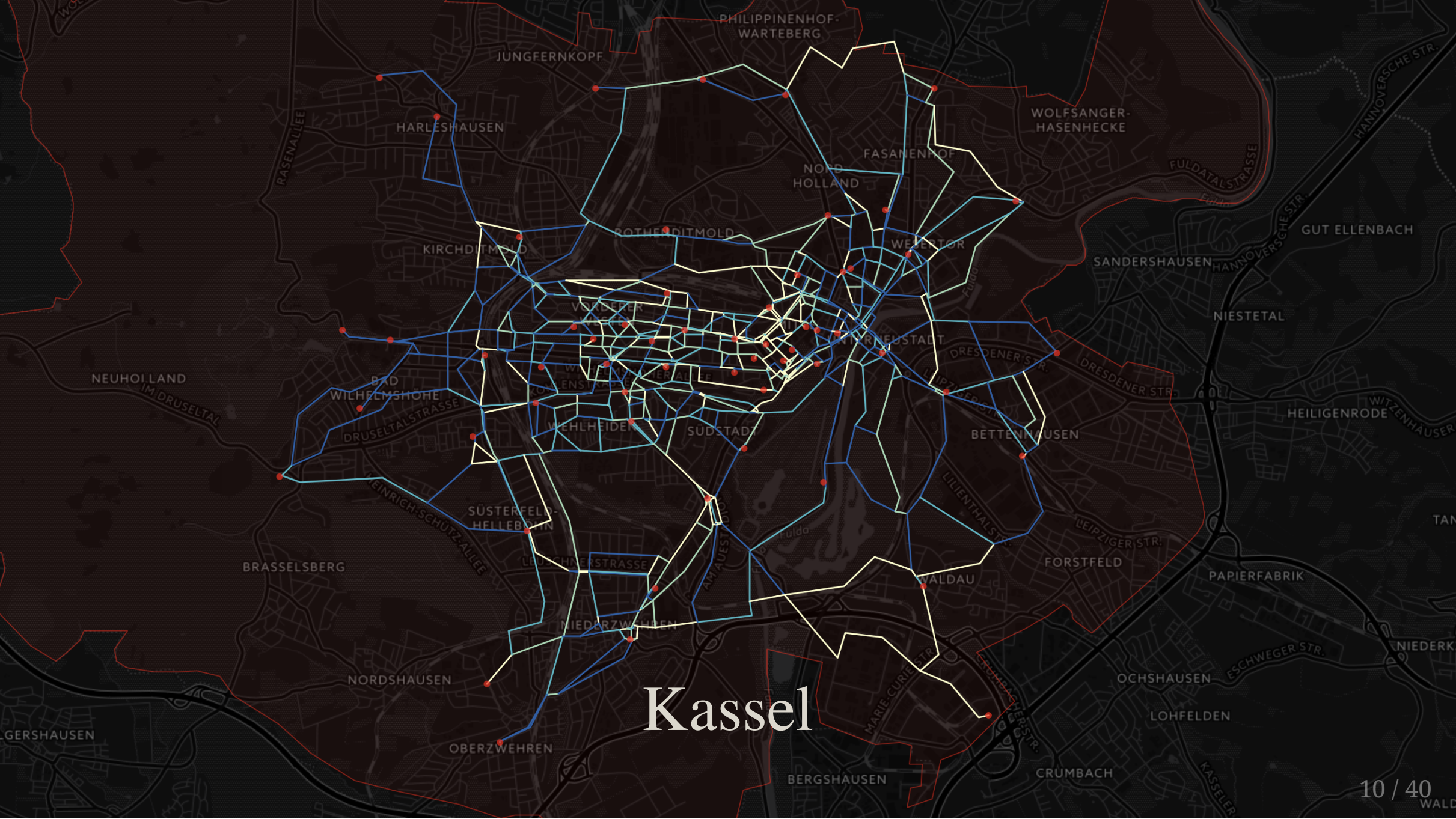
- Archive raw, pre/post process and final data
- Rewrite some parts of the code with `tidyverse` principles
- Benchmark reading and munging code
- Extend the analysis
- Documentation

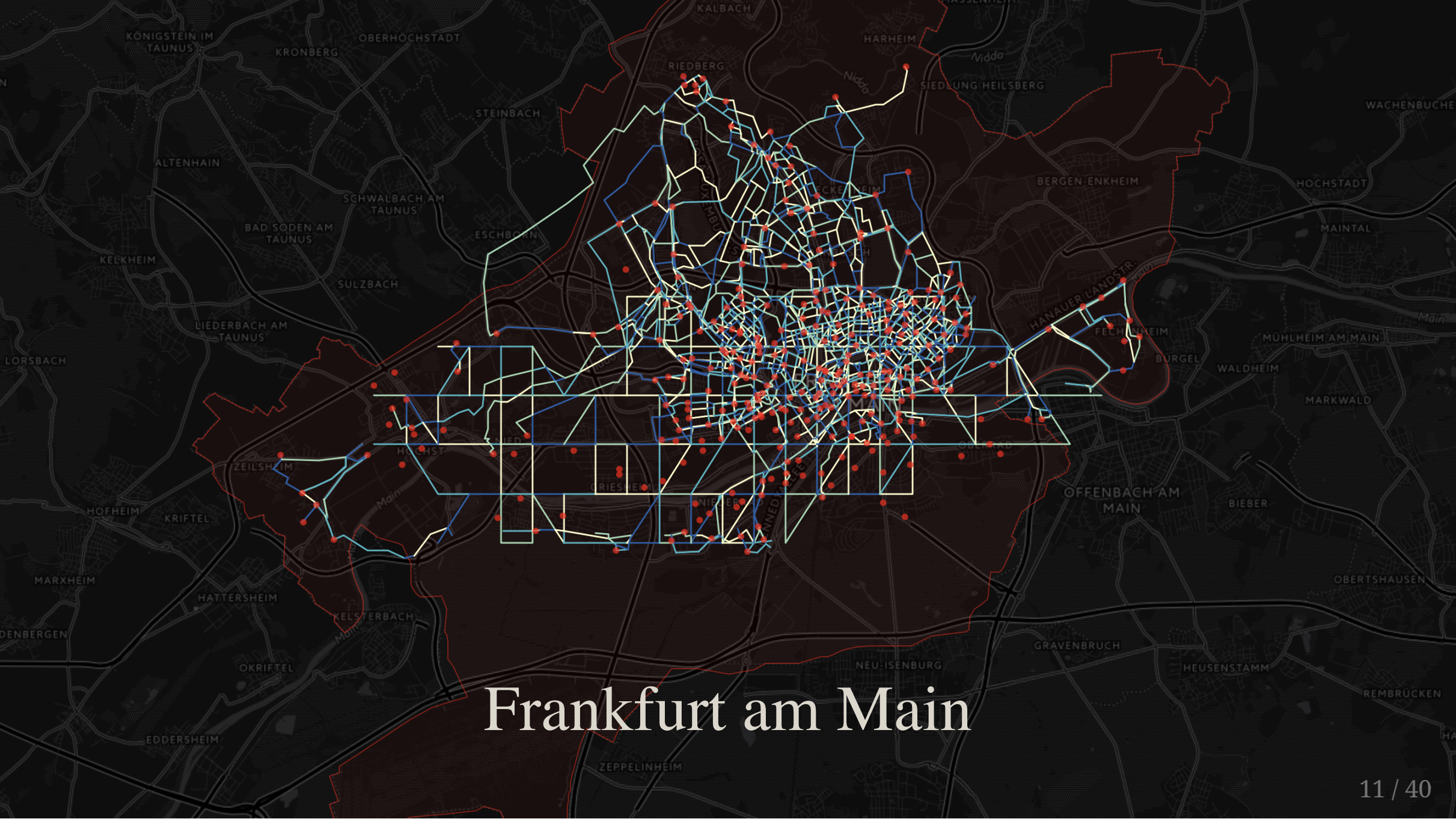
Extended... but how?

A series of extensions were made.

- Archive raw, pre/post process and final data
- Rewrite some parts of the code with `tidyverse` principles
- Benchmark reading and munging code
- Extend the analysis
- Documentation


Stuttgart
HAUSEN
WEILIMDORF
FEUERBACH
SOMMERRAIN
GIEBEL
WOLFBUSCH
BERGHEIM
STUTTGART-NORD
LUGINSLAND
HÖHE
BOTNANG
STUTTGART-OST
ROTENBERG
UNTERTÜRKHEIM
UHLBACH
WANGEN
OBERTÜRKHEIM
RÜDERN
FRAUENKOPF
ROHRACKER
HEDELFINGEN
LEDERBERG
BUSNAU
BRUDERRAIN
NECKARHALDE
METTINGEN
HEUMADEN
SILLENBUCH
SONNENBERG
KALTENTAL
HOFFELD
RIEDENBERG
BRÜHL
WEIL


Kassel
PHILIPPINENHOF-
WARTEBERG
JUNGFERNKOPF
HARLESHAUSEN
WOLFSANGER-
HASENHECKE
FASANENHOF
NORD-
HOLLAND
KIRCHDITMOLD
SANDERSHAUSEN
GUT ELLENBACH
NIESTETAL
NEUHOLLAND
BAD
WILHELMSHÖHE
WEHLHEIDEN
SÜDSTADT
BETTENHAUSEN
HEILIGENRODE
SÜSTERFELD-
HELLEBERG
BRASSELSBERG
FORSTFELD
WALDAU
PAPIERFABRIK
NORDSHAUSEN
OCHSHAUSEN
LOHFELDEN
OBERZWEHREN
BERGSHAUSEN
CRUMBACH



Frankfurt am Main


Berlin


Helmholtzplatz
VENLOER STRASSE
EHRENFELD
AGNESVIERTEL
INNERE KANALSTR.
Rhein
ZOOBRÜCKE
STEGERWALDSIEDLUNG
BUCHFORST
Rheinpark
MAARWEG
Mediapark
EIGELSTEIN
URSULAVIERTEL
STADTAUTOBAHN
E19 (a)
134
Barmer Platz
Bürgerpark
KALK
Innerer Grüngürtel
BELGISCHES VIERTEL
Von-Sandt-Platz
AACHENER STRASSE
BRAUNSFELD
DEUTZ
Clarenbachkanal
Aachener Weiher
Hiroshima-Nagasaki-Park
Karl-Schwering-Platz
KAPITOLVIERTEL
MAURITIUSVIERTEL
GÖTENRING
DEUTZER RING
Humboldtpark
HUMBOLDT
GEORGSVIERTEL
LINDENTHAL
Kahnweiher
Deutzer Stadtgarten
SIEGBURGER STR.
Lesegarten
PANTALEONSVIERTEL
Innenhof
GLEUELER STRASSE
LINDENTHALGÜRTEL
SEVERINSVIERTEL
De-Noël-Platz
LUXEMBURGER STR.
KRIEL
SÜLZ
Räuberwäldchen
SÜDSTADT
DECKSTEIN

Köln

Reproducibility... but why?

> “Reproducibility is the ability to take the code and data from a previous publication, rerun the code and get the same results”

https://simplystatistics.org/2017/03/02/rr-glossy/

Reproducibility benefits

Reproducibility benefits

- Make it easier for your future self. Data might be expanded in the future!

Reproducibility benefits

- Make it easier for your future self. Data might be expanded in the future!
- Review the basis that lead to decision

Reproducibility benefits

- Make it easier for your future self. Data might be expanded in the future!
- Review the basis that lead to decision
- Transparency

Reproducibility benefits

- Make it easier for your future self. Data might be expanded in the future!
- Review the basis that lead to decision
- Transparency
- Avoid manual errors

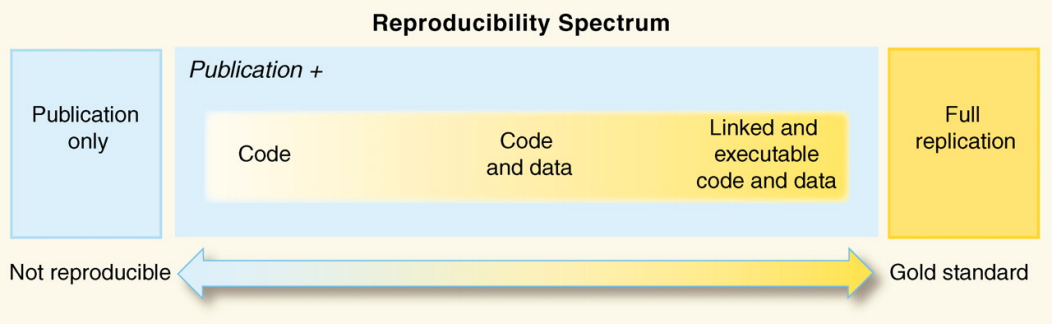
Reproducibility benefits

- Make it easier for your future self. Data might be expanded in the future!
- Review the basis that lead to decision
- Transparency
- Avoid manual errors
- Learn new skills

Reproducibility benefits

- Make it easier for your future self. Data might be expanded in the future!
- Review the basis that lead to decision
- Transparency
- Avoid manual errors
- Learn new skills
- Science! Reproducibility vs. Replication

Science



> "**Replication**: This is the act of repeating an entire study, independently of the original investigator without the use of original data (but generally using the same methods).
>
> **Reproducibility**: A study is reproducible if you can take the original data and the computer code used to analyze the data and reproduce all of the numerical findings from the study."

https://simplystatistics.org/2016/08/24/replication-crisis/

Reproducibility gone wrong

Growth in a Time of Debt

By Carmen M. Reinhart and Kenneth S. Rogoff*

American Economic Review: Papers & Proceedings 100 (May 2010): 573–578
http://www.aeaweb.org/articles.php?doi=10.1257/aer.100.2.573

```
# 23-class classification problem

skf=StratifiedKFold(labels,8)

if trainsvm:
    pred=N.zeros(len(labels))
    for train,test in skf:
        clf=LinearSVC()
        clf.fit(data[train],labels[train])          data[:,train]
        pred[test]=clf.predict(data[test])          data[:,test]
```

Results:
93% accuracy

Results:
53% accuracy

Keith Baggerly, left, and Kevin Coombes, statisticians at M. D. Anderson Cancer Center, found flaws in research on tumors. Michael Stravato for The New York Times

(Some)
Principles of reproduciblity

Checklist

Consider…

- Data

Checklist

Consider…

- Data
- Code

Checklist

Consider…

- Data
- Code
- Environment

Checklist

Consider…

- Data
- Code
- Environment
- Documentation

Checklist

Consider…

- Data
- Code
- Environment
- Documentation

Soft considerations

Checklist

Consider…

- Data
- Code
- Environment
- Documentation

Soft considerations

- How important is the output of the analysis?

Checklist

Consider…

- Data
- Code
- Environment
- Documentation

Soft considerations

- How important is the output of the analysis?
- Team effort vs. cowboy coder

Checklist

Consider…

- Data
- Code
- Environment
- Documentation

Soft considerations

- How important is the output of the analysis?
- Team effort vs. cowboy coder
- How much time should we invest to make it till some degrees reproducible?

Data Management

Long vs. short term archiving

Long vs. short term archiving

- Version Control services 

Long vs. short term archiving

- Version Control services 
- Academia services?

Long vs. short term archiving

- Version Control services 
- Academia services?
- Private internal repository (avoid silos, AirBnB case) airbnb / **knowledge-repo**

Long vs. short term archiving

- Version Control services 
- Academia services?
- Private internal repository (avoid silos, AirBnB case) airbnb / knowledge-repo
- Data archive services Archive.org, DataHub, Zenovo

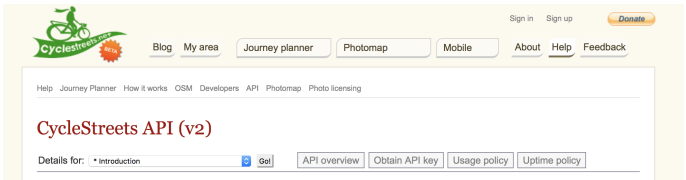
Long vs. short term archiving

- Version Control services 
- Academia services?
- Private internal repository (avoid silos, AirBnB case) airbnb / knowledge-repo
- Data archive services Archive.org, DataHub, Zenovo

Case

1. Corrupt CSV file after unzipping
   Had to upload the raw CSV file, 6Gb to archive.org to avoid unzipping problems

2. API
   External computations save intermediate results, before & after the API.



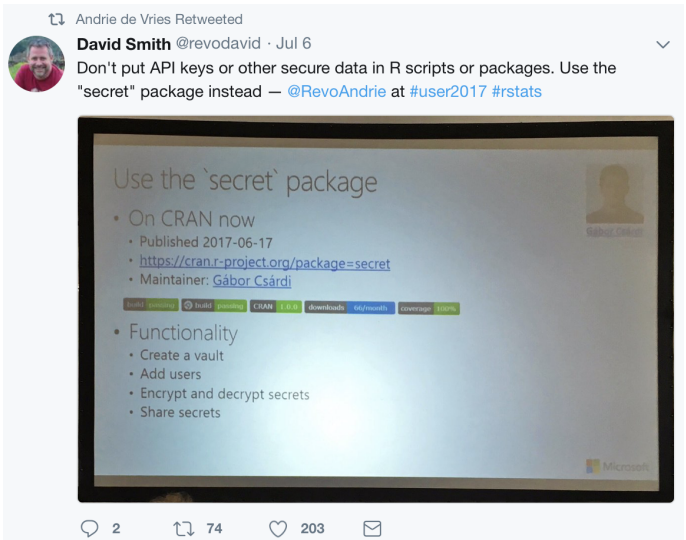
Index of /5/items/HACKATHONBOOKINGCALLABIKE/

../		
HACKATHONBOOKINGCALLABIKE_files.xml	17-Aug-2017 22:31	1.1K
HACKATHONBOOKINGCALLABIKE_meta.sqlite	15-Aug-2017 01:44	12.0K
HACKATHONBOOKINGCALLABIKE_meta.xml	17-Aug-2017 22:31	1.6K
HACKATHON_BOOKING_CALL_A_BIKE.csv	15-Aug-2017 01:42	6.0G

Case

API key never hardcoded! Two solutions:

1. Control your .Renviron
2. Check out the package `secret` by Gábor Csárdi [aut, cre], Andrie de Vries [aut]



.Renviron
To set global variables and or set API constants i.e. `Sys.getenv('CYCLESTREET')`

```
# Execute the command at the R console
file.edit('~/.Renviron')
```

```
# And then add your keys
CYCLESTREET=this_is_my_ip_secret
```

R Software

Session

- The environment is 
- Always include the session information in the documentation.

```
sessionInfo()
```

```
## R version 3.4.2 (2017-09-28)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS High Sierra 10.13.1
##
## Matrix products: default
## BLAS: /System/Library/Frameworks/Accelerate.framework/Versions/A/Frameworks/vecLib.framework/Versions/
## LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] tools     stats     graphics  grDevices utils     datasets  methods
## [8] base
##
```

Session

- Or with package `devtools`

```
devtools::session_info()
```

```
##  setting  value
##  version  R version 3.4.2 (2017-09-28)
##  system   x86_64, darwin15.6.0
##  ui       RStudio (1.1.345)
##  language (EN)
##  collate  en_US.UTF-8
##  tz       <NA>
##  date     2017-11-27
##
##  package     * version date       source
##  assertthat    0.2.0   2017-04-11 CRAN (R 3.4.0)
##  backports     1.1.0   2017-05-22 CRAN (R 3.4.0)
##  base        * 3.4.2   2017-10-04 local
##  bindr         0.1     2016-11-13 CRAN (R 3.4.0)
##  bindrcpp      0.2     2017-06-17 CRAN (R 3.4.0)
##  checkpoint  * 0.4.1   2017-06-26 CRAN (R 3.4.1)
##  compiler      3.4.2   2017-10-04 local
##  datasets    * 3.4.2   2017-10-04 local
```

Session

- *Advanced!*
  Package containerit

Markdown R

- Use Rmarkdown, jupiter notebooks or any other form of literate programming

- Use relative paths in favor of absolute paths

Relative

```
file.path("./LICENSE")
```

```
## [1] "./LICENSE"
```

Absolute

```
library(tools)
file_path_as_absolute("./LICENSE")
```

```
## [1] "/Users/RDynamics/Documents/R_folder/budape
```

Package versioning

- Package `Packrat` by RStudio

- Checkpoint by Microsoft

```
library(checkpoint)
checkpoint("2017-09-01")
```

Out of scope but important

1. Unit testing (code and data)
2. Code coverage
3. Continuous Integration / Continuous Deployment
4. …

Collaboration

Version control platforms

- Choose the best platform for the team.
- Add a rich `README.md` with an overview of the analysis
- 3rd party data upload. Check legal considerations!

34 lines (21 sloc) | 1.93 KB — Raw | Blame | History

pkgdown

build passing | CRAN not published | coverage unknown

pkgdown is designed to make it quick and easy to build a website for your package. You can see pkgdown in action at http://hadley.github.io/pkgdown/: this is the output of pkgdown applied to the latest version of pkgdown. Learn more in `vignette("pkgdown")` or `?build_site`.

Installation

pkgdown is not currently available from CRAN, but you can install the development version from github with:

```
# install.packages("devtools")
devtools::install_github("hadley/pkgdown")
```

Usage

Run pkgdown from the package directory each time you release your package:

```
pkgdown::build_site()
```

Structure

- Folder structure

```
.
|-- CITATION
|-- README
|-- LICENSE
|-- requirements.txt
|-- data
|    |-- birds_count_table.csv
|-- doc
|    |-- notebook.md
|    |-- manuscript.md
|    |-- changelog.txt
|-- results
|    |-- summarized_results.csv
|-- src
|    |-- sightings_analysis.py
|    |-- runall.py
```

from Good enough practices in scientific computing

- Naming conventions

The current state of naming conventions in R - UseR 2017 - YouTube

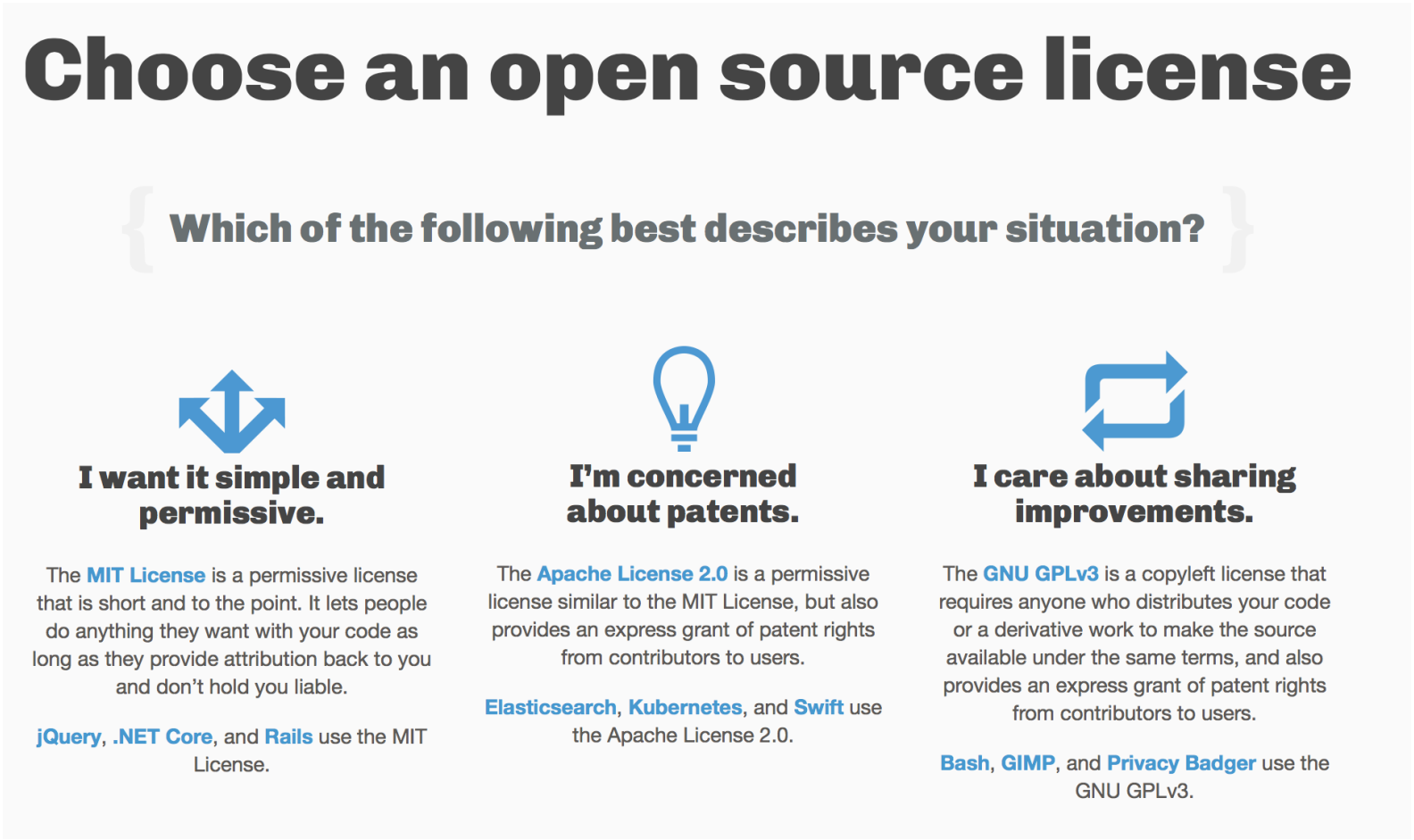
https://www.youtube.com/watch?v=Pv5dfsHBBKE

Jul 14, 2017 - Uploaded by rasmusab

This is a lightning talk I held at the **UseR 2017** conference in Brussels. I talk about **the current state of naming** ...

Rasmus Bååth 5 minutes video at User2017!

Licensing


Choose an open source license
Which of the following best describes your situation?
I want it simple and permissive.
The MIT License is a permissive license that is short and to the point. It lets people do anything they want with your code as long as they provide attribution back to you and don't hold you liable.
jQuery, .NET Core, and Rails use the MIT License.
I'm concerned about patents.
The Apache License 2.0 is a permissive license similar to the MIT License, but also provides an express grant of patent rights from contributors to users.
Elasticsearch, Kubernetes, and Swift use the Apache License 2.0.
I care about sharing improvements.
The GNU GPLv3 is a copyleft license that requires anyone who distributes your code or a derivative work to make the source available under the same terms, and also provides an express grant of patent rights from contributors to users.
Bash, GIMP, and Privacy Badger use the GNU GPLv3.

Cloud computing

Out of scope...but important!

- Calculations were done in Azure Data Science Virtual Machine on a `CentOS` linux distribution.
- Since I didn't use docker...the GIS packages have funny Unix library dependencies.

```
sudo yum update
sudo yum install gdal
sudo yum install proj-devel
sudo yum install proj-nad
sudo yum install proj-epsg
sudo yum install geos-devel
```

- MRAN is set up to 2017-09-01

```
library(checkpoint)
checkpoint("2017-09-01")
```

Reproducible Presentations

Different resources:

- Xaringan (RMarkdown presentation) <- This presentation!
- Rpres from RStudio
- Slidify

Summary

Data, Code, Environment & Documentation

Q&A

Reliable Dynamics

Carles CG
Data Scientist & freelanceR

carles@reliabledynamics.com
@carles_

Extra

More resources

- A Simple Explanation for the Replication Crisis in Science
- Good enough practices in scientific computing
- Package `rrtools`.
- Reproducibility guide

Aknowledgements

- Thanks to David Gohel (@DavidGohel) for sharing his presentation on GitHub. It made my intro to Xaringan much much easy!