Tidysq for Working with Biological Sequence Data in ML Driven Epitope Prediction in Cancer Immunotherapy

Leon Eyrich Jessen, PhD
Assistant Professor
Immunoinformatics and Machine Learning Group
Section for Bioinformatics
Department of Health Technology
Technical University of Denmark

Why R? Conference 2019
Warsaw, Poland
September 26th – 29th 2019
Sessions GEO, BIO 1, Room B, 15:05 - 15:25
Saturday September 28th 2019

Thank you to the organisers for inviting me!

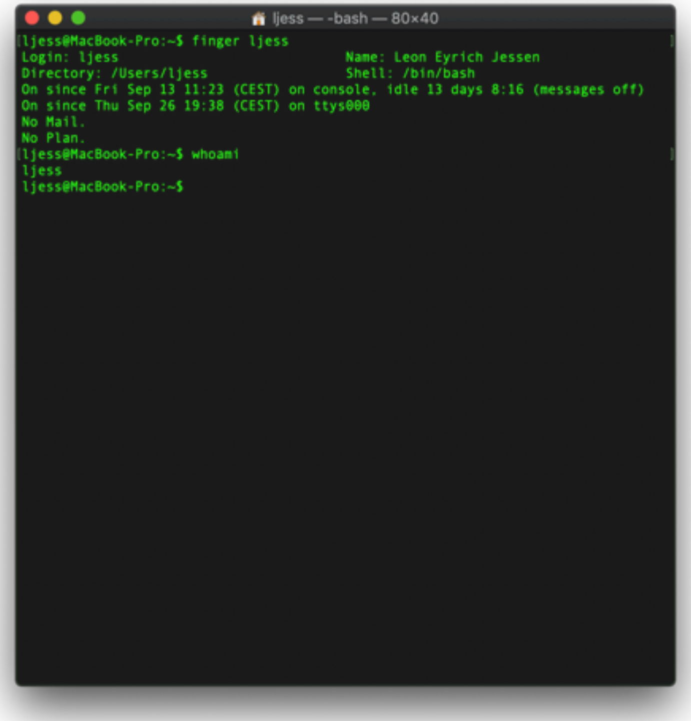
Technical University of Denmark

Part I

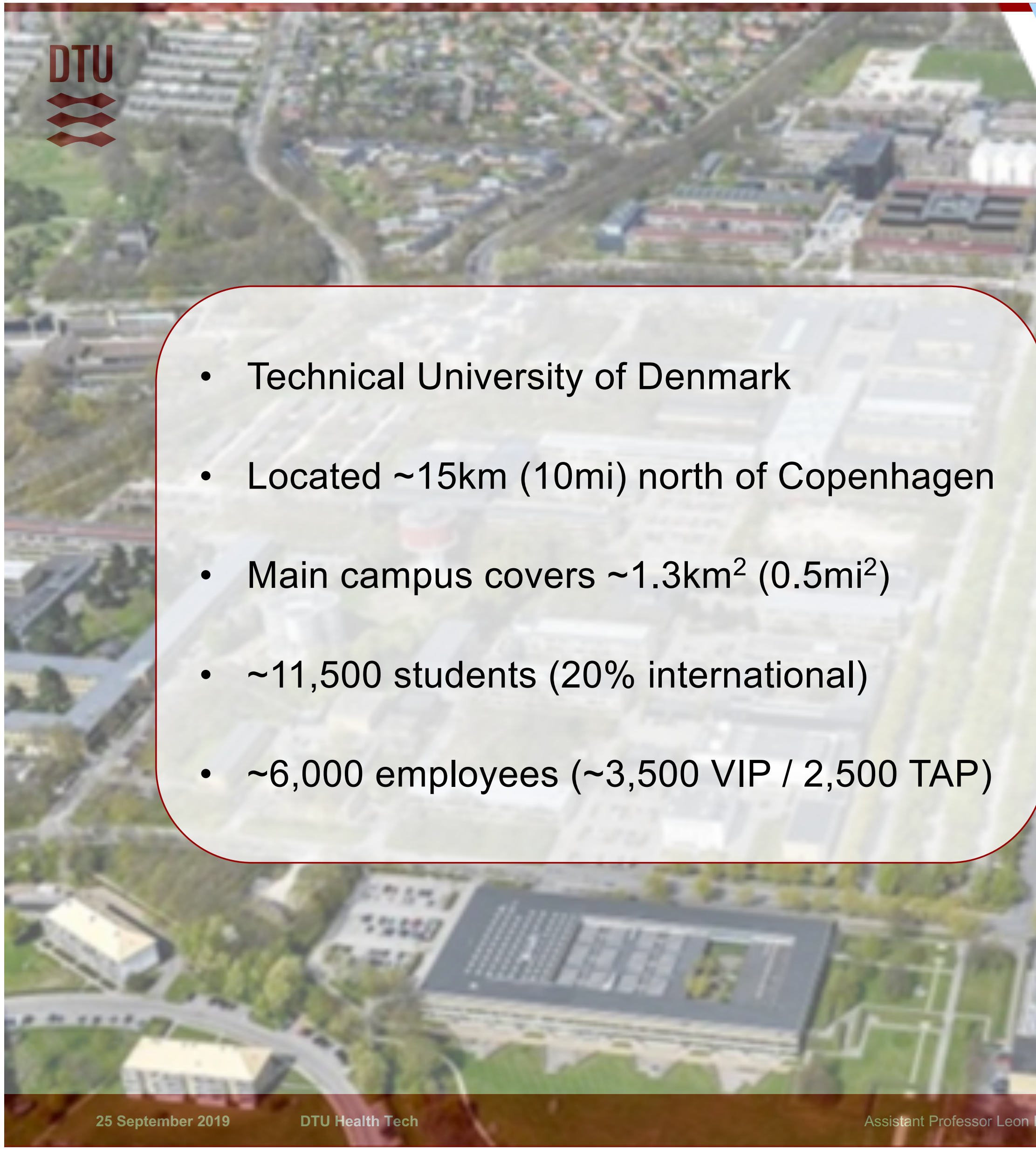
whoami

whoami

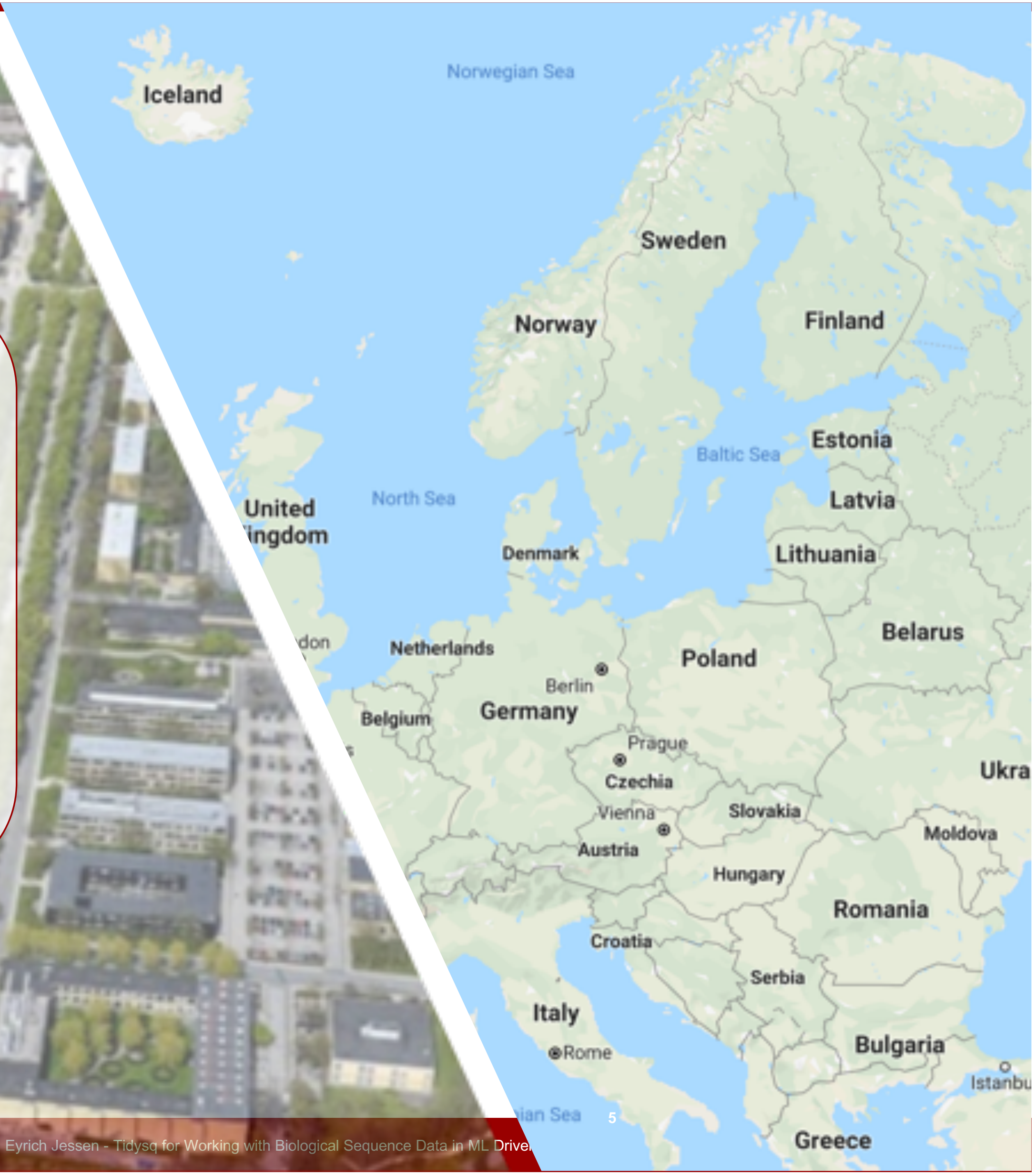
- Leon Eyrich Jessen
- BSc/MSc in biotech engineering
- PhD in Bioinformatics
- 3 Postdocs spanning clinical research (genetics/genomics) and machine learning
- Assistant professor of bioinformatics in the immunoinformatics and ML group
- Co-founder and CEO of nordicdatalab (Modern Data Science in R Training)



```
ljess@MacBook-Pro:~$ finger ljess
Login: ljess                            Name: Leon Eyrich Jessen
Directory: /Users/ljess                 Shell: /bin/bash
On since Fri Sep 13 11:23 (CEST) on console, idle 13 days 8:16 (messages off)
On since Thu Sep 26 19:38 (CEST) on ttys000
No Mail.
No Plan.
ljess@MacBook-Pro:~$ whoami
ljess
ljess@MacBook-Pro:~$
```

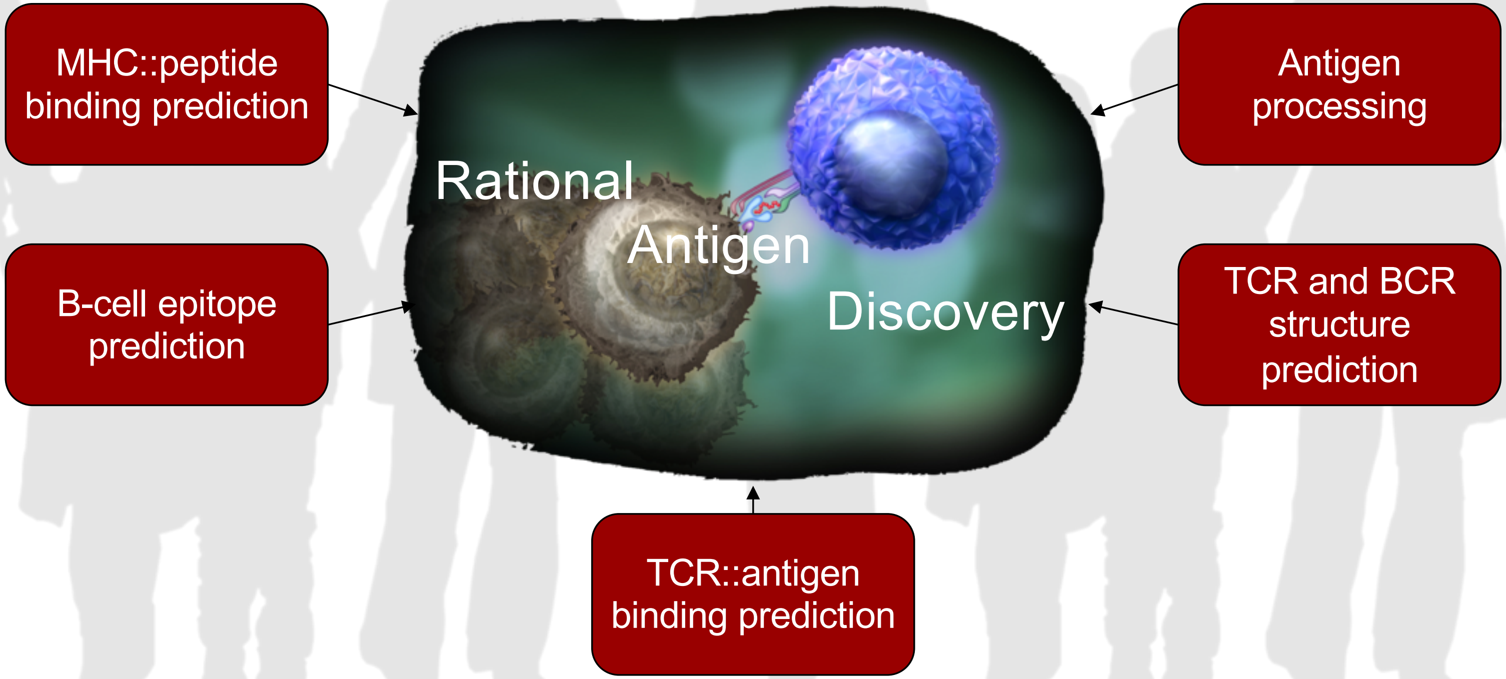


- Technical University of Denmark
- Located ~15km (10mi) north of Copenhagen
- Main campus covers ~1.3km^2 (0.5mi^2)
- ~11,500 students (20% international)
- ~6,000 employees (~3,500 VIP / 2,500 TAP)



“Development and application of mathematical models for important players of the human immune system”

From protein sequence to biomedical insights



Rational Antigen Discovery

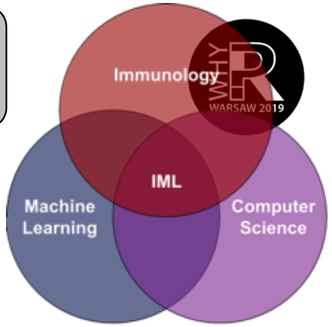
MHC::peptide binding prediction

B-cell epitope prediction

Antigen processing

TCR and BCR structure prediction

TCR::antigen binding prediction



<u>Pred servers</u>

NetMHC

NetMHCpan

NetMHCIIpan

Bepipred

Discotope

NetTCR

Lyra

TCRpMHC

NetChop

NetCTLpan

Immunoinformatics and Machine Learning Group

Part II

Okay, so why R?

Okay, so why R? (in ML)

- “I just feel that R is the wrong language for machine learning!”
- “No, I don’t use R, I mean I use ggplot, but not R!”
- ”R cannot be used in production and does not scale”
- “R was build for and by statisticians and that really shows”
- “Really, anything serious computing should be done in [Insert name of constrictor snake]”
- ”R is SO slow, try doing a nested for loop over elements in a tensor”
- “R has as many syntaxes and modes as there are packages” (Well, that’s kind of true)

Okay, so why R? (in ML)

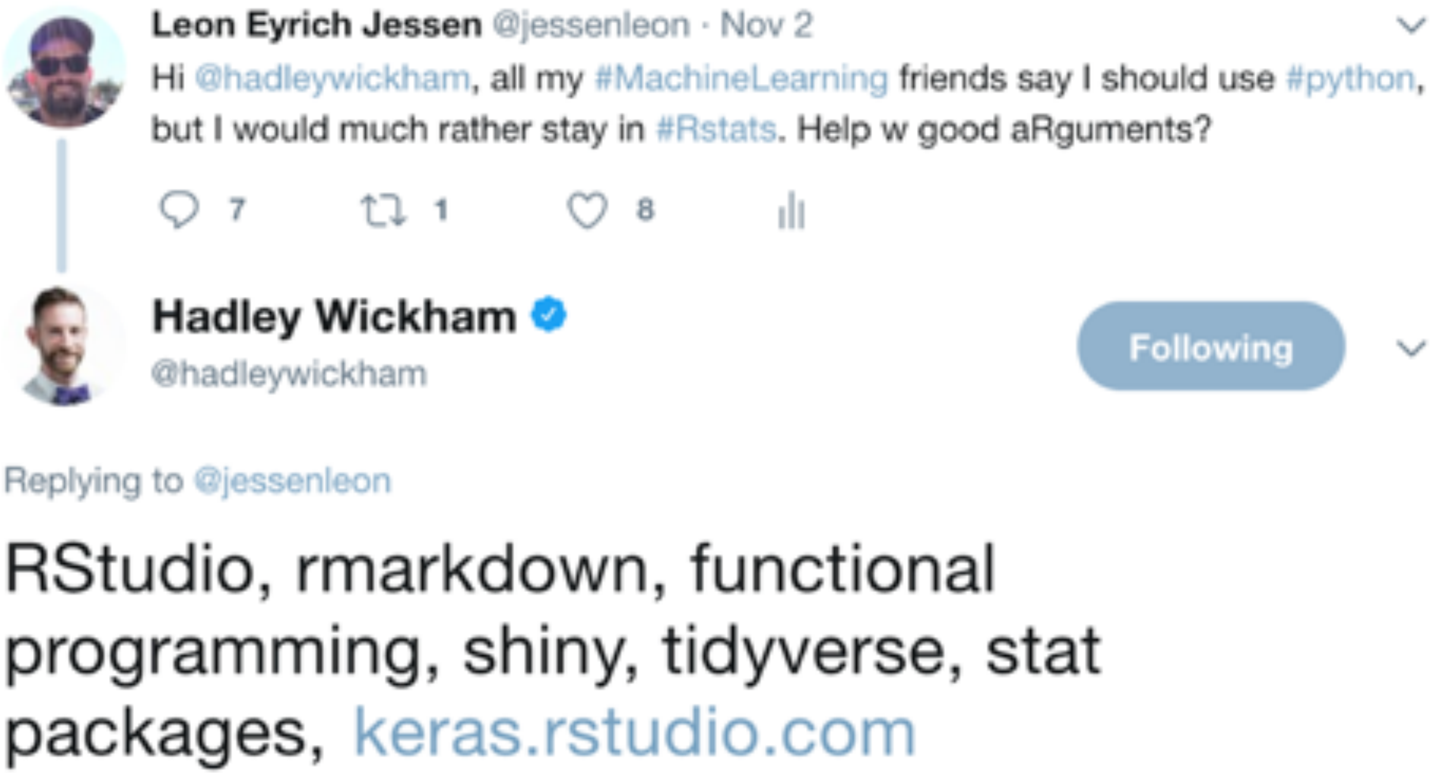
- “I just feel that R is the wrong language for machine learning!”
- “No, I don’t use R, I mean I use ggplot, but not R!”
- ”R cannot be used in production and does not scale”
- “R was build for and by statisticians and that really shows”
- “Really, anything serious computing should be done in [Insert name of constrictor snake]”
- ”R is SO slow, try doing a nested for loop over elements in a tensor”
- “R has as many syntaxes and modes as there are packages” (Well, that’s kind of true)

Prejudice, misconception and/or wrong usage!

Okay, so why R? (in ML)

- Let’s ask… Someone…

Okay, so why R? (in ML)



Leon Eyrich Jessen @jessenleon · Nov 2

Hi @hadleywickham, all my #MachineLearning friends say I should use #python, but I would much rather stay in #Rstats. Help w good aRguments?

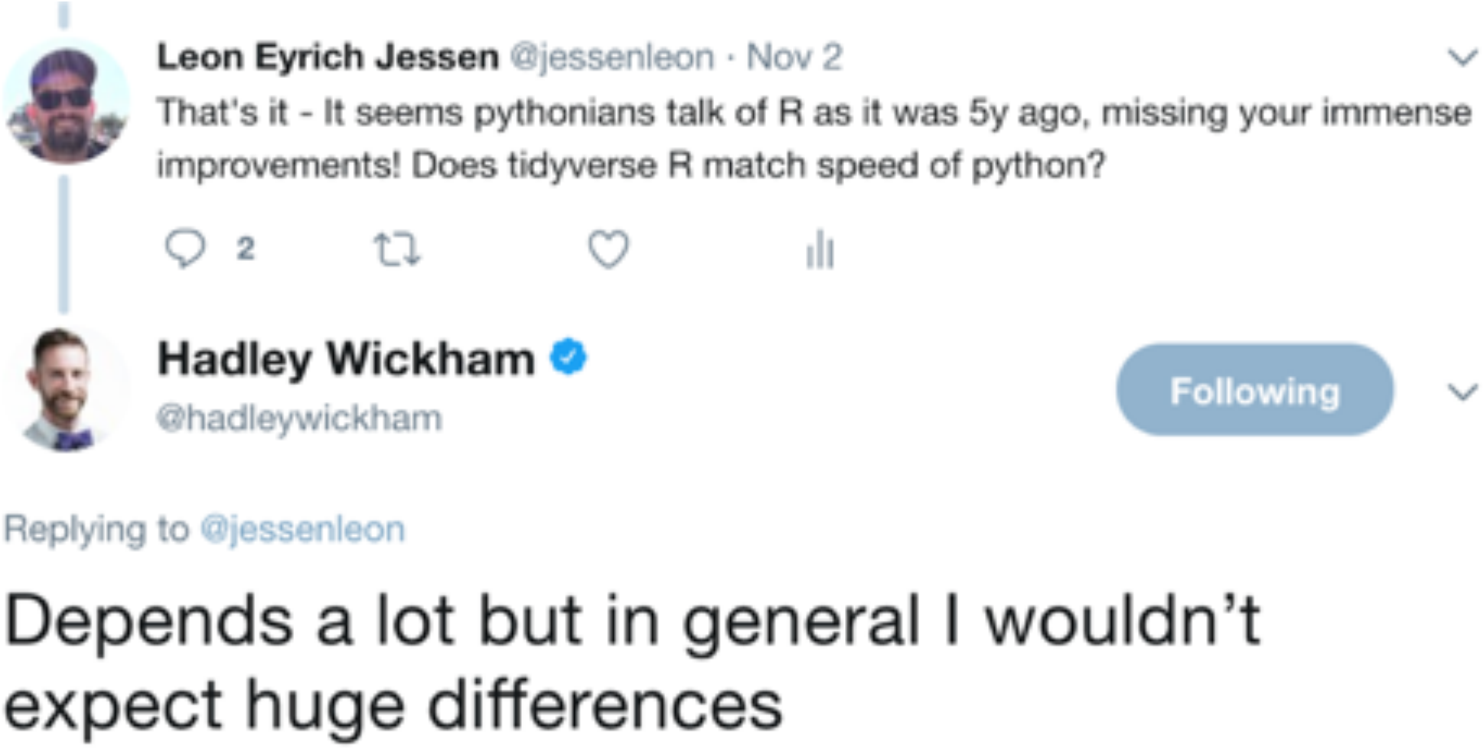
Hadley Wickham

@hadleywickham

Replying to @jessenleon

RStudio, rmarkdown, functional programming, shiny, tidyverse, stat packages, keras.rstudio.com

Okay, so why R? (in ML)



Leon Eyrich Jessen @jessenleon · Nov 2

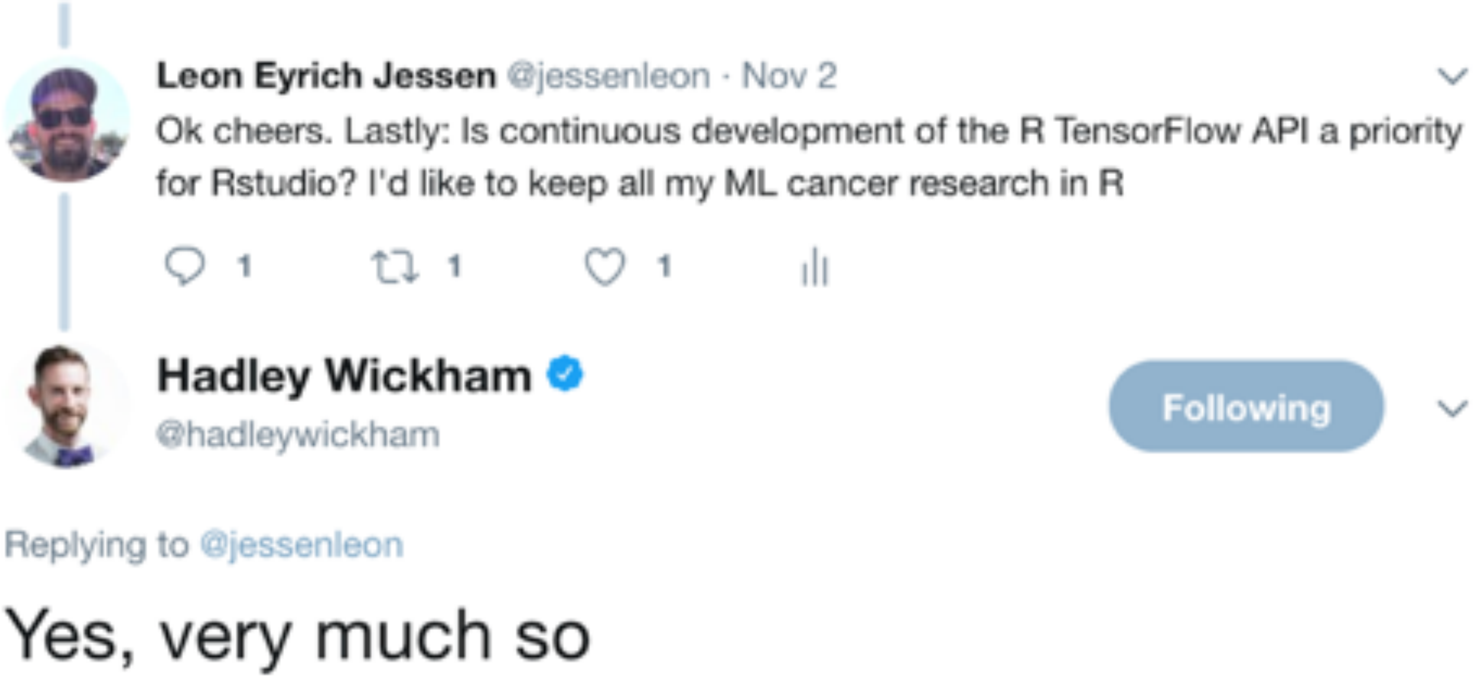
That's it - It seems pythonians talk of R as it was 5y ago, missing your immense improvements! Does tidyverse R match speed of python?

Hadley Wickham @hadleywickham

Replying to @jessenleon

Depends a lot but in general I wouldn't expect huge differences

Okay, so why R? (in ML)



Leon Eyrich Jessen @jessenleon · Nov 2

Ok cheers. Lastly: Is continuous development of the R TensorFlow API a priority for Rstudio? I'd like to keep all my ML cancer research in R

Hadley Wickham
@hadleywickham

Replying to @jessenleon

Yes, very much so

Bottomline

- R offers a full framework for reproducible end-to-end data science
- Rstudio is an extremely powerful productivity-increasing IDE
- The past ~3 years, R has become more unified and extended with state-of-the-art machine learning frameworks
- We all know this!
- Some people just have yet to discover! (…and that can be a biiit tiring)

Part III

Tidysq

We are amidst a data revolution

- Just the past 5 years, the cost of sequencing a human genome has gone down approximately 10-fold
- This development moves equally fast within areas such as mass spectrometry, in vitro immuno-peptide screening a.o.
- This facilitates the search for bio-markers, biologics, therapeutics, etc. but also redefines the requirements for storing, accessing and working with data

Tidysq - Work in Progress

- The aim of tidysq is to adapt the design philosophy, grammar, and data structures of the tidyverse to biological sequence data
- Thereby accessing the plethora of tidyverse tools available for application to sequence data
- With an emphasis on over-coming R object size challenges to allow laptop analysis of big NGS (and alike) data sets
- NB! What I am presenting here today, is work-in-progress

First step: Reading sequence data

- Standard bio-sequence format FASTA looks like so:
- Each sequence has an identifier > followed by the (multiline) sequence
- Obviously, not a standard row x column format

```
>AMY1|K19|T-Protein (Tau)
PGGGKVQIVYKPV
>AMY9|K19Gluc41|T-Protein (Tau)
NLKHQPGGGKVQIVYKPVDLSKVTSKCGSLGN
IHHKPGGGQVE
>AMY14|K19Gluc782|T-Protein
(Tau)
NLKHQPGGGKVQIVYKEVD
>AMY17|PHF8|T-Protein (Tau)
GKVQIVYK
>AMY18|PHF6|T-Protein (Tau)
VQIVYK
```

First step: Reading sequence data

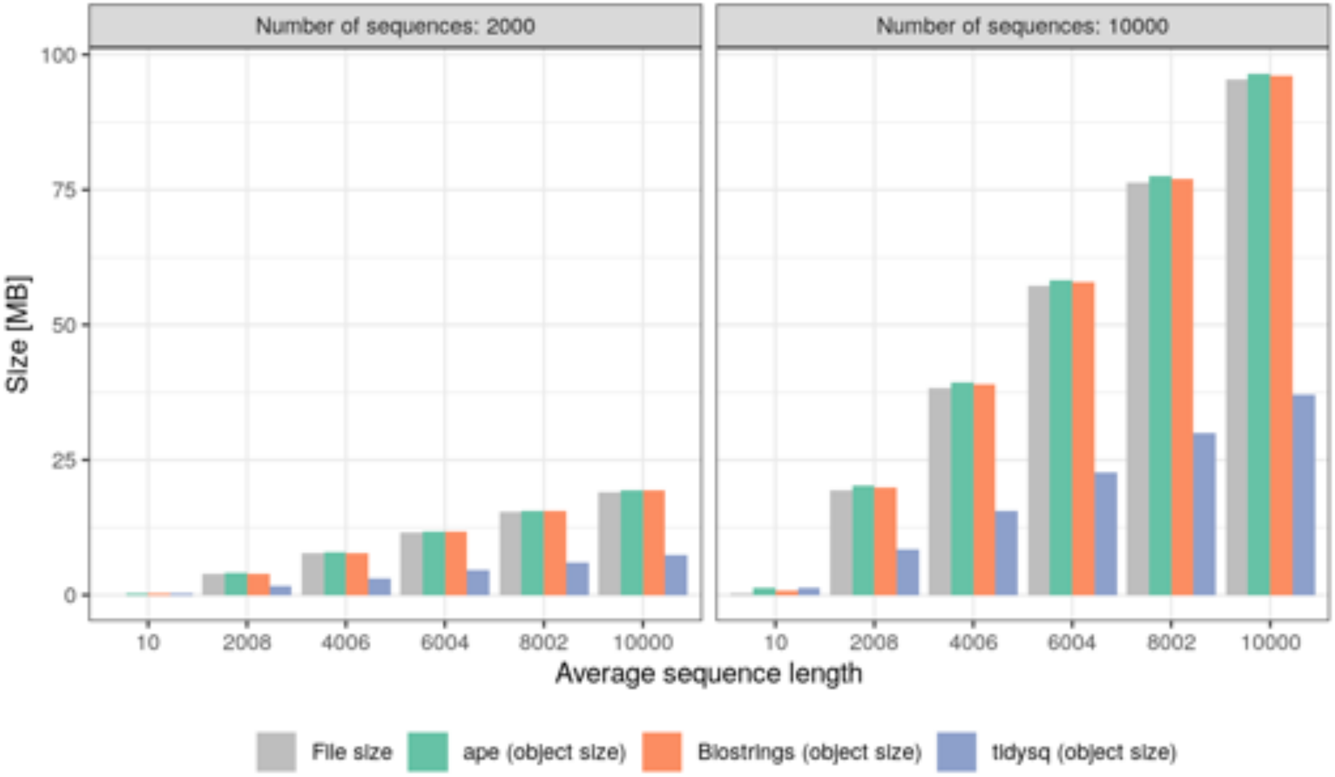
```
seq_dat = read_fasta(file = "data/seqs.fasta")
seq_dat
## # A tibble: 5 x 2
##   name                             sq
##   <chr>                            <(c)ami>
## 1 AMY1|K19|T-Protein (Tau)         PGGGKVQIVYKPV       <13>
## 2 AMY9|K19Gluc41|T-Protein (Tau)   NLKHQPGGGKVQIVY...  <43>
## 3 AMY14|K19Gluc782|T-Protein (Tau) NLKHQPGGGKVQIVY...  <19>
## 4 AMY17|PHF8|T-Protein (Tau)       GKVQIVYK             <8>
## 5 AMY18|PHF6|T-Protein (Tau)       VQIVYK               <6>
```

- FASTA file is split into two columns, the (observation) id and the sequence (variable)

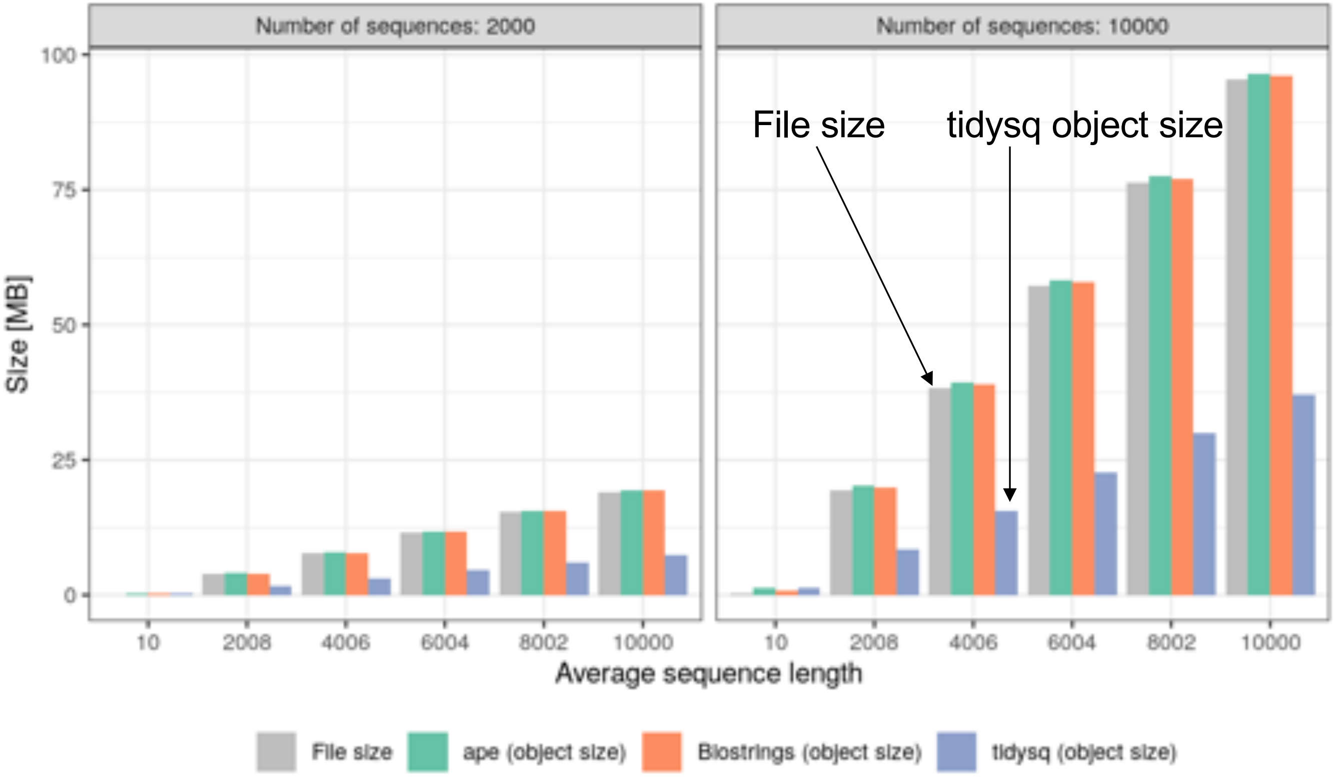
- So, tidy data in a tibble, that’s nice and familiar

- Note the special data type for the sequences, with a length indicator

Second step: Reducing data size



Second step: Reducing data size in memory



File size

tidysq object size

Find Motifs

- Implementing various sequence manipulation / extrapolation tools, capable of working on the compressed format

```
seq_dat %>% mutate(has_GG_motif = sq %has% "GG")
## # A tibble: 5 x 3
##   name                             sq                         has_GG_motif
##   <chr>                            <(c)ami>                   <lgl>
## 1 AMY1|K19|T-Protein (Tau)         PGGGKVQIVYKPV        <13> TRUE
## 2 AMY9|K19Gluc41|T-Protein (Tau)   NLKHQPGGGKVQIVY...   <43> TRUE
## 3 AMY14|K19Gluc782|T-Protein (Tau) NLKHQPGGGKVQIVY...   <19> TRUE
## 4 AMY17|PHF8|T-Protein (Tau)       GKVQIVYK              <8> FALSE
## 5 AMY18|PHF6|T-Protein (Tau)       VQIVYK                <6> FALSE
```

- Notice the custom %has% for (advanced) bio-motif search

- Also, notice the capability to be use tidysq with dplyr function

Advanced Motifs

- Implementing various sequence manipulation / extrapolation tools, capable of working on the compressed format

```
seq_dat %>% filter(sq %has% "^PXG")
## # A tibble: 1 x 2
##   name                     sq
##   <chr>                    <(c)ami>
## 1 AMY1|K19|T-Protein (Tau) PGGGKVQIVYKPV  <13>
```

Sequence Encoding

- For application in machine learning, we need to convert biology to a numerical representation

- This can be done using e.g. Hydropathy index (Kyte-Doolittle, 1982):

```
seq_dat  %>% encode(sq = sq, encoding = AAindex_norm["KYTJ820101",])
```

- Other encoding schemes, e.g. BLOSUM will follow

Export and import tidysq objects

- We are working on interfacing tidysq with ape, seqinr and Biostrings
- Aiming at enabling easy sequence manipulation / exploration in the tidyverse paradigm and then export to harvest the power of existing packages

Tidysq Team

- Open source (naturally) Tidysq beta is available at: https://github.com/michbur/tidysq
- Michal Burdukiewicz[1]
- Dominik Rafacz[1]
- Weronika Puchala[1]
- Filip Pietluch[1]
- Katarzyna Sidorczuk[1]
- Stefan Roediger[2]
- Leon Eyrich Jessen[3]

1. Warsaw University of Technology, Warsaw, Poland
2. Brandenburg University of Technology, Cottbus, Germany
3. Technical University of Denmark, Lyngby, Denmark

Part IV

ML Driven Epitope Prediction in Cancer Immunotherapy

Um actually…

- Now, I was supposed to talk about "ML Driven Epitope Prediction in Cancer Immunotherapy"

- But this I gave a keynote on at last years whyR:
  - http://rpubs.com/leonjessen/whyR_2018

- …and also you can read much more on my RStudio invited ML blogpost:
  - https://blogs.rstudio.com/tensorflow/posts/2018-01-29-dl-for-cancer-immunotherapy/

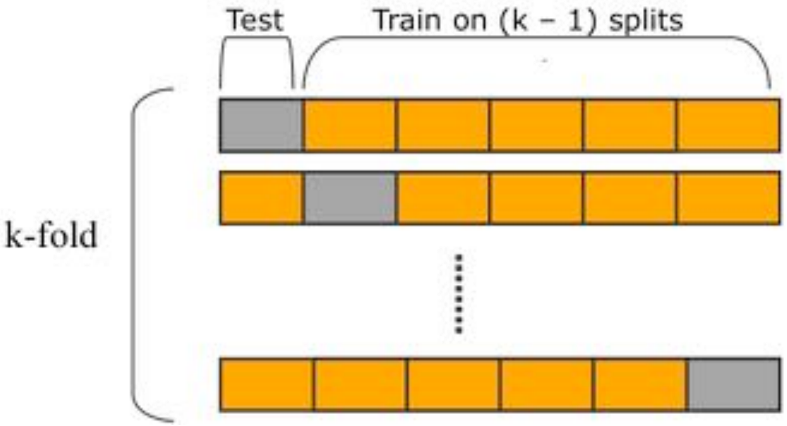
- So, instead…

Part IV

A Caveat of ML

“Then we randomly split the data in k-folds”

- “…and estimated the predictive performance of our model as etc.”
- Standard phrase in many many blog posts, tutorials, scientific papers and alike
- Using high level APIs various machine learning frameworks are readily available
- Just plug and play, choose your favourite model, test various hyperparameters, etc.
- …but what about the data? The data, not just the model, needs attention!



“Then we randomly split the data in k-folds”

- We easily spend 80% of a research project time on getting to love data: “You have to love your data, the order is:

Really

“Then we randomly split the data in k-folds”

- We easily spend 80% of a research project time on getting to love data: “You have to love your data, the order is:
  - Your wife/husband

Really

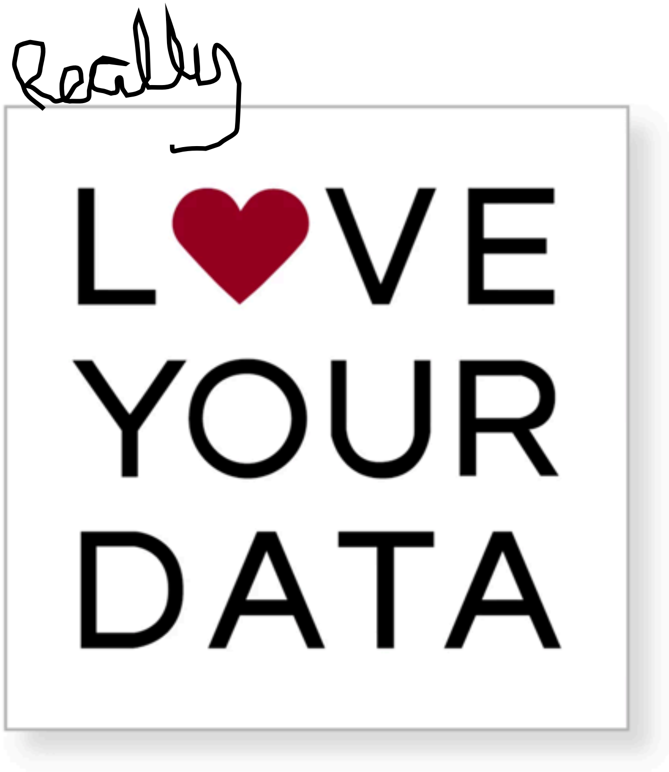
“Then we randomly split the data in k-folds”

- We easily spend 80% of a research project time on getting to love data: “You have to love your data, the order is:
  - Your wife/husband
  - Your kids

Really

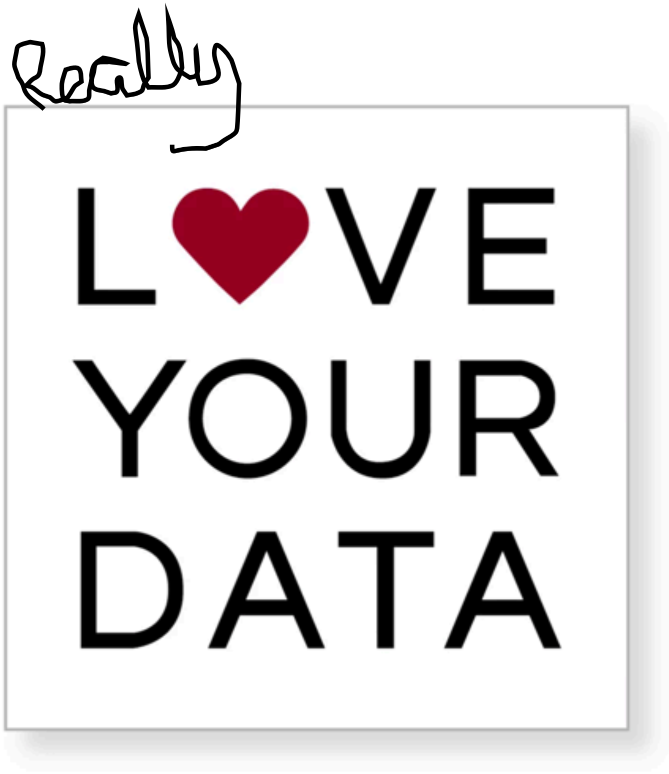
“Then we randomly split the data in k-folds”

- We easily spend 80% of a research project time on getting to love data: “You have to love your data, the order is:
  - Your wife/husband
  - Your kids
  - Then your data



“Then we randomly split the data in k-folds”

- We easily spend 80% of a research project time on getting to love data: “You have to love your data, the order is:
  - Your wife/husband
  - Your kids
  - Then your data
  - And other stuff: parents, family, friends, football, etc.


Really

Examples of data caveats

- Many repeats of similar data points
- Training data distribution (sample) is not representative of “natural” data (population)
- Really an issue in biology, model organisms, model proteins, etc. are often studied and studied and studied and that introduces an (“unnatural”) bias in the data
- Training performance even when using k-fold CV may be inflated compared with the actual extrapolatable mode prediction capabilities on true unseen data

Example

- Let us say you have some data

- You then decide on a scoring function using some similarity measure
  - If two data points are very similar, they are likely to have the same target value

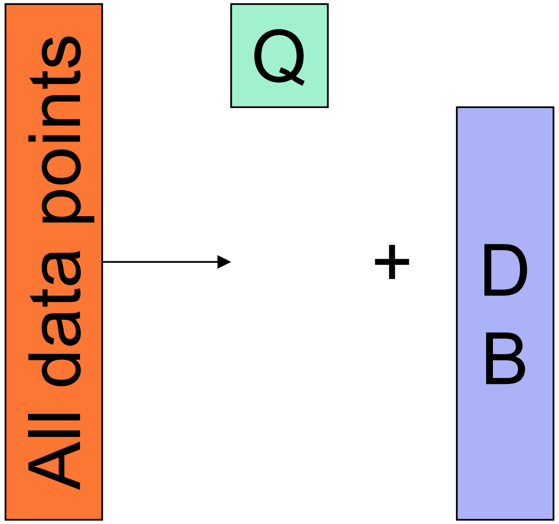
- You decide on a performance metric using a rank based approach
  - Given a split of the data, if for each data point in 1/5 of the data, the most similar data point in the 4/5 of the data has the right target value, then we’re happy!

- You then split your data randomly in 5 partitions and estimate the ability of your scoring measure to top-rank the correct target

Graphic illustration of scoring algorithm

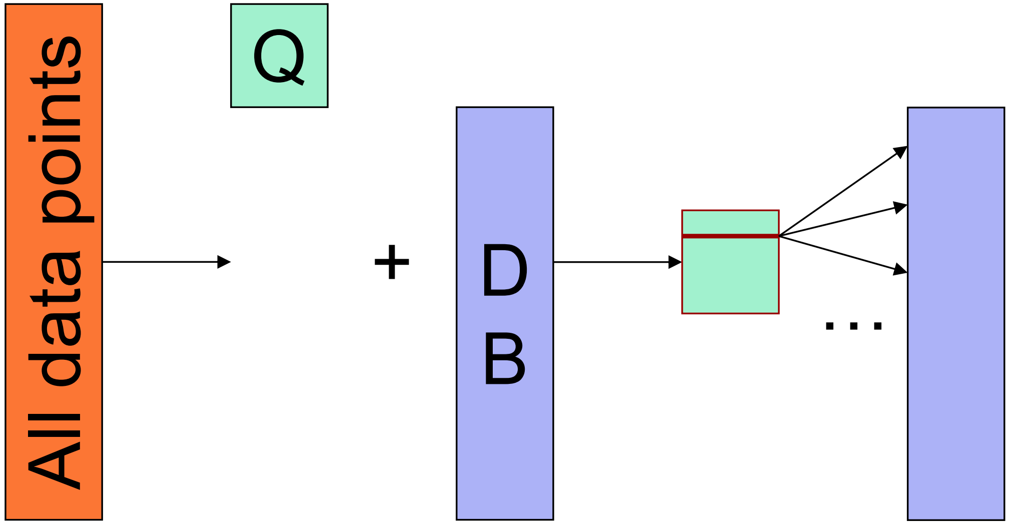
All data points

Graphic illustration of scoring algorithm



1. Split data

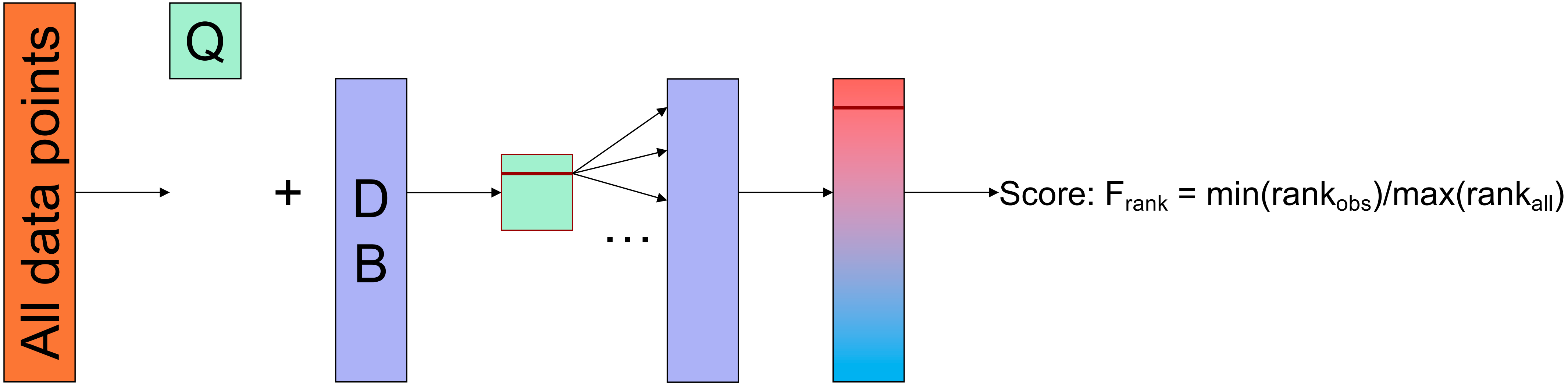
Graphic illustration of scoring algorithm



1. Split data

2. Score each DP in Q against all DPs in DB

Graphic illustration of scoring algorithm



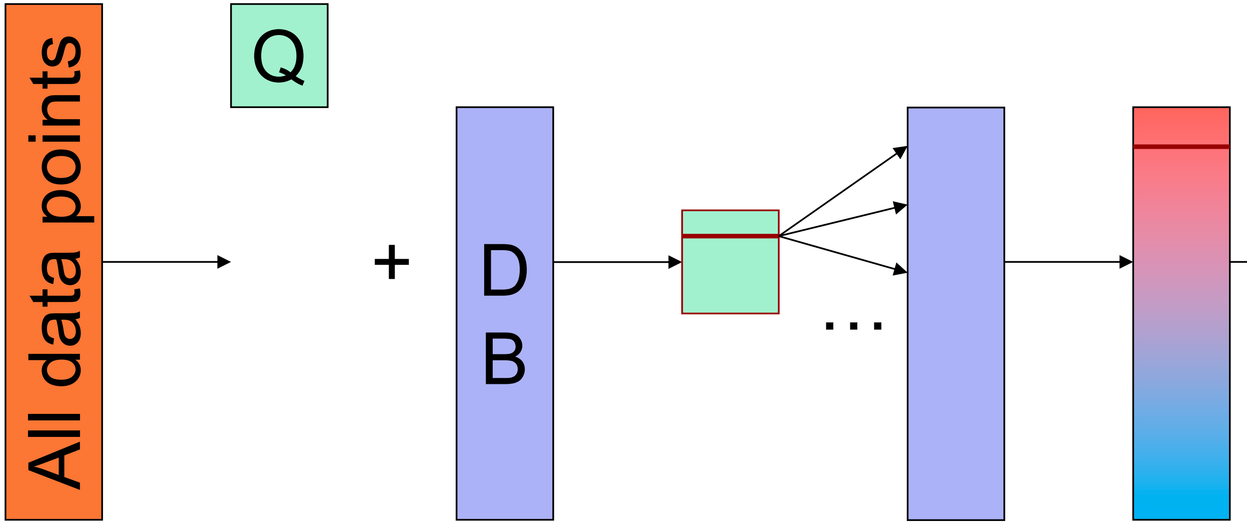
Score: $F_{rank} = \min(rank_{obs})/\max(rank_{all})$

1. Split data

2. Score each DP in Q against all DPs in DB

3. Rank scores and find the highest ranking DP with the right target and assign F_{rank} score

Graphic illustration of scoring algorithm



Score: $F_{rank} = \min(rank_{obs})/\max(rank_{all})$

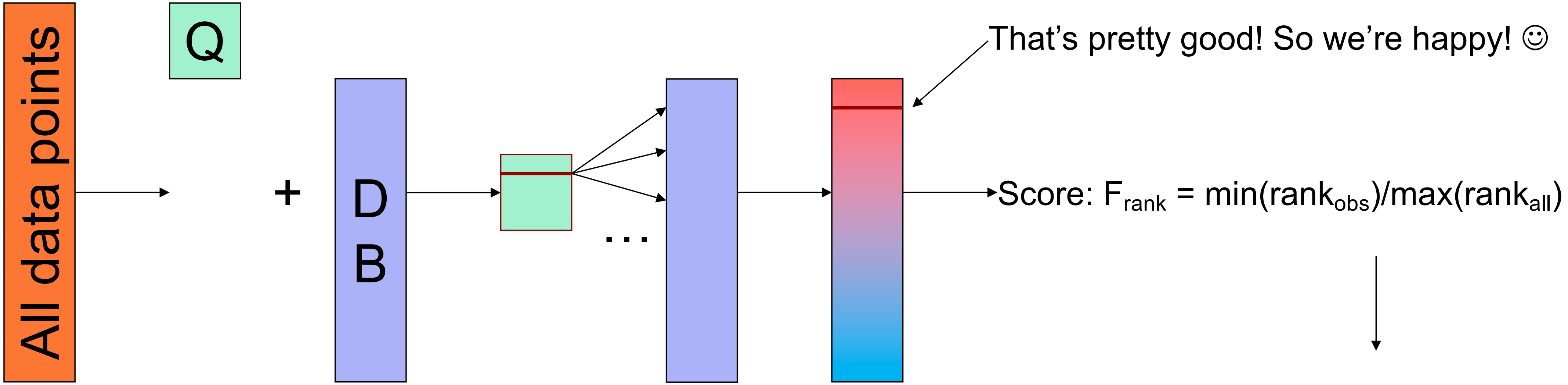
1. Split data

2. Score each DP in Q against all DPs in DB

3. Rank scores and find the highest ranking DP with the right target and assign F_{rank} score

4. Rotate and repeat for all splits of the data

Graphic illustration of scoring algorithm



That’s pretty good! So we’re happy! ☺

Score: $F_{rank} = \min(rank_{obs})/\max(rank_{all})$

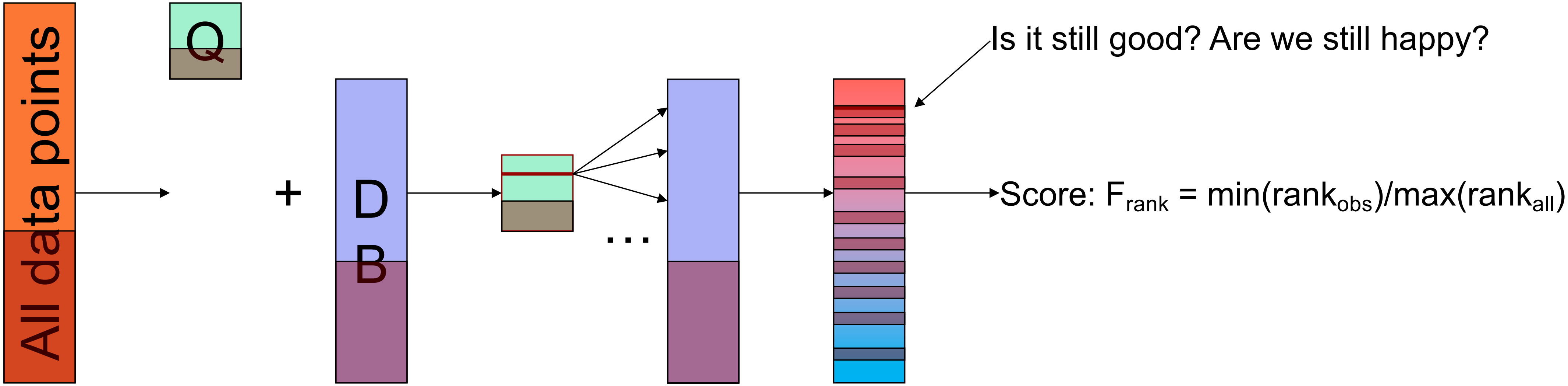
1. Split data

2. Score each DP in Q against all DPs in DB

3. Rank scores and find the highest ranking DP with the right target and assign F_{rank} score

4. Rotate and repeat for all splits of the data

Nice, but what if 40% of your data is “similar”?

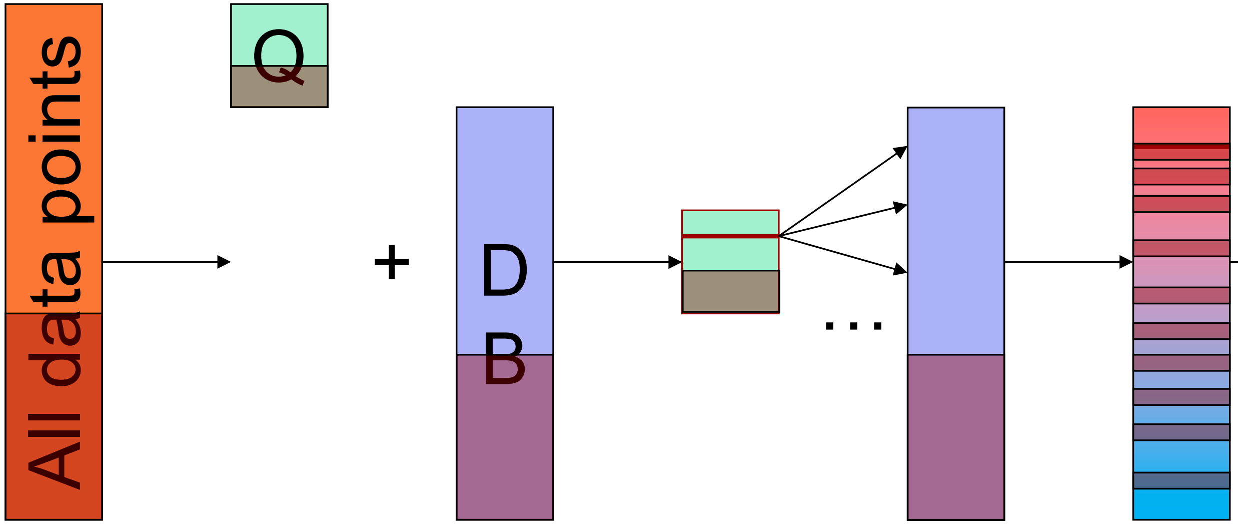


Is it still good? Are we still happy?

Score: F_{rank} = min($rank_{obs}$)/max($rank_{all}$)

What happened?

Nice, but what if 40% of your data is “similar”?



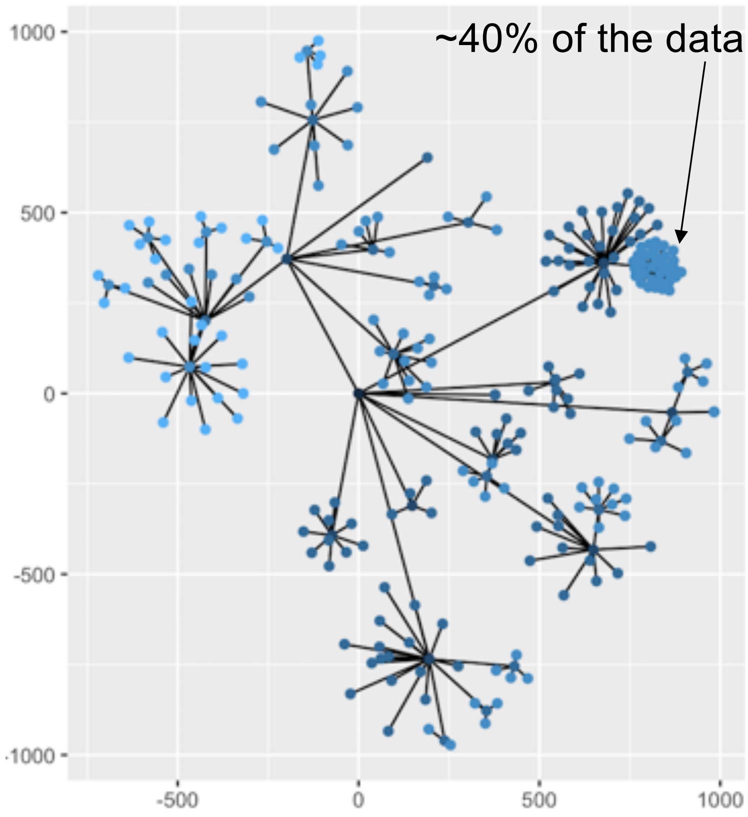
Is it still good? Are we still happy?
No, AUC ~0.5

Score: F_{rank} = min($rank_{obs}$)/max($rank_{all}$)

Then your nice rank score might simply be driven by the fact that higher abundance of similar data points leads to good scores as the probability of getting a good rank score by chance is dependent on data imbalance

The essence of this example…

- …is that if you simply randomise your data into 5 partitions, you risk “mixing” your training and your test data
- This leads to over-estimation of the predictive capabilities of your model
- Look for structure in your data and create partitions, which are as distinct as possible
- Consider what happens if you randomly select data points from this graph, versus taking the (feature) structure into account



Summary

- Part I
  - In my group we work with the development and application of mathematical models for important players of the human immune system, i.e. immunoinformatics

- Part II
  - R offers a full framework for reproducible end-to-end data science, that’s why R!

- Part III
  - Aim to adapt the design philosophy, grammar, and data structures of the tidyverse to biological sequence data enabling accessing the plethora of tidyverse tools available for application to bio-sequence data

- Part IV
  - Love your data, like really love it, so you understand what’s what and also, never underestimate the importance of domain specific knowledge

Acknowledgments

- The organisers for inviting me
- The tidysq team
- My group leader and research mentor Professor Morten Nielsen
- Funding agencies
- SoMe credentials

Twitter: jessenleon
GitHub: leonjessen
LinkedIn: leonjessen