Making your code faster:
コードを早く実行させている

Introduction to vectorisation and parallel computing
ベクトル化と並列計算の紹介する

#TokyoR 78th Meeting

2019-05-25

Tom Kelly

Postdoctoral Researcher

RIKEN Centre for Integrative Medical Sciences, Yokohama

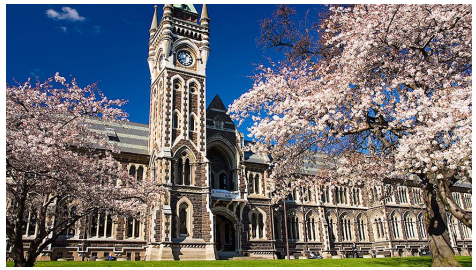
ケリー・トム

ポスドクで 研究者

国立研究開発法人理化学研究所の生命医科学研究センター、 横浜

自己紹介
ケリー・トム

- 名前はトム
- 苗字はケリー
- ニュージーランドから来た
- 二十七歳
- 妻の出身は東北
- 一年半ぐらい日本に住んでいた
(宮城県と神奈川県)



Twitter: @tomkXY

GitHub:TomKellyGenetics

自己紹介

ケリー・トム

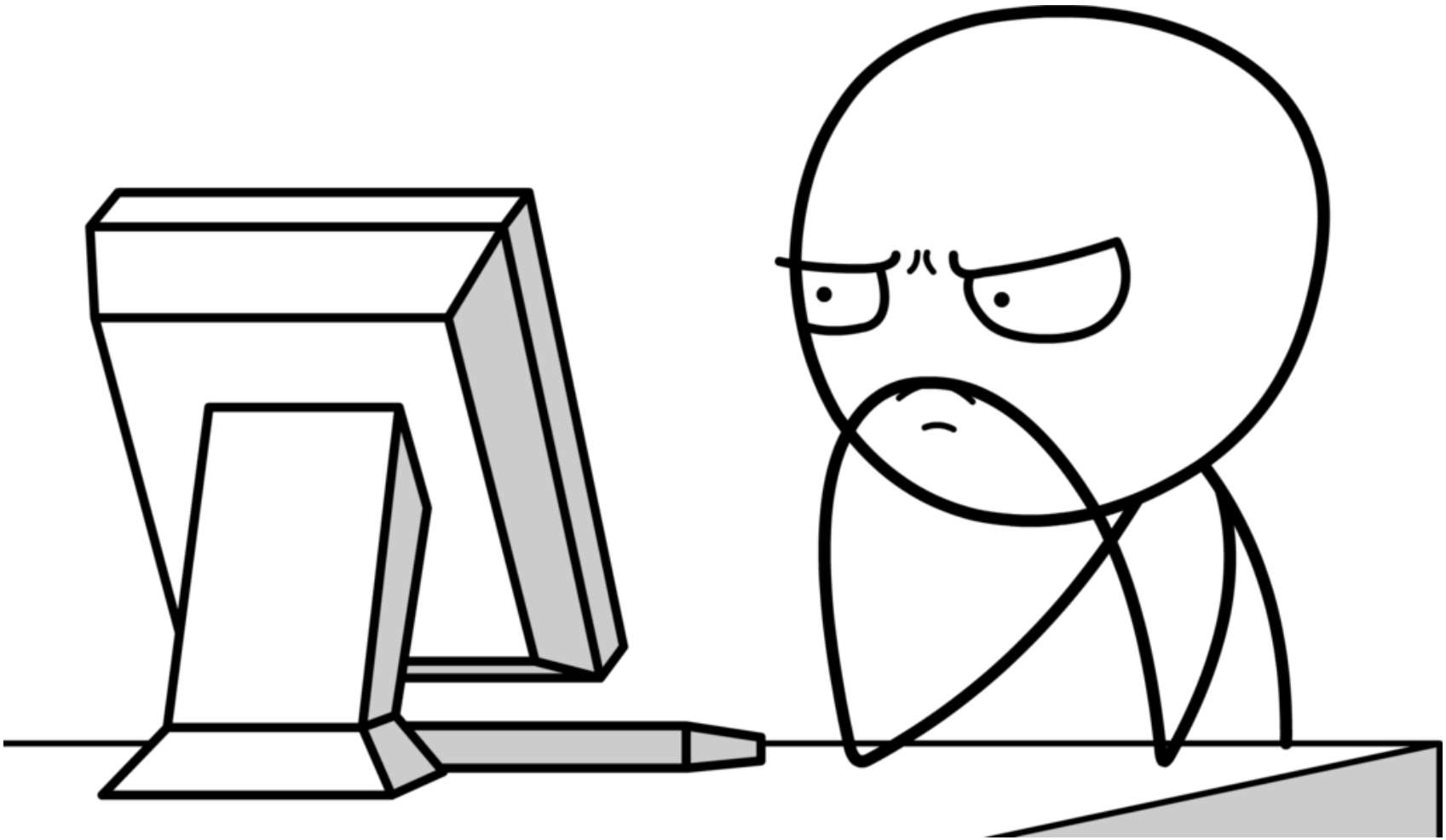
- 専攻は遺伝学と数学
- バイオインフォマティクスの研究者
- 主に統計解析を行う
- オタゴ大学で博士号を取った
- 横浜で理化学研究所でポスドク
- R言語を７年間ぐらい使っている

Twitter: @tomkXY

GitHub:TomKellyGenetics

R is too slow

R言語は遅いすぎる



R is too slow

R言語は遅いすぎる

How to make your code Faster

- Develop automated complex tasks with *Loops* and then optimise code (Loops are slow)
- Create your own *functions* to automate tasks (remove human-error)
- Use built-in *vectorised* and apply functions to process vectors, matrices, and lists more efficiently
- Pass functions to other languages (e.g., use C++ with Rcpp)
- Run independent tasks in *parallel*
- Use remote servers, clusters, and high-performance computing (HPC)

Key point

Running a task with multiple inputs
ダメです

- Don't copy-paste! (introduce human errors)
- Only use loops when needed
- Avoid "premature optimisation" (code first, speed up later)

オケです

- Write a function when you need to do something more than once
  - run a function with different inputs
  - share functions in packages
- evaluate in all elements at once
  - vectorised functions
  - the "apply" functions (and `plyr` package)
- *parallel* computing with `snow` package
  - set up dependent nodes (SOCKS/openmpi)
  - export input objects to cluster
  - run function in parallel (non-sequential) as multiple "threads"

Demonstration
デモ

Parallel Computing in R

Take home messages

How to make your code Faster

- Identify "bottleneck" points and optimise only this code
- Concepts in parallel computing can be applied to other package or languages languages (e.g., `dopar`, GNU parallel)
- Embarrassingly parallel processes *do not* run in order
- Parallel computing is *not always* faster (due to cluster set up "overheads" and communication)

Demonstration codes will be released on GitHub as a Rmarkdown document: `TomKellyGeneticsTokyoR78`

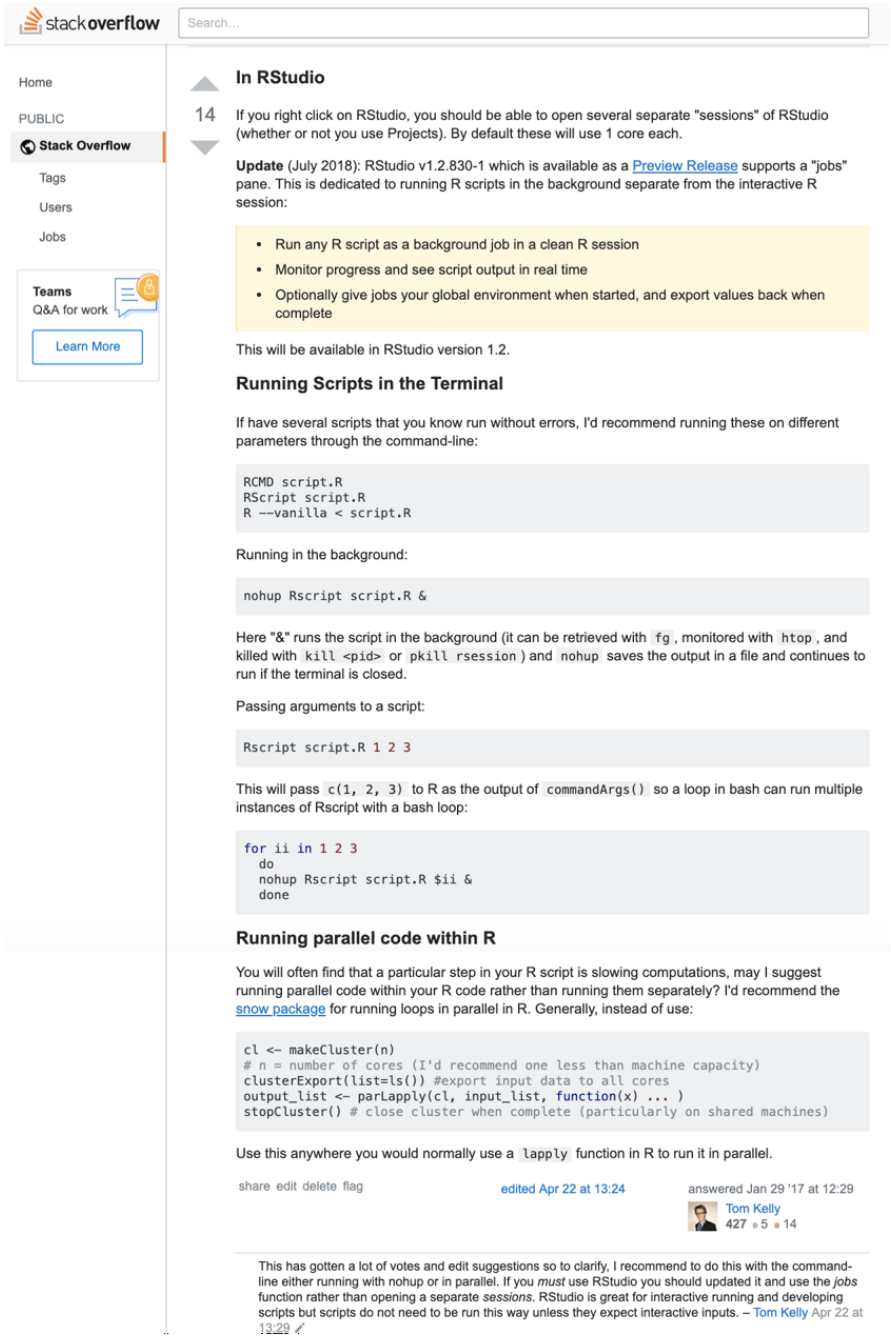
Automating tasks in R

How to run functions at the same time

- Running R in the background
  - RStudio jobs pane
  - running as a script (nohup)
  - pass arguments to script: `commandargs()` in R, `"$@"` in bash
- Submit jobs to a server

  "The Cloud is just someone else's computer"
  - Run on a local server or remote cluster (ssh)
  - Move input data to server (rsync or scp)
  - Make sure dependancies (packages) are installed
  - Queue jobs to the server with a scheduler
    - LoadLeveler: llq, llsubmit, llcancel
    - Slurm: squeue, srun, scancel
    - SGE: qstat, qsub, qdel
  - Run parallel scripts
  - Dependancy jobs (run after other jobs have finished)

See StackOverflow answer for more details



https://stackoverflow.com/questions/31137842/

Running R on remote systems

Take home messages

How to make your code Faster

- Ask about servers/clusters available and take opportunity to learn use them
- Be careful of "pre-mature optimisation"
- Automate tasks to save you work and perform reproducible analysis
- Run small "test" jobs to check it will run without errors
- Order of executing tasks is important
- CPU-hours cost money (but not as much as human-hours)

Demonstration codes will be released on GitHub as a Rmarkdown document

What is “Software Carpentry”?
「ソフトウェア・カーペントリー」は何ですか？

- A non-profit foundation based in the USA
  アメリカにある非営利団体
- A collection of collaboratively maintained lessons
  協働でレッスンを維持
- A 2-3 day series of hands-on workshops on tools for researchers
  研究者向けツールのハンズオン(体験型)ワークショップを2・3日間で教える
- A global community of instructors and member organisations
  指導者とメンバーからなる組織の国際的なコミュニティー

What is "Software Carpentry"
「ソフトウェア・カーペントリー」は何ですか

Organisations want to build research capacity
組織は研究能力を構築したい

- Consulting doesn't scale
  コンサルティングはスケールができない
- Tech support can't help with all research tools
  技術サポートは全部の研究ツールを手伝えない
- You get frustrated and isolated in online courses (MOOCs)
  オンラインコースでイライラして孤独になる

We aim to build a community and peer support
コミュニティとピアサポートを構築することを目指している

Please join “Software Carpentry”
「ソフトウェア・カーペントリー」参加してください

- To join the organisation
  組織に参加するには
  https://carpentries.org/join/
  - GitHub
    https://github.com/swcarpentry
  - Mailing list (Topicbox) メール
    https://carpentries.topicbox.com
- To become an instructor (online or in-person training)
  インストラクターになるには(ネットでか人で)
  https://carpentries.org/become-instructor/
- To help with the Japanese translations (contact me)
  日本語の翻訳を手伝うには(私に連絡)
  https://github.com/TomKellyGenetics
  tom.kelly@riken.jp @tomkXY

What would we need to do this?

「ソフトウェア・カーペントリー」をするには何が必要ですか？

Please join “Software Carpentry”
「ソフトウェア・カーペントリー」参加してください

- Japanese translation of the lessons
  レッスンを日本語に翻訳する
  demo: https://tomkellygenetics.github.io/git-novice/ja/index/index.html
  - Team to translate core lessons
    コアレッスンを翻訳するチーム
  - Maintainers to keep it up to date
    メンテナは最新の状態に保つ
- Instructors in Japan (who speak Japanese)
  日本でインストラクター(英語や日本語を話せる)
- Support from institutions
  学会からのサポート
- Then we can plan a workshop or conference
  そして、ワークショップや会議を計画できる

Please consider to volunteer

やってみたいかたはいませんか？

Contact: tom.kelly@riken.jp Twitter: @tomkXY

GitHub: TomKellyGenetics

Let's do our best for the community

コミュニティーのために頑張りましょう！

Thank you for your attention

よろしくお願いいたします

software carpentry