Analyzing TCGA data in the cloud

what you can do in just one day

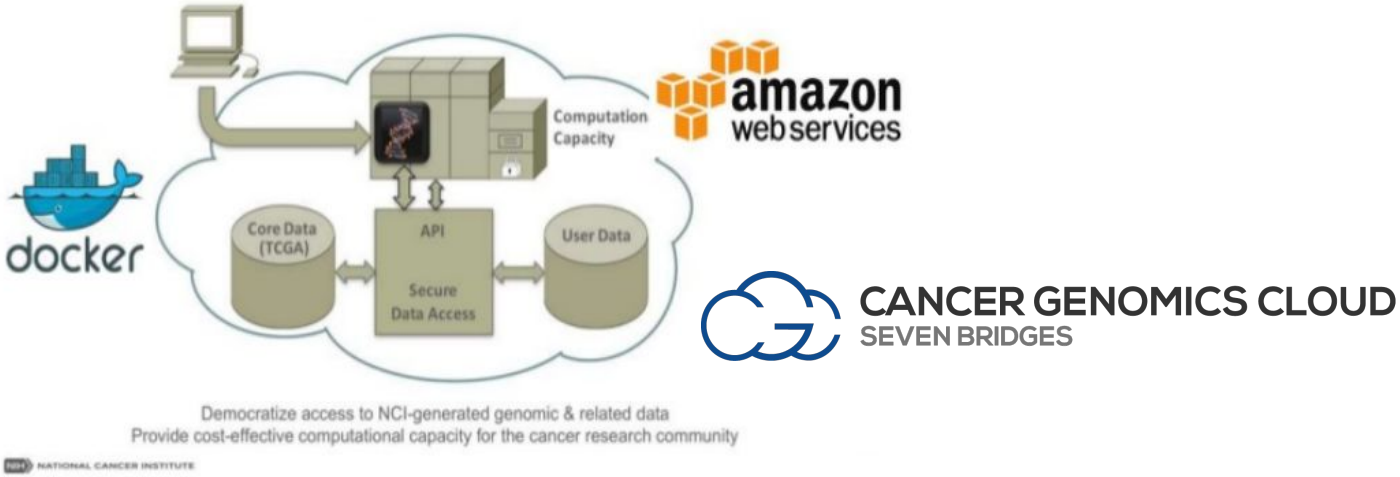
Dr. Matthieu Foll, Tiffany Delhomme and the participants from the course "Analyzing TCGA data in the cloud”.

OMICS discussions
04/06/2018

International Agency for Research on Cancer
World Health Organization

aim of the course

- **cloud computing** is a new paradigm in large scale computational research
- *idea*: bring the tool to the data
- *concepts*: elasticity, reproducibility, collaborative research



modified from I. Chandramouliswaran, NCI

conduct a project: main steps

1. Determine TCGA data to analyse and software to run on
2. Create a Dockerfile, build a Docker container and host it on DockerHub
3. Create a CGC project, add members
4. Add TCGA data on the project (GUI or API)
5. Create an app on the CGC (GUI or written in JSON/CWL)
6. Run the task on your files
    - one task per file
        - batch mode on the GUI
        - loop over files with the API
    - scatter mode to run multiple process in one task

course overview

- Agenda

Wednesday 28 February

09:00-10:00	Introduction to cloud computing and the SevenBridges architecture
10:00-10:30	Introduction to TCGA data
10:30-11:00	Break
11:00-11:30	Introduction to the SevenBridges web interface to run analyses
11:30-12:30	Practical application: run your first basic analysis in the cloud

Thursday 1 March

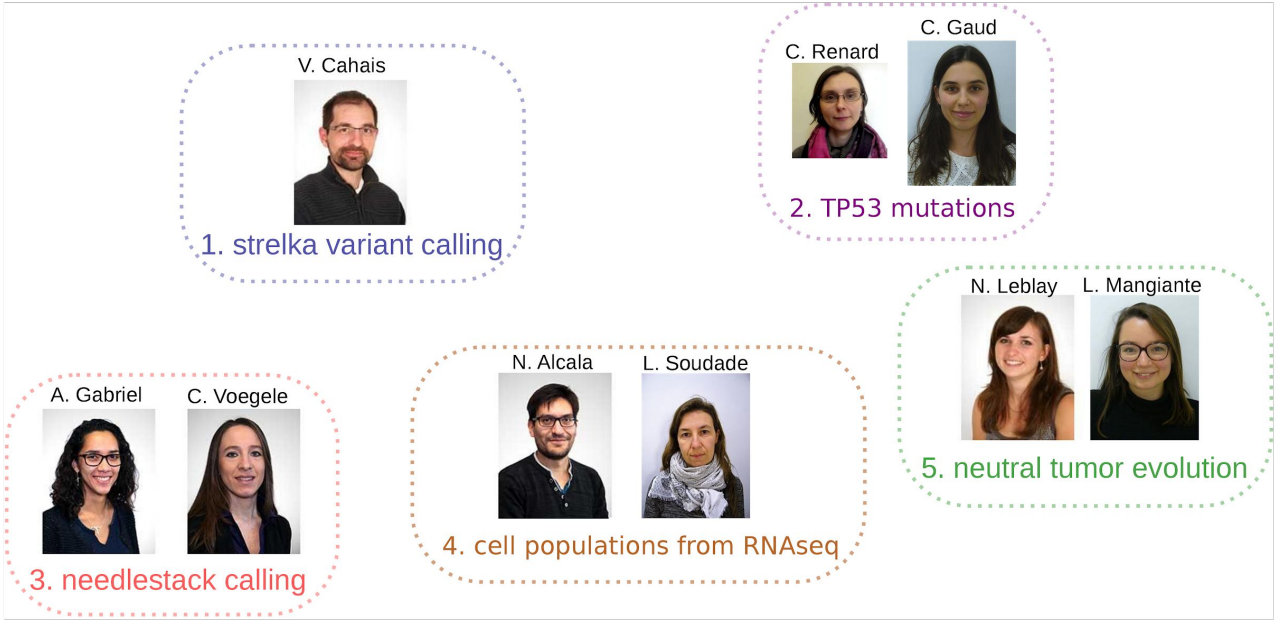
09:00-09:30	Introduction to Docker and DockerHub
09:30-11:00	Practical application: building your own Docker container and run it in the cloud
11:00-11:30	Break
11:30-12:30	Introduction to the R api and the CWL language

Friday 2 March

09:00-12:30	Practical application: running your own practical project in the cloud using the R api, CWL and Docker.
12:30-14:00	Lunch Break
14:00-17:00	Practical application: running your own practical project in the cloud using the R api, CWL and Docker.

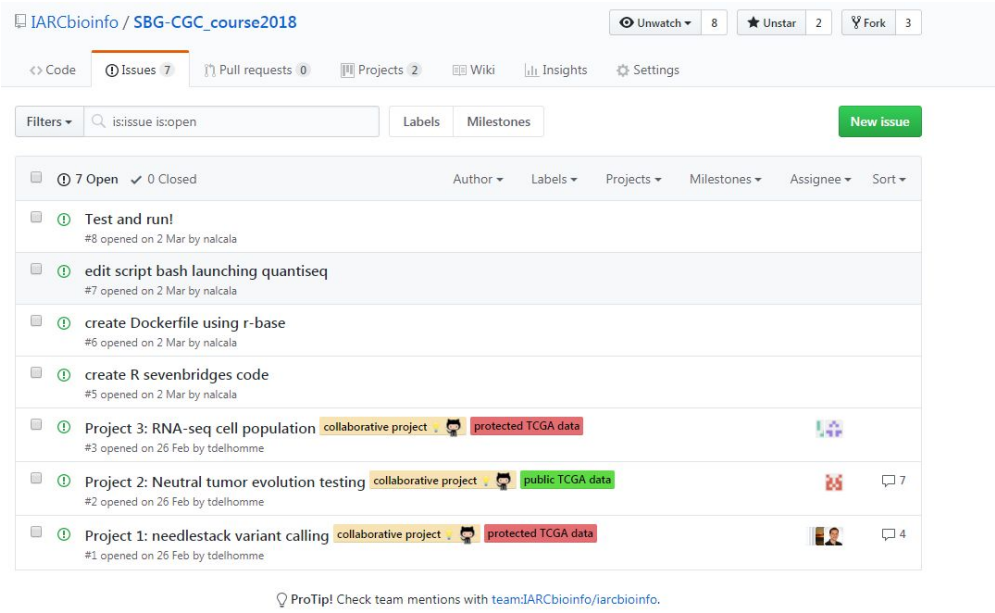
course overview

- Participants and projects



keep track of the course and record the code

- IARC bioinfo github webpage
  - page readme
    - agenda
    - guidelines
  - folders
    - store the code
    - store the slides
  - opened issues
    - discuss the projects
    - linked the code



https://github.com/IARCbioinfo/SBG-CGC_course2018

Projects

Project 1: strelka variant calling

Aim: Create wrapper to run Strelka2 on CGC

What was done:

- created a docker/IARC container for Strelka
- created via web interface + exported as JSON file (to use in R API) and available on the IARC github
- somatic and germline mode (2 wrappers)
- Can select: CPU, Memory

Vincent Cahais
MCA/EGE

Benchmark

- Data: 260 Go (18 BAM files)

	CGC	**Jupiter**
Exec time	33 min	1h 11min
Cores	4	2
Cost	0.20 $	?

- Transfer time between Jupiter and CGC: ~ 8 hours

Project 2: extraction of TP53 mutations

Aim: extract TP53 mutations (SNV) in all cancer types

- Public MAF (filtered for germline variants)
- MuTect outputs

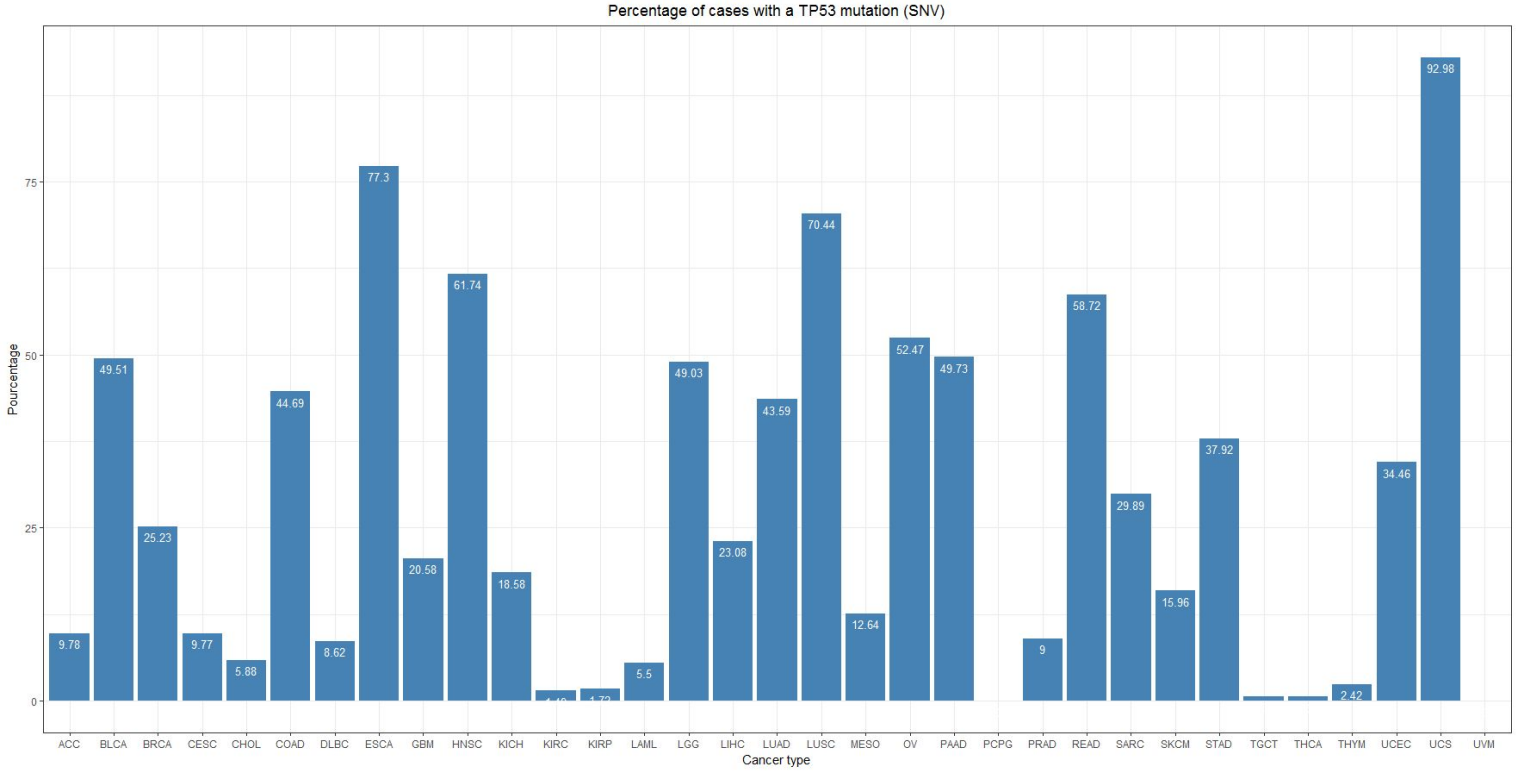
What was done:

- Create a wrapper for a bash command line on the web interface
- Loop over all cancer types

Execution time: <3 min | Cost: 0.25$

Claire Renard
Caroline Gaud
MCA/MMB

Project 2: extraction of TP53 mutations



International Agency for Research on Cancer

World Health Organization

Project 3: Needlestack variant calling

Aim: Perform a variant calling using Needlestack on each TCGA cohort

Why Needlestack ?

- Evaluate Needlestack on new pan-cancer data
- Find new variants

Why on the cloud ?

- Save downloading time and disk space on our cluster

Main challenge: parallelization without nextflow

Aurelie Gabriel
Catherine Voegele
Tiffany Delhomme
GEN/GCS

Project 3: Needlestack variant calling

Methods

- Adaptation of an existing docker file, creation of a project via the web interface and selection of TCGA BAM files and a reference genome available on cgc
- Creation of a task via the web interface using needlestack bash script and export in JSON format to run the task via the R API

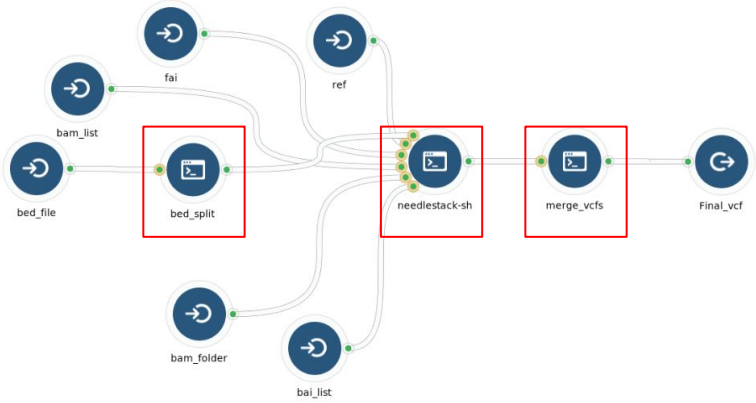
We did not finish in one day

Project 3: Needlestack variant calling

Problems and solutions

- Identify physical location of the BAM files (path)
- The BAM files are copied in an allocated machine
- Parallelize without Nextflow

→ Creation of a cgc **workflow** with **scatter option**



Workflow steps:

1) Bed file (genome regions) splitted for the parallelization. Scatter option applied to the splitted regions
2) Run needlestack on each region in parallel but in one task
3) Merge needlestack results (vcf files)

Project 3: Needlestack variant calling

<u>Results</u>

- Needlestack calling on TP53 exons and the UCS tumor BAM files (57 BAM files)
- Machine used: 32 CPUs, 1200 GB extra storage
- Cost: 0.61$ <=> 0.17$ for the storage and 0.44$ for the computation
- Time: 1h 03min

Project 4: estimation of cell populations from RNAseq

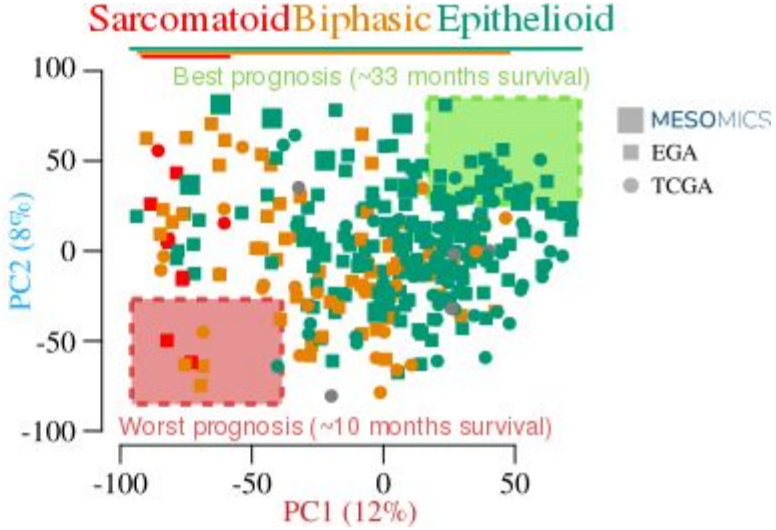
MESOMICS
Multi-omics characterization of
Malignant Pleural Mesothelioma

PIs: L. Fernandez-Cuesta & M. Foll

- **Preliminary results:** Strong variation in expression of *immune genes*

Pathways most associated with PC2 (GO)

immune response	($p=1\times10^{-41}$)
immune system process	($p=2\times10^{-31}$)
regulation of immune system process	($p=7\times10^{-47}$)
regulation of immune response	($p=2\times10^{-42}$)
+ regulation of immune system process	($p=8\times10^{-42}$)



Lorraine Soudade
Nicolas Alcala
GEN/GCS

Project 4: estimation of cell populations from RNAseq

MESOMICS
Multi-omics characterization of
Malignant Pleural Mesothelioma

PIs: L. Fernandez-Cuesta & M. Foll

- **Preliminary results:** Strong variation in expression of immune genes
- **Research question:** Is this variation due to different immune cell compositions?
- **Project:** Quantify immune cell in entire TCGA MESO cohort from bam files; cloud computing to avoid large downloads

Lorraine Soudade
Nicolas Alcala
GEN/GCS

Project 4: Methods

QuanTIseq

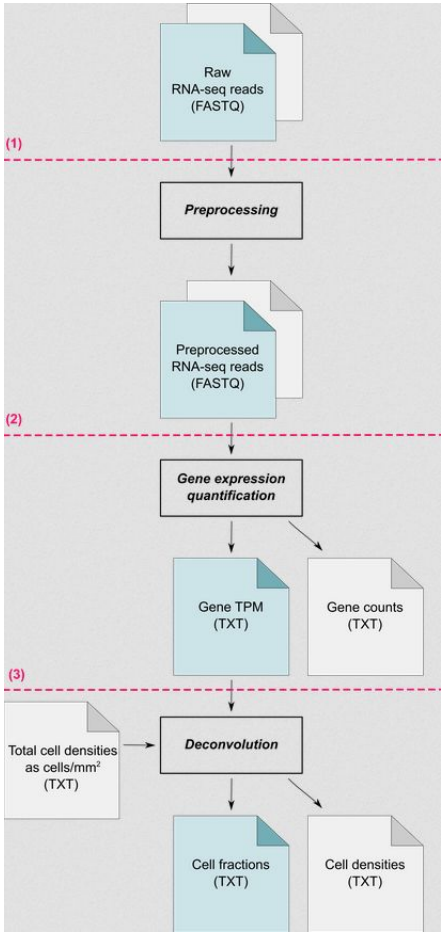
Pipeline for the **estimation of the proportion of cells** in the tumor sample from 10 different immune cell types (T cells, NK, ...)

Steps

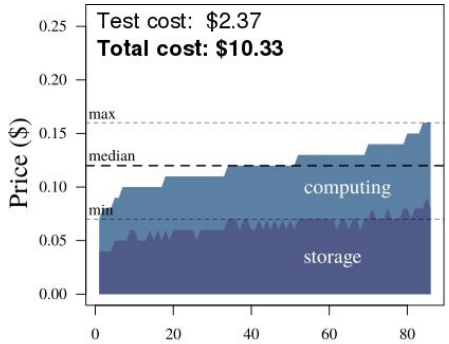
- Select **MESO RNAseq data** (*n*=86) via TCGA cloud UI
- Describe task **command & inputs/outputs** with CWL specification
- Implement SevenBridges R API to **run tasks in the cloud**
- Adapt & **Run QuanTIseq** on our data with docker

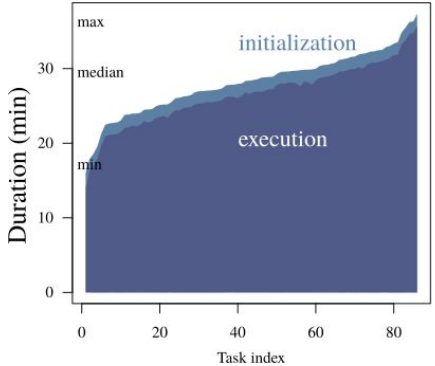
Difficulties

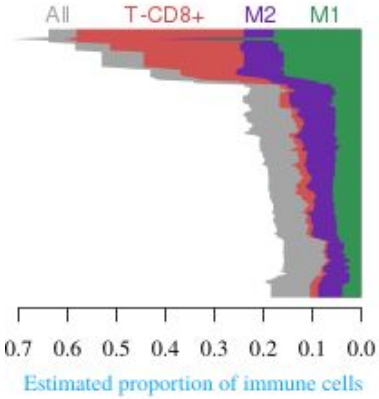
Adapt the analysis-ready pipeline to another input (BAM instead of FASTQ) and architecture (cloud instead of local machine), debug

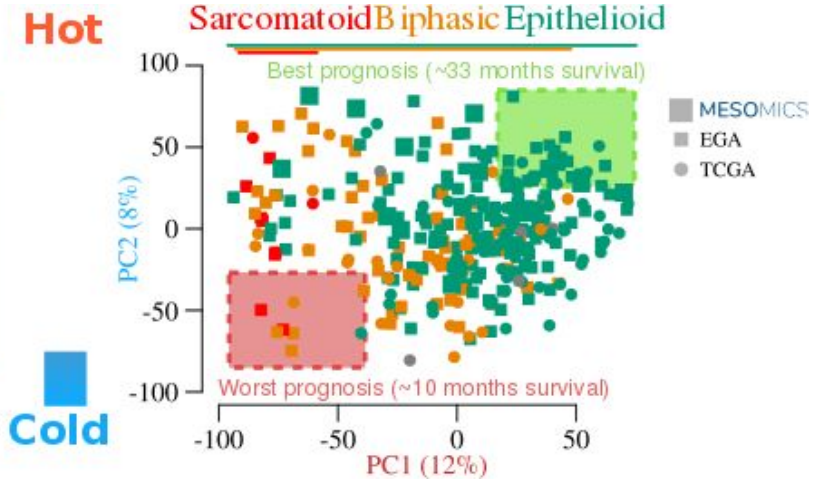


Project 4: Results









Conclusion: immune cell composition is an important source of variation in gene expression that influences survival

Project 5: neutral tumor evolution testing

Noémie Leblay
Lise Mangiante
GEN/GCS

MESOMICS
Multi-omics characterization of Malignant Pleural Mesothelioma

PIs: L. Fernandez-Cuesta & M. Foll
Nicolas Alcala

- **Research question:** Does Mesothelioma follow a neutral evolution model?

- **Project:** Test neutral tumor evolution model described in Williams et al.
  a. try to reproduce the method
  b. look at TCGA mesothelioma

ANALYSIS

nature genetics

Identification of neutral tumor evolution across cancer types

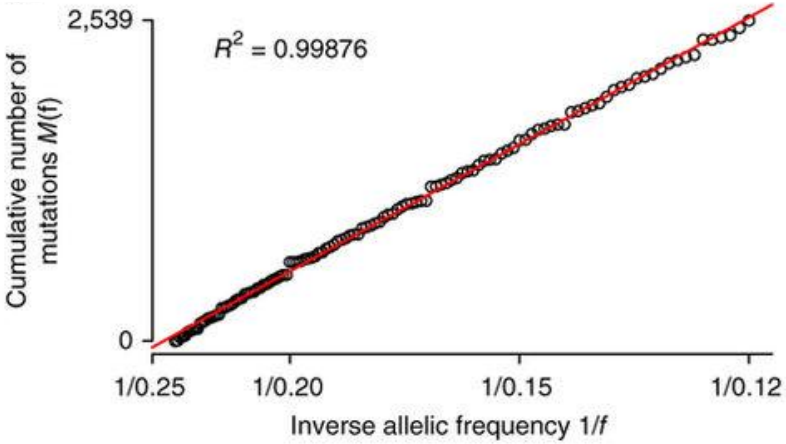
Marc J Williams[1–3,6], Benjamin Werner[4,6], Chris P Barnes[2,5], Trevor A Graham[1] & Andrea Sottoriva[4]

International Agency for Research on Cancer
World Health Organization

Project 5: neutral tumor evolution testing

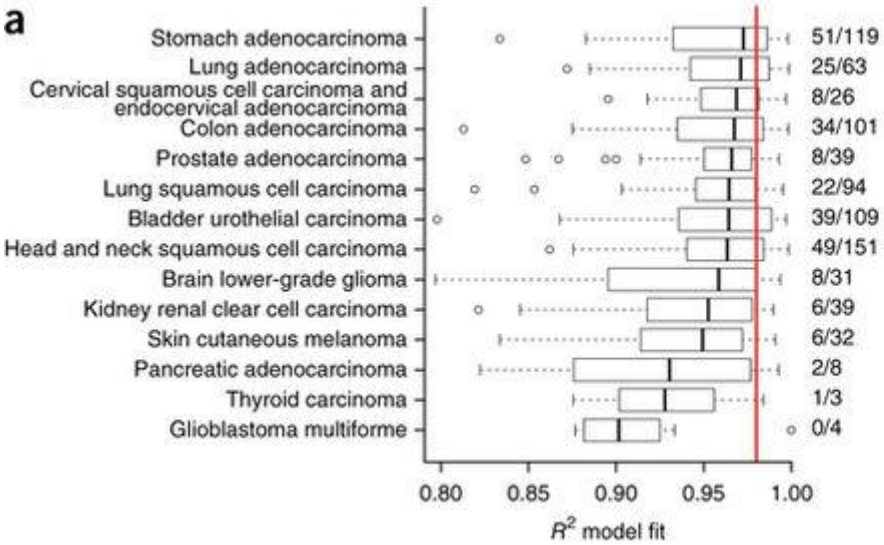
Noémie Leblay
Lise Mangiante
GEN/GCS

Neutral growth: subclones grow at the same rate (lack of stringent selection)



Neutral evolution across the whole genome of gastric cancers.

(Williams et al., 2016)



Neutral evolution and mutation rates across cancer types.
(Williams et al., 2016)

Project 5: neutral tumor evolution testing

Noémie Leblay
Lise Mangiante
GEN/GCS

Method:

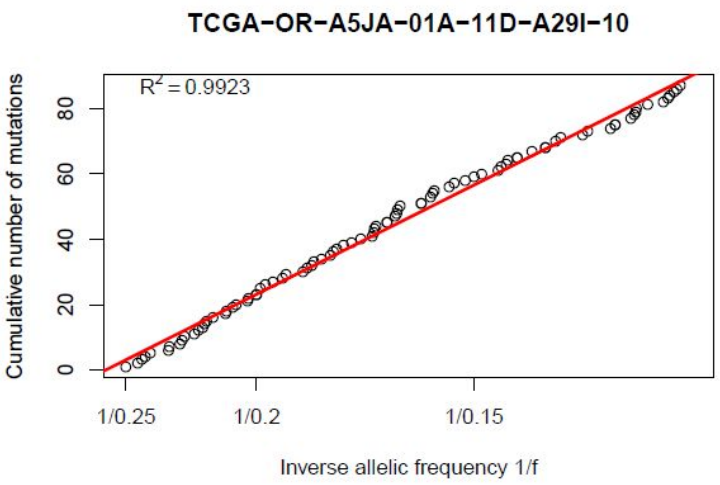
- Create R script which output regression coefficient and slope of the model by sample (Tiffany Delhomme)
- Adapt it for MAF files
- Put on GitHub the R script
- Create a Dockerfile using the GitHub link
- Create a SBG-CGC project:
  - ➔ Input: public MAF files from TCGA (Experimental Strategy = WXS)
  - ➔ App. Setting: dp_min, min_nb_point, min_r2, vaf_max, vaf_min
  - ➔ Output:
    - ◆ pdf file: all the cumulative number of mutations plot of samples present in the MAF file
    - ◆ R.data files: all the R^2 / slope coeficient per TCGA tumor type

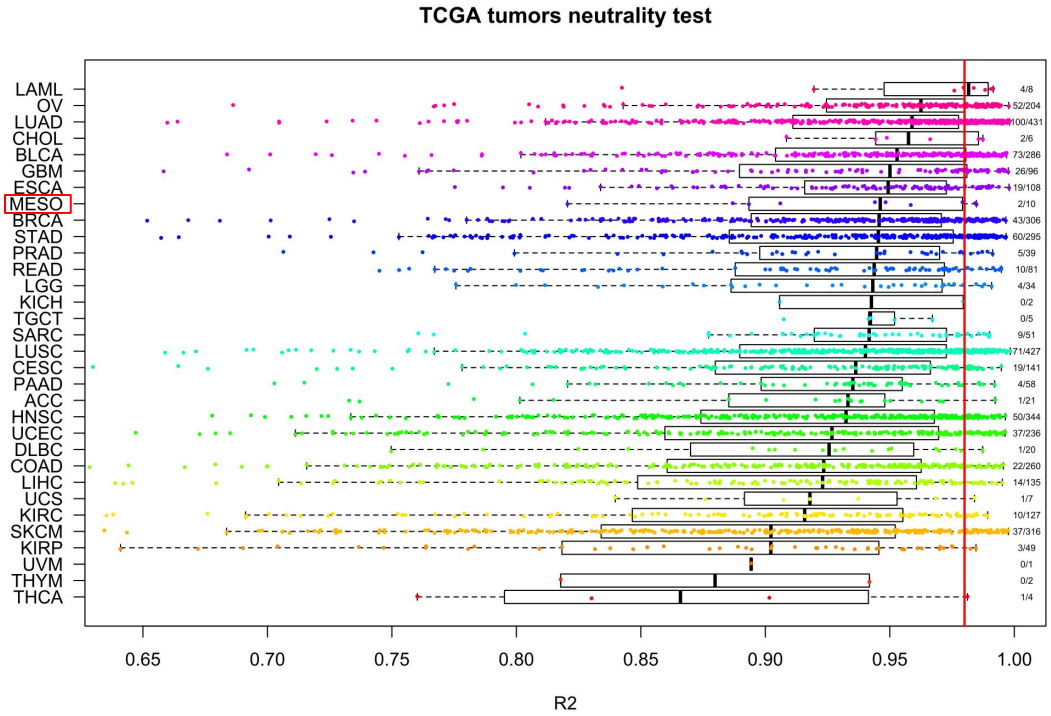
Problems:

- Errors in creating new files
  - MAF files are not physically on the server→ name of the file contain all the path of the file

Project 5: neutral tumor evolution testing

Results:





Conclusion: creation of a free, accessible, reproducible, and adaptable method

International Agency for Research on Cancer

World Health Organization

Conclusions

- Very powerful (but “With Great Power Comes Great Responsibility”)
- Cheap because you don't pay to store data
- The “cloud” relies on physical machines with their limits
- Requires IT skills
- Good training experience:
  - Rather knowledge sharing than training
  - Collaborative: a nice day working on interesting projects together
  - Using GitHub as a platform