

# Introduction to Bioinformatics

Chris Miller, Ph.D.  
Washington University in St. Louis

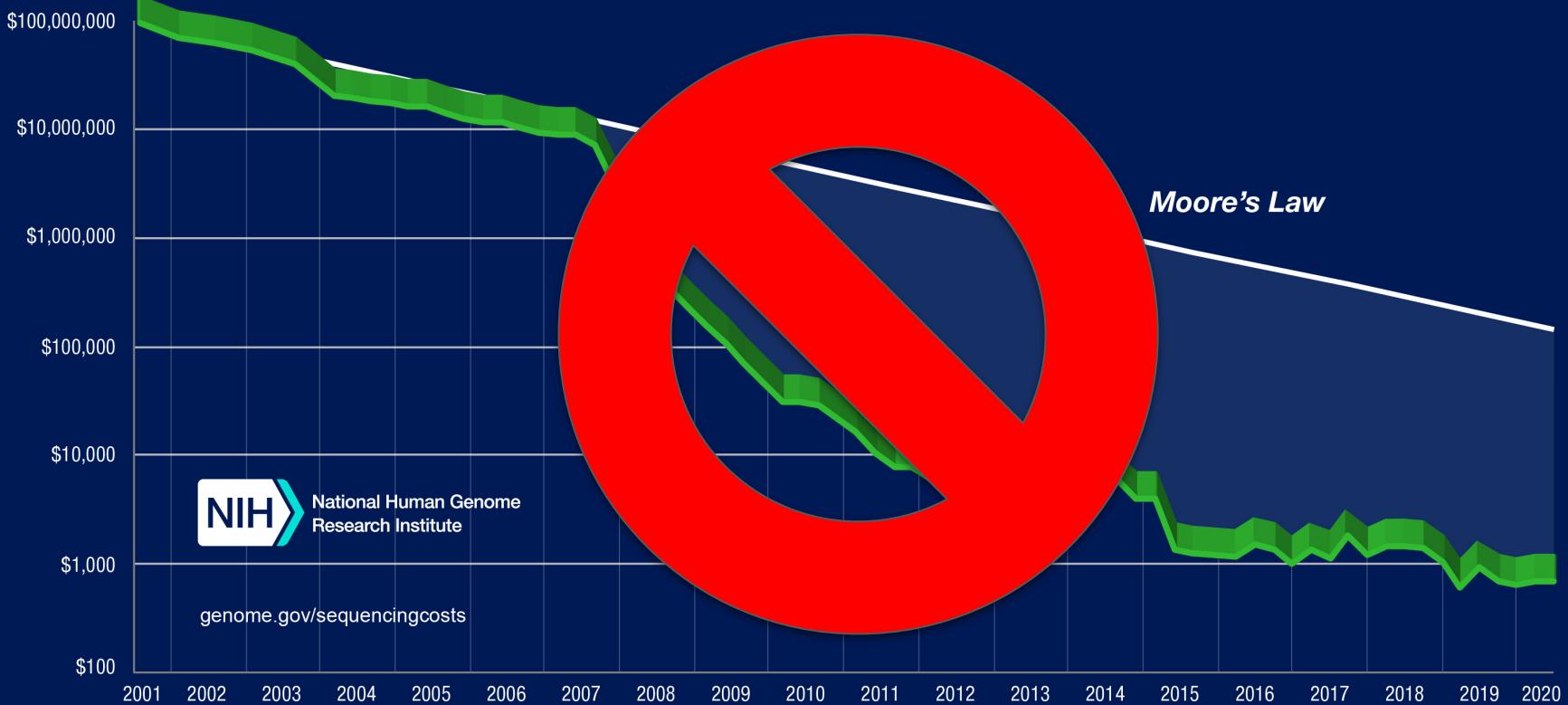
Some slides adapted from:  
<https://github.com/genome/bfx-workshop>  
<https://github.com/quinlan-lab/applied-computational-genomics>



# Why learn bioinformatics?

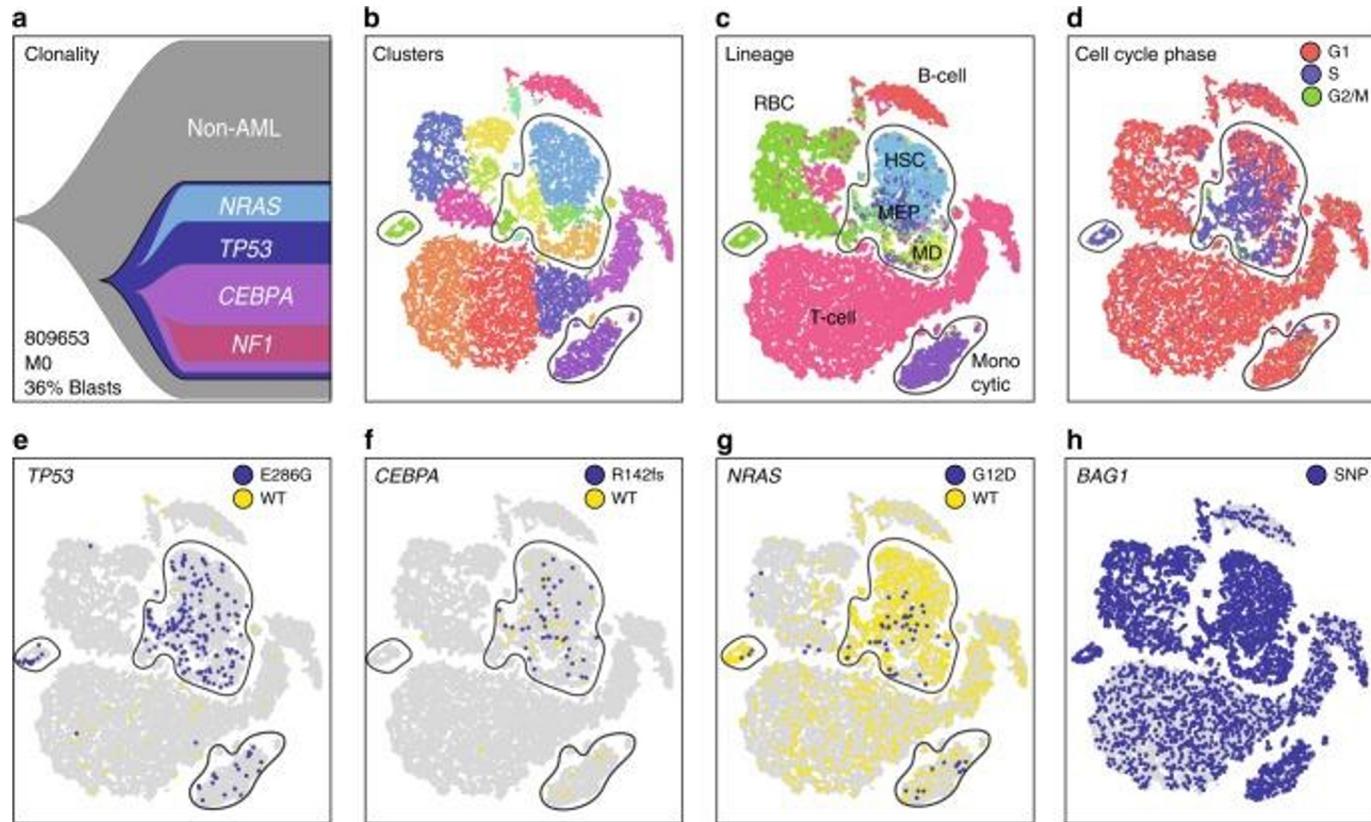
- Biology is now a quantitative discipline - especially genomics

## *Cost per Human Genome*



National Human Genome  
Research Institute

[genome.gov/sequencingcosts](http://genome.gov/sequencingcosts)



# Why learn bioinformatics?

- Biology is now a quantitative discipline - especially genomics
- Skills in programming, statistics, and visualization help you get the most out of your data



People who need complex data analysis



People who know how to do  
complex data analysis

# Why learn bioinformatics?

- Biology is now a quantitative discipline - especially genomics
- Skills in programming, statistics, and visualization help you get the most out of your data
- We're aiming to teach you the theory and practice of computational biology, with a focus on genomics but lessons that apply broadly

# What is bioinformatics?

**Chris Miller** @chrisamiller · Apr 8

I get that I'm not the arbiter of terminology but it all seems like meaningless distinctions to me. If you're using computers to study biological information, then use any combination of those words as a title. Then more importantly, tell me what you *\*actually do\**

**Liz Worhey** @lizworhey · Apr 8

Bioinformatics vs. Computational Biology: A Comparison  
[medicaltechnologyschools.com/biotechnology/...](http://medicaltechnologyschools.com/biotechnology/)

8 3 63

**“Michael”** @mikelove · Apr 8

Replying to [@chrisamiller](#)  
so you're cool with me saying I study Computer Bioinfology

2 6 40

**Rob Patro (@rob@genomic.social)** @nomad421 · Apr 8

Replying to [@mikelove](#) and [@chrisamiller](#)  
As long as you sound drunk whenever you say it, I'd be fine with this.

9

# What is bioinformatics?

- Application of computational techniques to biological data
- Covers a lot of ground!
  - Population genetics
  - Cancer genomics
  - Microbial genomics
  - Proteomics
  - Ecology/Evolution
  - Medical informatics/EHR mining
  - computational behavioral biology
  - Epidemiology
  - Protein folding
  - CryoEM or tomography
  - Drug design/molecular dynamics
  - Algorithmic design/optimization
  - Metabolomics
  - Mathematical Biology

# What is bioinformatics?

More Computational

More biological



Algorithmic Design

Dataviz

Biological expertise

Machine Learning

Statistics

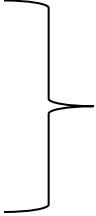
Scientific rigor

High performance/cloud  
computing

Data munging

Communication

# Common skills

- Statistics
  - Programming
  - Visualization
- 
- “Data science”
- Deep understanding of the biological system and experiments

# Goals of this course

- To empower you to improve and expedite your research
- To expose you to new ideas and techniques that may advance your research program

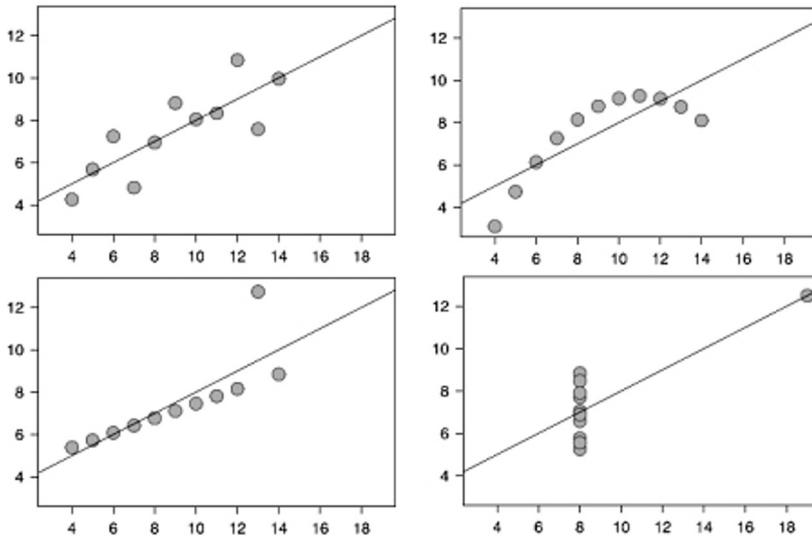
# Course structure

- Command-line basics to get you up to speed
- Generation of sequencing data, formats, alignment
- ChIP-seq and methylation, peak calling
- Introduction to the R programming language
- Bulk RNA-seq and differential gene expression
- Introduction to Python
- Single-cell RNA-seq
- Bedtools/genome arithmetic
- Variant calling and interpretation
- Statistics and probability
- Single-cell epigenomics, ATAC-seq

**Don't trust your data**

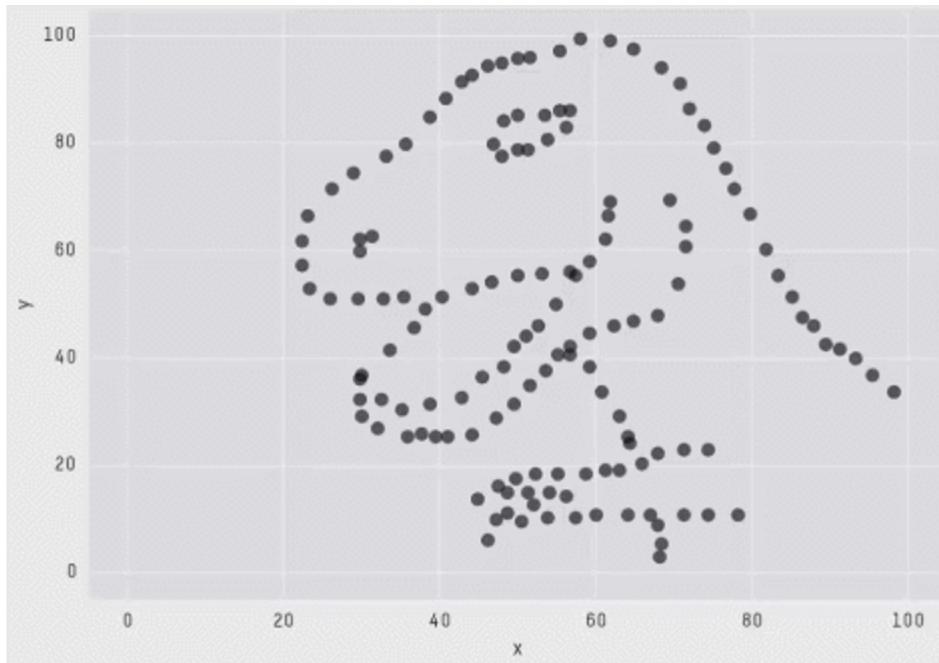
# Trusting your data

Anscombe's quartet



Property	Value	Accuracy
Mean of $x$	9	exact
Sample variance of $x : \sigma^2$	11	exact
Mean of $y$	7.50	to 2 decimal places
Sample variance of $y : \sigma^2$	4.125	$\pm 0.003$
Correlation between $x$ and $y$	0.816	to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	to 2 and 3 decimal places, respectively
Coefficient of determination of the linear regression : $R^2$	0.67	to 2 decimal places

# Datasaurus Dozen



X Mean: 54.2659224  
Y Mean: 47.8313999  
X SD : 16.7649829  
Y SD : 26.9342120  
Corr. : -0.0642526

# Summary statistics are dangerous

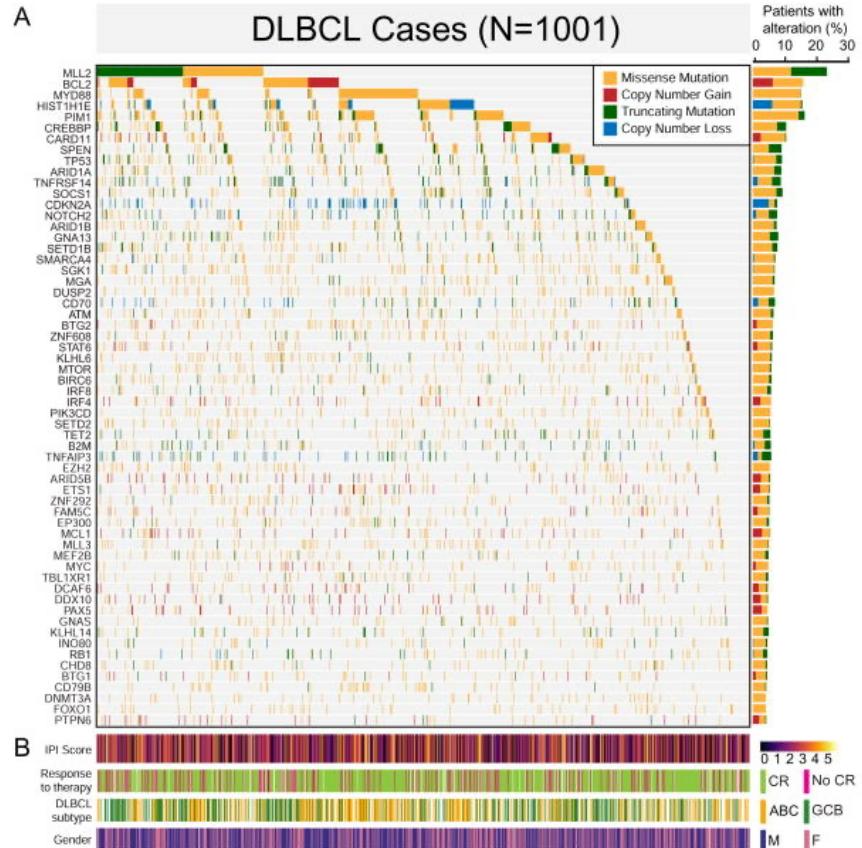
- Visualize your data!
- A picture is worth a thousand p-values

"If your experiment needs statistics, you ought to have done a better experiment"

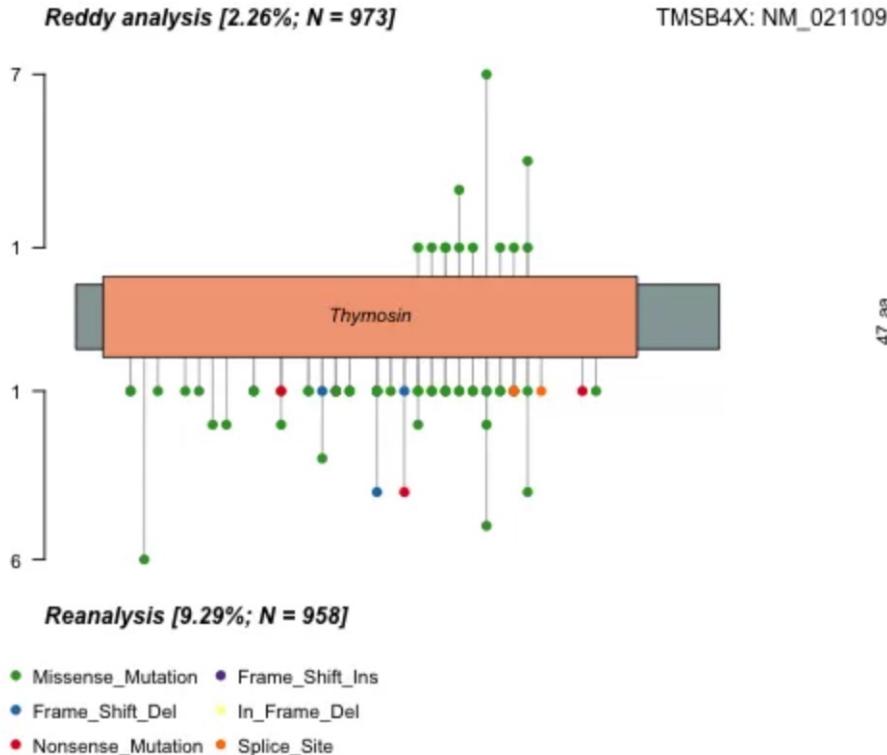
- Ernest Rutherford

- The bioinformatics core aligned the data and sent me a list of differentially expressed genes. I'm done, right?
- We ran Mutect to call somatic mutations in this tumor genome. Let's take it to the bank

# Real world consequences



# Real world consequences



# Real world consequences

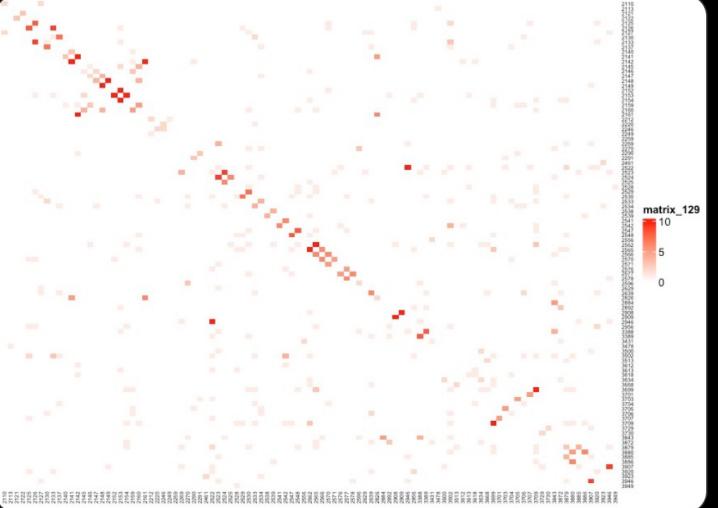
Ryan D Morin @morinryan · Oct 2

RNA/DNA mismatches (sample swaps) affecting at least 10% of the patients in Reddy et al, a Cell paper with over 700 citations. Same issue was described in a more recent paper from this group. #lymphoma  
#genomics #goodresearchpractice  
[pubpeer.com/publications/E...](http://pubpeer.com/publications/E...)

1 1 7

Ryan D Morin @morinryan · Oct 2

Sharing of variants between RNA and DNA. Red should be on the diagonal. Most swaps seem to be between adjacent or nearby IDs.

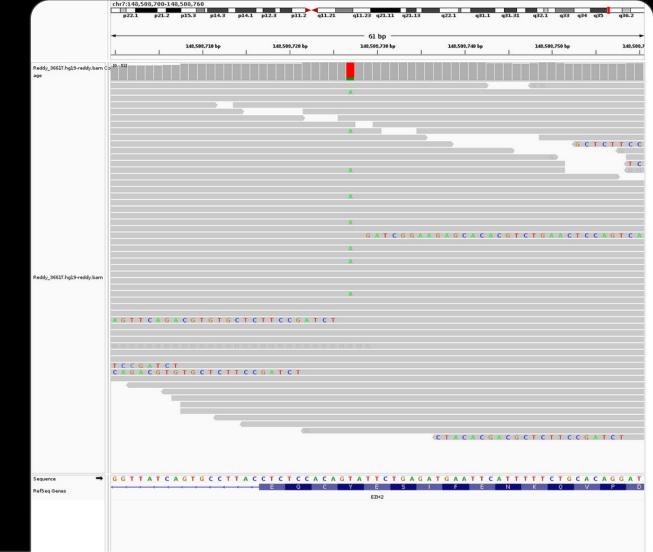


Ryan D Morin @morinryan · Nov 4

There are over 3,600 examples of variants like this, supported by at least 3 somatic variant callers (i.e. by consensus, they're real) and yet Reddy didn't report them. All of these are coding variants in the DLBCL genes described in Reddy but all were absent for some reason.

Ryan D Morin @morinryan · Nov 4

24/33 (this is the limit imposed by Twitter). This is just the first 24. If someone still thinks I'm cherry-picking examples. This is a clinically relevant hot spot that was described 7 years before the Reddy study. Inexcusable to miss this many of them, and yet excuses are made!



# Real world consequences

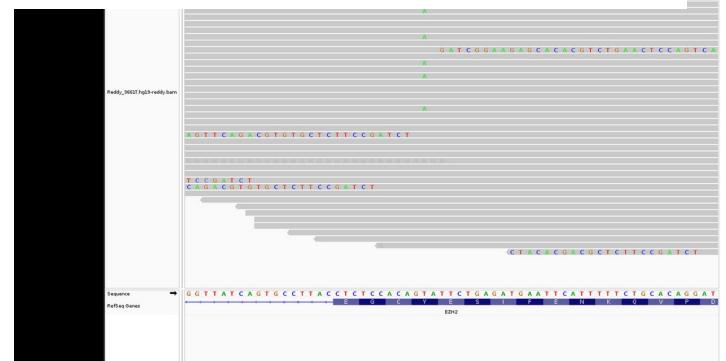
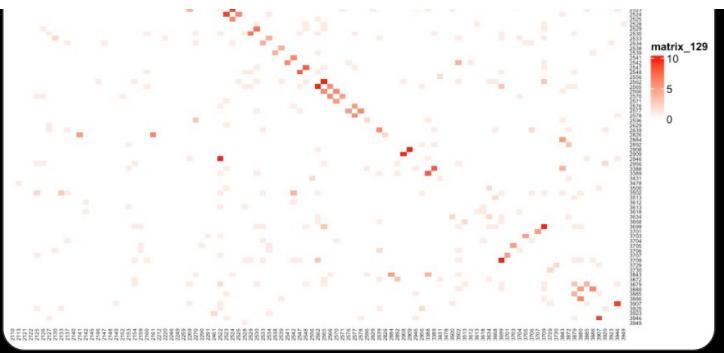
Ryan D Morin @morinryan · Oct 2

RNA/DNA mismatches (sample swaps) affecting at least 10% of the patients in Reddy et al, a Cell paper with over 700 citations. Same issue was described in a more recent paper from this group. #lymphoma  
#genomics #goodresearchpractice  
[pubpeer.com/publications/E...](http://pubpeer.com/publications/E...)

Ryan D Morin @morinryan · Nov 4

There are over 3,600 examples of variants like this, supported by at least 3 somatic variant callers (i.e. by consensus, they're real) and yet Reddy didn't report them. All of these are coding variants in the DLBCL genes described in Reddy but all were absent for some reason.

Although the effects on each conclusion from Panea *et al* has not been evaluated, we demonstrated that ~30% of the reported mutations are not supported by their WGS data, which caused a significant inflation of the mutation prevalence of at least 16 genes and the rate of coding mutations in 9 genes (Supplemental Figure S3). These lead to



# Lessons to be learned

- Check and double check and triple check your data and your scripts
- Visualize your data!
- Admit when mistakes are made

# Errors

- Will happen!
- Errors of commission vs omission
- Type 1 errors – False positives
- Type 2 errors – False negatives

“Analyzing your data means inherently distrusting your data until you have exhausted yourself into giving up and trusting it.”

-Aaron Quinlan

# Reproducibility

- If you're doing bioinformatics right, reproducibility should be "easy"!
- Data will be well organized and stored safely

sample637.tsv  
sample647.tsv  
sample983.tsv

Mouse\_TP53\_WT\_637.tsv  
Mouse\_TP53\_KO\_647.tsv  
Mouse\_TP53\_KO\_983.tsv

Laptop vs compute cluster (both need backups)!

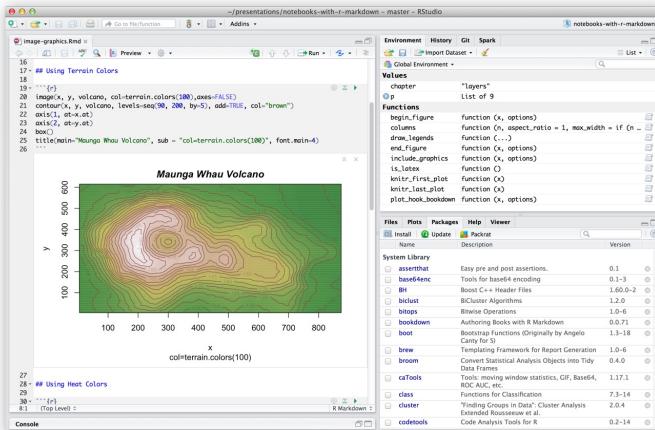
# Reproducibility

- Your tools/parameters/settings should all be stored in scripts
- The ideal to aim for is that you could send someone a link to your data and scripts, and they could sit down, run it, and reproduce your figures/tables
- Hell is other people's data.  
Hell is also your own data 6 months later.

# Reproducibility

- Your tools/parameters/settings should all be stored in scripts
- The ideal to aim for is that you could send someone a link to your data and scripts, and they could sit down, run it, and reproduce your figures/tables

R Markdown



The screenshot shows an RStudio interface with an R Markdown file open. The code includes sections for terrain colors and heat colors, resulting in a volcano plot titled "Maunga Whau Volcano". The RStudio environment tab shows a "Layers" section with a list of 9 items.

```
...{r}
16 # Using Terrain Colors
17 ...
18 ##(r)
19 ...
20 # Input(x, y, volcano, col=terrain.colors(100), axes=FALSE)
21 # contour(x, y, volcano, levels=seq(0, 200, by=5), add=TRUE, col="brown")
22 axis(x, at=x.ot)
23 axis(2, at=y.ot)
24 title(main="Maunga Whau Volcano", sub = "col=terrain.colors(100)", font.main=4)
25 ...
26 ...

Maunga Whau Volcano

y
100 200 300 400 500 600
100 200 300 400 500 600
x
100 200 300 400 500 600 700 800

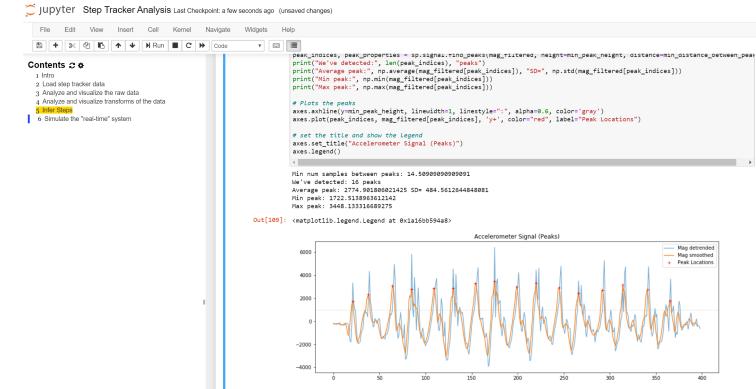
27 ## Using Heat Colors
28 ##(r)
29 ...
30 ...{r}
31 ...{Top Level}:
```

Environment tab: Values, chapter, layers, List of 9

Code tab: Values, chapter, layers, List of 9

Console tab: (Top Level):

Jupyter notebooks



The screenshot shows a Jupyter Notebook interface with a "Step Tracker Analysis" notebook. It contains a table of contents and several code cells. One cell displays a line plot of an accelerometer signal with peaks highlighted in red, blue, and orange, corresponding to detected steps. The plot is titled "Accelerometer Signal (Peaks)".

Contents & #

- 1 Intro
- 2 Load step tracker data
- 3 Analyze and visualize the raw data
- 4 Analyze and visualize transforms of the data
- 5 Visualize the peaks
- 6 Simulate the "real-time" system

```
# Step Tracker Analysis Last Checkpoint: a few seconds ago (unsaved changes)

Contents & #
```

- 1 Intro
- 2 Load step tracker data
- 3 Analyze and visualize the raw data
- 4 Analyze and visualize transforms of the data
- 5 Visualize the peaks
- 6 Simulate the "real-time" system

```
peak_indices, peak_properties = nn.signal_rising_peaks(mag_filtered, height_min_peak_height, distance_min_distance_between_peaks)
print("We've detected: ", len(peak_indices), "peaks")
print("Average peak: ", np.average(nn.get_mean(peak_properties['mag'][peak_indices])), "SD: ", np.std(mag_filtered[peak_indices]))
print("Max peak: ", np.max(nn.get_mean(peak_properties['mag'][peak_indices])))

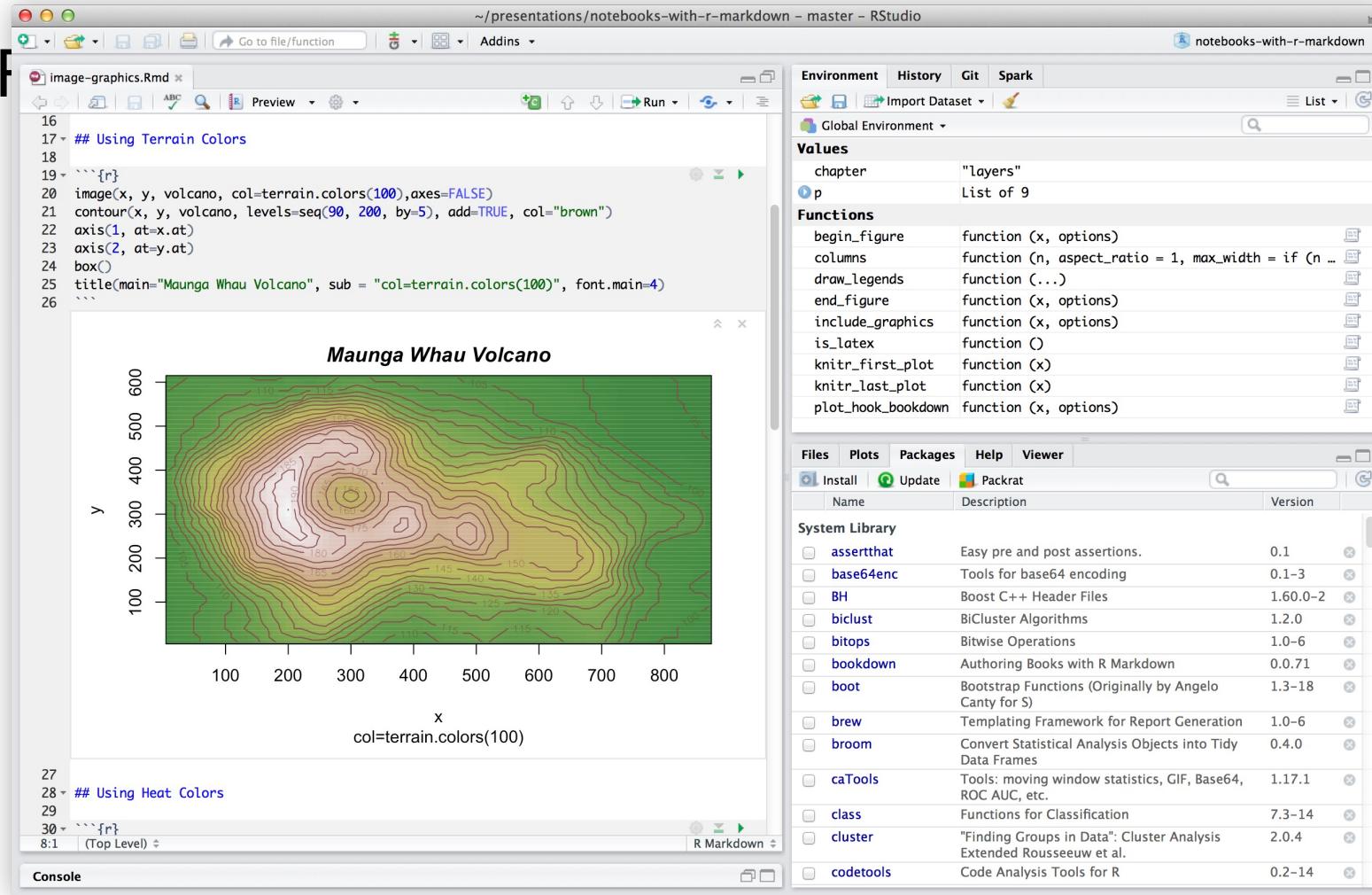
# Plot the peaks
plt.figure(figsize=(10, 5))
plt.plot(mag, mag_filtered, linewidth=1, linestyle='-', alpha=0.5, color='gray')
axes.plot(peak_indices, mag_filtered[peak_indices], 'yo', color='red', label='Peak Locations')

# set the title and show the legend
axes.set_title("Accelerometer Signal (Peaks)")
axes.legend()

# Run
nn.run_samples_between_peaks(14_5000000000000001)
Average peak: 2774.003000021425 SD: 464.5612044848801
Min peak: 1722.5130000000002
Max peak: 3448.1331000000007
Out[100]: StepTrackerSignal(legend:Legend at 0x1a1600059448)
```

Accelerometer Signal (Peaks)

Legend: Mag derivative (blue), Mag smoothed (orange), Peak Locations (red)





R

File Edit View Insert Cell Kernel Help

Not Trusted

Python 3



## Inputs

We are using a toy example data set based on the HCC1395 blood normal cell line. The sequence reads and genome reference are a subset targeting chr6, genes HLA-A and HLA-B-C, and chr17, genes TP53 and BRCA1.

[FASTA Normal Reads Lane 3](#) [Normal Reads Lane 4](#)

All inputs and additional resources can be viewed at: <https://console.cloud.google.com/storage/browser/analysis-workflows-example-data>

In this example, each file was downloaded to ~/Downloads. If you saved the downloaded files in another folder or location, the following paths will need to be updated to account for those differences.

```
In [ ]: !mv ~/Downloads/hla_and_brca_genes.fa $PWD/ref/.  
!mv ~/Downloads/2895499331.bam $PWD/unaligned/normal/.  
!mv ~/Downloads/2895499399.bam $PWD/unaligned/normal/.
```

## Index

Index files of various formats and data structures are used commonly in genomics to provide random access to specific records, locations, or content within much larger domains and coordinate spaces. Ex. entire genome nucleotide sequences, billions of sequence reads, millions of variant records, etc.

## Samtools

Using samtools faidx, we create an index (.fai) of the reference FASTA (.fa) which provides random access to the nucleotides at specific positions within the complete genome reference. The FAIDX index is used both by the samtools faidx command as well as other toolkits, algorithms, and libraries that require random access to specific coordinates in a timely manner.

```
In [ ]: ls $PWD/ref
```

We only have the FASTA sequence, let's make a FAIDX format file using Samtools by running the command with no additional arguments (other than the FASTA file):

# What happens to my data after analysis?

- Journals will not publish data that isn't accessible
  - NOT just “Available upon reasonable request”!
- Every NIH grant now requires a Data Sharing plan
- Covers more than just sequence data – gel images, textual qPCR readouts, flow plots/data, etc
- Needs to go into a repository

# FAIR principles of data

- Findable
- Accessible
- Interoperable
- Reproducible

# Examples of good places to deposit data

- NCBI repositories
  - SRA – Short Read Archive
  - dbGaP – front-end/access control for human data in SRA
  - GEO – rich metadata associated with experiments (RNA, scRNA, arrays, etc)
- Organism specific repos (Flybase, etc)
- General data repos that assign a doi (zenodo.org)
- Institutional repositories – your university library probably runs one
- **NOT** a lab website or a Google bucket (what happens in 5 years?)

# What happens when this works well?

National Heart, Lung, and Blood Institute | **BioData CATALYST** | Powered by Gen3

Browse Data | Documentation | ICMILLER | Logout

Data File

Explorer Filters | Data Tools | Summary Statistics | Table of Records

Data Access ^

Data with Access  
 Data without Access  
 All Data

Export to Seven Bridges · Export All to Terra · Export to PFB · Export to Workspace

Subjects: 53,964 Projects: 23

Filters

Harmonized Variables

Project Subject

Collapse all

Program 1 selected

topmed 53,964  
 parent 186,592  
 tutorial 14,433  
 open\_access 3,202

Annotated Sex

no data 53,964 (100%)

Race

no data 100%

Showing 1 - 20 of 53,964 subjects Show Empty Columns

Project Id	Data Format
topmed-CAMP_DS-AST-COPD	CRAM
topmed-BioMe_HMB-NPU	CRAM
topmed-BioVU_AF_HMB-GSO	

# What happens when this works well?

The screenshot shows the BioData CATALYST interface. At the top, there are tabs for 'Discovery', 'Expression', 'Analysis', 'Networks', and 'Profile'. Below the tabs, there's a search bar with the word 'analysis' and several filters on the left, including 'Data Name', 'Project ID', and 'Status'. The main area displays a table with columns 'Subject', 'Project ID', 'Title', and 'Status'. There are 53,964 rows in total, with 23 projects. A red circle highlights the 'Status' column header.

The screenshot shows the Dockstore interface. The left sidebar has navigation links: Dashboard, Workflows (which is selected), Tools, Services, Starred, Account, and Help Desk. The main content area is titled 'My Workflows' and shows a list of GitHub organizations: chrisamiller, genome, griffithlab, and leyabdotorg. Under 'chrisamiller', there are two workflows: 'analysis-workflows' (Published) and 'five-dollar-genome-analysis-pipeline' (Unpublished). Under 'genome', there is one workflow: 'analysis-workflows' (Published). Under 'griffithlab' and 'leyabdotorg', there are no unpublished workflows. On the right, a specific workflow from 'github.com/genome/analysis-workflows:master' is shown. It has tabs for 'Info' (selected), 'Launch', 'Versions', 'Files', and 'Tools'. The 'Info' tab contains a message about keeping the workflow in sync with GitHub. Below that is a section titled 'Workflow Information' with fields for 'Source Code' (set to 'github.com/genome/analysis-workflows'), 'Workflow Path' ('/Dockstore.cwl'), 'Test File Path' ('/test.json'), and 'Topic Automatic' (described as 'Open workflow definitions for genomic analysis from MGI at WUSM'). There are 'Edit' buttons for each of these fields.

# What happens when this works well?

The screenshot shows the Terra WORKSPACES interface. The top navigation bar includes 'WORKSPACES' (Beta), 'Workspaces > leylab.terra/ch\_fusion\_topmed > workflows > wdl\_samtools.manta', and a 'COVID-19 Data & Tools' button. Below the navigation is a toolbar with 'DASHBOARD', 'DATA', 'ANALYSES', 'WORKFLOWS' (highlighted in green), and 'JOB HISTORY'. On the left, a sidebar titled 'BioData CATALYST' displays 'Data Notes' (53,964) and 'Filters' (selected 'Annotated Genes'). The main content area shows the workflow details for 'wdl\_samtools.manta' (Version: v1.3, Source: [github.com/saimukund20/wdl\\_samtools.manta](https://github.com/saimukund20/wdl_samtools.manta)). It includes a synopsis: 'No documentation provided' and two execution options: 'Run workflow with inputs defined by file paths' (radio button) and 'Run workflow(s) with inputs defined by data table' (radio button, selected). The 'Step 1' section shows 'pharmu\_set' selected as the root entity type. The 'Step 2' section has a 'SELECT DATA' button and a message 'No data selected'. Below these are checkboxes for 'Use call caching', 'Delete intermediate outputs', 'Use reference disks', 'Retry with more memory', and 'Ignore empty outputs'. At the bottom are tabs for 'SCRIPT', 'INPUTS' (highlighted in green), 'OUTPUTS', and 'RUN ANALYSIS'. The 'INPUTS' tab shows three variables: 'wf' for 'fusion\_sites' (File type, attribute: "gs://fc-6248b026-3011-4591-9688-255a248b35b9/sites\_merged\_with\_solid\_tumor.bed"), 'wf' for 'manta\_config' (File type, attribute: "gs://fc-6248b026-3011-4591-9688-255a248b35b9/configManta\_1.py.ini"), and 'wf' for 'reference' (File type, attribute: "workspace.referenceData\_hg38\_ref\_fasta"). There are also 'Download json' and 'SEARCH INPUTS' buttons.

# What happens when this works well?

The screenshot shows the BioData CATALYST interface. At the top, there's a navigation bar with 'Data', 'File', 'Discovery', 'Expression', 'Analysis', 'Workflows', and 'Profile'. Below this is a search bar with placeholder text 'Search for samples, analysis, or projects...'. Under the search bar, there are sections for 'Data Names' (with filters for 'Data with Metrics', 'All Data', and 'All analysis'), 'Filters' (with a red circle highlighting the 'Advanced' button), and 'Programs' (with a red circle highlighting 'Project ID'). The main area displays a table with columns 'Subject', 'Project', 'Sample ID', 'Barcode', and 'Status'. A large red circle highlights the 'Barcode' column header. The table shows 53,964 rows.



The screenshot shows the Dockstore Workflows interface. At the top, there's a navigation bar with 'Dashboard', 'Workflows', 'Services', 'Help Desk', and 'Account'. Below this is a search bar with 'Search' and a GitHub icon. The main area shows a workflow named 'wgl\_zarrtools\_manta'. It has two steps: 'Step 1' (using 'zarrtools') and 'Step 2' (using 'manta'). The 'Step 1' step is highlighted with a green background. The table below lists tasks with columns 'Task name', 'Variable', 'Type', and 'Ansible'. There are two tasks listed: 'public\_ip' (variable 'ip', type 'Ansible', value '192.168.122.111:4500:4500') and 'reference' (variable 'reference', type 'Ansible', value 'https://raw.githubusercontent.com/GenomeAnalysisTools/wgl\_zarrtools\_manta/main/reference/hg38\_vcf.vcf').



The screenshot shows the Dockstore interface. At the top, there's a navigation bar with 'Dashboard', 'Workflows', 'Services', 'Help Desk', and 'Account'. Below this is a search bar with 'Search' and a GitHub icon. The main area shows a workflow named 'wgl\_zarrtools\_manta'. It has two steps: 'Step 1' (using 'zarrtools') and 'Step 2' (using 'manta'). The 'Step 1' step is highlighted with a green background. The table below lists tasks with columns 'Task name', 'Variable', 'Type', and 'Ansible'. There are two tasks listed: 'public\_ip' (variable 'ip', type 'Ansible', value '192.168.122.111:4500:4500') and 'reference' (variable 'reference', type 'Ansible', value 'https://raw.githubusercontent.com/GenomeAnalysisTools/wgl\_zarrtools\_manta/main/reference/hg38\_vcf.vcf').



Analysis results for >50,000 Whole Genome Samples in a few hours

# Infrastructure – where do I analyze my data?

- Laptop
  - Pro: Easy – it's sitting in front of you!
  - Pro: You have root access (can install anything you need)
  - Con: Power is limited – number of cores, amount of RAM
  - Con: amount of disk is limited (a single WGS experiment can be >50 Gb)
  - Con: what if it's stolen? (you do have automatic backups, right?!)
  - Con: what happens when you close the lid?



# Infrastructure – where do I analyze my data?

- Big desktop machine or blade in your lab
  - Pro: moderate amounts of CPUs/RAM for big jobs
  - Pro: You have priority access
  - Pro: Can submit jobs and walk away
  - Con: You have to become a sysadmin and take care of it.  
(Who applies security updates? What happens if the power supply fails? Backups?)



# Infrastructure – where do I analyze my data?

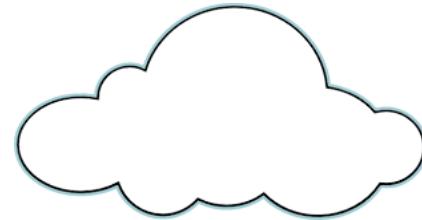
- Local Compute Cluster

- Pro: Lots of CPUs/RAM for big jobs
- Pro: probably has dedicated disk with good backups
- Pro: Can submit jobs and walk away
- Con: You may have to contact administrators to do installs
- Con: you have to share resources, and you may not have priority!



# Infrastructure – where do I analyze my data?

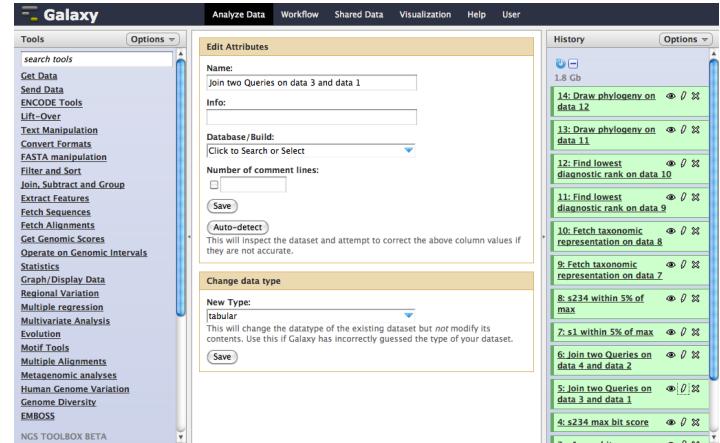
- Cloud (remote compute cluster)
  - Pro: As much CPU/RAM as you can imagine
  - Pro: secure disk, backups, etc
  - Pro: no reasonable limits on access
  - Con: You may have to transfer your data up/down
  - Con: can be pricey (and you have to be so careful!)
  - Con: unless you have institutional support, you have to learn to administer it



# Infrastructure – where do I analyze my data?

- Web analysis portals

- Backed by cloud, more friendly front ends
- Still have to pay for it, learn the system
- Have to transfer your data up/down
- May have GUIs for common tools



# Infrastructure

- Bioinformatics requires infrastructure just like lab work
  - pipette tips don't appear and cell cultures don't feed themselves
  - servers don't appear and software doesn't install itself

# Funding

- Traditionally has been very hard to get grants for software development
- even harder for maintenance
- large amount of "abandonware"

# What is Unix?

**Definition 1:** Unix is not an acronym; it is a pun on "Multics". Multics was a large multi-user operating system that was being developed at Bell Labs shortly before Unix was created in the early '70s. Brian Kernighan is credited with the name.

**Definition 2:** Where computational genomics is done.

**Definition 3:** Your dear friend.



Recommended reading: "The Evolution of the Unix Time-sharing system", Dennis M. Ritchie  
<https://pdfs.semanticscholar.org/f64f/6e66da16e93ebf4221fc8915b2420fd56b66.pdf>

# Unix history

Ken Thompson (sitting) and Dennis Ritchie working together at a PDP-11

- File system, command interpreter (shell), and process management started by Ken Thompson
- Device files and further dev from Dennis Ritchie, as well as McIlroy and Ossanna (to a lesser degree)
- **Vast array of simple, dependable tools that each do one simple task.**
- By combining these tools, one can conduct rather sophisticated analyses
- Wildly popular platform for high performance computing. Supports parallelism.
- SunOS/Solaris, IBM's AIX, Hewlett-Packard HP-UX, OSX, Linux, Android, etc.



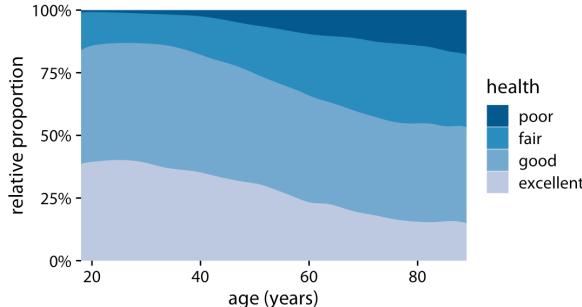
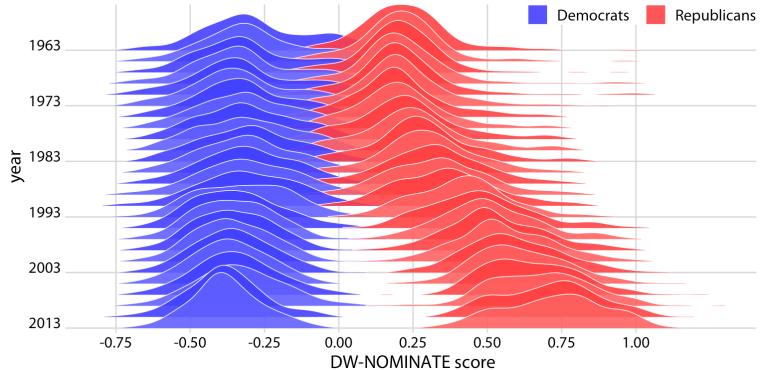
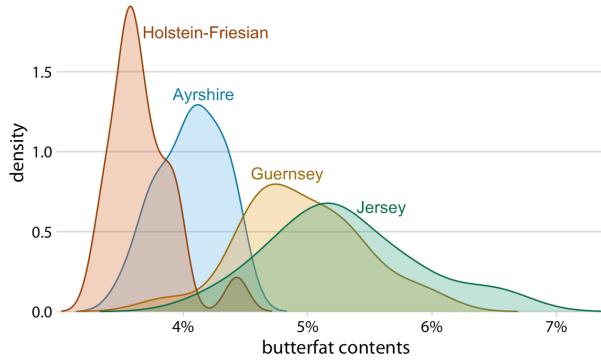
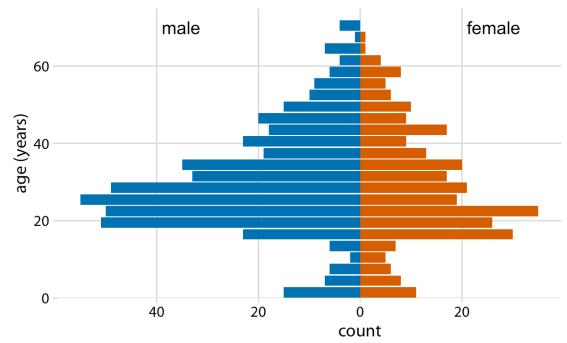
Credit:[https://en.wikipedia.org/wiki/History\\_of\\_Unix](https://en.wikipedia.org/wiki/History_of_Unix)

# Why Unix?

- **Power!**
- The big event had to be postponed due to COVID and now we have to change every instance of "Apr 2020" to "Oct 2022". Problem is, there's a huge nested set of directories containing over 10,000 files!
- Clicking around in Windows explorer is not going to get the job done
- On a Unix system, that's just one short line of code:

```
find . -name "*.txt" | xargs -n 1 sed -i.bak 's/Apr 2020/Oct 2022/g'
```

# Turning data into insight



# Course structure

- Command-line basics to get you up to speed
- Generation of sequencing data, formats, alignment
- ChIP-seq and methylation, peak calling
- Introduction to the R programming language and data visualization
- Bulk RNA-seq and differential gene expression
- Introduction to Python
- Single-cell RNA-seq
- Bedtools/genome arithmetic
- Variant calling and interpretation
- Statistics and probability
- Single-cell epigenomics, ATAC-seq