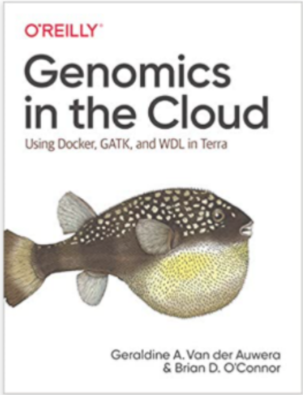
Genomics in the Cloud

The Semi-Official Companion Booklet

Author: Geraldine A. Van der Auwera

Date: December 31, 2020

Version: 1.0



O'REILLY

Genomics in the Cloud

Using Docker, GATK, and WDL in Terra

Geraldine A. Van der Auwera
& Brian D. O'Connor

Original book: https://oreil.ly/genomics-cloud

About this booklet

This booklet contains the figures used in Genomics in the Cloud (in full color) and their captions. Its primary purpose is to provide a way for readers of the print version of the book (which is in grayscale) to access the full color versions of the figures, either by browsing the PDF or printing it out. The booklet also includes a list of chapters as well as a table of contents for each chapter, which might be helpful as a quick reference. Note that this first version is a little rough around the edges; there is a lot of opportunity for improvement, but it's going to take some wrestling with LaTeX... All feedback and offers of help are welcome!

Figure re-use policy

The figures in this booklet are subject to the same rules/copyright as the original book. You are welcome to use any of these figures in teaching materials, presentations and so on, without express permission (except for the two figures credited to others, 6.15 and 3.3; please check their respective authors' requirements). Attribution is always appreciated, e.g. "*Genomics in the Cloud* by Geraldine A. Van der Auwera and Brian D. O'Connor (O'Reilly), https://oreil.ly/genomics-cloud".

Additional resources

- Book: https://oreil.ly/genomics-cloud
- Blog: https://broadinstitute.github.io/genomics-in-the-cloud
- Github: https://github.com/broadinstitute/genomics-in-the-cloud
- Figures: https://console.cloud.google.com/storage/browser/genomics-in-the-cloud/figures/

Contact information

- Twitter: @VdaGeraldine
- LinkedIn: Geraldine Van der Auwera

Cover image credit

Brocken Inaglory, Wikimedia Commons

List of Chapters

List of Figures

Chapter 1 Introduction

Why you should care about the cloud, and how bioinformatics / life sciences research benefits from moving to a cloud-based ecosystem for data sharing and analysis. No, the cloud environment is not perfect; yes, it really is a game changer.

1.1 The Promises and Challenges of Big Data in Biology and Life Sciences

1.2 Infrastructure Challenges

1.3 Toward a Cloud-Based Ecosystem for Data Sharing and Analysis

1.3.1 Cloud-Hosted Data and Compute

1.3.2 Platforms for Research in the Life Sciences

1.3.3 Standardization and Reuse of Infrastructure

1.4 Being FAIR

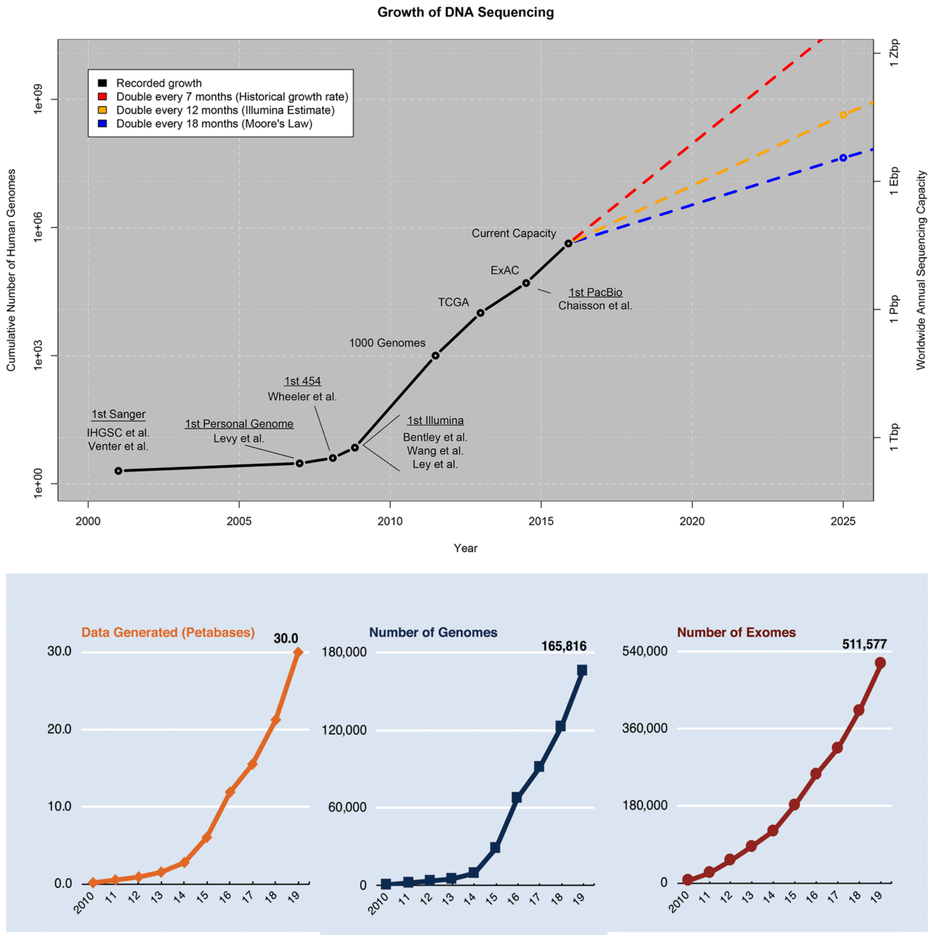


Figure 1.1: Recorded growth of sequencing datasets up to 2015 and projected growth for the next decade (top); growth in data production at the Broad Institute (bottom).

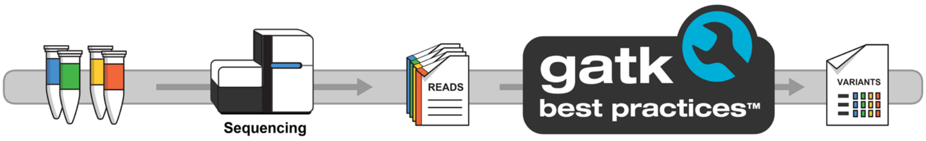


Figure 1.2: GATK provides a series of Best Practices to process sequence data for a variety of experimental designs.

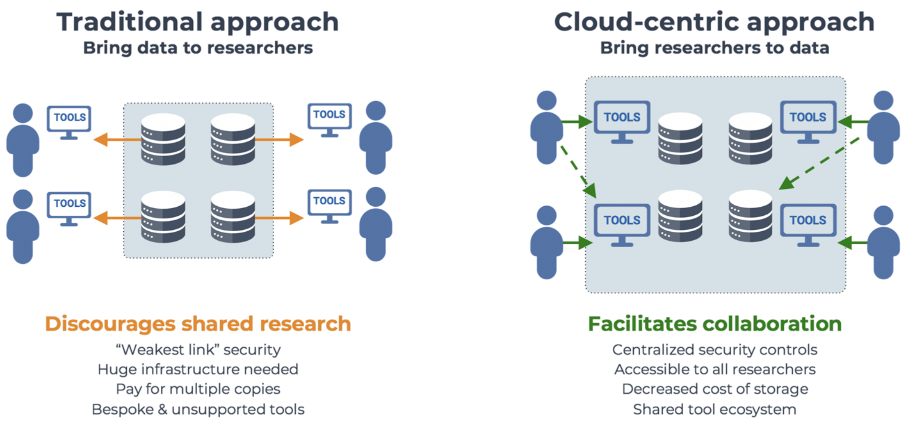

Traditional approach
Bring data to researchers
TOOLS
TOOLS
TOOLS
TOOLS
Discourages shared research
"Weakest link" security
Huge infrastructure needed
Pay for multiple copies
Bespoke & unsupported tools
Cloud-centric approach
Bring researchers to data
TOOLS
TOOLS
TOOLS
TOOLS
Facilitates collaboration
Centralized security controls
Accessible to all researchers
Decreased cost of storage
Shared tool ecosystem

Figure 1.3: Inverting the model for data sharing.

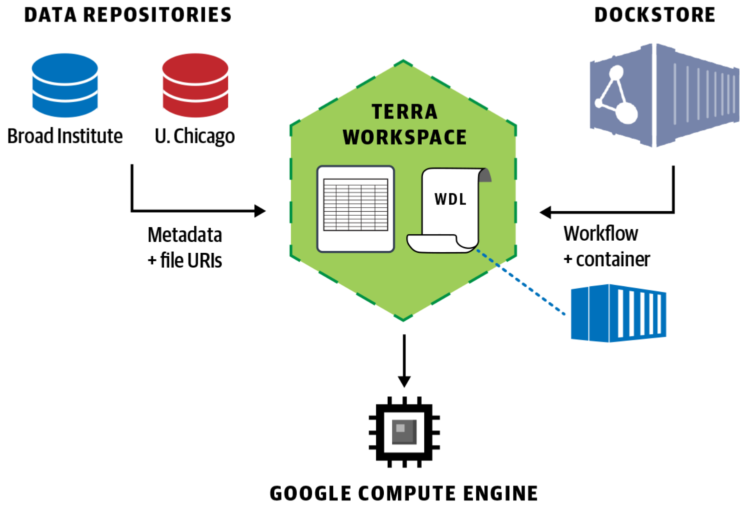


Figure 1.4: Data Biosphere principles in action: federated data analysis across multiple datasets in Terra using a workflow imported from Dockstore and executed in GCP.

Chapter 2 Genomics in a Nutshell: A Primer for Newcomers to the Field

A primer for newcomers to the field of genomics, covering foundational terms and concepts such as genes, DNA and genomic variation, plus the technical basics of sequencing and handling genomic data.

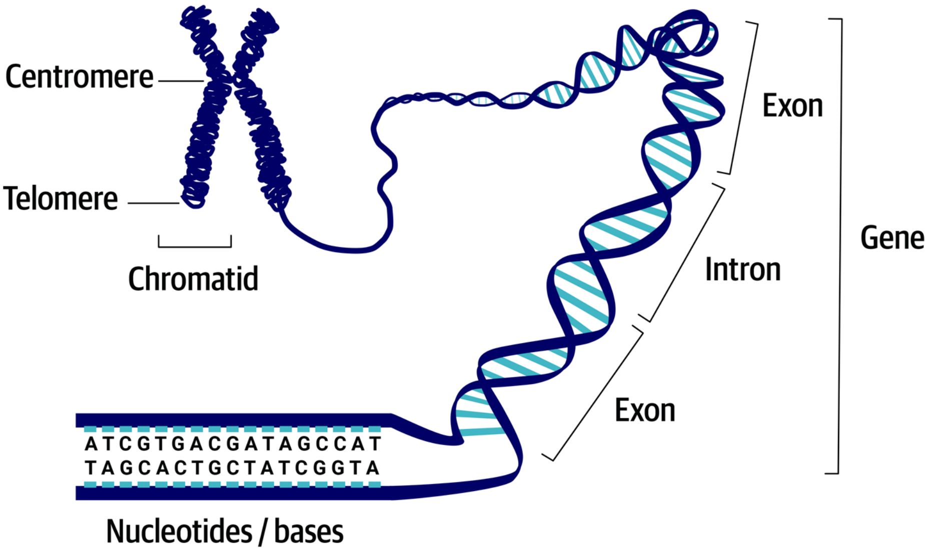


Figure 2.1: The chromosome (shown here in the form of two sister chromatids, each composed of one incredibly long molecule of double-stranded DNA) on which we delineate genes composed of exons and introns.

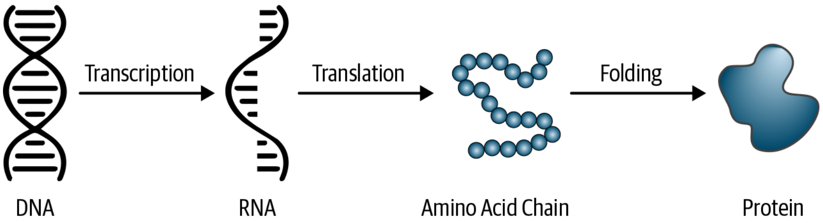
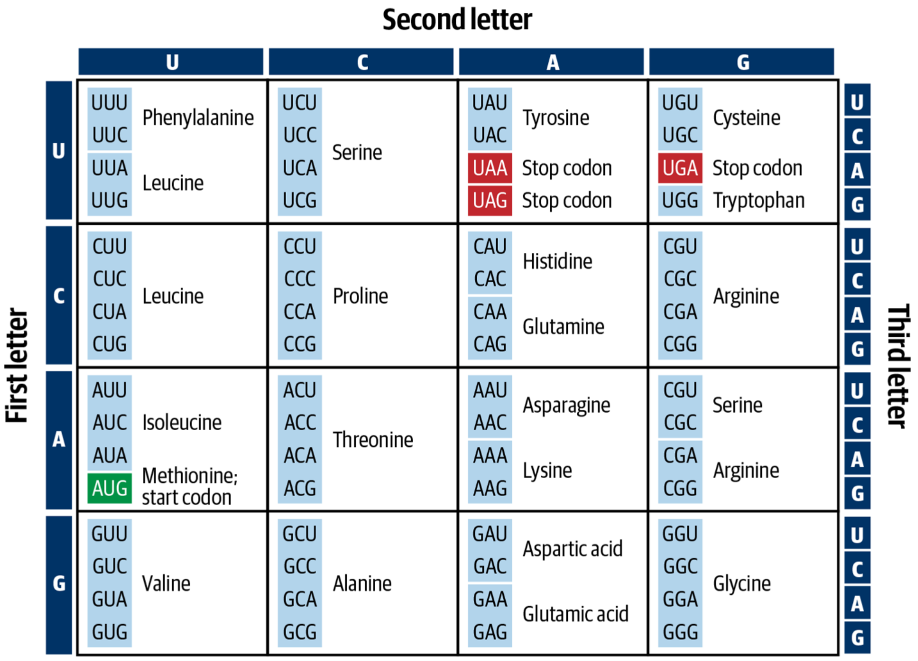


Figure 2.2: The central dogma of biology: DNA leads to RNA; RNA leads to amino acids; amino acids lead to protein.



Second letter

First letter	U		C		A		G		Third letter
U	UUU	Phenylalanine	UCU	Serine	UAU	Tyrosine	UGU	Cysteine	U
	UUC		UCC		UAC		UGC		C
	UUA	Leucine	UCA		UAA	Stop codon	UGA	Stop codon	A
	UUG		UCG		UAG	Stop codon	UGG	Tryptophan	G
C	CUU	Leucine	CCU	Proline	CAU	Histidine	CGU	Arginine	U
	CUC		CCC		CAC		CGC		C
	CUA		CCA		CAA	Glutamine	CGA		A
	CUG		CCG		CAG		CGG		G
A	AUU	Isoleucine	ACU	Threonine	AAU	Asparagine	CGU	Serine	U
	AUC		ACC		AAC		CGC		C
	AUA		ACA		AAA	Lysine	CGA	Arginine	A
	AUG	Methionine; start codon	ACG		AAG		CGG		G
G	GUU	Valine	GCU	Alanine	GAU	Aspartic acid	GGU	Glycine	U
	GUC		GCC		GAC		GGC		C
	GUA		GCA		GAA	Glutamic acid	GGA		A
	GUG		GCG		GAG		GGG		G

Figure 2.3: The genetic code connects three-letter codons in a messenger RNA sequence to specific amino acids.

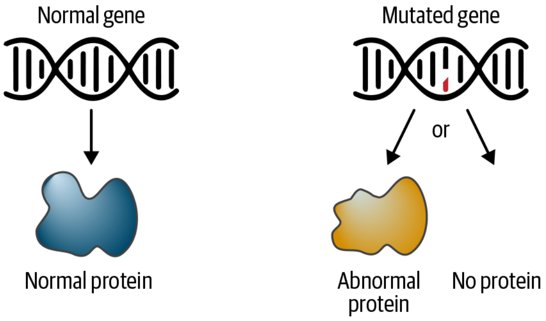


Figure 2.4: A mutation in the DNA sequence can cause the gene's protein product to function abnormally or disable its production entirely.

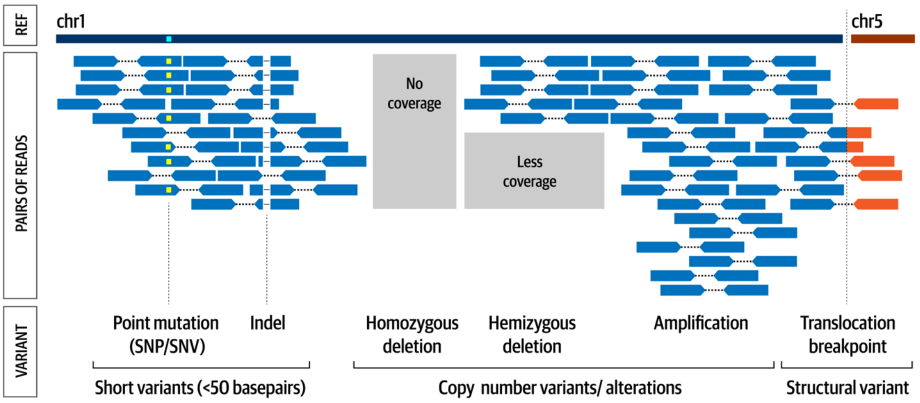


Figure 2.5: The major types of variant classified by physical changes to the DNA.

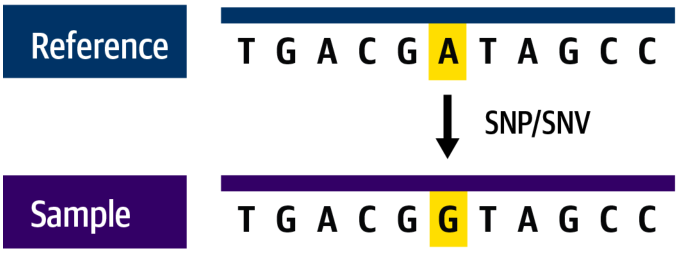


Figure 2.6: A single-nucleotide variant.

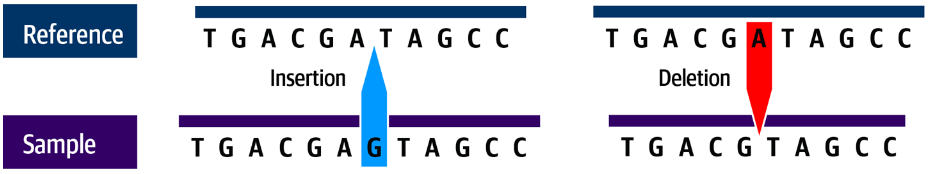


Figure 2.7: Indels can be insertions (left) or deletions (right).

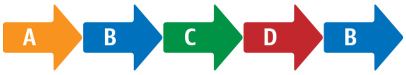


Figure 2.8: Example of copy-number variant caused by a duplication.

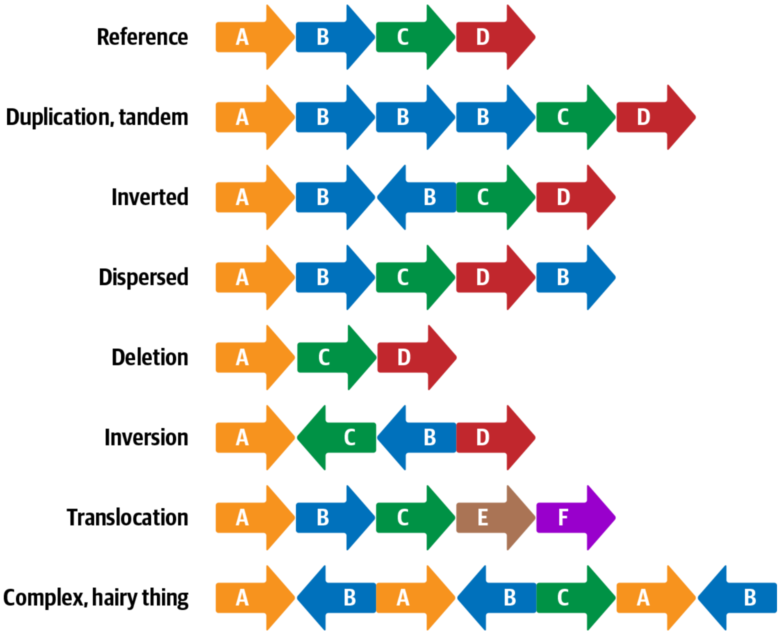


Figure 2.9: Examples of structural variants.

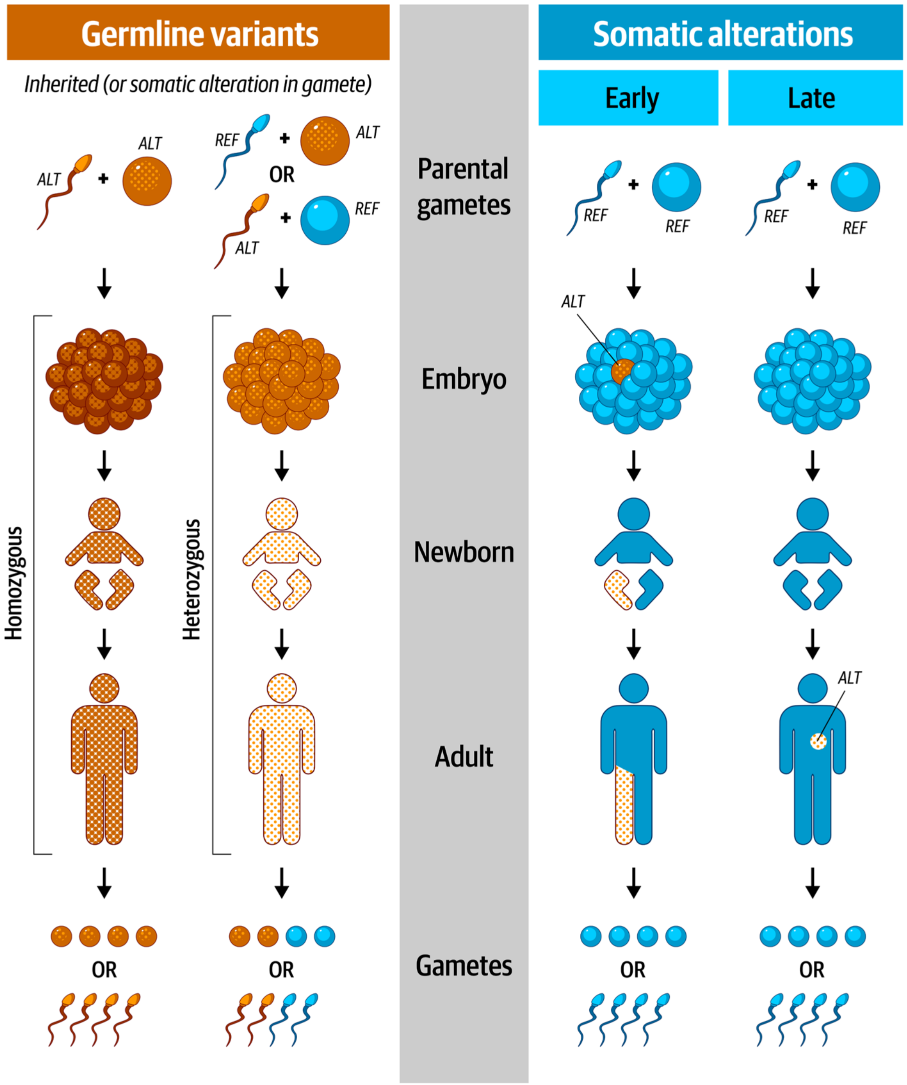


Figure 2.10: Germline variants are present in all cells of the body (left) while somatic alterations are present only in a subset of cells (right).

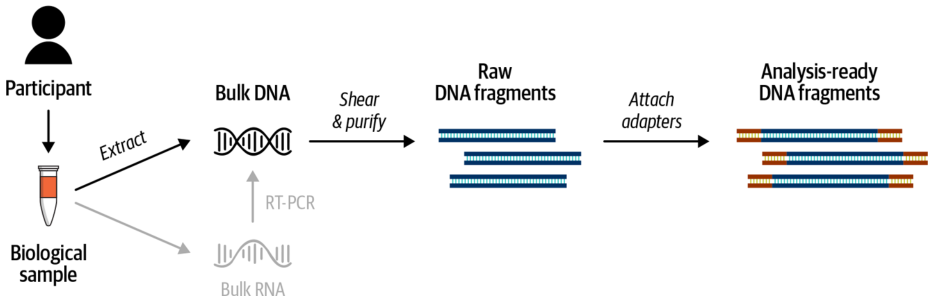


Figure 2.11: Library preparation process for bulk DNA (top); alternative pathway for bulk RNA (bottom).

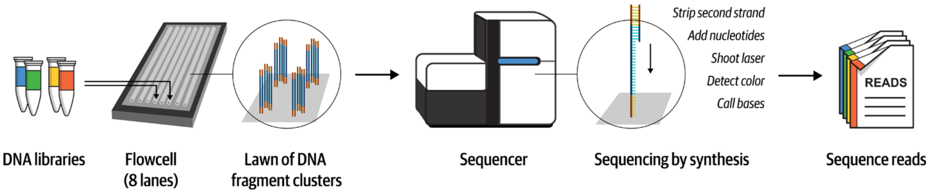
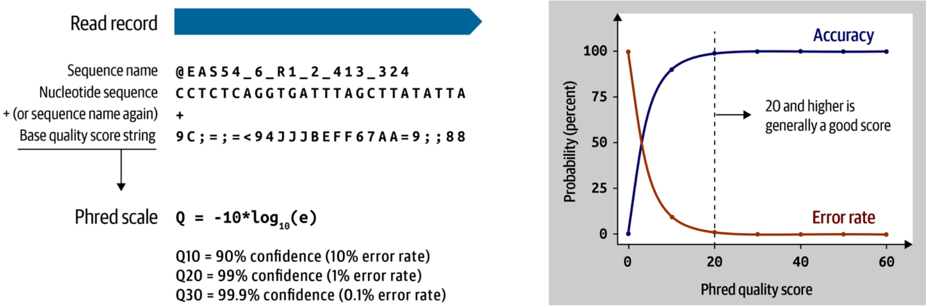


Figure 2.12: Overview of Illumina short read sequencing.

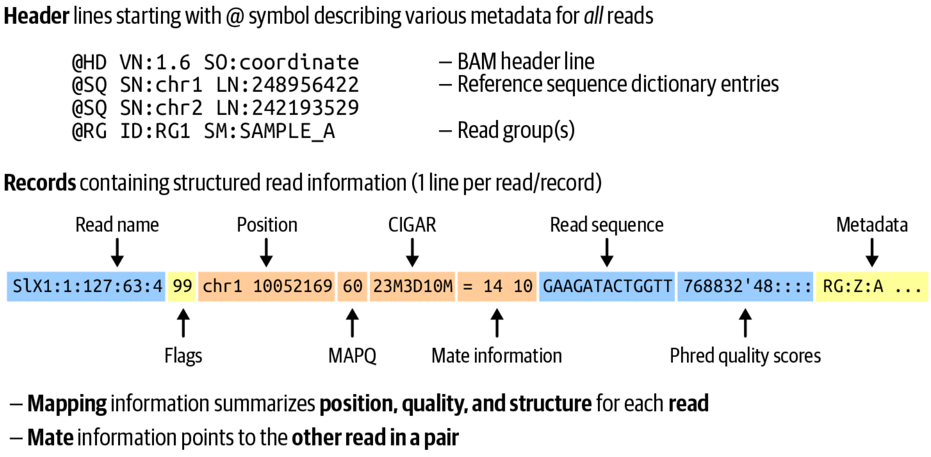


```
@EAS54_6_R1_2_413_324
CCTCTCAGGTGATTTAGCTTATATTA
+
9C;=;=<94JJJBEFF67AA=9;;88
```

$Q = -10*\log_{10}(e)$

Q10 = 90% confidence (10% error rate)
Q20 = 99% confidence (1% error rate)
Q30 = 99.9% confidence (0.1% error rate)

Figure 2.13: FASTQ and Phred scale.


Header lines starting with @ symbol describing various metadata for *all* reads

```
@HD VN:1.6 SO:coordinate       – BAM header line
@SQ SN:chr1 LN:248956422       – Reference sequence dictionary entries
@SQ SN:chr2 LN:242193529
@RG ID:RG1 SM:SAMPLE_A         – Read group(s)
```

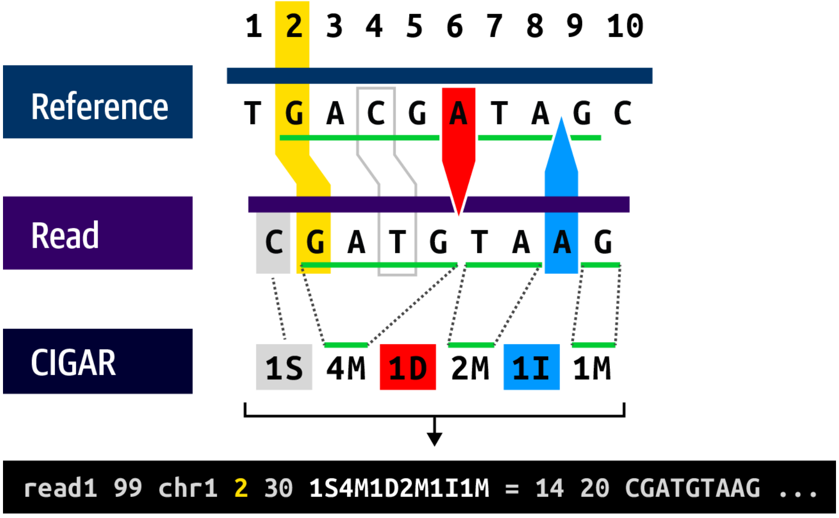
Records containing structured read information (1 line per read/record)

```
SlX1:1:127:63:4 99 chr1 10052169 60 23M3D10M = 14 10 GAAGATACTGGTT 768832'48:::: RG:Z:A ...
```

– **Mapping** information summarizes **position, quality, and structure** for each **read**

– **Mate** information points to the **other read in a pair**

Figure 2.14: Key elements of the SAM format: file header and read record structure.



```
read1 99 chr1 2 30 1S4M1D2M1I1M = 14 20 CGATGTAAG ...
```

Figure 2.15: The CIGAR string describes the structure of the read alignment.

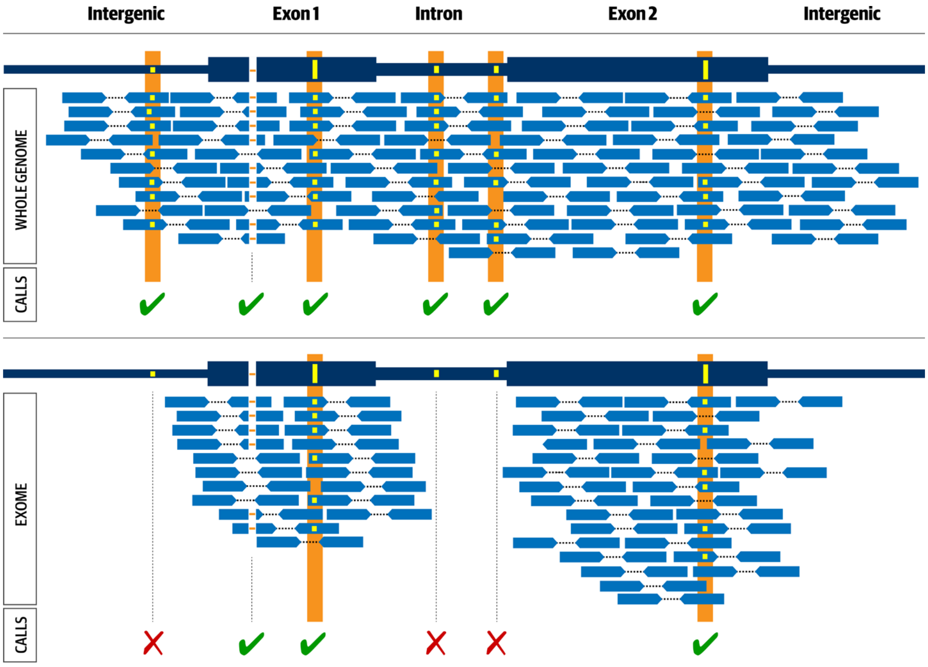


Figure 2.16: Experimental design comparison between whole genome (top) and exome (bottom).

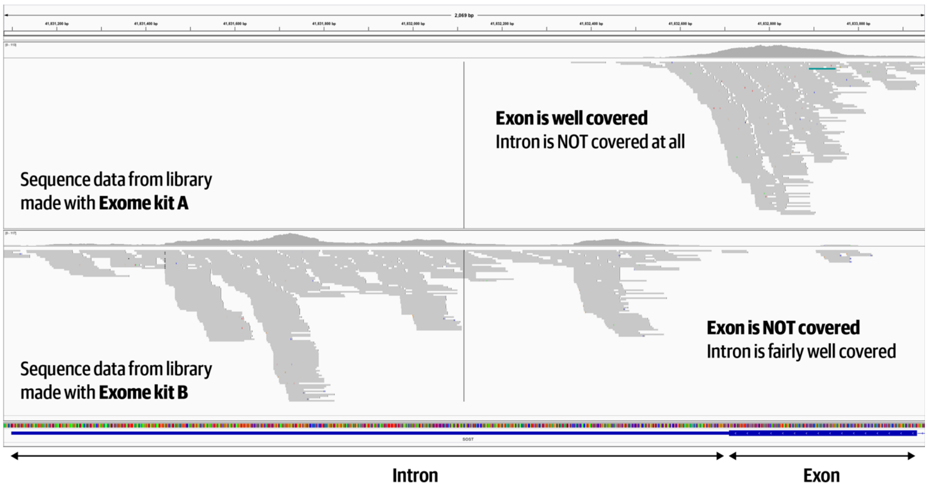

Exon is well covered
Intron is NOT covered at all
Sequence data from library made with Exome kit A
Exon is NOT covered
Intron is fairly well covered
Sequence data from library made with Exome kit B
Intron
Exon

Figure 2.17: Different exome preparation kits can lead to important differences in coverage location and quantity.

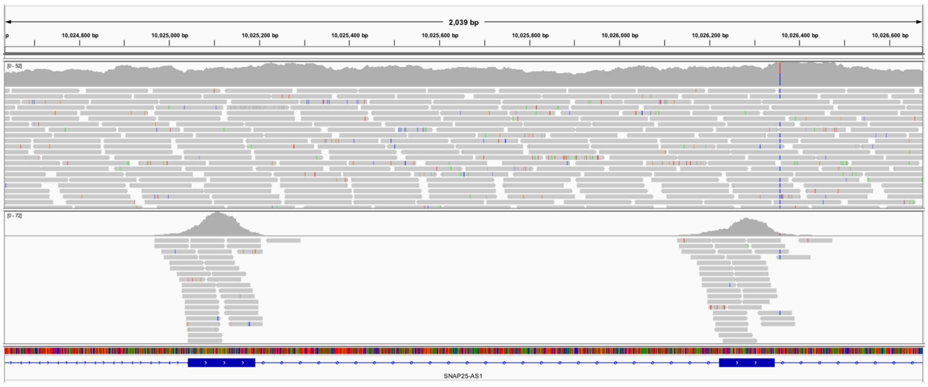


Figure 2.18: Visual appearance of whole genome sequence (WGS, top) and exome sequence (bottom) in a genome browser.

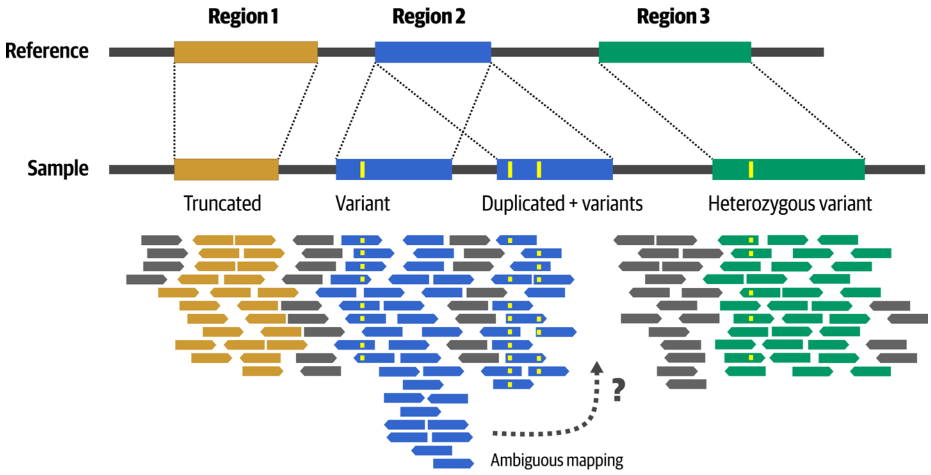


Figure 2.19: Sequence divergence introduces mapping challenges and ambiguity.

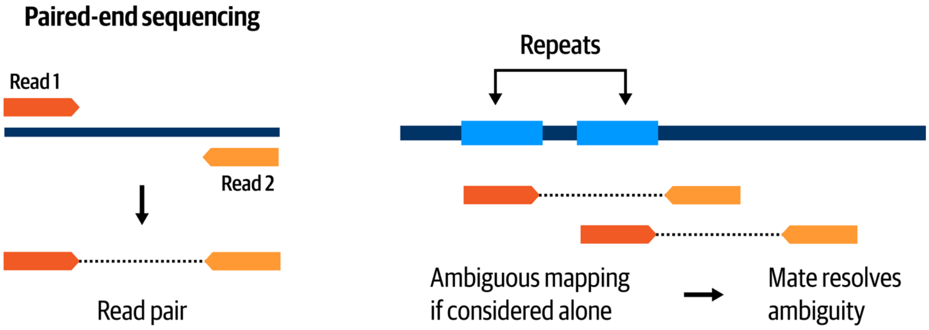
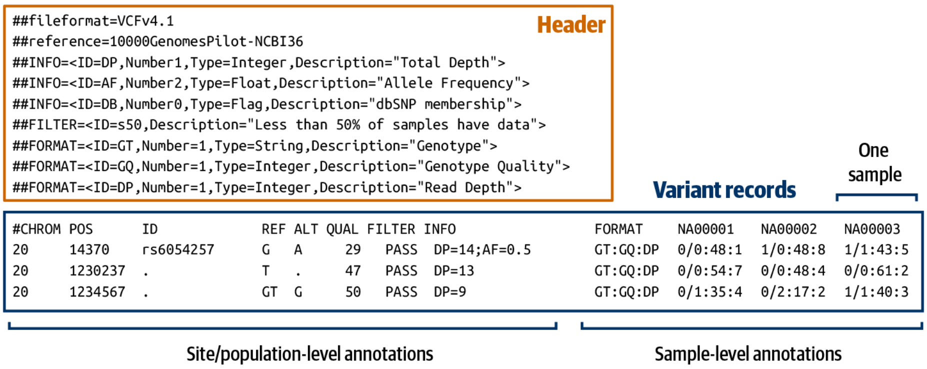


Figure 2.20: Paired-end sequencing helps resolve mapping ambiguity.



```
##fileformat=VCFv4.1
##reference=10000GenomesPilot-NCBI36
##INFO=<ID=DP,Number1,Type=Integer,Description="Total Depth">
##INFO=<ID=AF,Number2,Type=Float,Description="Allele Frequency">
##INFO=<ID=DB,Number0,Type=Flag,Description="dbSNP membership">
##FILTER=<ID=s50,Description="Less than 50% of samples have data">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
##FORMAT=<ID=GQ,Number=1,Type=Integer,Description="Genotype Quality">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Read Depth">
```

```
#CHROM POS      ID         REF ALT QUAL FILTER INFO            FORMAT   NA00001   NA00002   NA00003
20     14370    rs6054257  G   A     29   PASS  DP=14;AF=0.5   GT:GQ:DP 0/0:48:1  1/0:48:8  1/1:43:5
20     1230237  .          T   .     47   PASS  DP=13          GT:GQ:DP 0/0:54:7  0/0:48:4  0/0:61:2
20     1234567  .          GT  G     50   PASS  DP=9           GT:GQ:DP 0/1:35:4  0/2:17:2  1/1:40:3
```

Figure 2.21: Basic structure of a VCF file.

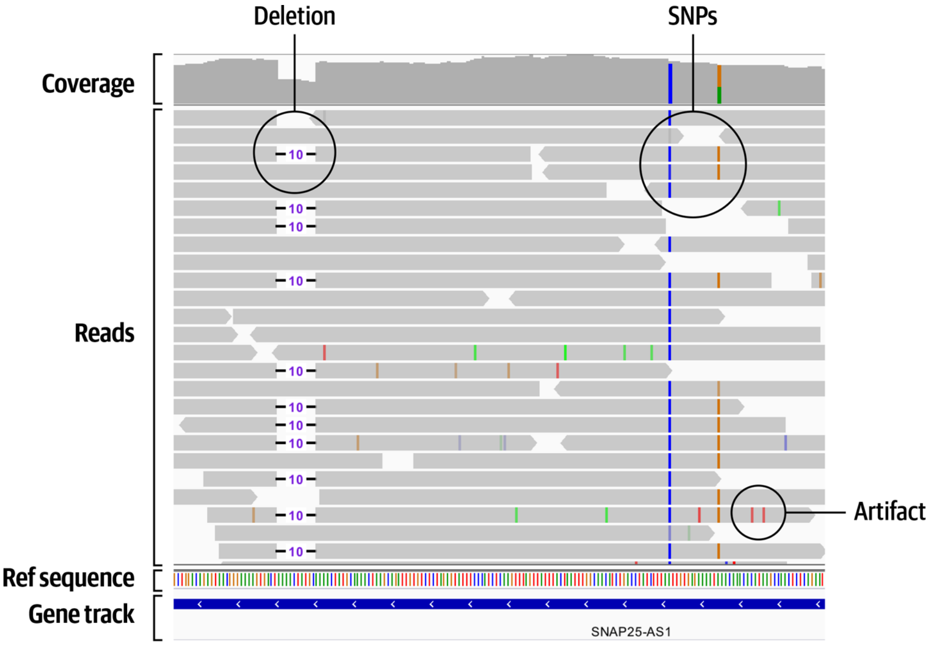


Figure 2.22: Pileup of reads in IGV showing several probable short variants.

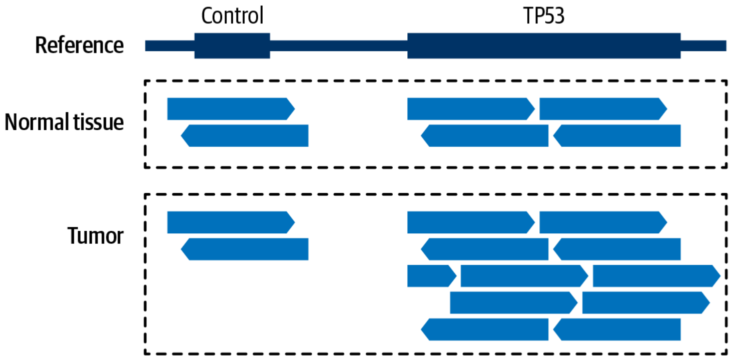


Figure 2.23: Relative amounts of coverage provide evidence for copy-number modeling.

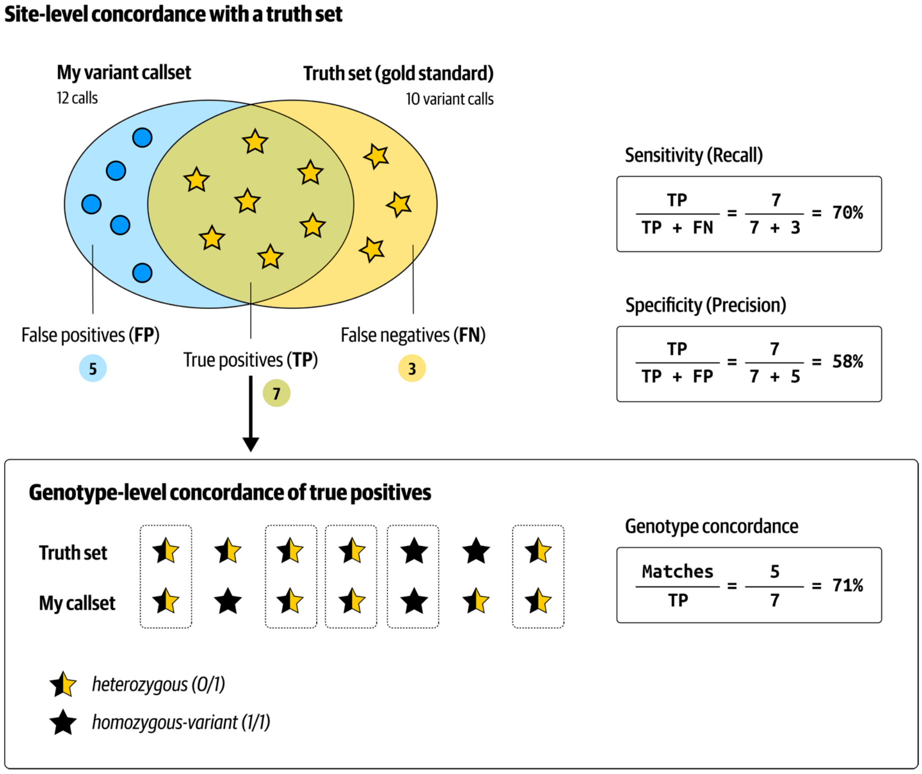


Figure 2.24: Cheat sheet of variant metrics.

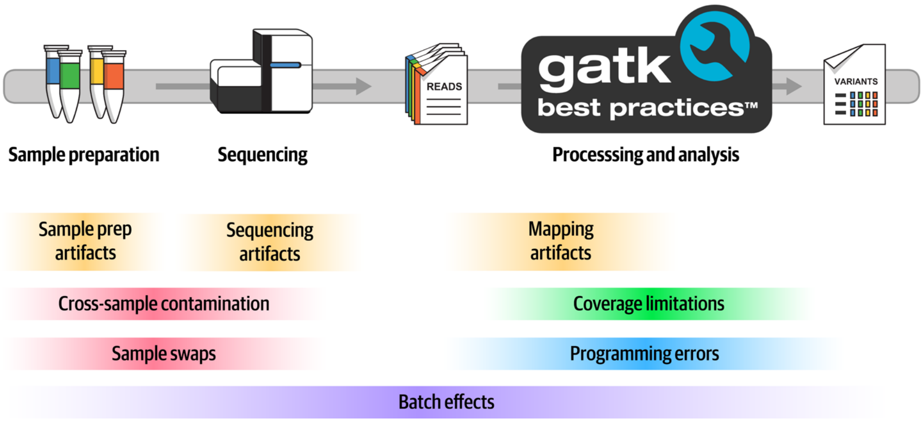


Figure 2.25: Common sources of error in variant discovery.

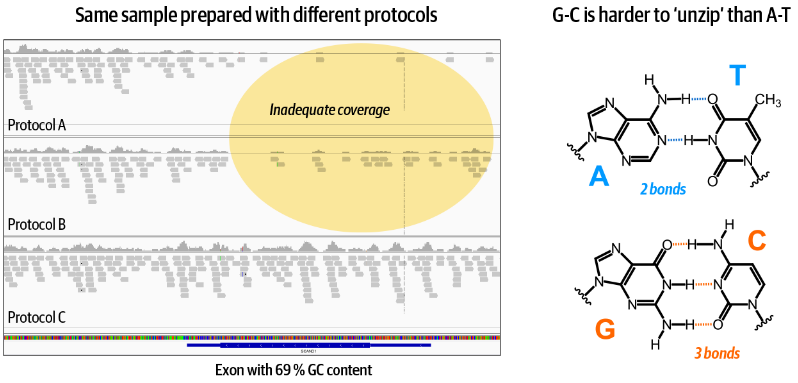

Same sample prepared with different protocols
Inadequate coverage
Protocol A
Protocol B
Protocol C
Exon with 69 % GC content
G-C is harder to 'unzip' than A-T
T
A
2 bonds
C
G
3 bonds

Figure 2.26: Some biochemical properties of the DNA itself cause biases in certain regions.

Chapter 3 Computing Technology Basics for Life Scientists

CPU, GPU, TPU, FPGA, OMG GTFO – no really, just some basic hardware terminology, plus an introduction to key concepts like parallelism, pipelining, containers and virtual machines in fairly plain language.

3.1 Basic Infrastructure Components and Performance Bottlenecks

3.1.1 Types of Processor Hardware: CPU, GPU, TPU, FPGA, OMG

3.1.2 Levels of Compute Organization: Core, Node, Cluster, and Cloud

3.1.3 Addressing Performance Bottlenecks

3.2 Parallel Computing

3.2.1 Parallelizing a Simple Analysis

3.2.2 From Cores to Clusters and Clouds: Many Levels of Parallelism

3.2.3 Trade-Offs of Parallelism: Speed, Efficiency, and Cost

3.3 Pipelining for Parallelization and Automation

3.3.1 Workflow Languages

3.3.2 Popular Pipelining Languages for Genomics

3.3.3 Workflow Management Systems

3.4 Virtualization and the Cloud

3.4.1 VMs and Containers

3.4.2 Introducing the Cloud

3.4.3 Categories of Research Use Cases for Cloud Services

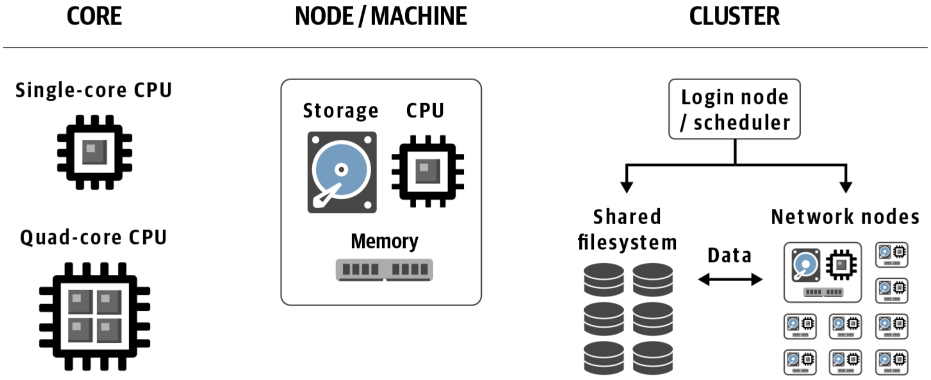


Figure 3.1: Levels of compute organization.

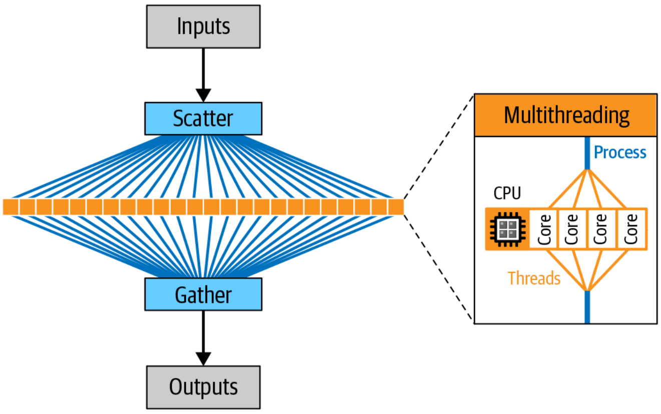


Figure 3.2: Scatter-gather allows parallel execution of tasks on different CPU cores (on a single machine or multiple machines, depending on how it's implemented).

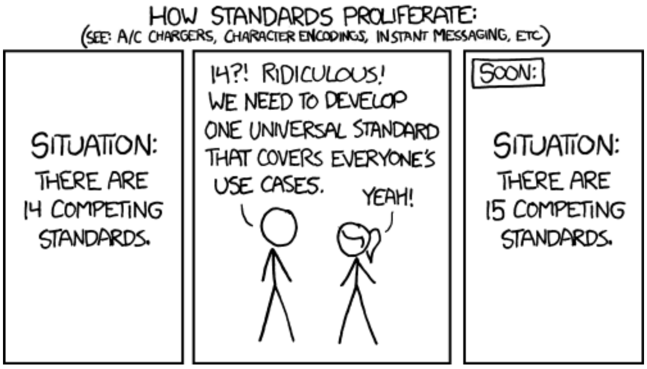

HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

Figure 3.3: XKCD comic on the proliferation of standards (source: https://xkcd.com/927).

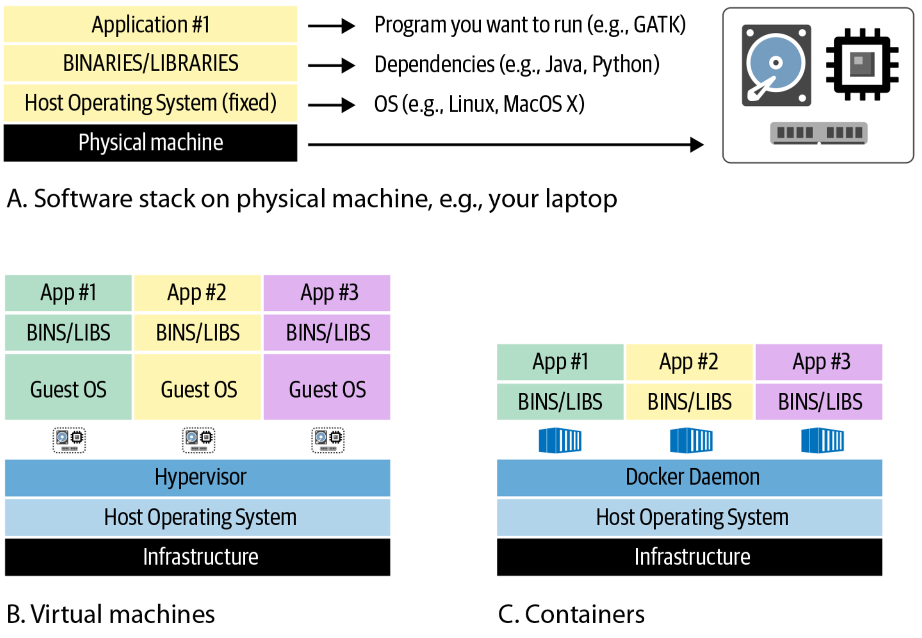


Figure 3.4: A) The software stack installed on a physical machine; B) a system hosting multiple VMs; C) a system hosting multiple containers.

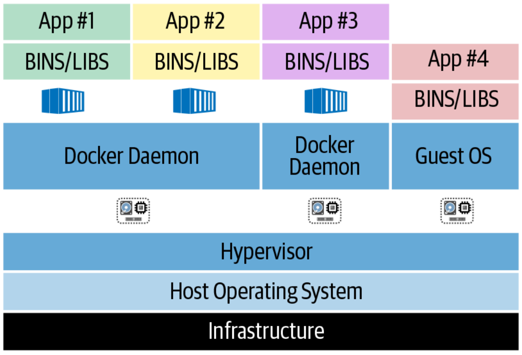


Figure 3.5: A system with three VMs: the one on the left is running two containers, serving App 1 and App 2; the middle is running a single container, serving App 3; the right is serving App 4 directly (no container).

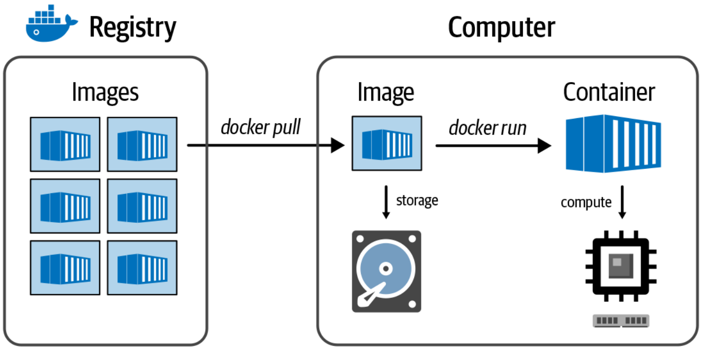


Figure 3.6: The relationship between registry, image, and container.

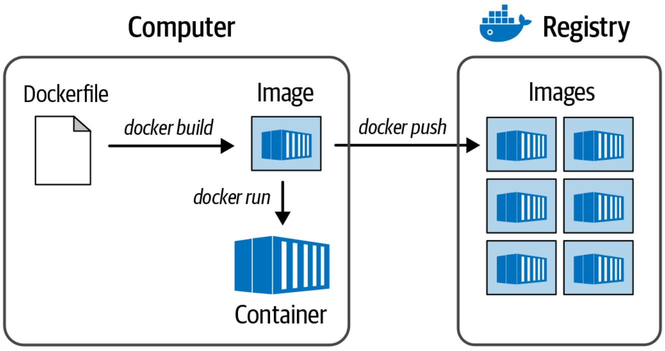


Figure 3.7: The process for creating a Docker image.

Chapter 4 First Steps in the Cloud

Finally we get to do some hands-on work (on Google Cloud). Set up an account, get free credits, practice managing data in storage buckets and interacting with a Docker container, get a nice custom VM set up to do some genomics.

4.1 Setting Up Your Google Cloud Account and First Project

4.1.1 Creating a Project

4.1.2 Checking Your Billing Account and Activating Free Credits

4.2 Running Basic Commands in Google Cloud Shell

4.2.1 Logging in to the Cloud Shell VM

4.2.2 Using gsutil to Access and Manage Files

4.2.3 Pulling a Docker Image and Spinning Up the Container

4.2.4 Mounting a Volume to Access the Filesystem from Within the Container

4.3 Setting Up Your Own Custom VM

4.3.1 Creating and Configuring Your VM Instance

4.3.2 Logging into Your VM by Using SSH

4.3.3 Checking Your Authentication

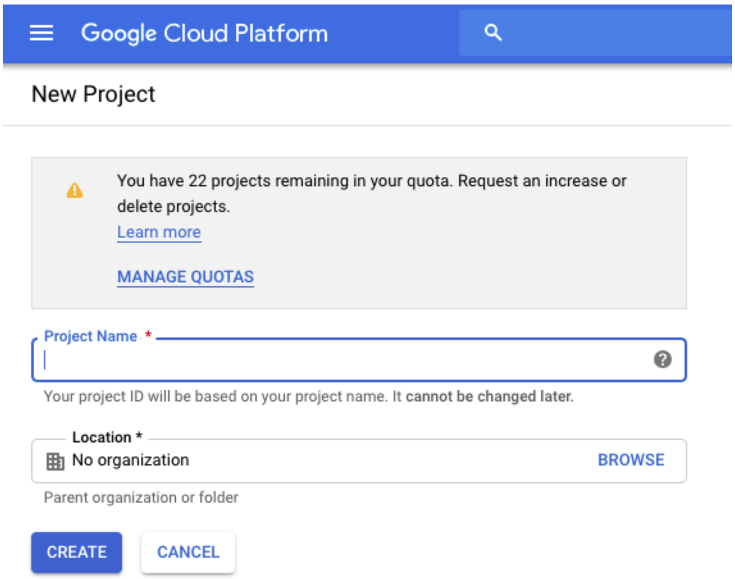
4.3.4 Copying the Book Materials to Your VM

4.3.5 Installing Docker on Your VM

4.3.6 Setting Up the GATK Container Image

4.3.7 Stopping Your VM. . . to Stop It from Costing You Money

4.4 Configuring IGV to Read Data from GCS Buckets



New Project

You have 22 projects remaining in your quota. Request an increase or delete projects.
Learn more

MANAGE QUOTAS

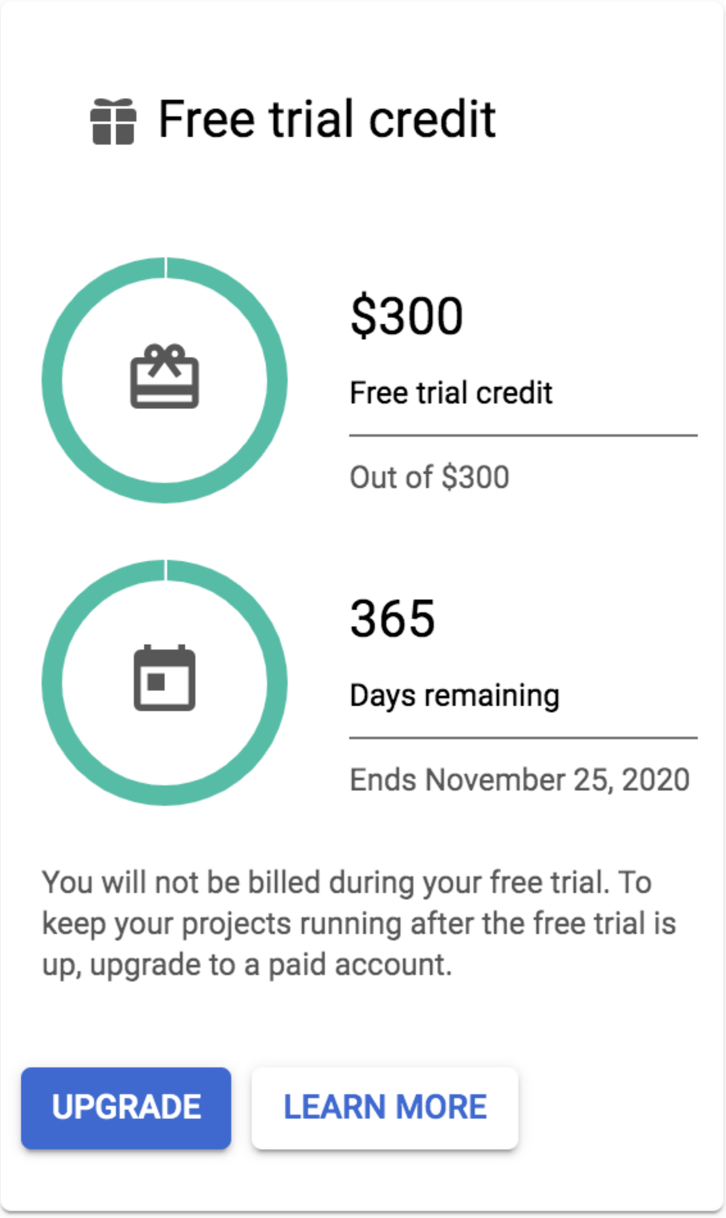
Project Name *

Your project ID will be based on your project name. It **cannot be changed later**.

Location *
No organization BROWSE

Parent organization or folder

Figure 4.1: Creating a new project.



Free trial credit

$300

Free trial credit

Out of $300

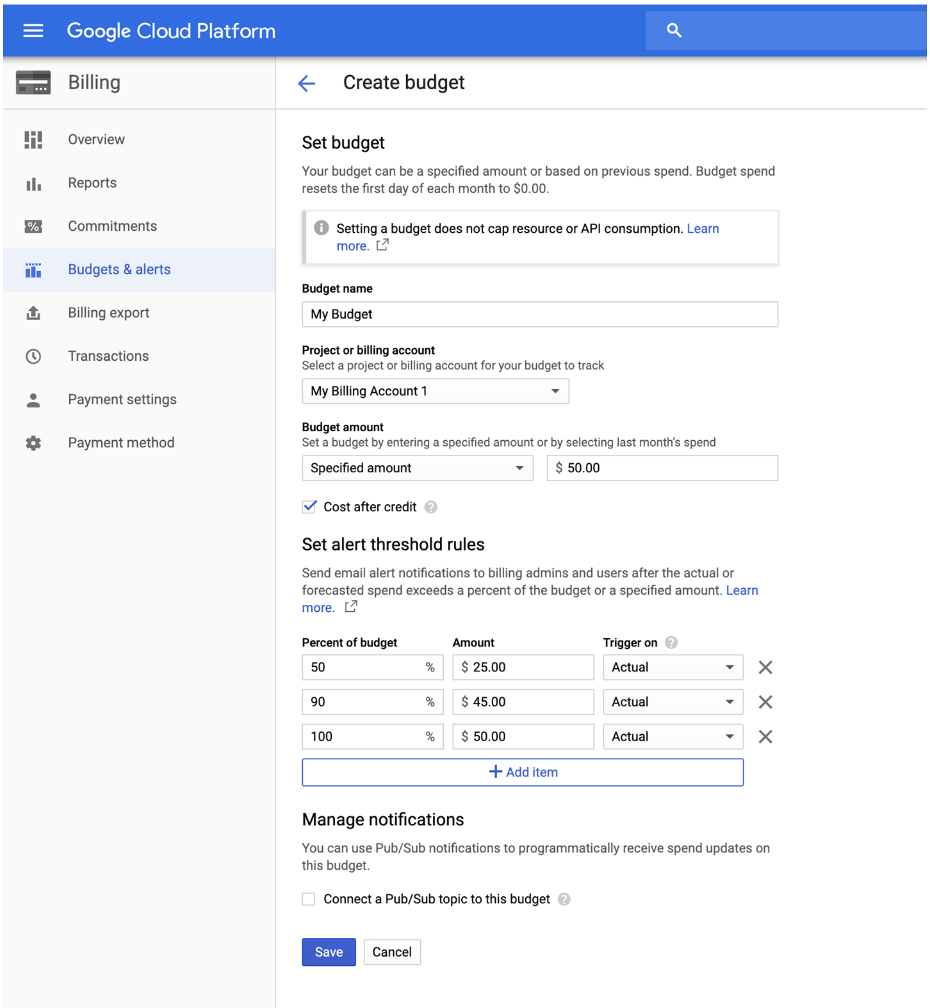
365

Days remaining

Ends November 25, 2020

You will not be billed during your free trial. To keep your projects running after the free trial is up, upgrade to a paid account.

Figure 4.2: The panel in the Billing console summarizing free trial credits availability.



Create budget

Set budget

Your budget can be a specified amount or based on previous spend. Budget spend resets the first day of each month to $0.00.

Setting a budget does not cap resource or API consumption. Learn more.

Budget name
My Budget

Project or billing account
Select a project or billing account for your budget to track
My Billing Account 1

Budget amount
Set a budget by entering a specified amount or by selecting last month's spend
Specified amount $ 50.00

☑ Cost after credit

Set alert threshold rules

Send email alert notifications to billing admins and users after the actual or forecasted spend exceeds a percent of the budget or a specified amount. Learn more.

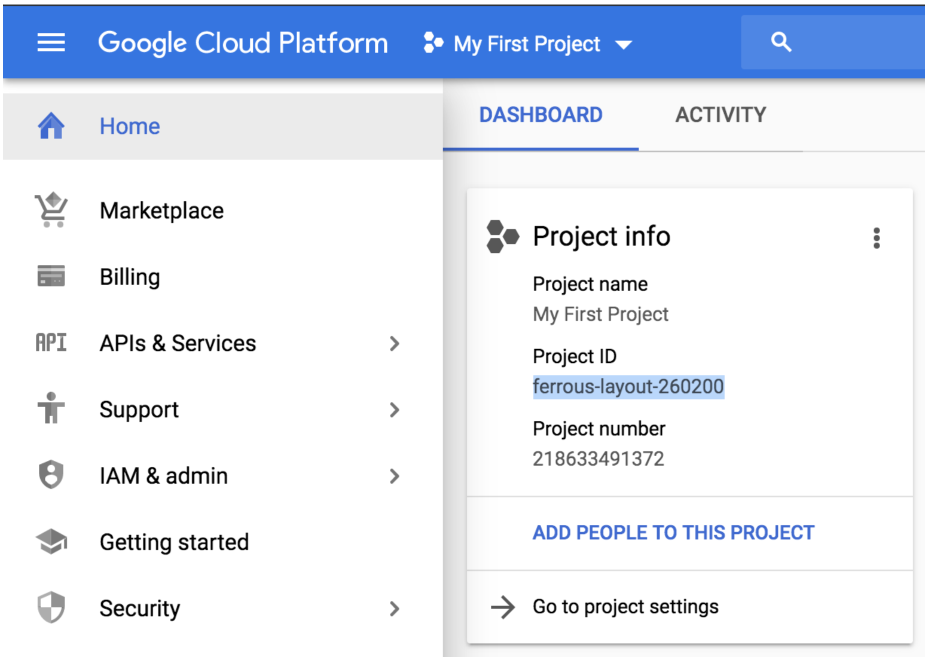
Percent of budget	Amount	Trigger on
50 %	$ 25.00	Actual
90 %	$ 45.00	Actual
100 %	$ 50.00	Actual

Manage notifications

You can use Pub/Sub notifications to programmatically receive spend updates on this budget.

□ Connect a Pub/Sub topic to this budget

Figure 4.3: Budget and alert threshold administration.



Project name
My First Project

Project ID
ferrous-layout-260200

Project number
218633491372

Figure 4.4: Location of the Project ID in the GCP console.

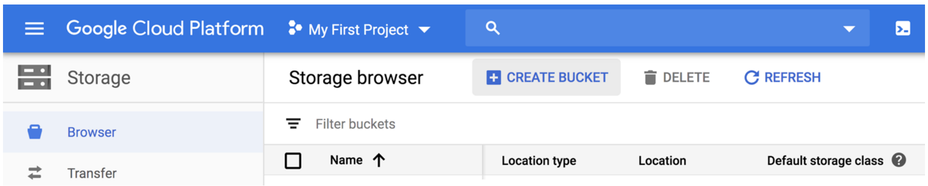
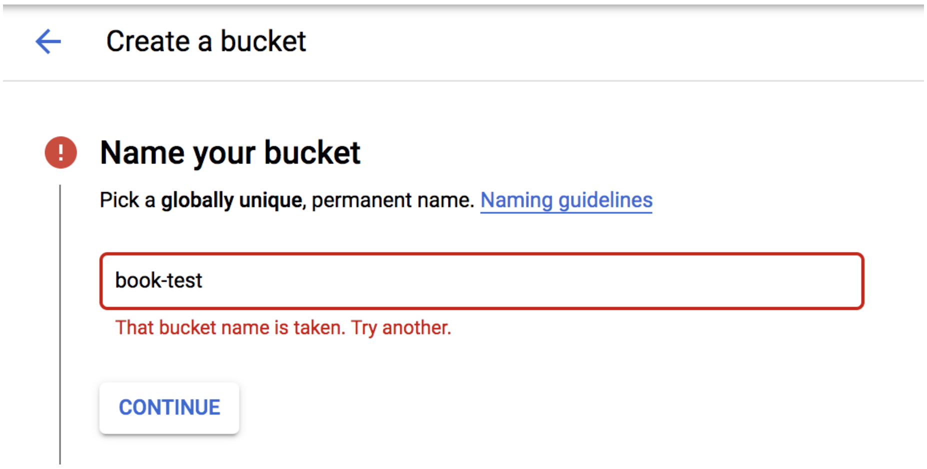


Figure 4.5: GCP console storage browser.



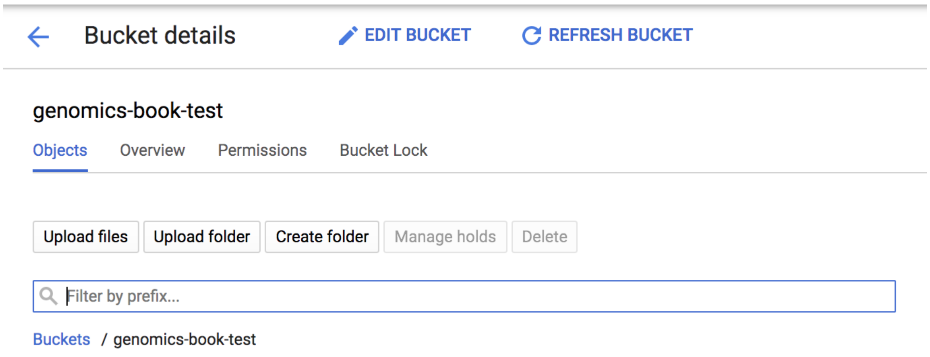
Name your bucket

Pick a **globally unique**, permanent name. Naming guidelines

book-test

That bucket name is taken. Try another.

Figure 4.6: Naming your bucket.



Bucket details

genomics-book-test

Buckets / genomics-book-test

Figure 4.7: Viewing the contents of your bucket.

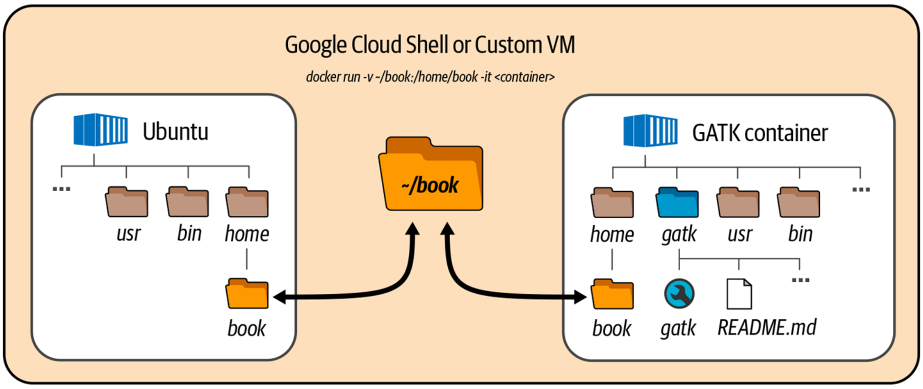
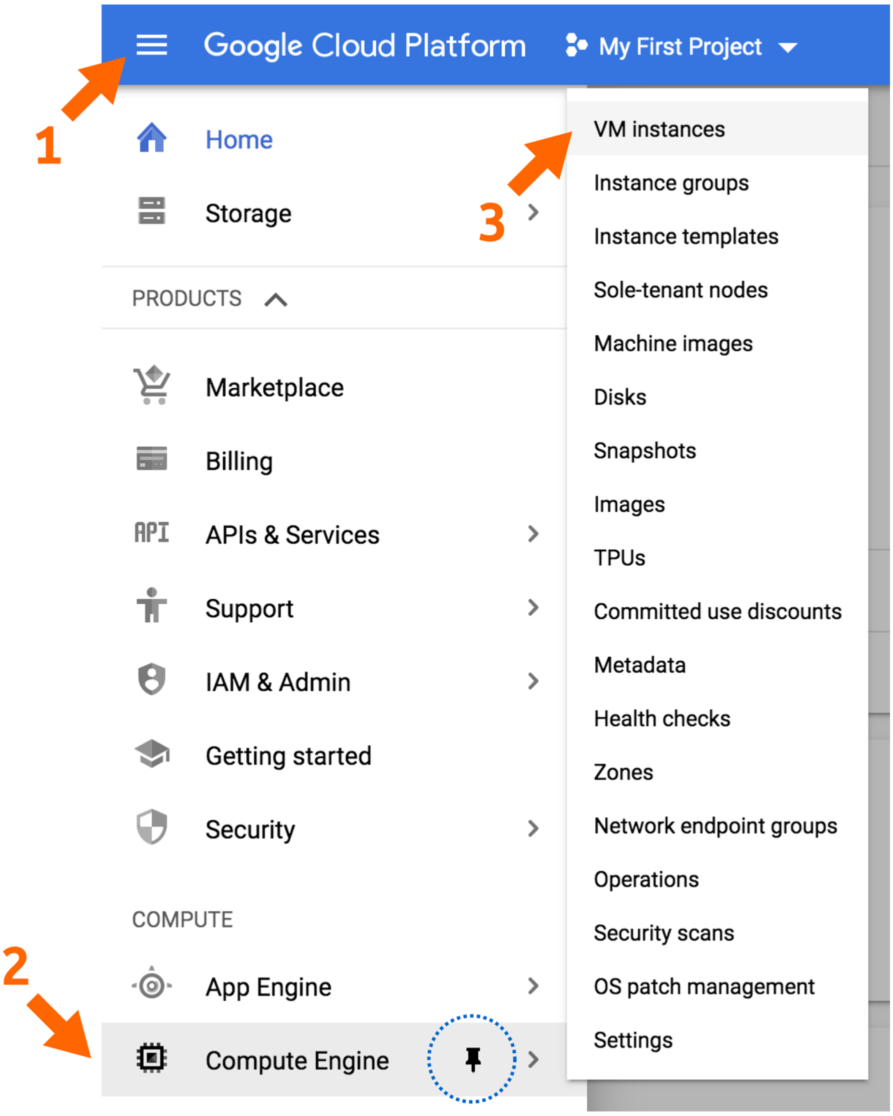


Figure 4.8: Mounting a directory from your Google Cloud Shell VM into a Docker container: Ubuntu container used in this chapter (left); GATK container introduced in First Steps with GATK (right).



TIP: Click the pin symbol to "pin" this service in the shortcuts menu

Figure 4.9: Compute Engine menu showing the VM instances menu item.

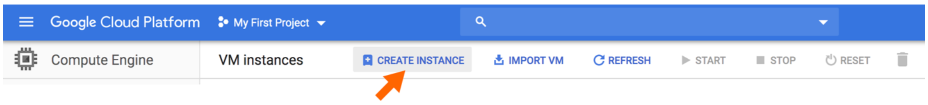
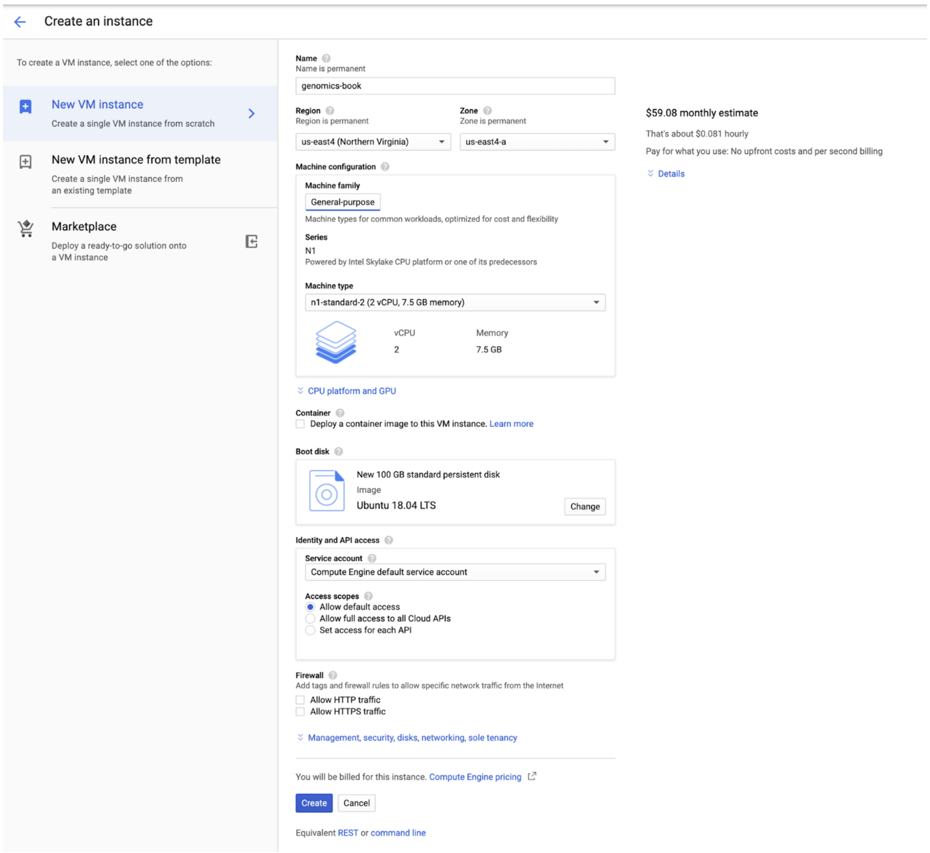


Figure 4.10: Create a VM instance.



Create an instance

To create a VM instance, select one of the options:

New VM instance
Create a single VM instance from scratch

New VM instance from template
Create a single VM instance from an existing template

Marketplace
Deploy a ready-to-go solution onto a VM instance

Name
Name is permanent
genomics-book

Region
Region is permanent
us-east4 (Northern Virginia)

Zone
Zone is permanent
us-east4-a

Machine configuration

Machine family
General-purpose
Machine types for common workloads, optimized for cost and flexibility

Series
N1
Powered by Intel Skylake CPU platform or one of its predecessors

Machine type
n1-standard-2 (2 vCPU, 7.5 GB memory)

vCPU	Memory
2	7.5 GB

CPU platform and GPU

Container
□ Deploy a container image to this VM instance. Learn more

Boot disk
New 100 GB standard persistent disk
Image
Ubuntu 18.04 LTS
Change

Identity and API access

Service account
Compute Engine default service account

Access scopes
● Allow default access
○ Allow full access to all Cloud APIs
○ Set access for each API

Firewall
Add tags and firewall rules to allow specific network traffic from the Internet
□ Allow HTTP traffic
□ Allow HTTPS traffic

Management, security, disks, networking, sole tenancy

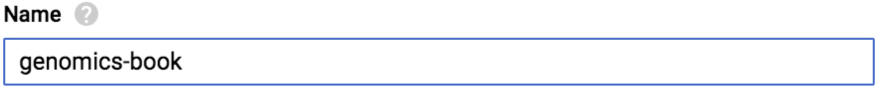
You will be billed for this instance. Compute Engine pricing

Create Cancel

Equivalent REST or command line

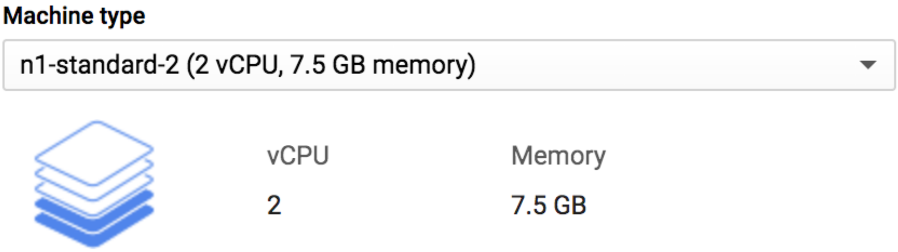
$59.08 monthly estimate
That's about $0.081 hourly
Pay for what you use: No upfront costs and per second billing
Details

Figure 4.11: The VM instance configuration panel.



Name
genomics-book

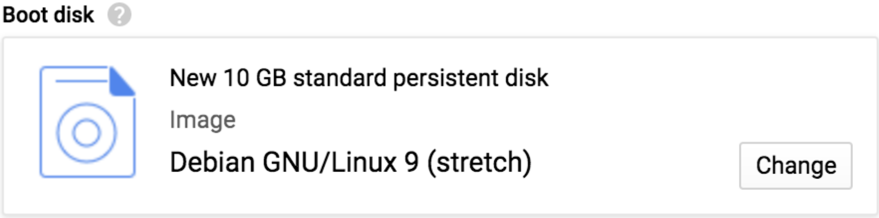
Figure 4.12: Name your VM instance.



Machine type
n1-standard-2 (2 vCPU, 7.5 GB memory)

vCPU	Memory
2	7.5 GB

Figure 4.13: Selecting a machine type.



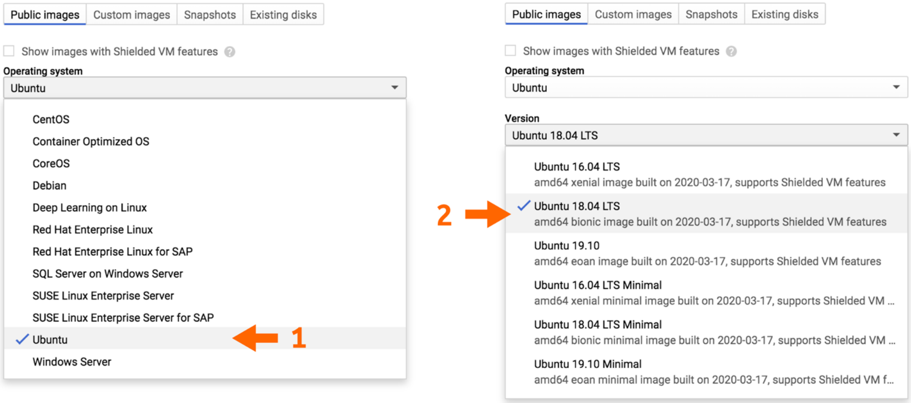
Boot disk

New 10 GB standard persistent disk

Image

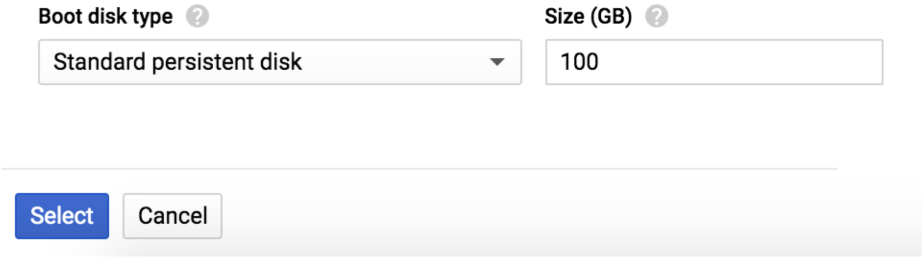
Debian GNU/Linux 9 (stretch)

Figure 4.14: Choosing a boot disk size and image.



Public images
Custom images
Snapshots
Existing disks
□ Show images with Shielded VM features
Operating system
Ubuntu
CentOS
Container Optimized OS
CoreOS
Debian
Deep Learning on Linux
Red Hat Enterprise Linux
Red Hat Enterprise Linux for SAP
SQL Server on Windows Server
SUSE Linux Enterprise Server
SUSE Linux Enterprise Server for SAP
✓ Ubuntu
1
Windows Server
Public images
Custom images
Snapshots
Existing disks
□ Show images with Shielded VM features
Operating system
Ubuntu
Version
Ubuntu 18.04 LTS
Ubuntu 16.04 LTS
amd64 xenial image built on 2020-03-17, supports Shielded VM features
2
✓ Ubuntu 18.04 LTS
amd64 bionic image built on 2020-03-17, supports Shielded VM features
Ubuntu 19.10
amd64 eoan image built on 2020-03-17, supports Shielded VM features
Ubuntu 16.04 LTS Minimal
amd64 xenial minimal image built on 2020-03-17, supports Shielded VM ...
Ubuntu 18.04 LTS Minimal
amd64 bionic minimal image built on 2020-03-17, supports Shielded VM ...
Ubuntu 19.10 Minimal
amd64 eoan minimal image built on 2020-03-17, supports Shielded VM f...

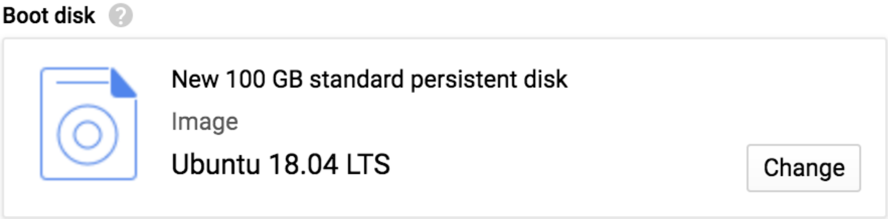
Figure 4.15: Selecting a base image.



Boot disk type: Standard persistent disk

Size (GB): 100

Figure 4.16: Setting the boot disk size.



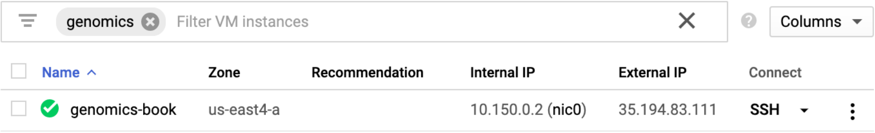
Boot disk

New 100 GB standard persistent disk

Image

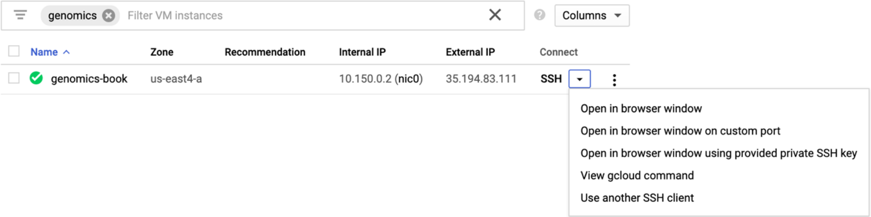
Ubuntu 18.04 LTS

Figure 4.17: The updated boot disk selection.



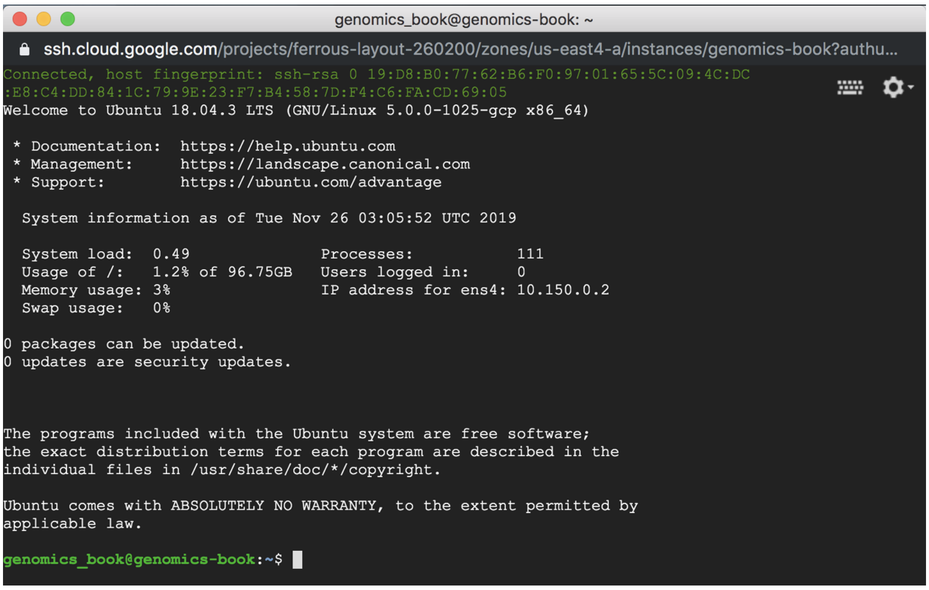
□	Name	Zone	Recommendation	Internal IP	External IP	Connect
□	✓ genomics-book	us-east4-a		10.150.0.2 (nic0)	35.194.83.111	SSH

Figure 4.18: Viewing the VM status.



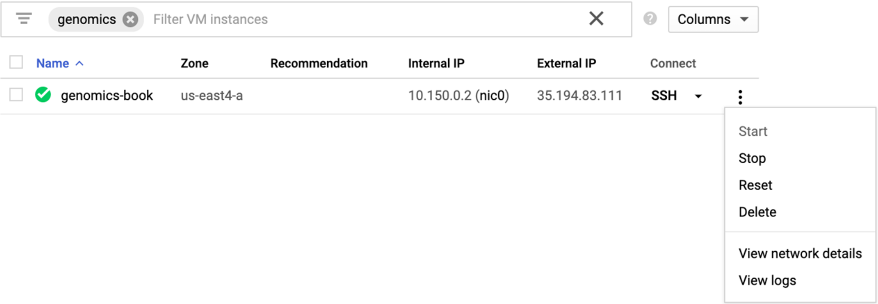
□	Name	Zone	Recommendation	Internal IP	External IP	Connect
□	✓ genomics-book	us-east4-a		10.150.0.2 (nic0)	35.194.83.111	SSH

Figure 4.19: Options for SSHing into your VM.


genomics_book@genomics-book: ~
ssh.cloud.google.com/projects/ferrous-layout-260200/zones/us-east4-a/instances/genomics-book?authu...

```
Connected, host fingerprint: ssh-rsa 0 19:D8:B0:77:62:B6:F0:97:01:65:5C:09:4C:DC
:E8:C4:DD:84:1C:79:9E:23:F7:B4:58:7D:F4:C6:FA:CD:69:05
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 5.0.0-1025-gcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Tue Nov 26 03:05:52 UTC 2019

  System load:  0.49              Processes:           111
  Usage of /:   1.2% of 96.75GB   Users logged in:     0
  Memory usage: 3%                IP address for ens4: 10.150.0.2
  Swap usage:   0%

0 packages can be updated.
0 updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

genomics_book@genomics-book:~$ 
```

Figure 4.20: VM instance terminal.



□	Name	Zone	Recommendation	Internal IP	External IP	Connect
□	✓ genomics-book	us-east4-a		10.150.0.2 (nic0)	35.194.83.111	SSH

Figure 4.21: Stopping, starting, or deleting your VM instance.

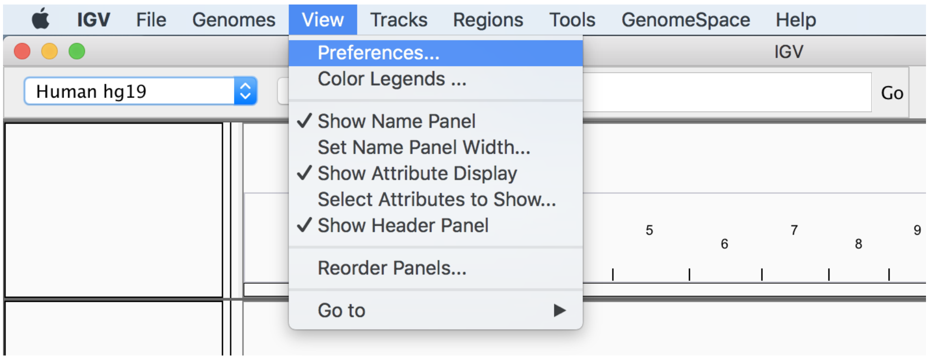
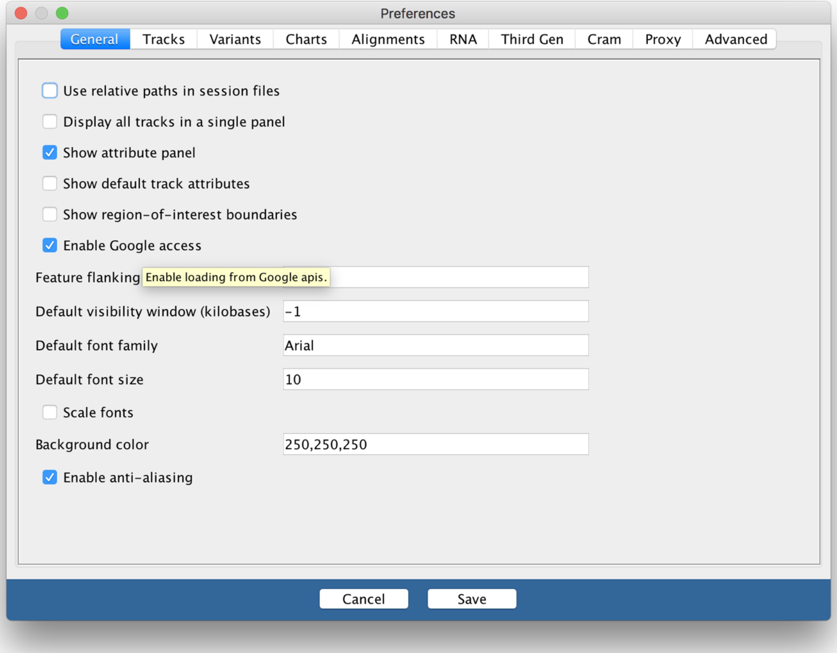


Figure 4.22: Selecting the Preferences menu item.



☐ Use relative paths in session files

☐ Display all tracks in a single panel

☑ Show attribute panel

☐ Show default track attributes

☐ Show region-of-interest boundaries

☑ Enable Google access

Feature flanking Enable loading from Google apis.

Default visibility window (kilobases) -1

Default font family Arial

Default font size 10

☐ Scale fonts

Background color 250,250,250

☑ Enable anti-aliasing

Figure 4.23: The IGV Preferences pane.

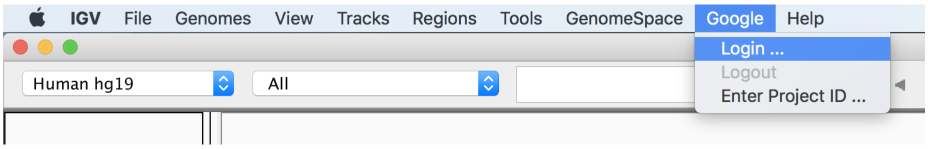


Figure 4.24: Selecting the Google Login menu item.

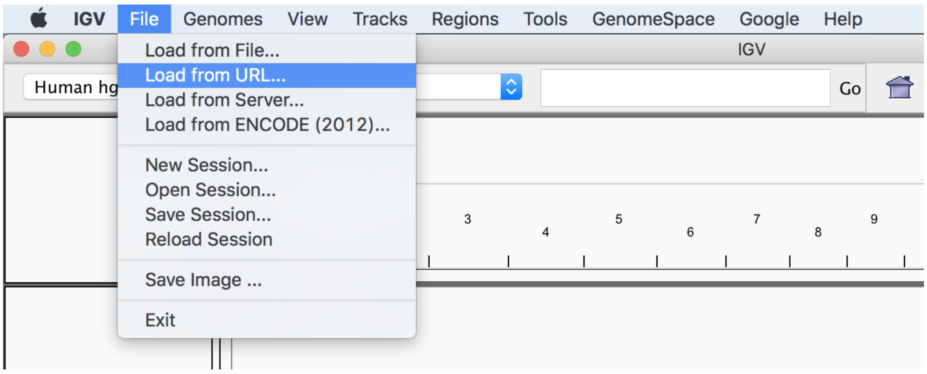
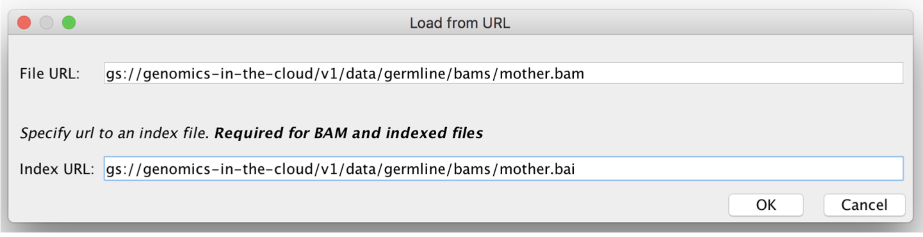


Figure 4.25: The Load from URL menu item.


Load from URL

File URL: gs://genomics-in-the-cloud/v1/data/germline/bams/mother.bam

Specify url to an index file. ***Required for BAM and indexed files***

Index URL: gs://genomics-in-the-cloud/v1/data/germline/bams/mother.bai

OK Cancel

Figure 4.26: The Load from URL dialog box.

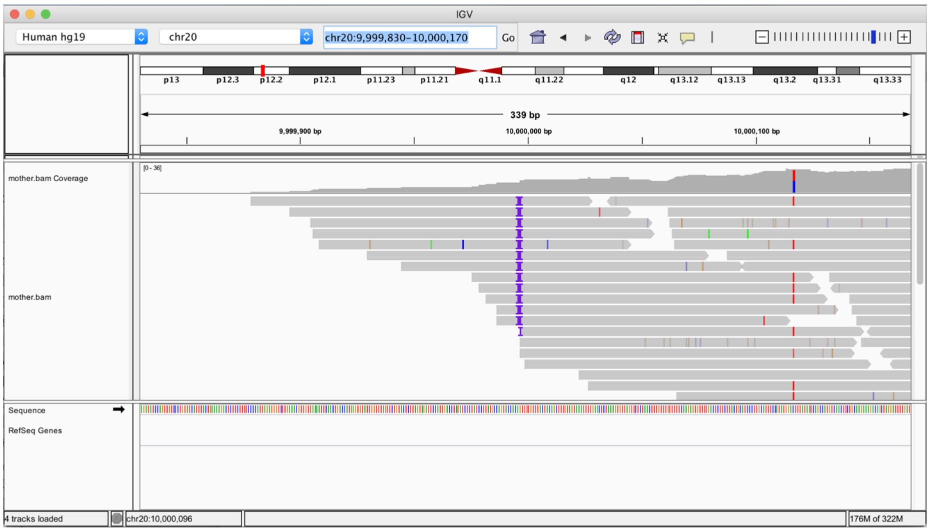


Figure 4.27: IGV view of a BAM file located in a GCS bucket.

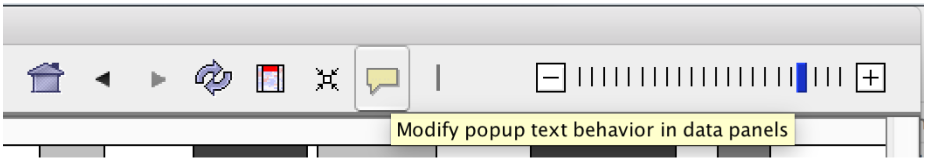
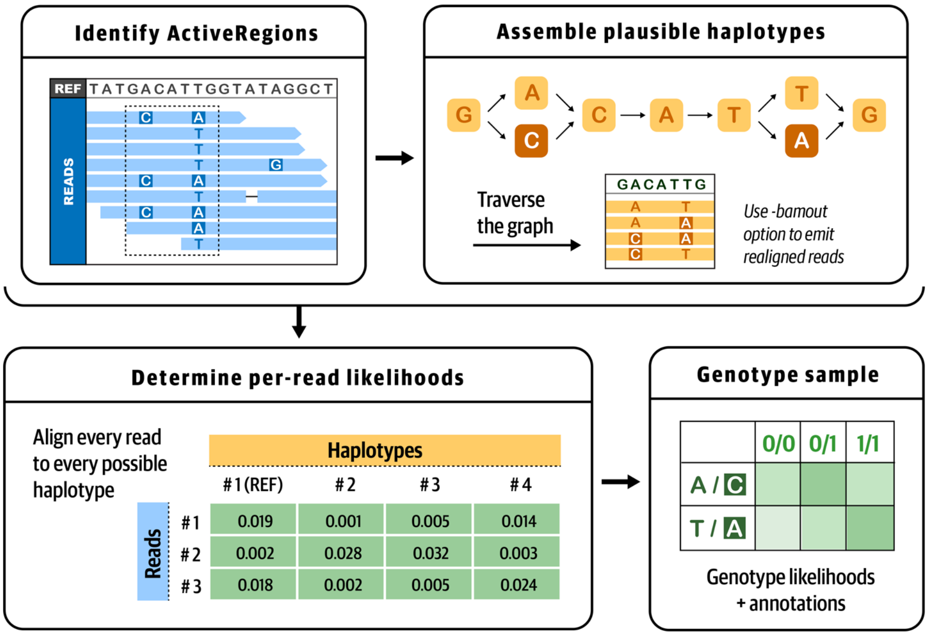


Figure 4.28: Changing the behavior of the detail viewer from "on Hover" to "on Click."

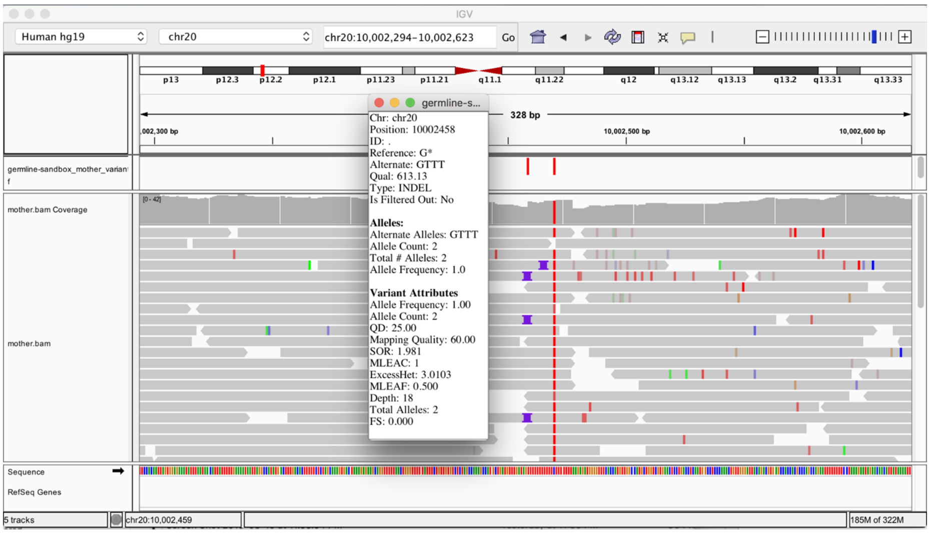
Chapter 5 First Steps with GATK

Let's meet the workhorse of genomics! We start with a general overview, requirements, command line syntax, the usual – then dive into calling variants with HaplotypeCaller, plus some visual troubleshooting and variant filtering concepts.


Identify ActiveRegions
REF TATGACATTGGTATAGGCT
READS
Assemble plausible haplotypes
G A C C A T T A G
Traverse the graph
GACATTG
Use -bamout option to emit realigned reads
Determine per-read likelihoods
Align every read to every possible haplotype
Reads
Genotype sample
0/0 0/1 1/1
A / C
T / A
Genotype likelihoods + annotations

	Haplotypes #1 (REF)	#2	#3	#4
#1	0.019	0.001	0.005	0.014
#2	0.002	0.028	0.032	0.003
#3	0.018	0.002	0.005	0.024

Figure 5.1: The four stages of HaplotypeCaller's operation.

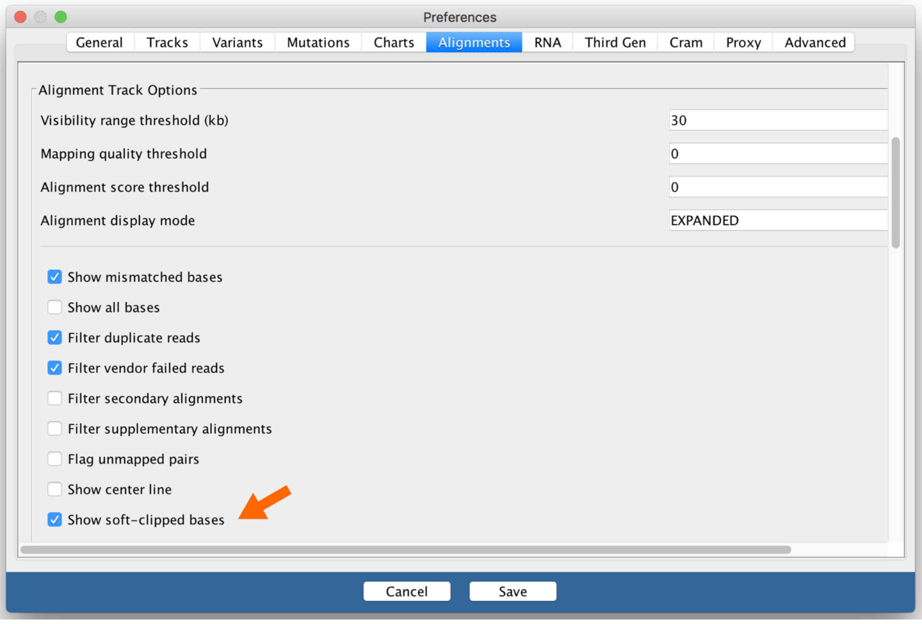


Chr: chr20
Position: 10002458
ID: .
Reference: G*
Alternate: GTTT
Qual: 613.13
Type: INDEL
Is Filtered Out: No

Alleles:
Alternate Alleles: GTTT
Allele Count: 2
Total # Alleles: 2
Allele Frequency: 1.0

Variant Attributes
Allele Frequency: 1.00
Allele Count: 2
QD: 25.00
Mapping Quality: 60.00
SOR: 1.981
MLEAC: 1
ExcessHet: 3.0103
MLEAF: 0.500
Depth: 18
Total Alleles: 2
FS: 0.000

Figure 5.2: The original BAM file and output VCF file loaded in IGV.



Alignment Track Options

Visibility range threshold (kb) 30

Mapping quality threshold 0

Alignment score threshold 0

Alignment display mode EXPANDED

☑ Show mismatched bases

□ Show all bases

☑ Filter duplicate reads

☑ Filter vendor failed reads

□ Filter secondary alignments

□ Filter supplementary alignments

□ Flag unmapped pairs

□ Show center line

☑ Show soft-clipped bases

Figure 5.3: IGV alignment settings.

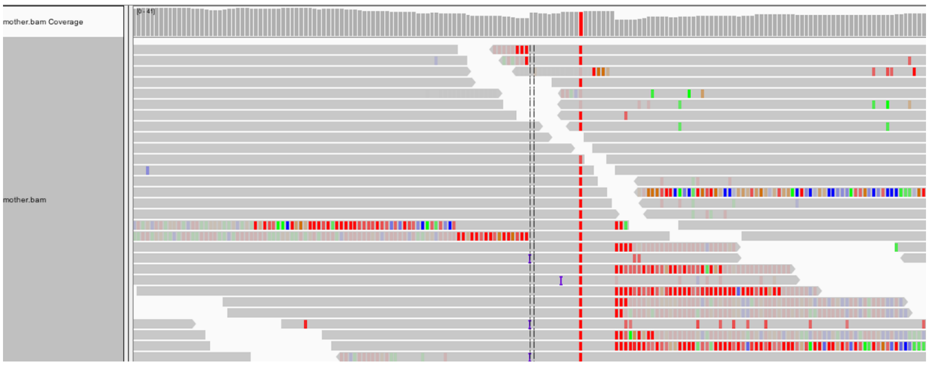


Figure 5.4: Turning on the display of soft clips shows a lot of information that was hidden.

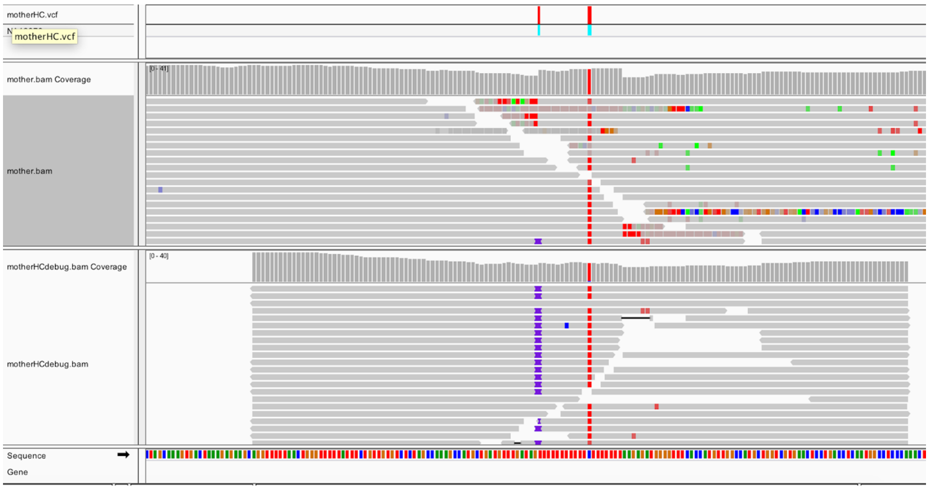


Figure 5.5: Realigned reads in the bamout file (bottom track).

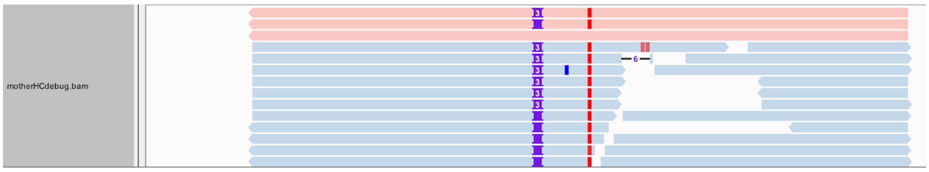


Figure 5.6: Bamout shows artificial haplotypes constructed by HaplotypeCaller.

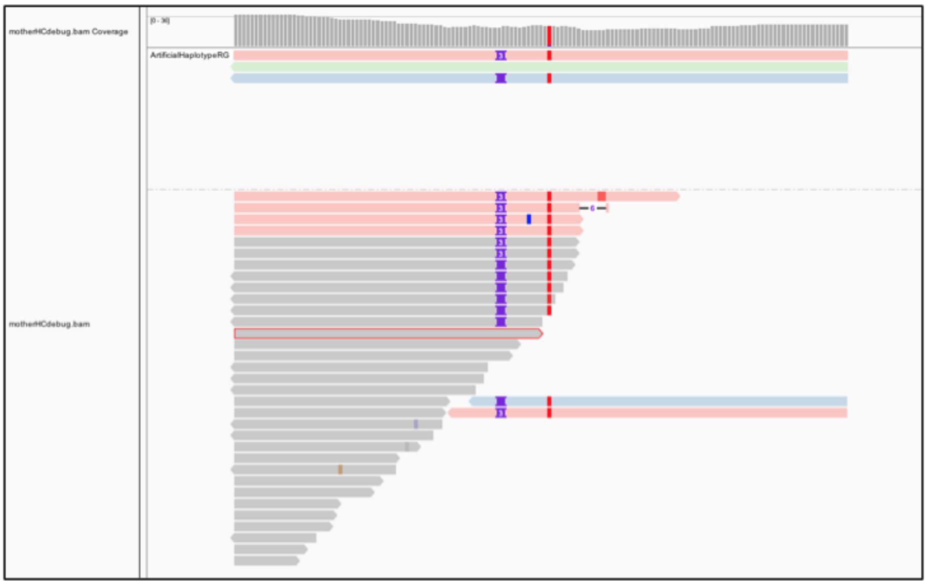


Figure 5.7: Bamout shows support per haplotype.

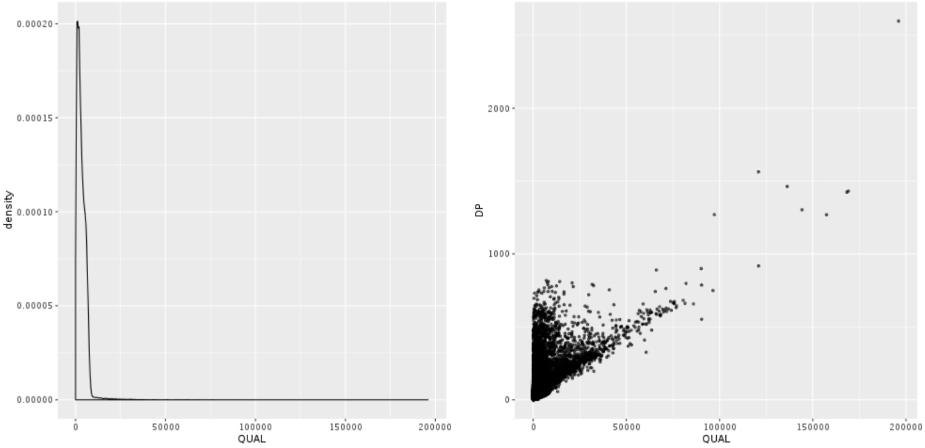


Figure 5.8: Density plot of QUAL (left); scatter plot of QUAL versus DP (right).

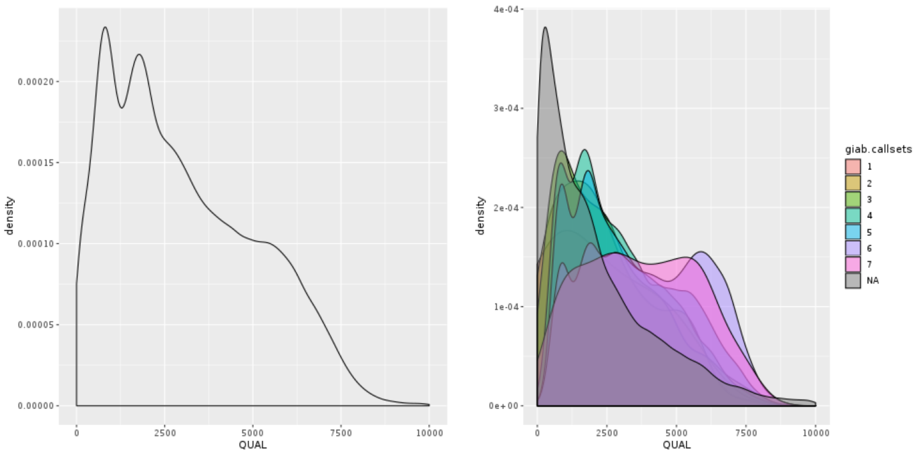


Figure 5.9: Density plot of QUAL: all calls together (left); stratified by callsets annotation (right).

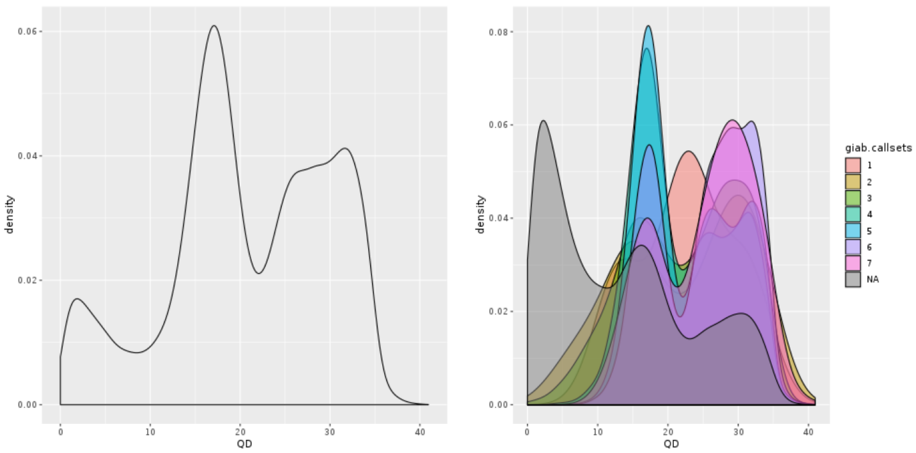


Figure 5.10: Density plot of QD: all calls together (left); stratified by callsets annotation (right).

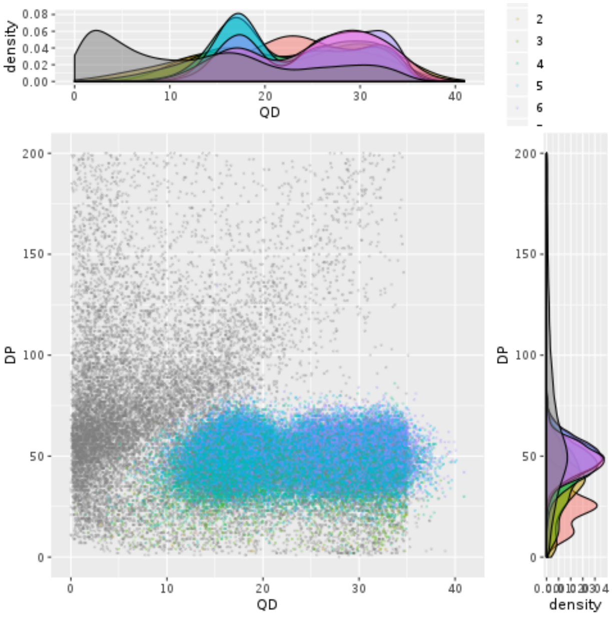
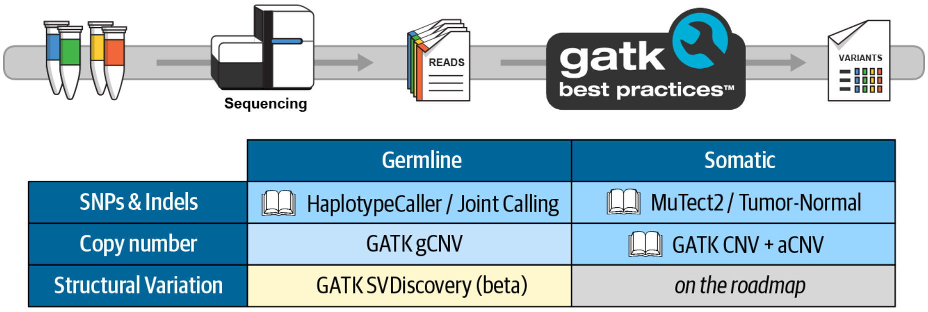


Figure 5.11: A scatter plot with marginal densities of QD versus DP.



	Germline	Somatic
SNPs & Indels	🕮 HaplotypeCaller / Joint Calling	🕮 MuTect2 / Tumor-Normal
Copy number	GATK gCNV	🕮 GATK CNV + aCNV
Structural Variation	GATK SVDiscovery (beta)	*on the roadmap*

Figure 5.12: Table of standard variant discovery use cases covered by GATK Best Practices.

Chapter 6 GATK Best Practices for Germline Short Variant Discovery

Step by step examination of what may be the most commonly run genomics pipeline in the world, with highlights on joint calling for populations and deep learning for single-sample analysis.

6.1 Data Preprocessing

6.1.1 Mapping Reads to the Genome Reference

6.1.2 Marking Duplicates

6.1.3 Recalibrating Base Quality Scores

6.2 Joint Discovery Analysis

6.2.1 Overview of the Joint Calling Workflow

6.2.2 Calling Variants per Sample to Generate GVCFs

6.2.3 Consolidating GVCFs

6.2.4 Applying Joint Genotyping to Multiple Samples

6.2.5 Filtering the Joint Callset with Variant Quality Score Recalibration

6.2.6 Refining Genotype Assignments and Adjusting Genotype Confidence

6.2.7 Next Steps and Further Reading

6.3 Single-Sample Calling with CNN Filtering

6.3.1 Overview of the CNN Single-Sample Workflow

6.3.2 Applying 1D CNN to Filter a Single-Sample WGS Callset

6.3.3 Applying 2D CNN to Include Read Data in the Modeling

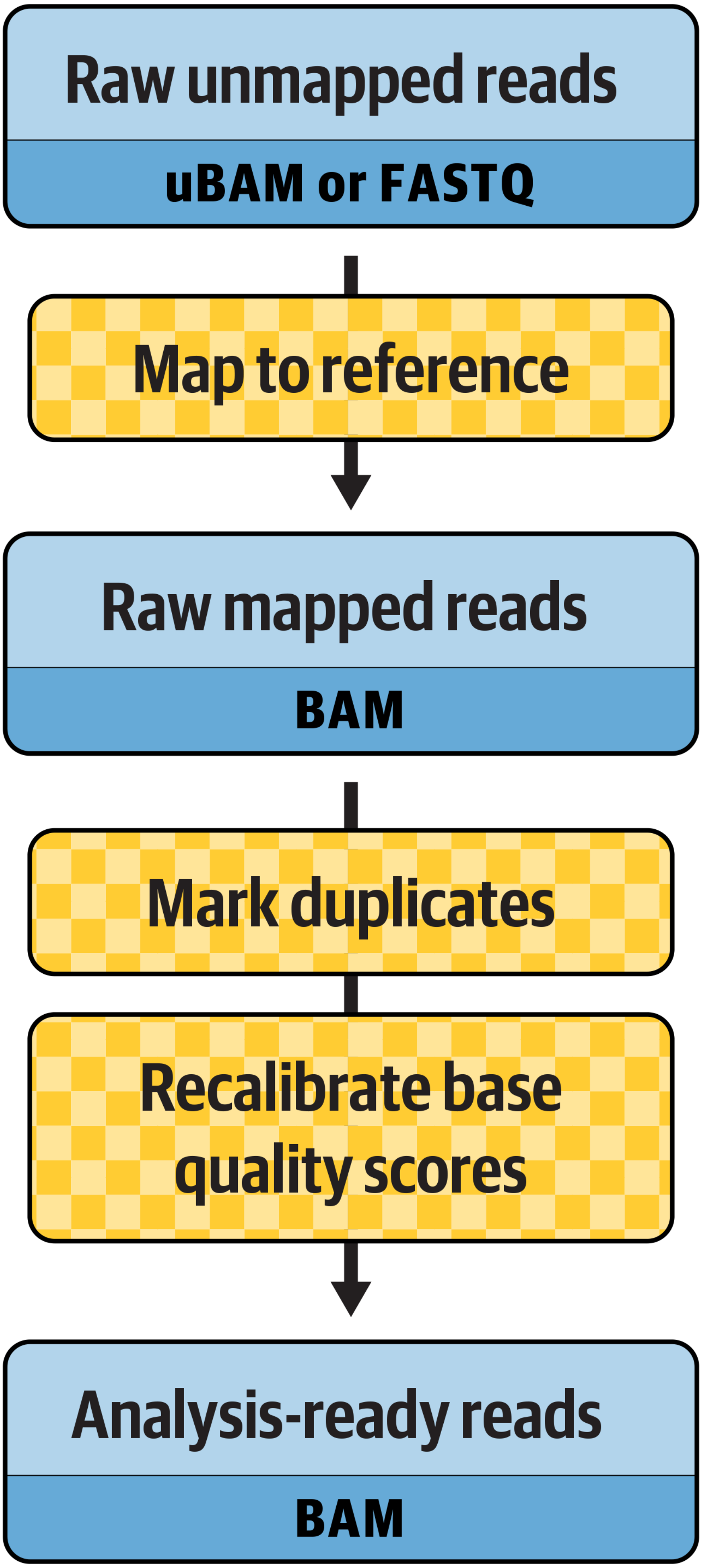


Figure 6.1: The main steps in the preprocessing workflow.

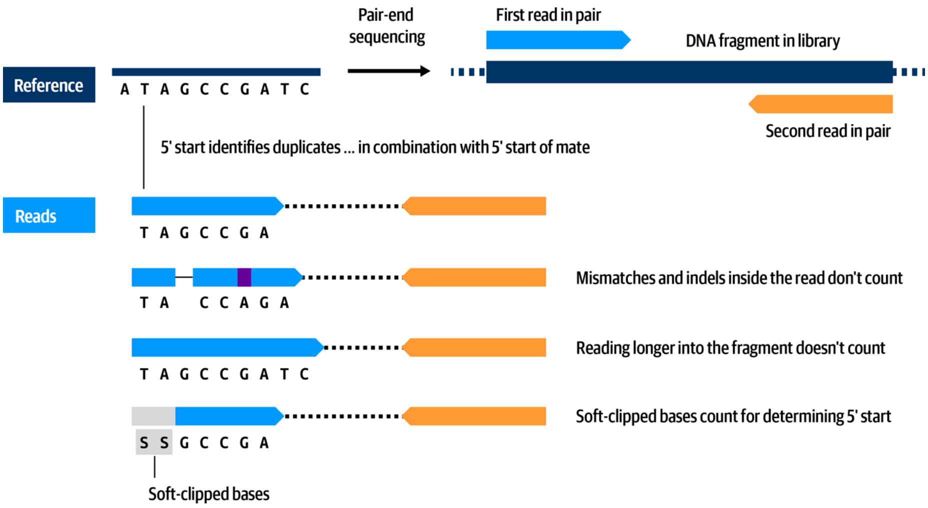

Pair-end sequencing
First read in pair
DNA fragment in library
Second read in pair
Reference
A T A G C C G A T C
5' start identifies duplicates ... in combination with 5' start of mate
Reads
T A G C C G A
T A C C A G A
Mismatches and indels inside the read don't count
T A G C C G A T C
Reading longer into the fragment doesn't count
S S G C C G A
Soft-clipped bases count for determining 5' start
Soft-clipped bases

Figure 6.2: Reads marked as duplicates because they originated from the same DNA fragment in the library.

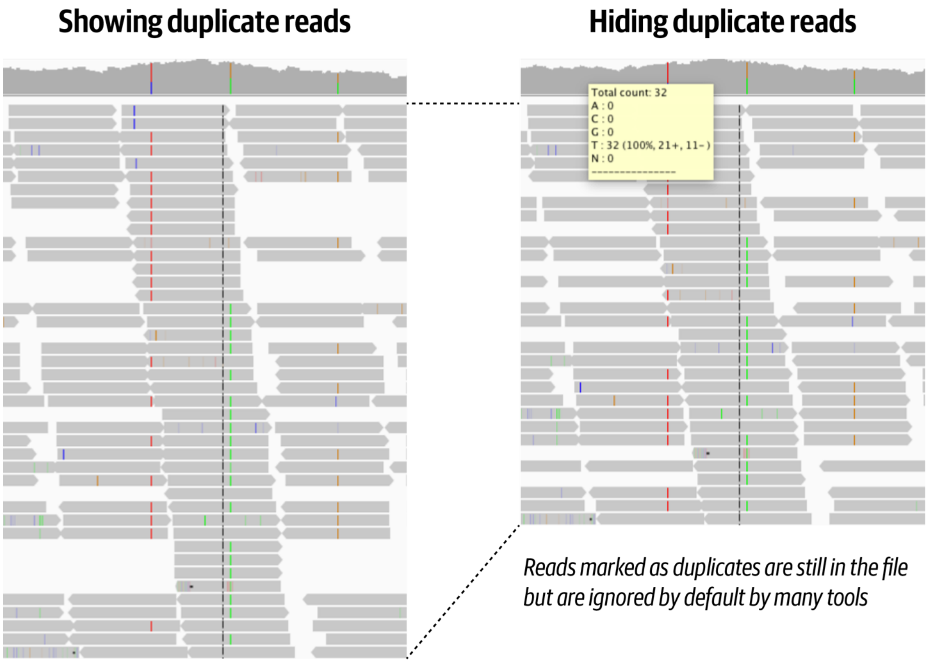


Figure 6.3: The effect of duplicate marking visualized in Integrated Genome Viewer.

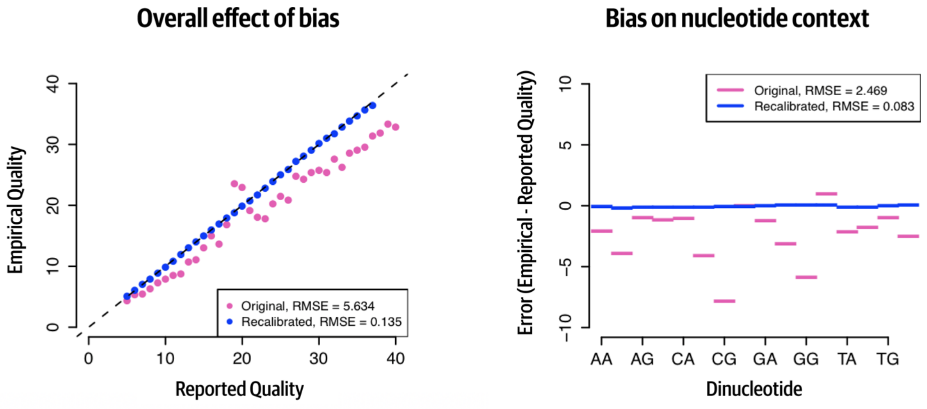


Figure 6.4: Visualizing the effect of BQSR.

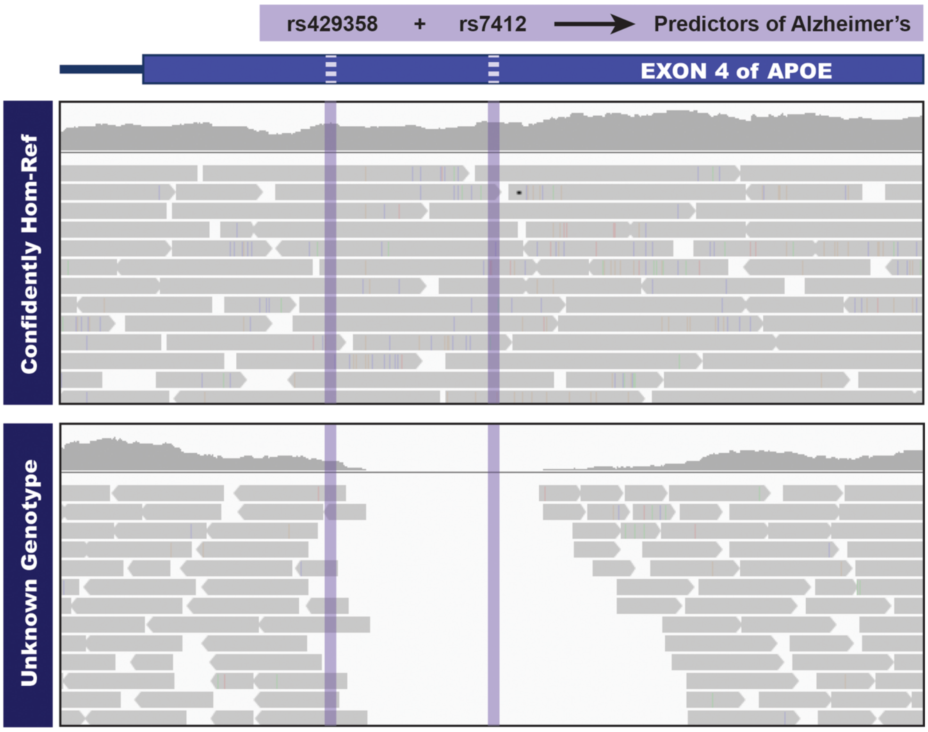


Figure 6.5: Sites that would be omitted from the VCF in a single-sample callset.

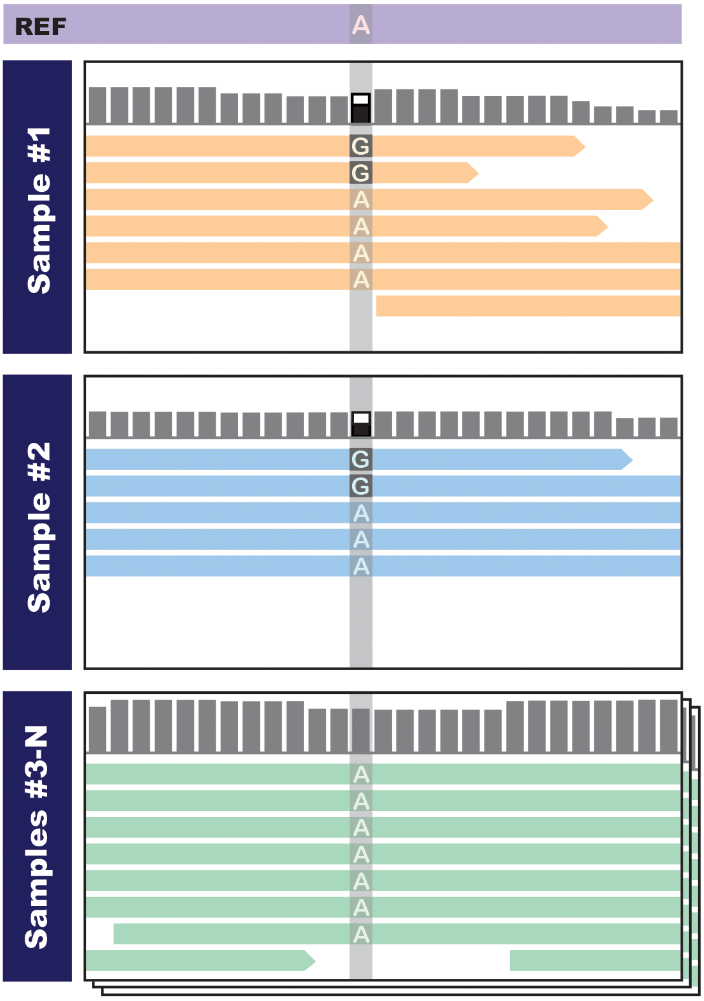


Figure 6.6: Seeing concordant evidence in multiple samples boosts our confidence that there is real variation.

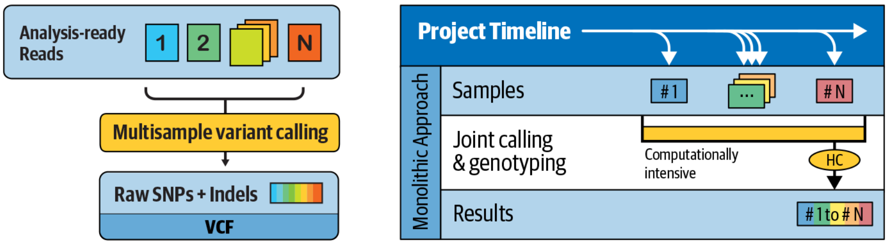
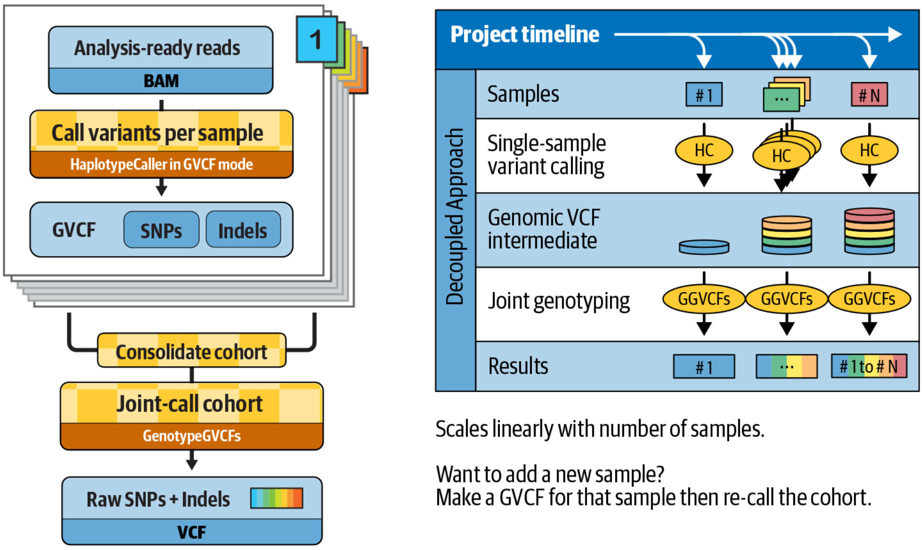


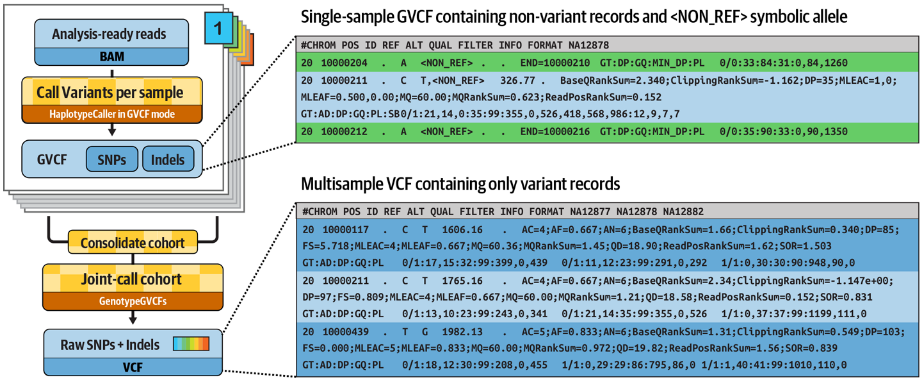
Figure 6.7: Traditional multisample analysis scales poorly and causes the N + 1 problem.



Scales linearly with number of samples.

Want to add a new sample?
Make a GVCF for that sample then re-call the cohort.

Figure 6.8: The GVCF workflow improves the scaling of joint calling and solves the N + 1 problem.



Single-sample GVCF containing non-variant records and <NON_REF> symbolic allele

```
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA12878
20  10000204   .   A   <NON_REF> .   .   END=10000210  GT:DP:GQ:MIN_DP:PL   0/0:33:84:31:0,84,1260
20  10000211   .   C   T,<NON_REF>   326.77 .   BaseQRankSum=2.340;ClippingRankSum=-1.162;DP=35;MLEAC=1,0;
MLEAF=0.500,0.00;MQ=60.00;MQRankSum=0.623;ReadPosRankSum=0.152
GT:AD:DP:GQ:PL:SB0/1:21,14,0:35:99:355,0,526,418,568,986:12,9,7,7
20  10000212   .   A   <NON_REF> .   .   END=10000216  GT:DP:GQ:MIN_DP:PL   0/0:35:90:33:0,90,1350
```

Multisample VCF containing only variant records

```
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT NA12877 NA12878 NA12882
20  10000117   .   C   T   1606.16    .   AC=4;AF=0.667;AN=6;BaseQRankSum=1.66;ClippingRankSum=0.340;DP=85;
FS=5.718;MLEAC=4;MLEAF=0.667;MQ=60.36;MQRankSum=1.45;QD=18.90;ReadPosRankSum=1.62;SOR=1.503
GT:AD:DP:GQ:PL   0/1:17,15:32:99:399,0,439   0/1:11,12:23:99:291,0,292   1/1:0,30:30:90:948,90,0
20  10000211   .   C   T   1765.16    .   AC=4;AF=0.667;AN=6;BaseQRankSum=2.34;ClippingRankSum=-1.147e+00;
DP=97;FS=0.809;MLEAC=4;MLEAF=0.667;MQ=60.00;MQRankSum=1.21;QD=18.58;ReadPosRankSum=0.152;SOR=0.831
GT:AD:DP:GQ:PL   0/1:13,10:23:99:243,0,341   0/1:21,14:35:99:355,0,526   1/1:0,37:37:99:1199,111,0
20  10000439   .   T   G   1982.13    .   AC=5;AF=0.833;AN=6;BaseQRankSum=1.31;ClippingRankSum=0.549;DP=103;
FS=0.000;MLEAC=5;MLEAF=0.833;MQ=60.00;MQRankSum=0.972;QD=19.82;ReadPosRankSum=1.56;SOR=0.839
GT:AD:DP:GQ:PL   0/1:18,12:30:99:208,0,455   1/1:0,29:29:86:795,86,0 1/1:1,40:41:99:1010,110,0
```

Figure 6.9: Progression from per-sample GVCFs to final cohort VCF.

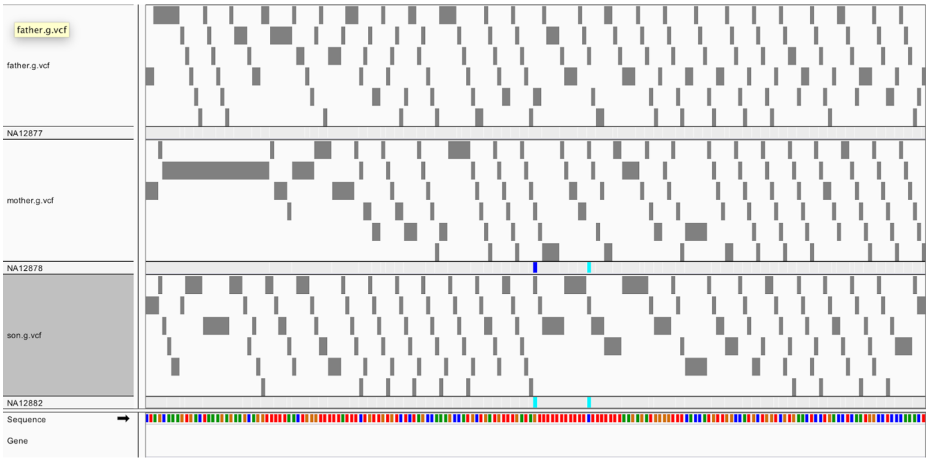


Figure 6.10: GVCFs viewed in IGV show tiled nonvariant blocks.

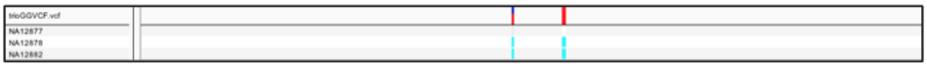


Figure 6.11: Variant call with genotype assignment for the three samples.

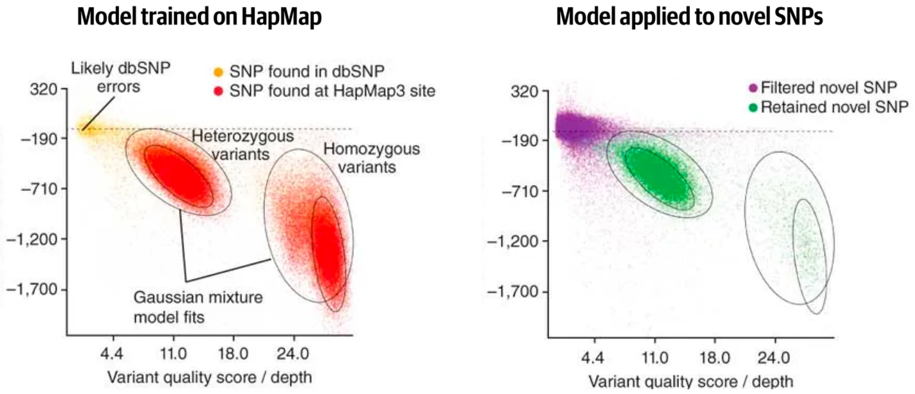


Figure 6.12: Gaussian clusters learned from a training set are applied to novel variant calls.

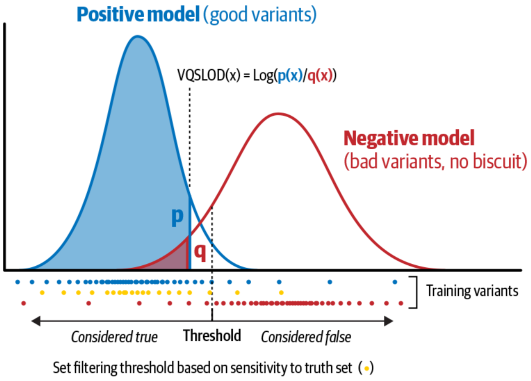
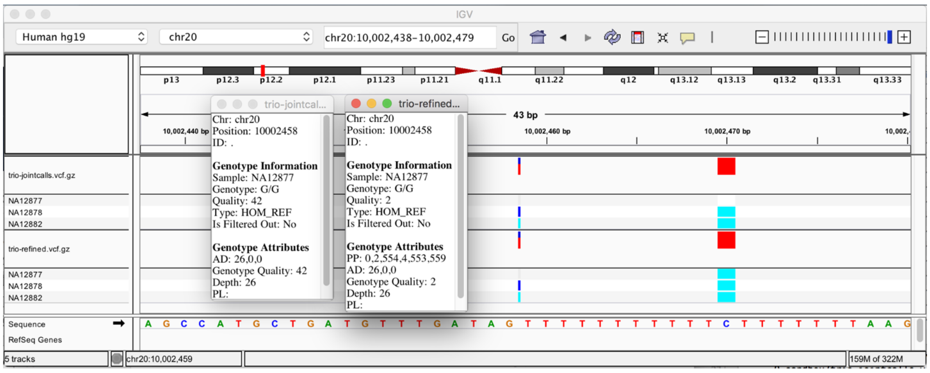

Positive model (good variants)
VQSLOD(x) = Log(p(x)/q(x))
Negative model
(bad variants, no biscuit)
p
q
Training variants
Considered true
Threshold
Considered false
Set filtering threshold based on sensitivity to truth set (•)

Figure 6.13: How the VQSLOD score is calculated for an individual annotation.



trio-jointcal...

Chr: chr20
Position: 10002458
ID: .

Genotype Information
Sample: NA12877
Genotype: G/G
Quality: 42
Type: HOM_REF
Is Filtered Out: No

Genotype Attributes
AD: 26,0,0
Genotype Quality: 42
Depth: 26
PL:

trio-refined...

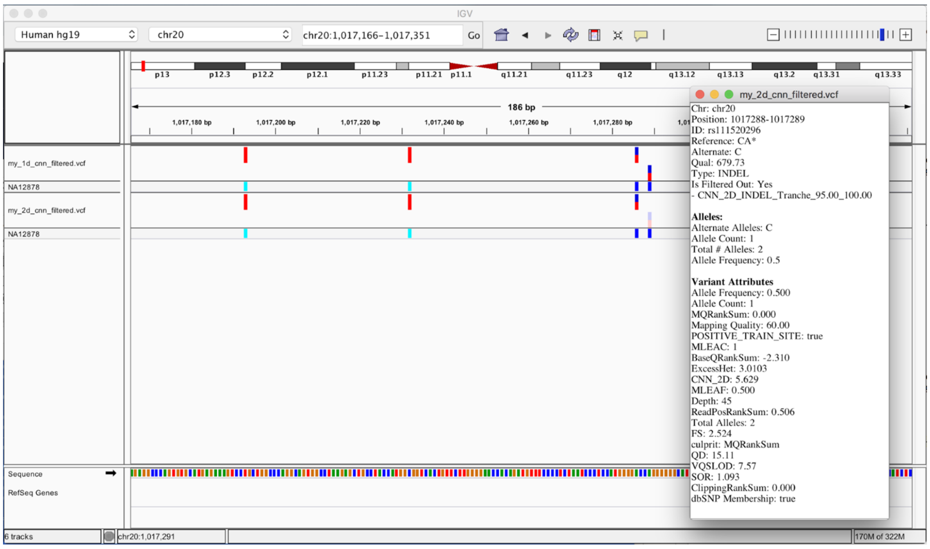
Chr: chr20
Position: 10002458
ID: .

Genotype Information
Sample: NA12877
Genotype: G/G
Quality: 2
Type: HOM_REF
Is Filtered Out: No

Genotype Attributes
PP: 0,2,554,4,553,559
AD: 26,0,0
Genotype Quality: 2
Depth: 26
PL:

Figure 6.14: Genotype assignments corrected on the basis of pedigree and population priors.

Figure 6.15: Labradoodle or fried chicken? (Source: Karen Zack, @teenybiscuit).



my_2d_cnn_filtered.vcf

```
Chr: chr20
Position: 1017288-1017289
ID: rs111520296
Reference: CA*
Alternate: C
Qual: 679.73
Type: INDEL
Is Filtered Out: Yes
- CNN_2D_INDEL_Tranche_95.00_100.00

Alleles:
Alternate Alleles: C
Allele Count: 1
Total # Alleles: 2
Allele Frequency: 0.5

Variant Attributes
Allele Frequency: 0.500
Allele Count: 1
MQRankSum: 0.000
Mapping Quality: 60.00
POSITIVE_TRAIN_SITE: true
MLEAC: 1
BaseQRankSum: -2.310
ExcessHet: 3.0103
CNN_2D: 5.629
MLEAF: 0.500
Depth: 45
ReadPosRankSum: 0.506
Total Alleles: 2
FS: 2.524
culprit: MQRankSum
QD: 15.11
VQSLOD: 7.57
SOR: 1.093
ClippingRankSum: 0.000
dbSNP Membership: true
```

Figure 6.16: Different calls made by 1D and 2D CNN models.

Chapter 7 GATK Best Practices for Somatic Variant Discovery

Switching gears to cancer genomics with a rundown of how somatic calling is different; step by step through the pipelines for somatic short variants (Mutect2) and copy number alterations.

7.1 Challenges in Cancer Genomics

7.2 Somatic Short Variants (SNVs and Indels)

7.2.1 Overview of the Tumor-Normal Pair Analysis Workflow

7.2.2 Creating a Mutect2 PoN

7.2.3 Running Mutect2 on the Tumor-Normal Pair

7.2.4 Estimating Cross-Sample Contamination

7.2.5 Filtering Mutect2 Calls

7.2.6 Annotating Predicted Functional Effects with Funcotator

7.3 Somatic Copy-Number Alterations

7.3.1 Overview of the Tumor-Only Analysis Workflow

7.3.2 Collecting Coverage Counts

7.3.3 Creating a Somatic CNA PoN

7.3.4 Applying Denoising

7.3.5 Performing Segmentation and Call CNAs

7.3.6 Additional Analysis Options

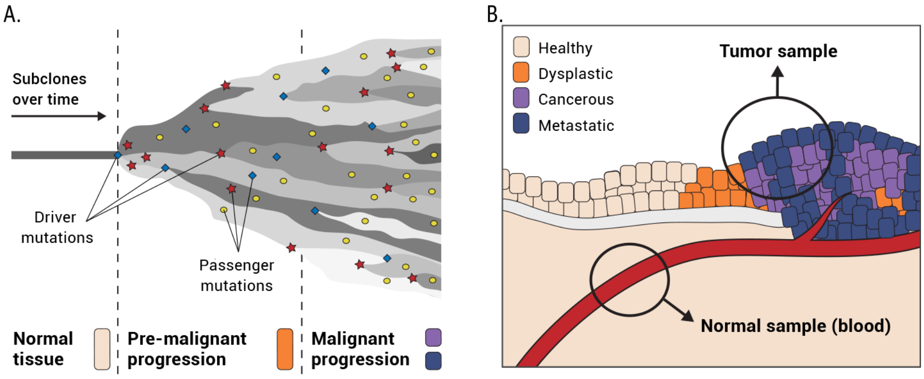


Figure 7.1: Tumor progression leads to heterogeneity (left); sampling is difficult (right).

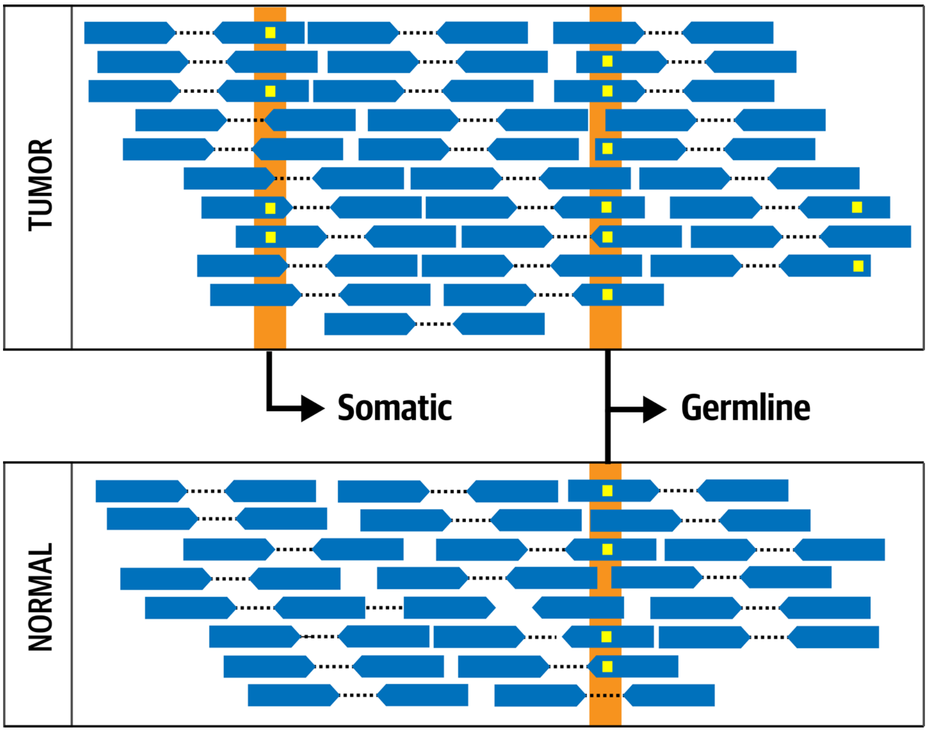


Figure 7.2: The fundamental concept of Tumor-Normal comparison.

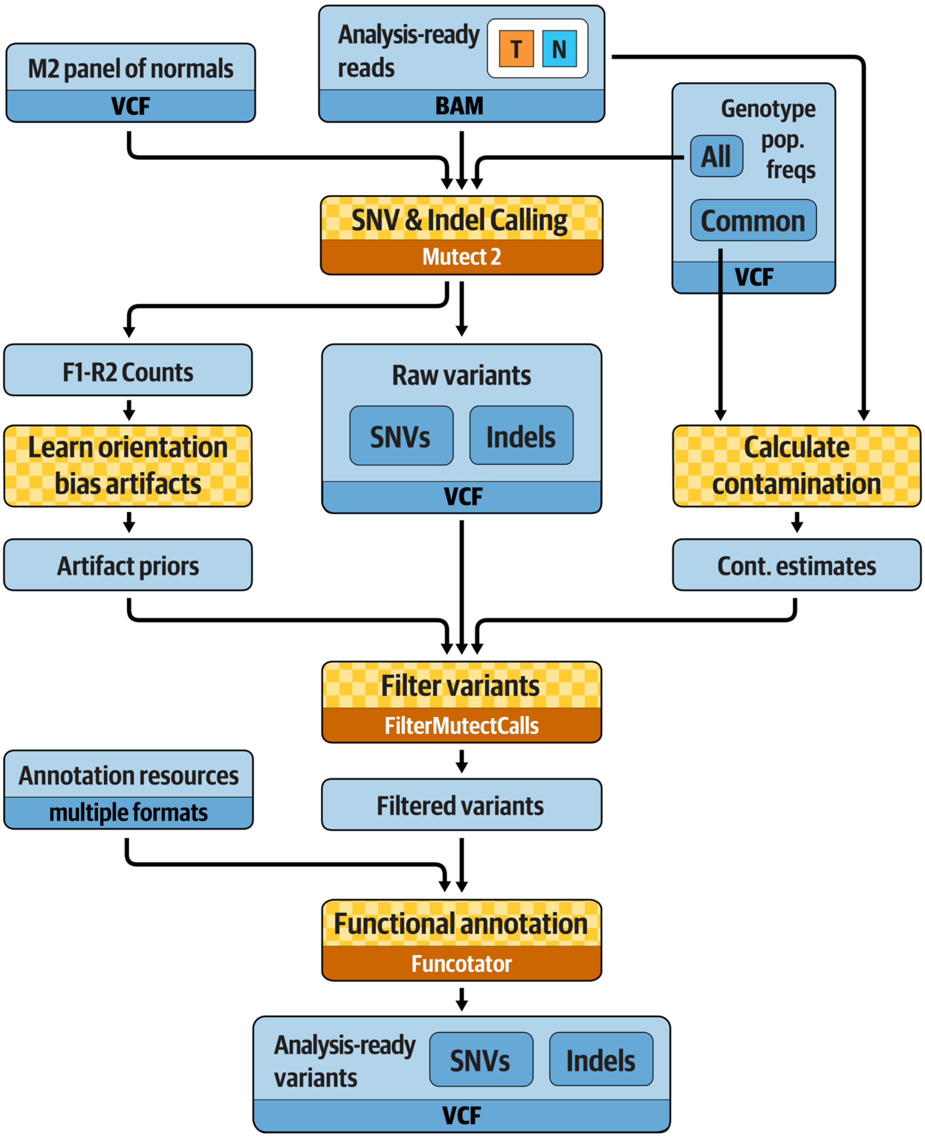
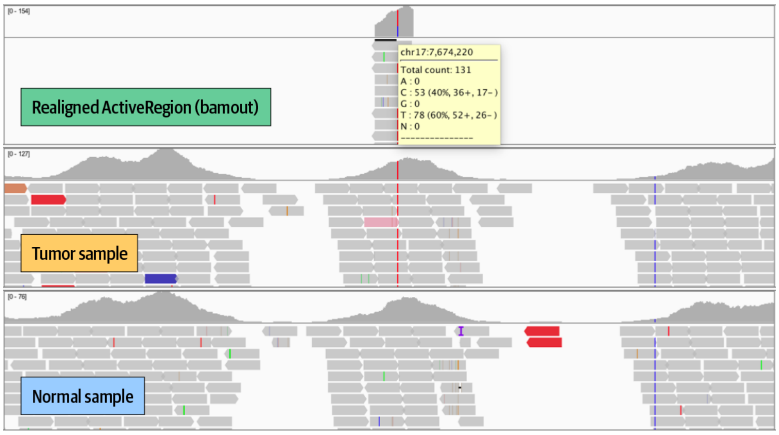
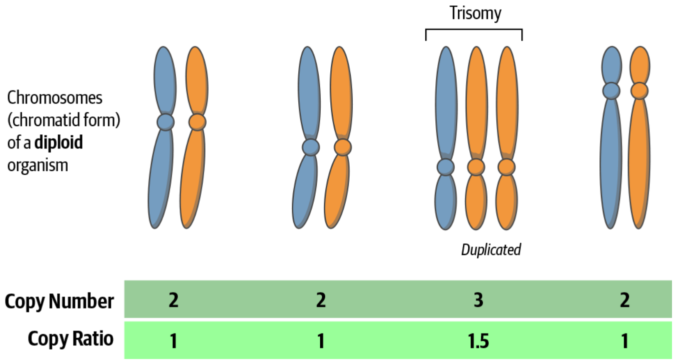


Figure 7.3: Best Practices for somatic short variant discovery.



```
chr17:7,674,220
Total count: 131
A : 0
C : 53 (40%, 36+, 17- )
G : 0
T : 78 (60%, 52+, 26- )
N : 0
---------------
```

Figure 7.4: Zooming in on TP53 in IGV.



Copy Number	2	2	3	2
Copy Ratio	1	1	1.5	1

Figure 7.5: Difference between copy number and copy ratio.

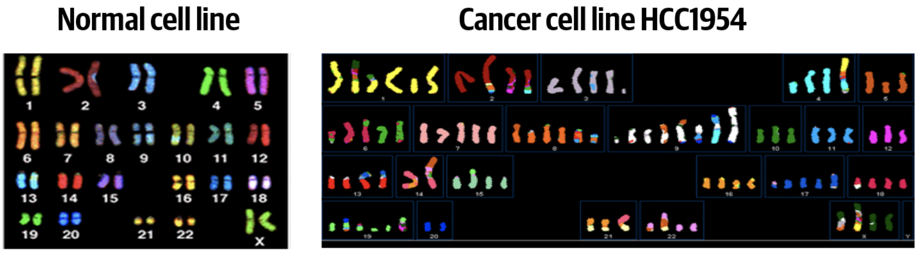


Figure 7.6: Spectral karyotyping paints each chromosome pair with a color, showing various chromosomal segments that are amplified or missing (colors in left and right panels are not expected to match).

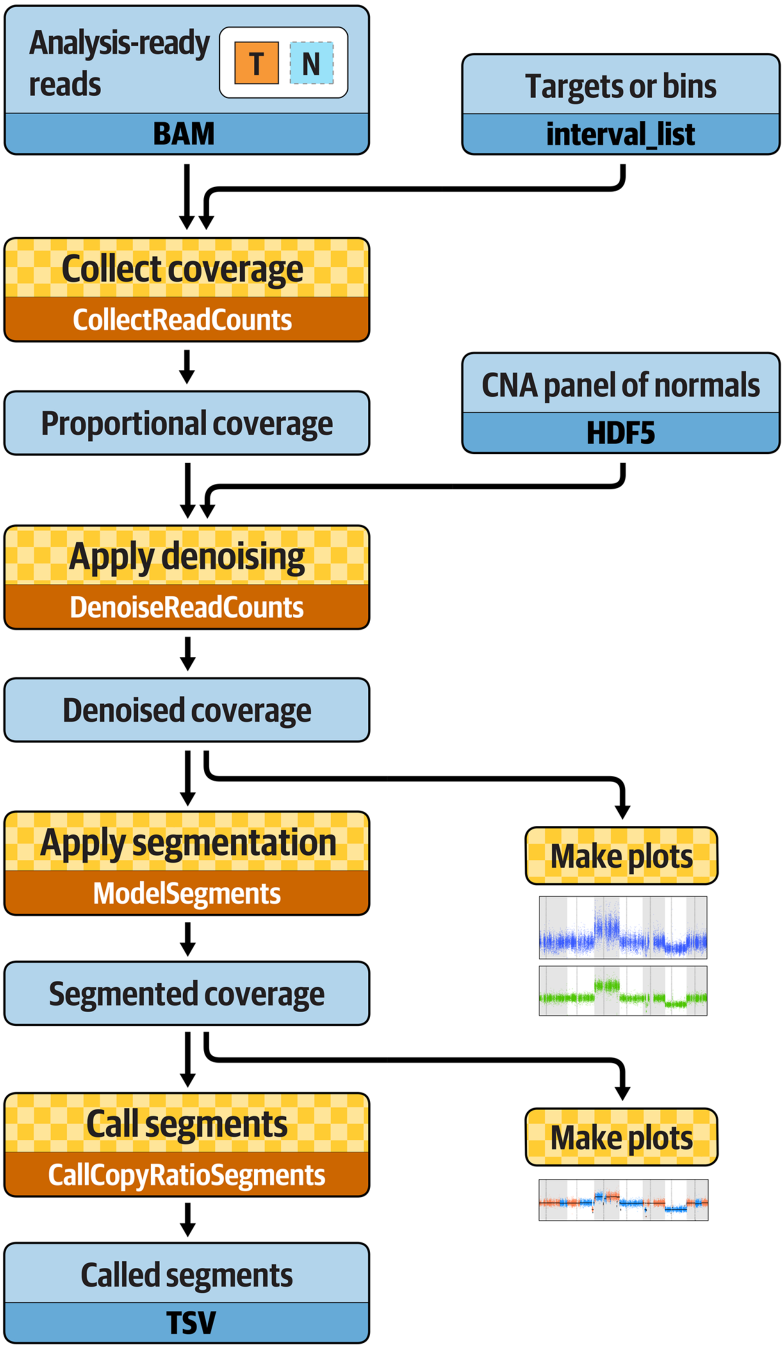


Figure 7.7: Best Practices workflow for somatic copy-number alteration discovery.

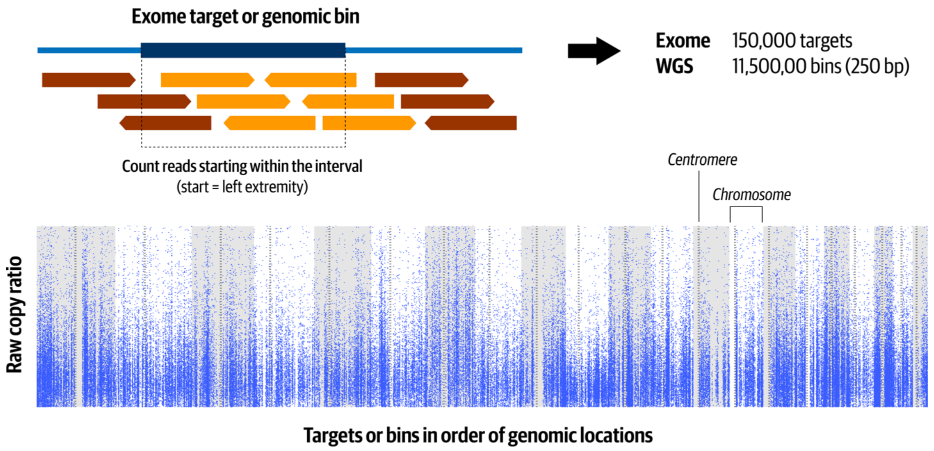

Exome target or genomic bin
Exome 150,000 targets
WGS 11,500,00 bins (250 bp)
Count reads starting within the interval
(start = left extremity)
Centromere
Chromosome
Raw copy ratio
Targets or bins in order of genomic locations

Figure 7.8: Read counts in each genomic target or bin form the basis for estimating segmented copy ratio, and each dot is the value for a single target or bin.

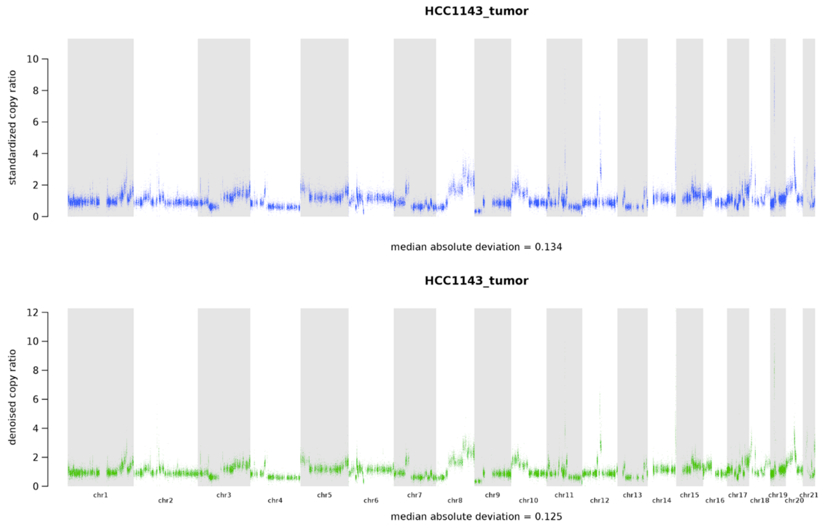


Figure 7.9: Copy-number alteration analysis plots showing the standardized copy ratios after the first step of denoising (top) and the fully denoised copy ratios after the second round (bottom).

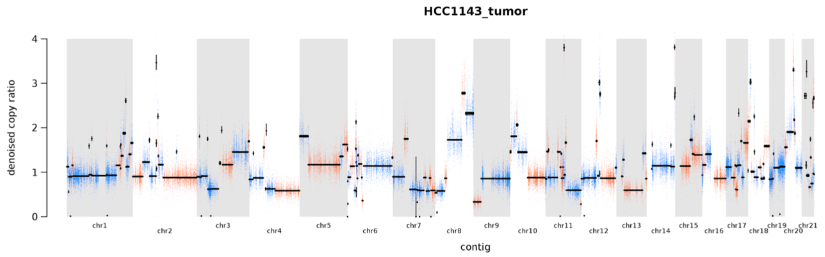


Figure 7.10: Plot of segments modeled based on denoised copy ratios.

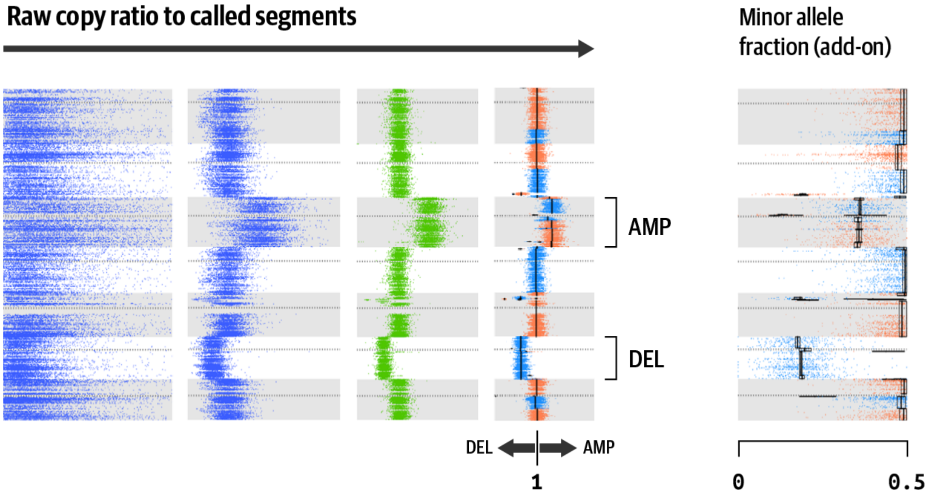


Figure 7.11: Full progression from raw data to results.

Chapter 8 Automating Analysis Execution with Workflows

Halfway point; we pivot to the challenges of automating and scaling up these analyses, introducing the Cromwell workflow system and the portable Workflow Description Language (WDL).

8.1 Introducing WDL and Cromwell

8.2 Installing and Setting Up Cromwell

8.3 Your First WDL: Hello World

8.3.1 Learning Basic WDL Syntax Through a Minimalist Example

8.3.2 Running a Simple WDL with Cromwell on Your Google VM

8.3.3 Interpreting the Important Parts of Cromwell's Logging Output

8.3.4 Adding a Variable and Providing Inputs via JSON

8.3.5 Adding Another Task to Make It a Proper Workflow

8.4 Your First GATK Workflow: Hello HaplotypeCaller

8.4.1 Exploring the WDL

8.4.2 Generating the Inputs JSON

8.4.3 Running the Workflow

8.4.4 Breaking the Workflow to Test Syntax Validation and Error Messaging

8.5 Introducing Scatter-Gather Parallelism

8.5.1 Exploring the WDL

8.5.2 Generating a Graph Diagram for Visualization

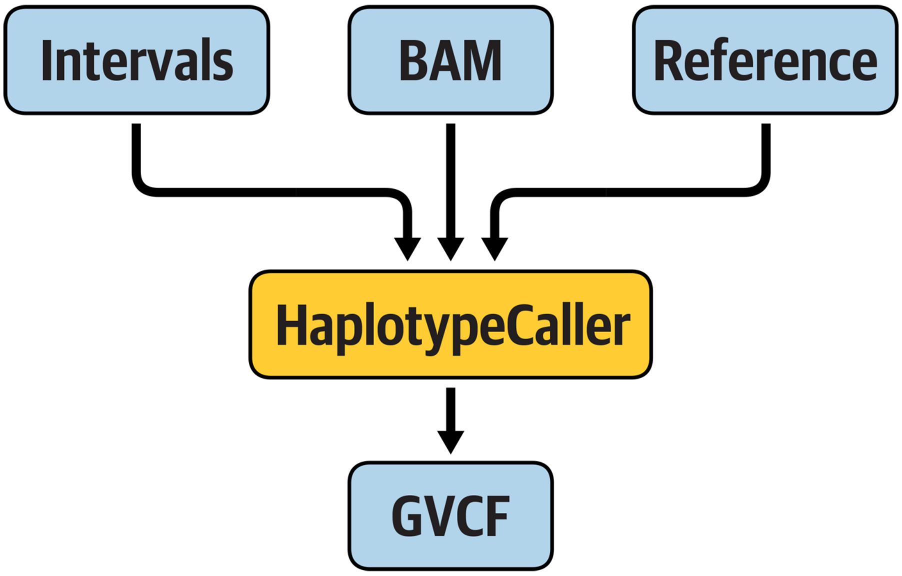


Figure 8.1: A hypothetical workflow that runs HaplotypeCaller.

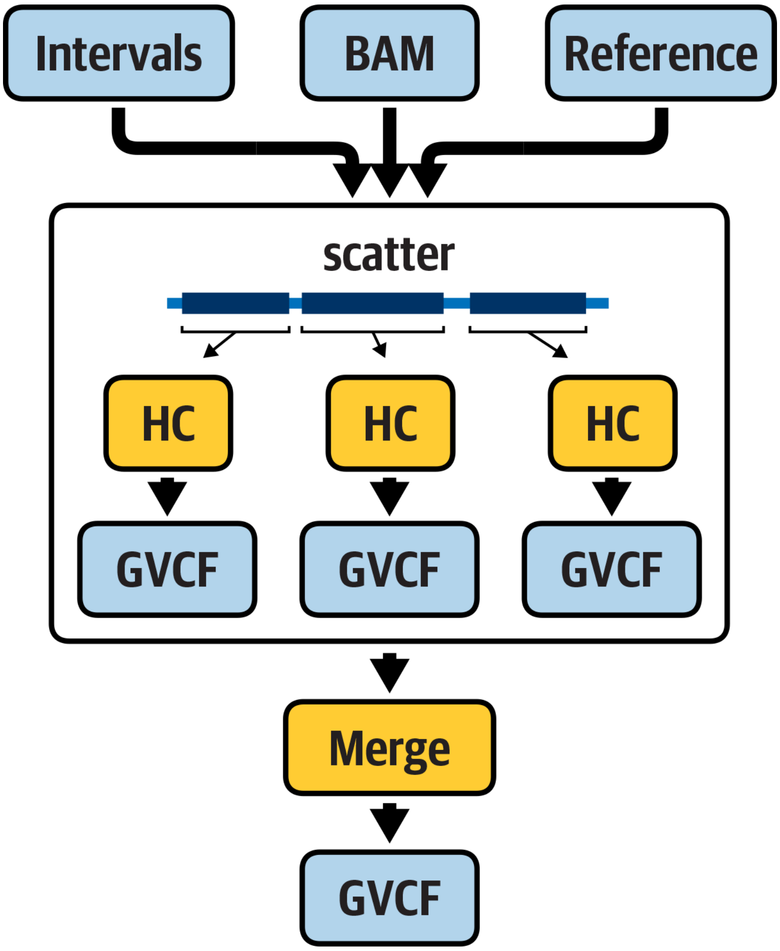
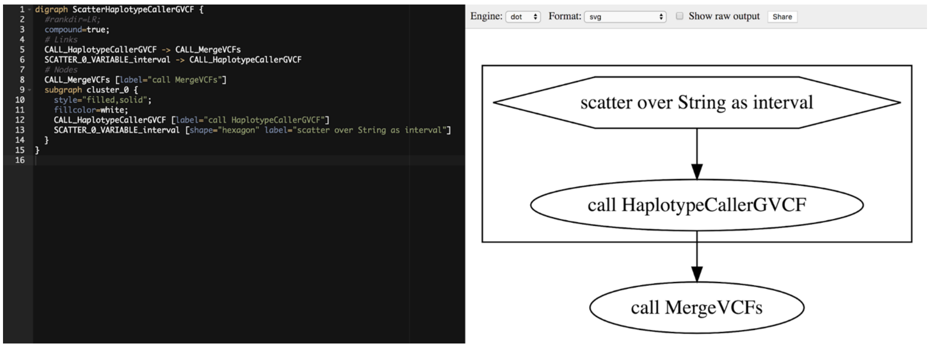


Figure 8.2: A workflow that parallelizes the execution of HaplotypeCaller.



```
digraph ScatterHaplotypeCallerGVCF {
  #rankdir=LR;
  compound=true;
  # Links
  CALL_HaplotypeCallerGVCF -> CALL_MergeVCFs
  SCATTER_0_VARIABLE_interval -> CALL_HaplotypeCallerGVCF
  # Nodes
  CALL_MergeVCFs [label="call MergeVCFs"]
  subgraph cluster_0 {
    style="filled,solid";
    fillcolor=white;
    CALL_HaplotypeCallerGVCF [label="call HaplotypeCallerGVCF"]
    SCATTER_0_VARIABLE_interval [shape="hexagon" label="scatter over String as interval"]
  }
}
```

Engine: dot Format: svg □ Show raw output

Figure 8.3: Visualizing the workflow graph in an online Graphviz application.

Chapter 9 Deciphering Real Genomics Workflows

We pretend to stumble across 2 mystery workflows, go through a systematic process of investigating their content to understand what they do and how they do it, learning useful WDL features along the way.

9.1 Mystery Workflow #1: Flexibility Through Conditionals

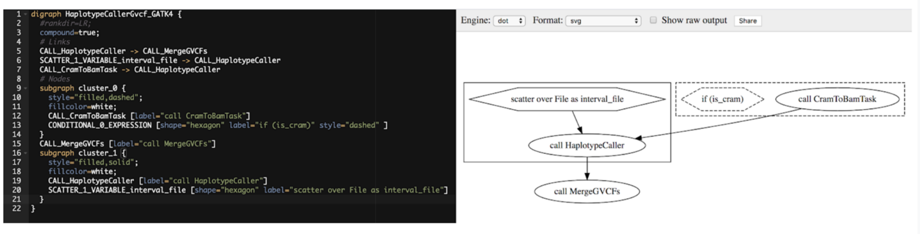
9.1.1 Mapping Out the Workflow

9.1.2 Reverse Engineering the Conditional Switch

9.2 Mystery Workflow #2: Modularity and Code Reuse

9.2.1 Mapping Out the Workflow

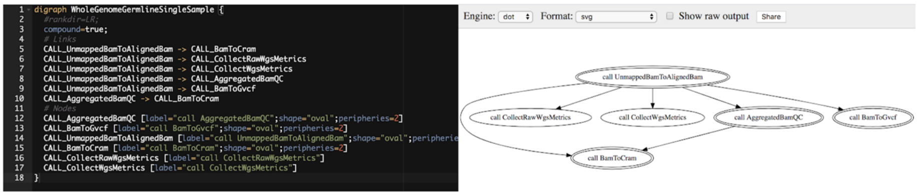
9.2.2 Unpacking the Nesting Dolls



```
digraph HaplotypeCallerGvcf_GATK4 {
  #rankdir=LR;
  compound=true;
  # Links
  CALL_HaplotypeCaller -> CALL_MergeGVCFs
  SCATTER_1_VARIABLE_interval_file -> CALL_HaplotypeCaller
  CALL_CramToBamTask -> CALL_HaplotypeCaller
  # Nodes
  subgraph cluster_0 {
    style="filled,dashed";
    fillcolor=white;
    CALL_CramToBamTask [label="call CramToBamTask"]
    CONDITIONAL_0_EXPRESSION [shape="hexagon" label="if (is_cram)" style="dashed" ]
  }
  CALL_MergeGVCFs [label="call MergeGVCFs"]
  subgraph cluster_1 {
    style="filled,solid";
    fillcolor=white;
    CALL_HaplotypeCaller [label="call HaplotypeCaller"]
    SCATTER_1_VARIABLE_interval_file [shape="hexagon" label="scatter over File as interval_file"]
  }
}
```

Engine: dot Format: svg □ Show raw output

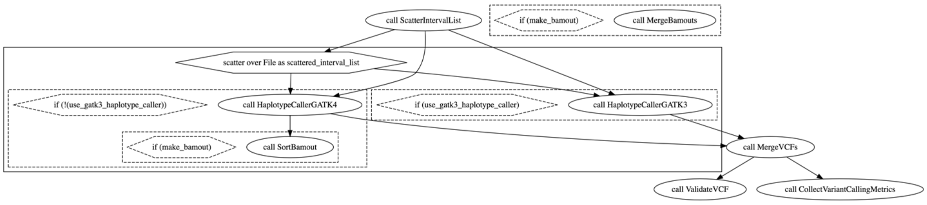
Figure 9.1: Graph description in JSON (left) and visual rendering (right).



```
digraph WholeGenomeGermlineSingleSample {
  #rankdir=LR;
  compound=true;
  # Links
  CALL_UnmappedBamToAlignedBam -> CALL_BamToCram
  CALL_UnmappedBamToAlignedBam -> CALL_CollectRawWgsMetrics
  CALL_UnmappedBamToAlignedBam -> CALL_CollectWgsMetrics
  CALL_UnmappedBamToAlignedBam -> CALL_AggregatedBamQC
  CALL_UnmappedBamToAlignedBam -> CALL_BamToGvcf
  CALL_AggregatedBamQC -> CALL_BamToCram
  # Nodes
  CALL_AggregatedBamQC [label="call AggregatedBamQC";shape="oval";peripheries=2]
  CALL_BamToGvcf [label="call BamToGvcf";shape="oval";peripheries=2]
  CALL_UnmappedBamToAlignedBam [label="call UnmappedBamToAlignedBam";shape="oval";peripherie
  CALL_BamToCram [label="call BamToCram";shape="oval";peripheries=2]
  CALL_CollectRawWgsMetrics [label="call CollectRawWgsMetrics"]
  CALL_CollectWgsMetrics [label="call CollectWgsMetrics"]
}
```

Engine: dot Format: svg □ Show raw output

Figure 9.2: Visual rendering of the workflow graph.



call ScatterIntervalList
if (make_bamout)
call MergeBamouts
scatter over File as scattered_interval_list
if (!(use_gatk3_haplotype_caller))
call HaplotypeCallerGATK4
if (use_gatk3_haplotype_caller)
call HaplotypeCallerGATK3
if (make_bamout)
call SortBamout
call MergeVCFs
call ValidateVCF
call CollectVariantCallingMetrics

Figure 9.3: Graph diagram of the VariantCalling.wdl workflow.

Chapter 10 Running Single Workflows at Scale with Pipelines API

So far we've been running everything on our little custom VM. Now it's time to unleash the full power of the cloud by dispatching workflow tasks to multiple machines – with surprisingly little effort.

10.1 Introducing the GCP Genomics Pipelines API Service

10.1.1 Enabling Genomics API and Related APIs in Your Google Cloud Project

10.2 Directly Dispatching Cromwell Jobs to PAPI

10.2.1 Configuring Cromwell to Communicate with PAPI

10.2.2 Running Scattered HaplotypeCaller via PAPI

10.2.3 Monitoring Workflow Execution on Google Compute Engine

10.3 Understanding and Optimizing Workflow Efficiency

10.3.1 Granularity of Operations

10.3.2 Balance of Time Versus Money

10.3.3 Suggested Cost-Saving Optimizations

10.3.4 Platform-Specific Optimization Versus Portability

10.4 Wrapping Cromwell and PAPI Execution with WDL Runner

10.4.1 Setting Up WDL Runner

10.4.2 Running the Scattered HaplotypeCaller Workflow with WDL Runner

10.4.3 Monitoring WDL Runner Execution

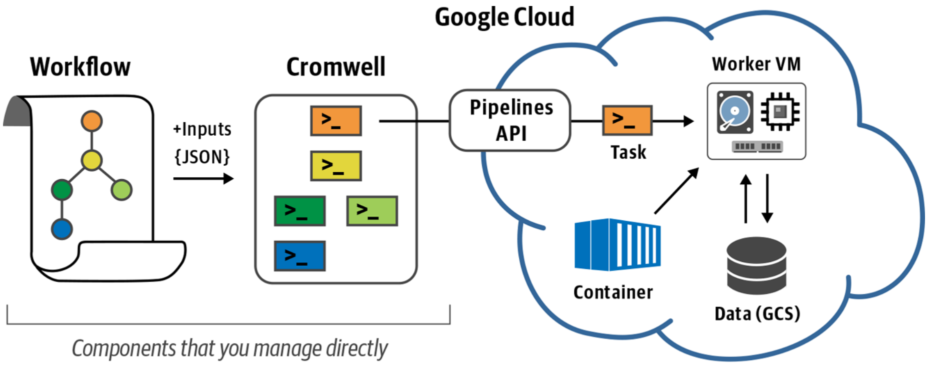
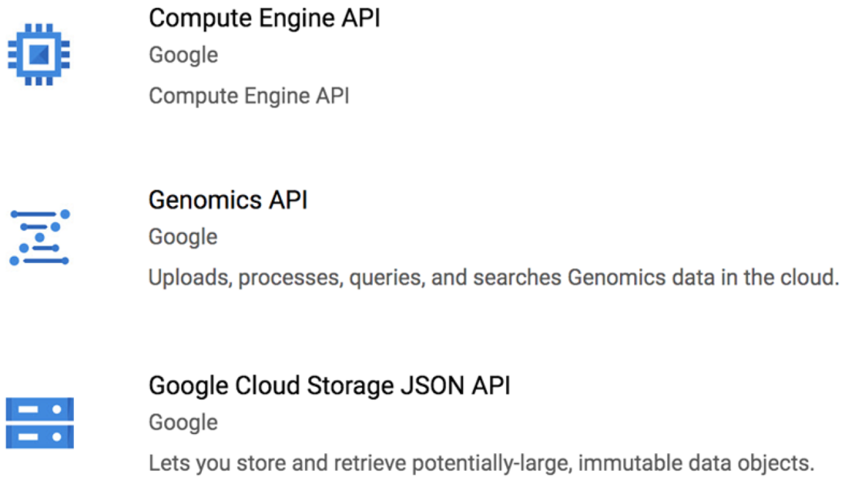


Figure 10.1: Overview of Cromwell + PAPI operation.



Compute Engine API
Google
Compute Engine API

Genomics API
Google
Uploads, processes, queries, and searches Genomics data in the cloud.

Google Cloud Storage JSON API
Google
Lets you store and retrieve potentially-large, immutable data objects.

Figure 10.2: Logos and descriptions for the three required APIs: Genomics API, Cloud Storage JSON API, and Compute Engine API.

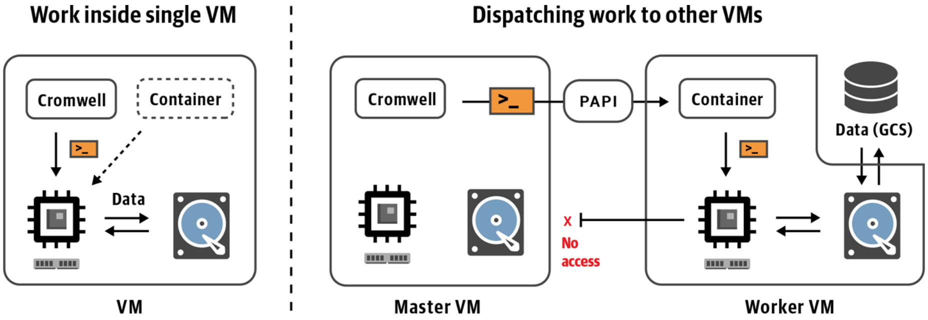
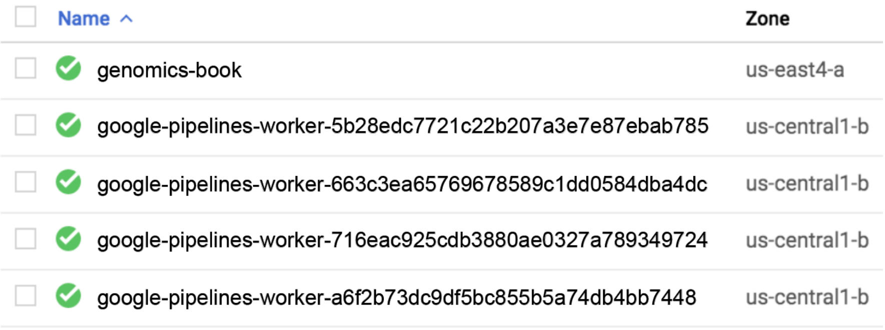


Figure 10.3: Side-by-side comparison of local versus PAPI execution.



□	Name ^	Zone
□	✓ genomics-book	us-east4-a
□	✓ google-pipelines-worker-5b28edc7721c22b207a3e7e87ebab785	us-central1-b
□	✓ google-pipelines-worker-663c3ea65769678589c1dd0584dba4dc	us-central1-b
□	✓ google-pipelines-worker-716eac925cdb3880ae0327a789349724	us-central1-b
□	✓ google-pipelines-worker-a6f2b73dc9df5bc855b5a74db4bb7448	us-central1-b

Figure 10.4: List of active VM instances.

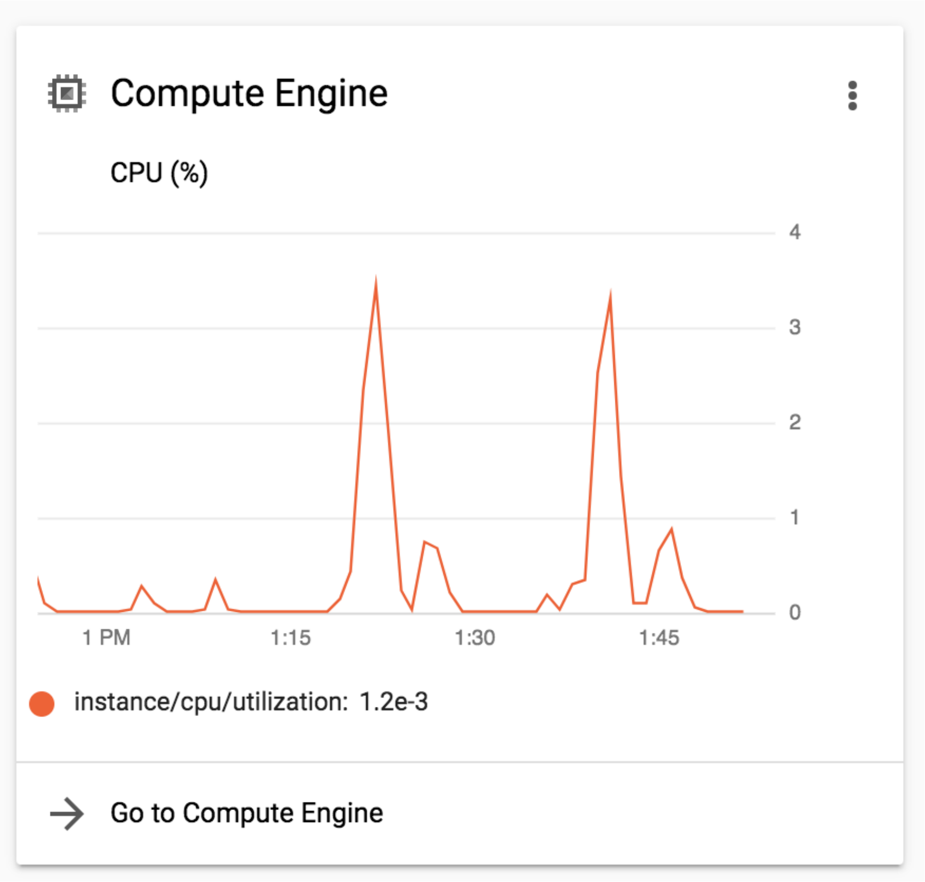


Figure 10.5: Overview of Compute Engine activity.

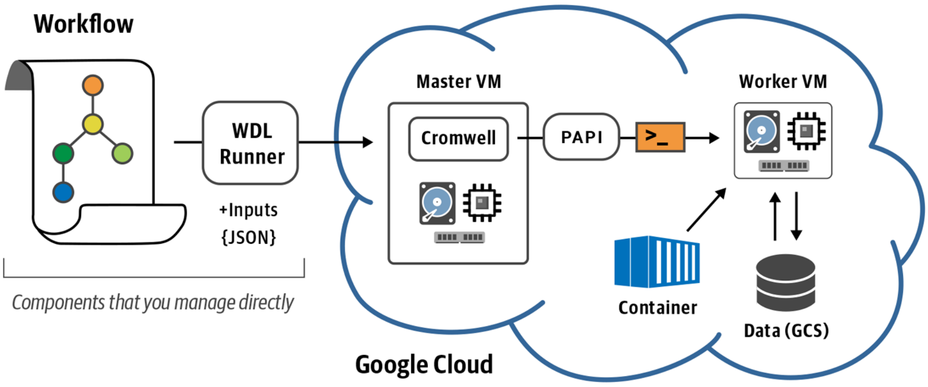
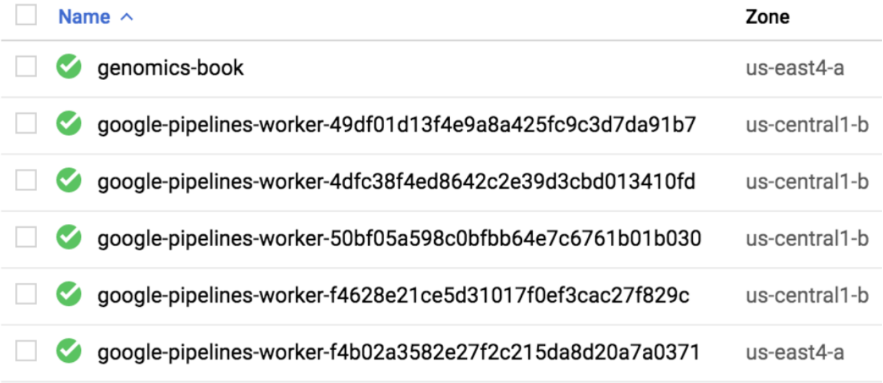
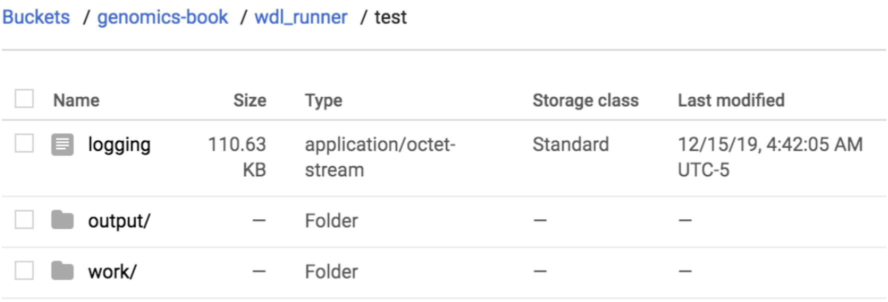


Figure 10.6: Overview of WDL Runner operation.



□	Name ^	Zone
□	✓ genomics-book	us-east4-a
□	✓ google-pipelines-worker-49df01d13f4e9a8a425fc9c3d7da91b7	us-central1-b
□	✓ google-pipelines-worker-4dfc38f4ed8642c2e39d3cbd013410fd	us-central1-b
□	✓ google-pipelines-worker-50bf05a598c0bfbb64e7c6761b01b030	us-central1-b
□	✓ google-pipelines-worker-f4628e21ce5d31017f0ef3cac27f829c	us-central1-b
□	✓ google-pipelines-worker-f4b02a3582e27f2c215da8d20a7a0371	us-east4-a

Figure 10.7: List of active VM instances (WDL Runner submission).



Buckets / genomics-book / wdl_runner / test

□	Name	Size	Type	Storage class	Last modified
□	logging	110.63 KB	application/octet-stream	Standard	12/15/19, 4:42:05 AM UTC-5
□	output/	—	Folder	—	—
□	work/	—	Folder	—	—

Figure 10.8: Output from the WDL Runner submission.

Chapter 11 Running Many Workflows Conveniently in Terra

Now we're scaling up to arbitrary numbers of samples, using the managed Cromwell server in Terra, an open platform for secure data access and analysis.

11.1 Getting Started with Terra

11.1.1 Creating an Account

11.1.2 Creating a Billing Project

11.1.3 Cloning the Preconfigured Workspace

11.2 Running Workflows with the Cromwell Server in Terra

11.2.1 Running a Workflow on a Single Sample

11.2.2 Running a Workflow on Multiple Samples in a Data Table

11.2.3 Monitoring Workflow Execution

11.2.4 Locating Workflow Outputs in the Data Table

11.2.5 Running the Same Workflow Again to Demonstrate Call Caching

11.3 Running a Real GATK Best Practices Pipeline at Full Scale

11.3.1 Finding and Cloning the GATK Best Practices Workspace for Germline Short Variant Discovery

11.3.2 Examining the Preloaded Data

11.3.3 Selecting Data and Configuring the Full-Scale Workflow

11.3.4 Launching the Full-Scale Workflow and Monitoring Execution

11.3.5 Options for Downloading Output Data—or Not

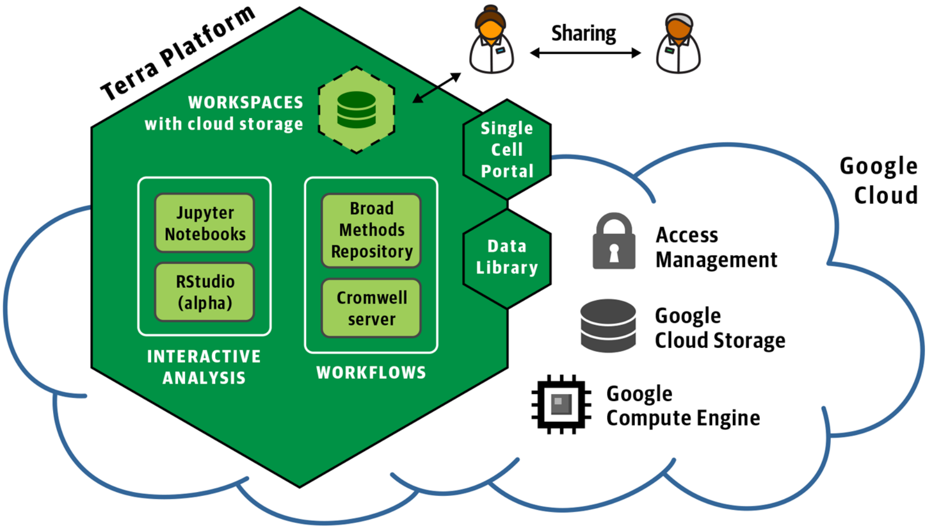


Figure 11.1: Overview of the Terra platform.

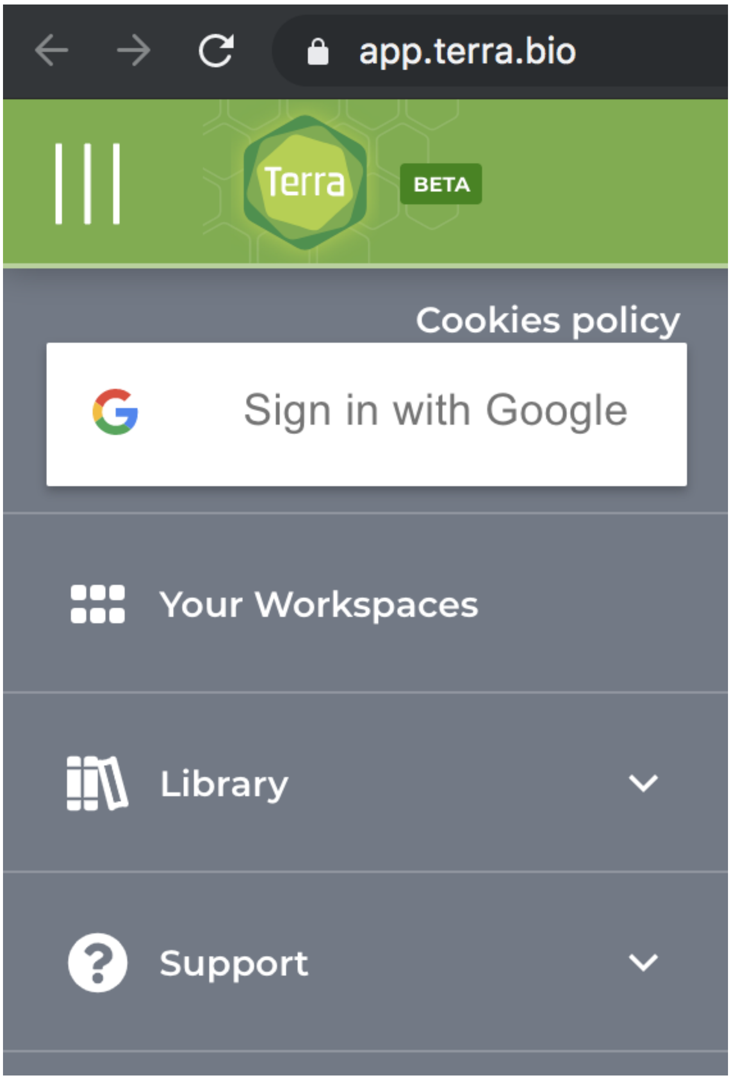


Figure 11.2: Expanded side menu showing sign-in button.

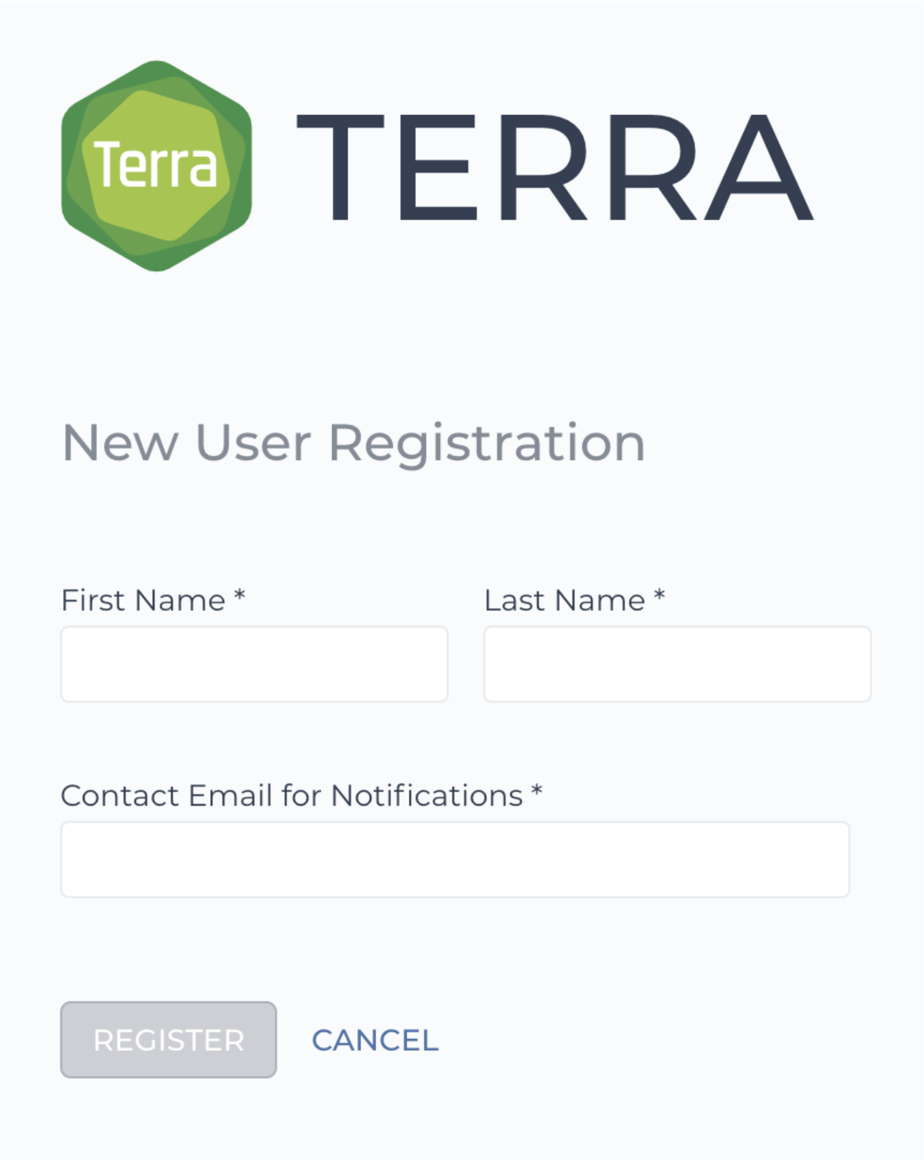
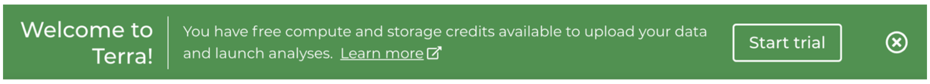
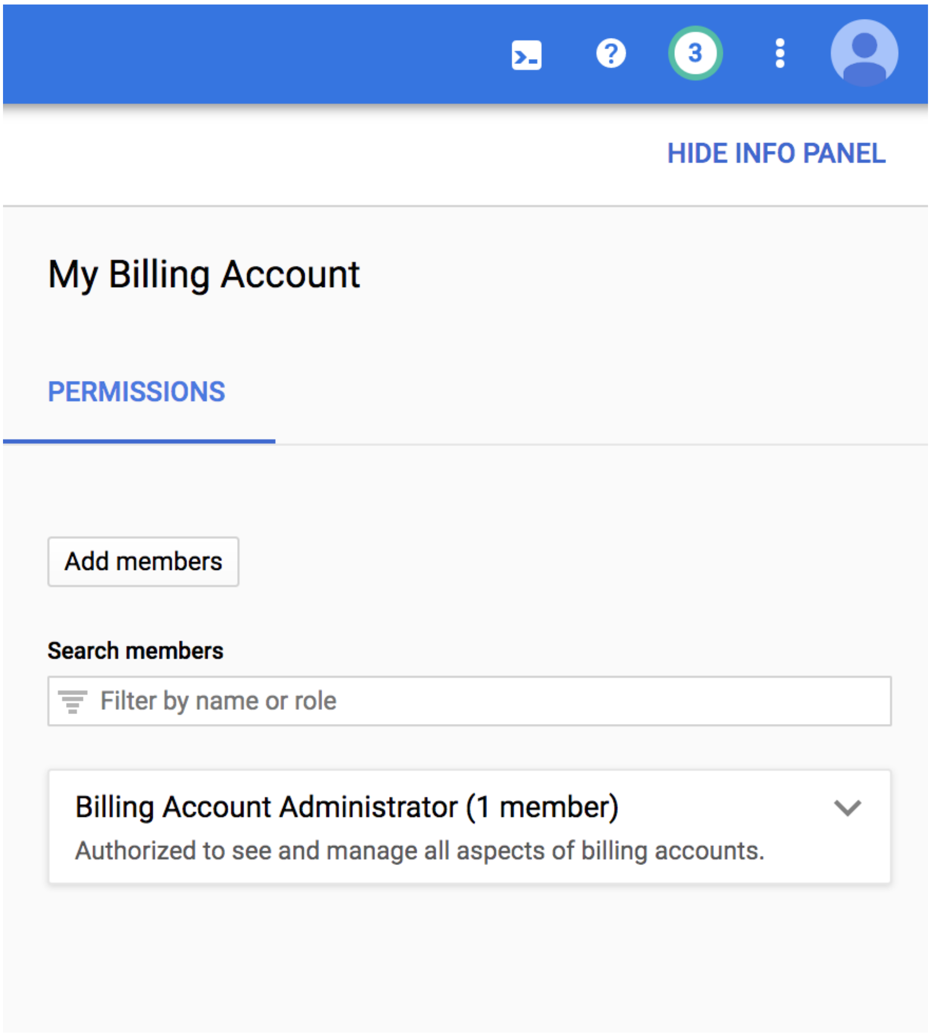


Figure 11.3: The New User Registration form.



You have free compute and storage credits available to upload your data and launch analyses. Learn more

Figure 11.4: The GCP console Billing account permissions panel.



HIDE INFO PANEL

My Billing Account

PERMISSIONS

Add members

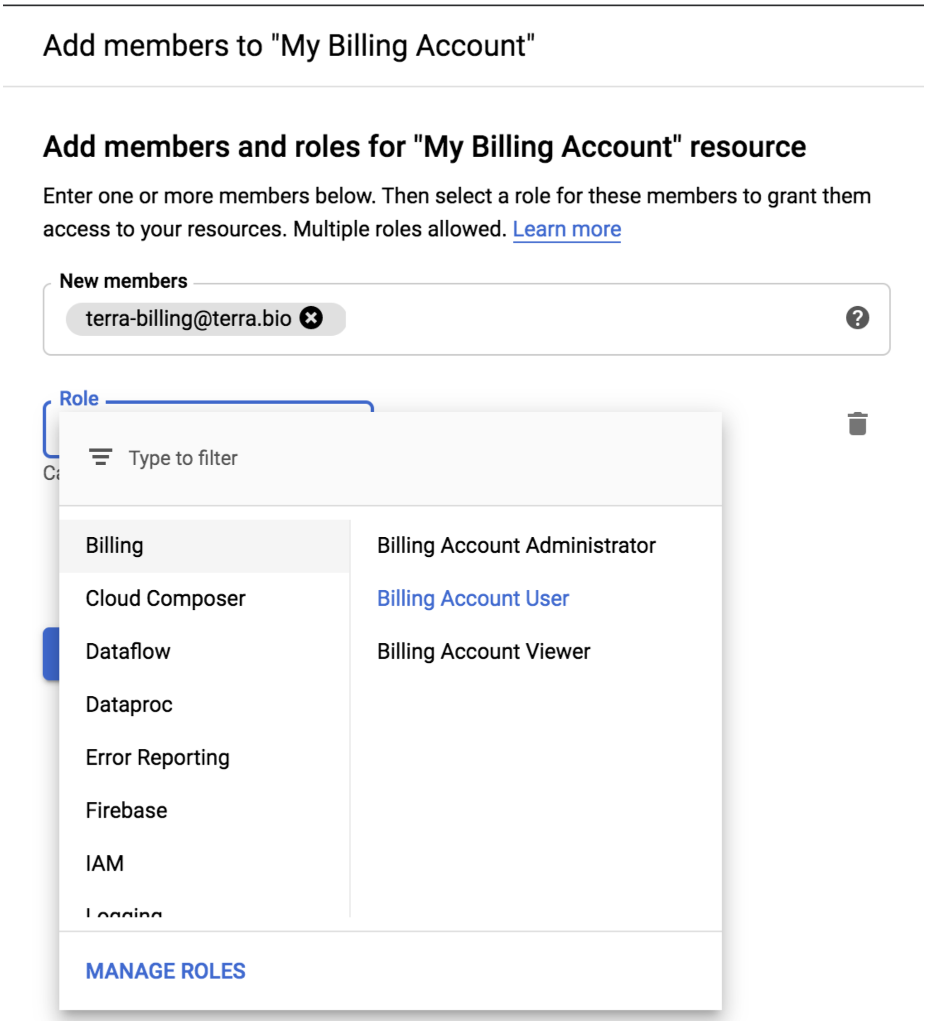
Search members

Filter by name or role

Billing Account Administrator (1 member)

Authorized to see and manage all aspects of billing accounts.

Figure 11.5: Adding the Terra billing user account as a user on a GCP billing account.



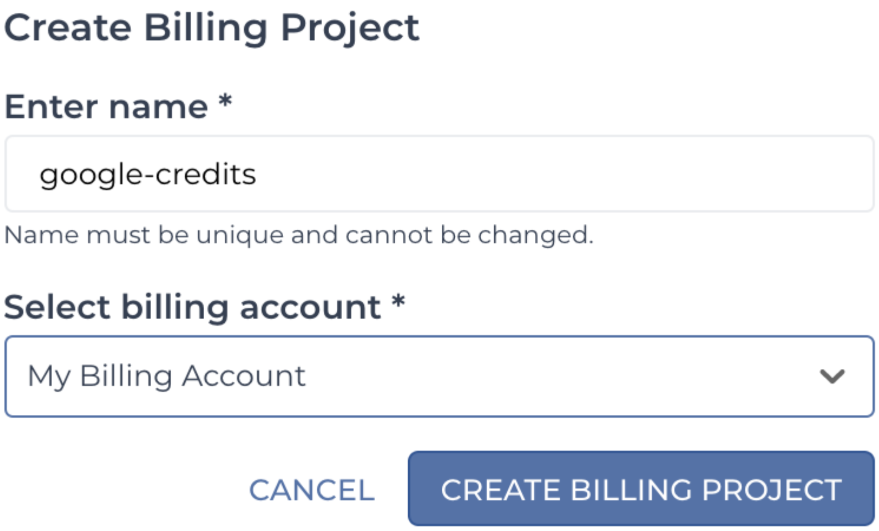
Add members to "My Billing Account"

Add members and roles for "My Billing Account" resource

Enter one or more members below. Then select a role for these members to grant them access to your resources. Multiple roles allowed. Learn more

New members: terra-billing@terra.bio

Figure 11.6: Using an existing billing account to create a billing project in Terra.



Create Billing Project

Enter name *

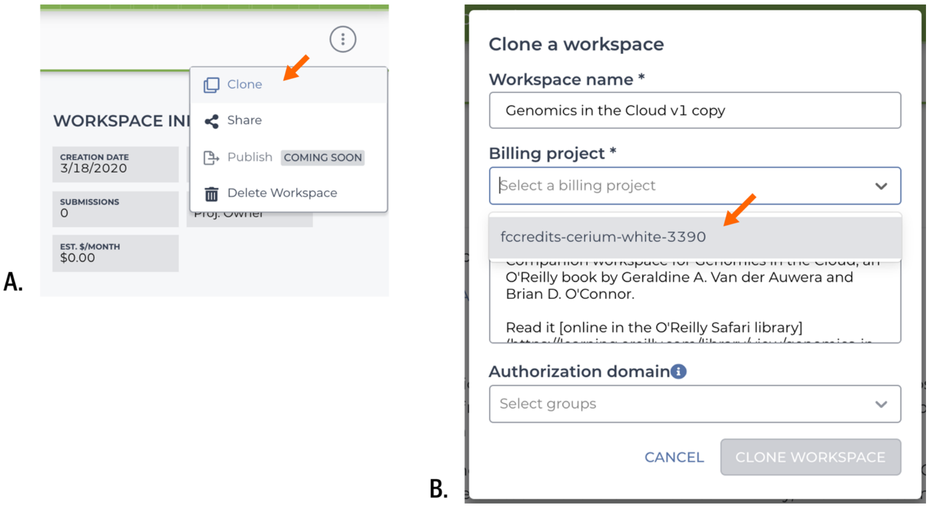
google-credits

Name must be unique and cannot be changed.

Select billing account *

My Billing Account

Figure 11.7: Cloning the preconfigured workspace. A) List of available actions; B) cloning form.



Clone a workspace

Workspace name *

Genomics in the Cloud v1 copy

Billing project *

Select a billing project

fccredits-cerium-white-3390

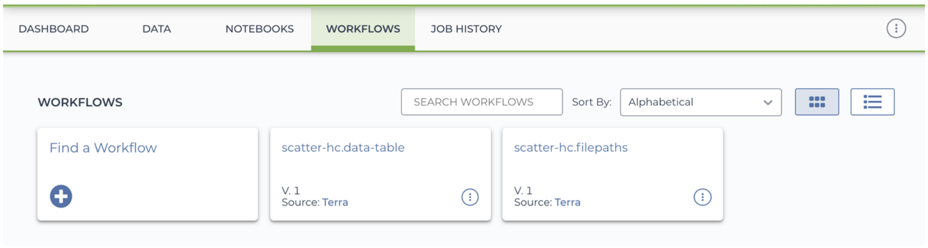
Companion workspace for Genomics in the Cloud, an O'Reilly book by Geraldine A. Van der Auwera and Brian D. O'Connor.

Read it [online in the O'Reilly Safari library]

Authorization domain

Select groups

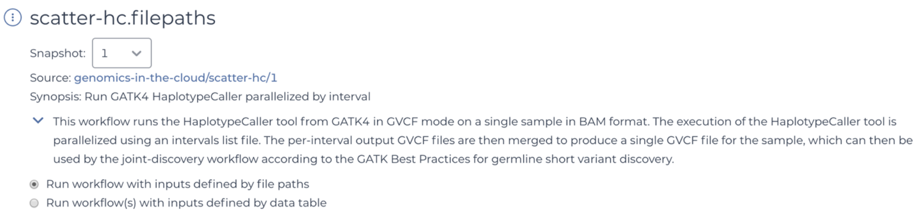
Figure 11.8: List of available workflow configurations.



scatter-hc.data-table
V. 1
Source: Terra

scatter-hc.filepaths
V. 1
Source: Terra

Figure 11.9: Viewing the workflow information summary.



scatter-hc.filepaths

Snapshot: 1

Source: genomics-in-the-cloud/scatter-hc/1

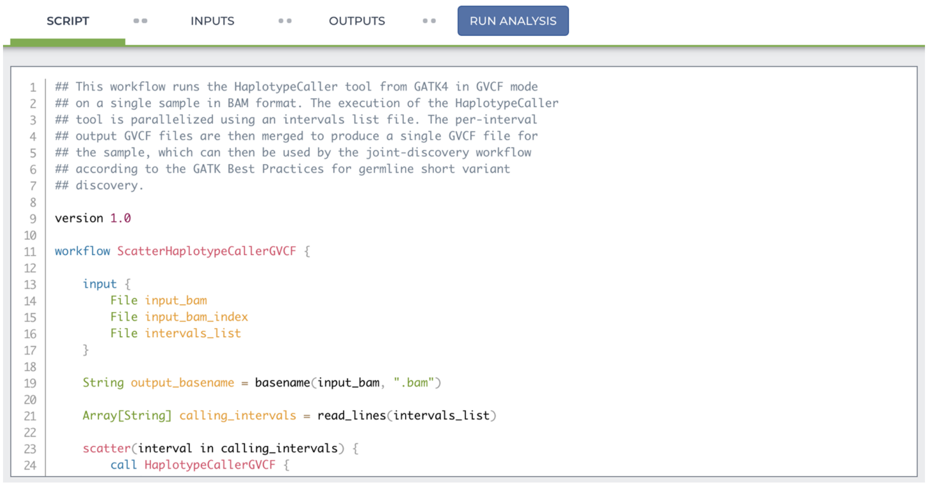
Synopsis: Run GATK4 HaplotypeCaller parallelized by interval

This workflow runs the HaplotypeCaller tool from GATK4 in GVCF mode on a single sample in BAM format. The execution of the HaplotypeCaller tool is parallelized using an intervals list file. The per-interval output GVCF files are then merged to produce a single GVCF file for the sample, which can then be used by the joint-discovery workflow according to the GATK Best Practices for germline short variant discovery.

● Run workflow with inputs defined by file paths

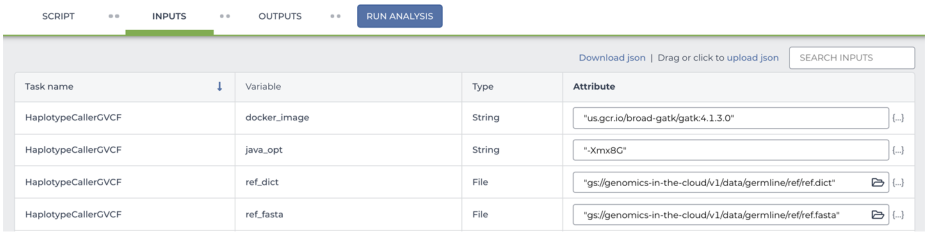
○ Run workflow(s) with inputs defined by data table

Figure 11.10: Viewing the workflow script.



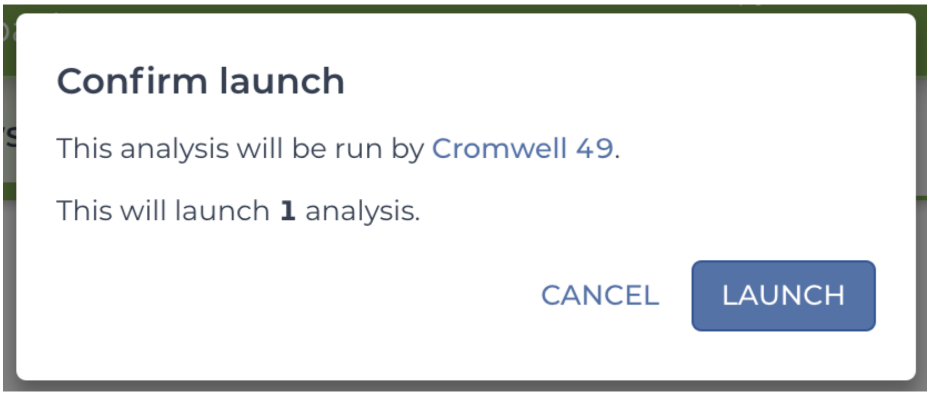
```
## This workflow runs the HaplotypeCaller tool from GATK4 in GVCF mode
## on a single sample in BAM format. The execution of the HaplotypeCaller
## tool is parallelized using an intervals list file. The per-interval
## output GVCF files are then merged to produce a single GVCF file for
## the sample, which can then be used by the joint-discovery workflow
## according to the GATK Best Practices for germline short variant
## discovery.

version 1.0

workflow ScatterHaplotypeCallerGVCF {

    input {
        File input_bam
        File input_bam_index
        File intervals_list
    }

    String output_basename = basename(input_bam, ".bam")

    Array[String] calling_intervals = read_lines(intervals_list)

    scatter(interval in calling_intervals) {
        call HaplotypeCallerGVCF {
```

Figure 11.11: Viewing the workflow inputs.



Task name	Variable	Type	Attribute
HaplotypeCallerGVCF	docker_image	String	"us.gcr.io/broad-gatk/gatk:4.1.3.0"
HaplotypeCallerGVCF	java_opt	String	"-Xmx8G"
HaplotypeCallerGVCF	ref_dict	File	"gs://genomics-in-the-cloud/v1/data/germline/ref/ref.dict"
HaplotypeCallerGVCF	ref_fasta	File	"gs://genomics-in-the-cloud/v1/data/germline/ref/ref.fasta"

Figure 11.12: The workflow launch dialog.



Confirm launch

This analysis will be run by Cromwell 49.

This will launch **1** analysis.

Figure 11.13: Overview of workflow submission in Terra.

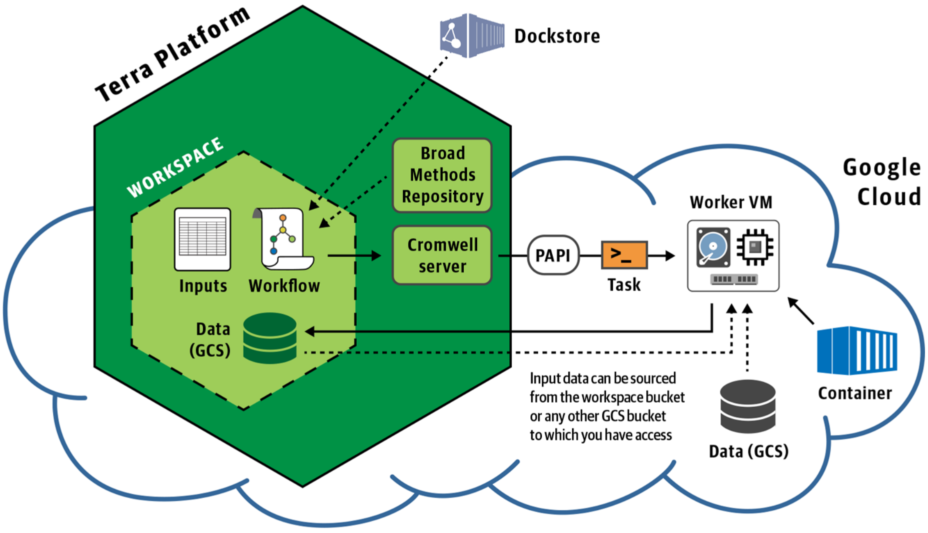
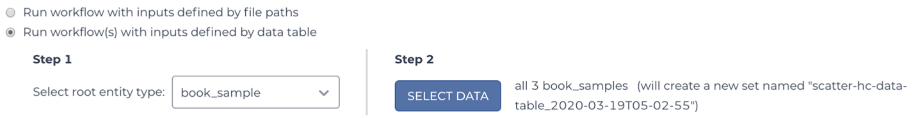


Figure 11.14: The second workflow is set to run on rows in a data table.



○ Run workflow with inputs defined by file paths

● Run workflow(s) with inputs defined by data table

Step 1

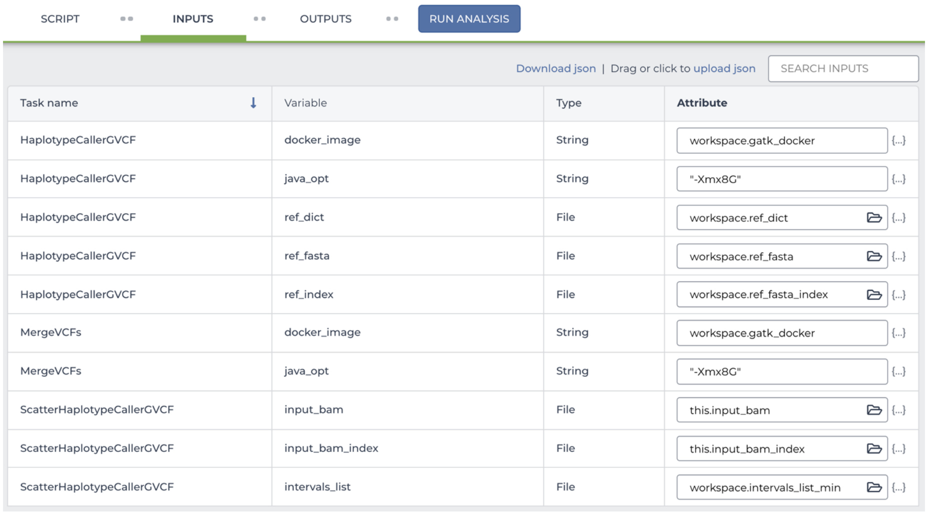
Select root entity type: book_sample

Step 2

SELECT DATA

all 3 book_samples (will create a new set named "scatter-hc-data-table_2020-03-19T05-02-55")

Figure 11.15: The workflow input configuration references data tables.



Task name	Variable	Type	Attribute
HaplotypeCallerGVCF	docker_image	String	workspace.gatk_docker
HaplotypeCallerGVCF	java_opt	String	"-Xmx8G"
HaplotypeCallerGVCF	ref_dict	File	workspace.ref_dict
HaplotypeCallerGVCF	ref_fasta	File	workspace.ref_fasta
HaplotypeCallerGVCF	ref_index	File	workspace.ref_fasta_index
MergeVCFs	docker_image	String	workspace.gatk_docker
MergeVCFs	java_opt	String	"-Xmx8G"
ScatterHaplotypeCallerGVCF	input_bam	File	this.input_bam
ScatterHaplotypeCallerGVCF	input_bam_index	File	this.input_bam_index
ScatterHaplotypeCallerGVCF	intervals_list	File	workspace.intervals_list_min

Figure 11.16: Viewing the menu of data tables on the DATA tab.

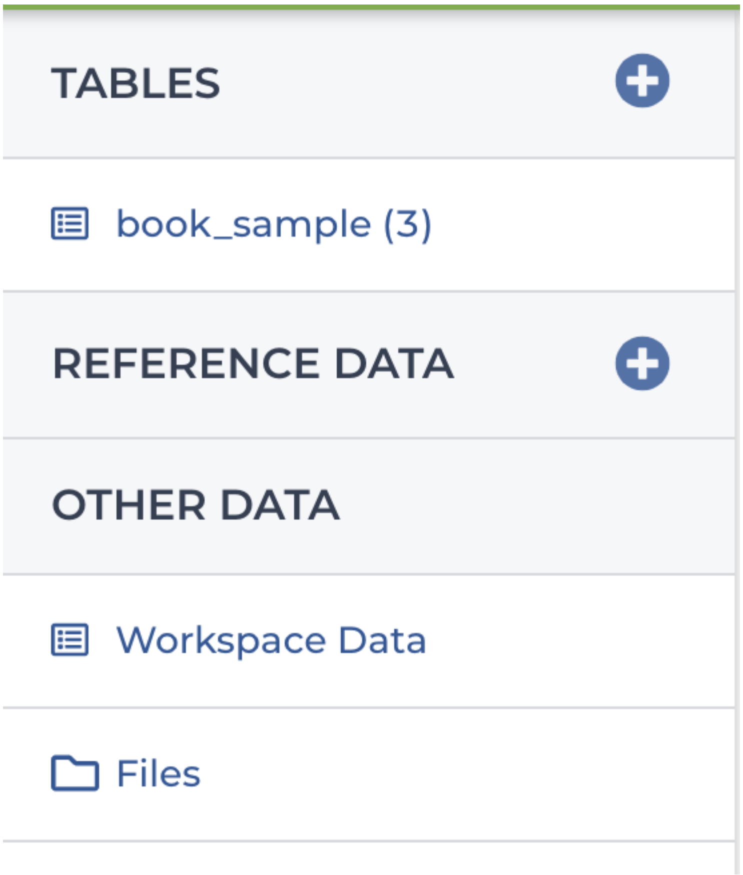
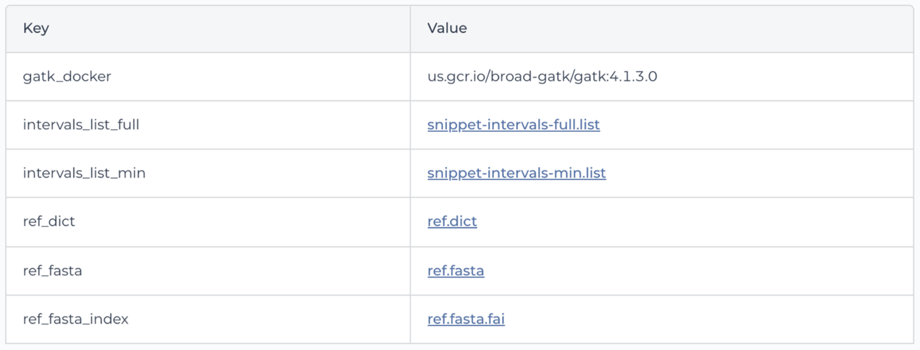
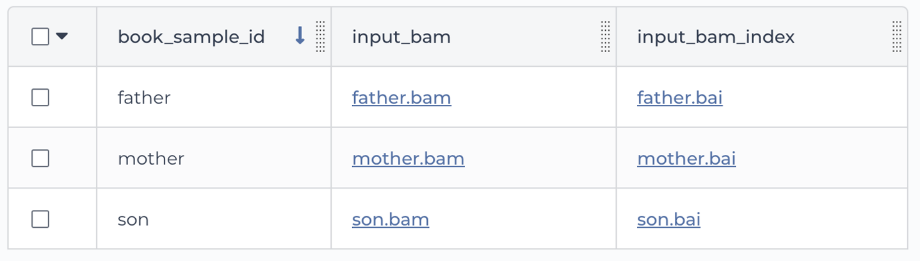


Figure 11.17: The Workspace Data table.



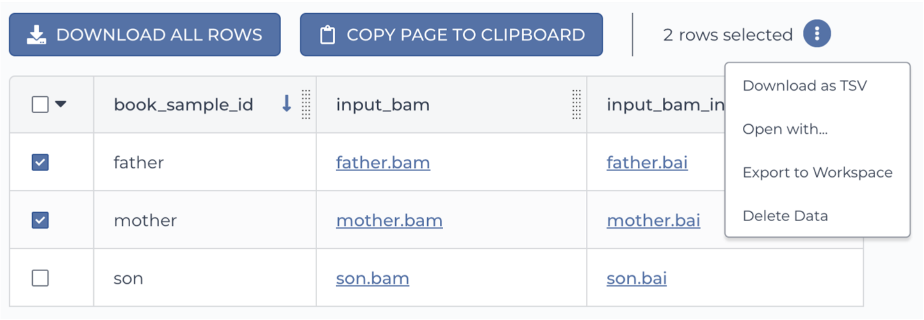
Key	Value
gatk_docker	us.gcr.io/broad-gatk/gatk:4.1.3.0
intervals_list_full	snippet-intervals-full.list
intervals_list_min	snippet-intervals-min.list
ref_dict	ref.dict
ref_fasta	ref.fasta
ref_fasta_index	ref.fasta.fai

Figure 11.18: The book$_sampletable$.



□ ▾	book_sample_id ↓	input_bam	input_bam_index
□	father	father.bam	father.bai
□	mother	mother.bam	mother.bai
□	son	son.bam	son.bai

Figure 11.19: Initiating an analysis directly on a subset of data.



□	book_sample_id	input_bam	input_bam_in
☑	father	father.bam	father.bai
☑	mother	mother.bam	mother.bai
□	son	son.bam	son.bai

Figure 11.20: Specifying a workflow to run on the selected data.

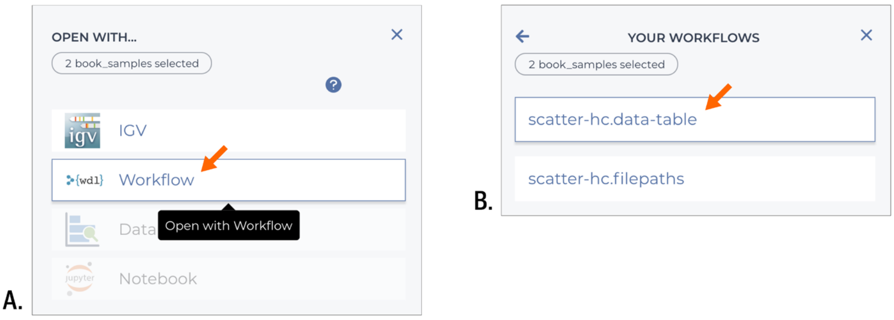
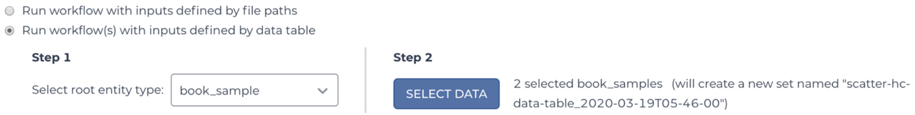


Figure 11.21: Configuration updated with data selection.



○ Run workflow with inputs defined by file paths
● Run workflow(s) with inputs defined by data table

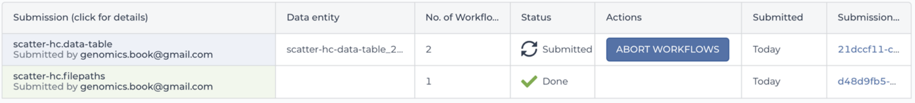
Step 1

Select root entity type: book_sample

Step 2

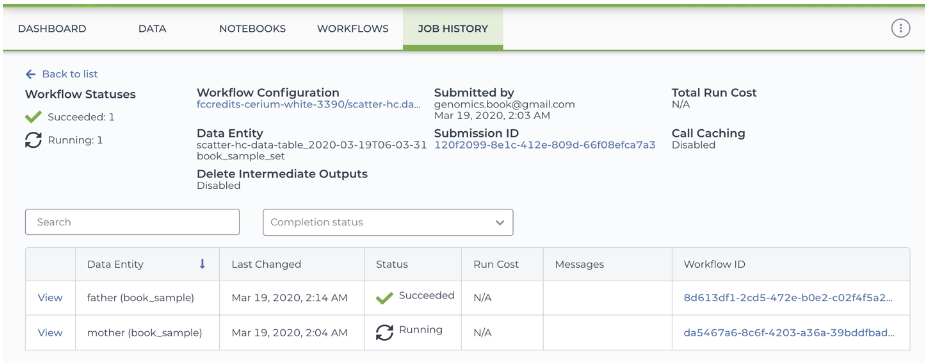
2 selected book_samples (will create a new set named "scatter-hc-data-table_2020-03-19T05-46-00")

Figure 11.22: List of submissions in the Job History.



Submission (click for details)	Data entity	No. of Workflo...	Status	Actions	Submitted	Submission...
scatter-hc.data-table Submitted by genomics.book@gmail.com	scatter-hc-data-table_2...	2	Submitted	ABORT WORKFLOWS	Today	21dccf11-c...
scatter-hc.filepaths Submitted by genomics.book@gmail.com		1	✓ Done		Today	d48d9fb5-...

Figure 11.23: The workflow submission summary page.



Workflow Statuses
✓ Succeeded: 1
Running: 1

Workflow Configuration
fccredits-cerium-white-3390/scatter-hc.da...

Data Entity
scatter-hc-data-table_2020-03-19T06-03-31
book_sample_set

Delete Intermediate Outputs
Disabled

Submitted by
genomics.book@gmail.com
Mar 19, 2020, 2:03 AM

Submission ID
120f2099-8e1c-412e-809d-66f08efca7a3

Total Run Cost
N/A

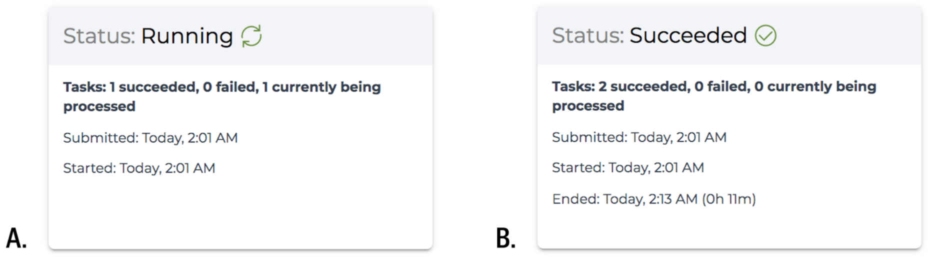
Call Caching
Disabled

Search

Completion status

	Data Entity	Last Changed	Status	Run Cost	Messages	Workflow ID
View	father (book_sample)	Mar 19, 2020, 2:14 AM	✓ Succeeded	N/A		8d613df1-2cd5-472e-b0e2-c02f4f5a2...
View	mother (book_sample)	Mar 19, 2020, 2:04 AM	Running	N/A		da5467a6-8c6f-4203-a36a-39bddfbad...

Figure 11.24: Workflow in A) Running state and, B) Succeeded state.



A.

Status: Running

Tasks: 1 succeeded, 0 failed, 1 currently being processed

Submitted: Today, 2:01 AM

Started: Today, 2:01 AM

B.

Status: Succeeded

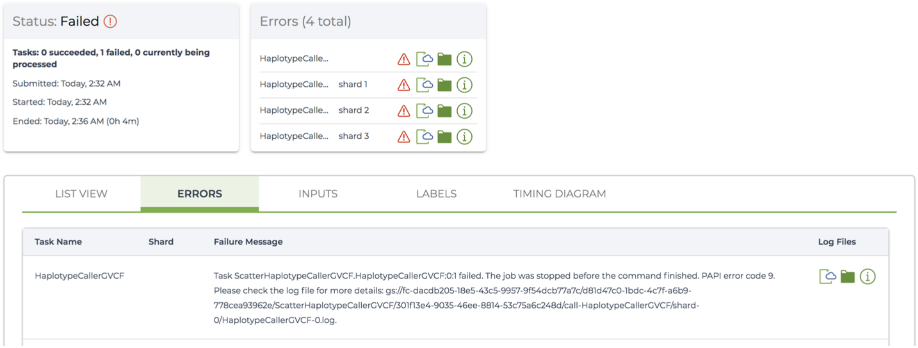
Tasks: 2 succeeded, 0 failed, 0 currently being processed

Submitted: Today, 2:01 AM

Started: Today, 2:01 AM

Ended: Today, 2:13 AM (0h 11m)

Figure 11.25: A workflow in Failed state with ERRORS summary and Failure Message.



Status: Failed

Tasks: 0 succeeded, 1 failed, 0 currently being processed

Submitted: Today, 2:32 AM

Started: Today, 2:32 AM

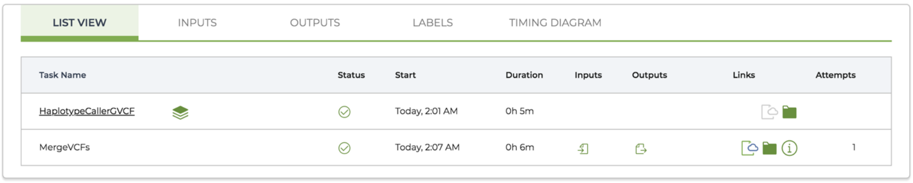
Ended: Today, 2:36 AM (0h 4m)

Errors (4 total)

HaplotypeCalle...	
HaplotypeCalle...	shard 1
HaplotypeCalle...	shard 2
HaplotypeCalle...	shard 3

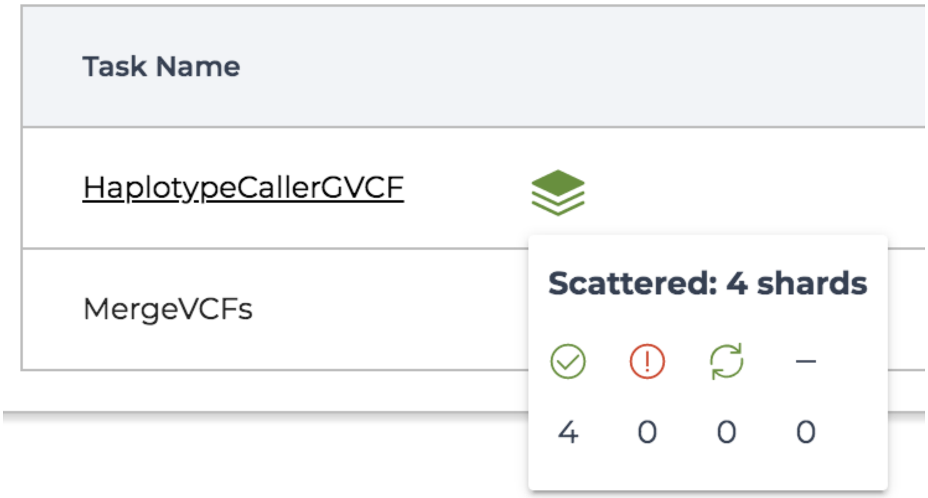
Task Name	Shard	Failure Message	Log Files
HaplotypeCallerGVCF		Task ScatterHaplotypeCallerGVCF.HaplotypeCallerGVCF:0:1 failed. The job was stopped before the command finished. PAPI error code 9. Please check the log file for more details: gs://fc-dacdb205-18e5-43c5-9957-9f54dcb77a7c/d81d47c0-1bdc-4c7f-a6b9-778cea93962e/ScatterHaplotypeCallerGVCF/301f13e4-9035-46ee-8814-53c75a6c248d/call-HaplotypeCallerGVCF/shard-0/HaplotypeCallerGVCF-0.log.	

Figure 11.26: List of tasks and related resources.



Task Name	Status	Start	Duration	Inputs	Outputs	Links	Attempts
HaplotypeCallerGVCF	✓	Today, 2:01 AM	0h 5m				
MergeVCFs	✓	Today, 2:07 AM	0h 6m				1

Figure 11.27: Viewing the status of shards for a scattered task.

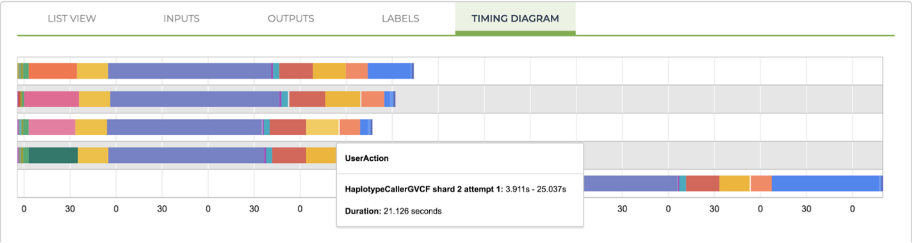


Task Name
HaplotypeCallerGVCF
MergeVCFs

Scattered: 4 shards

✓	!	↻	–
4	0	0	0

Figure 11.28: A timing diagram showing the breakdown of runtime per stage of execution for each task call.

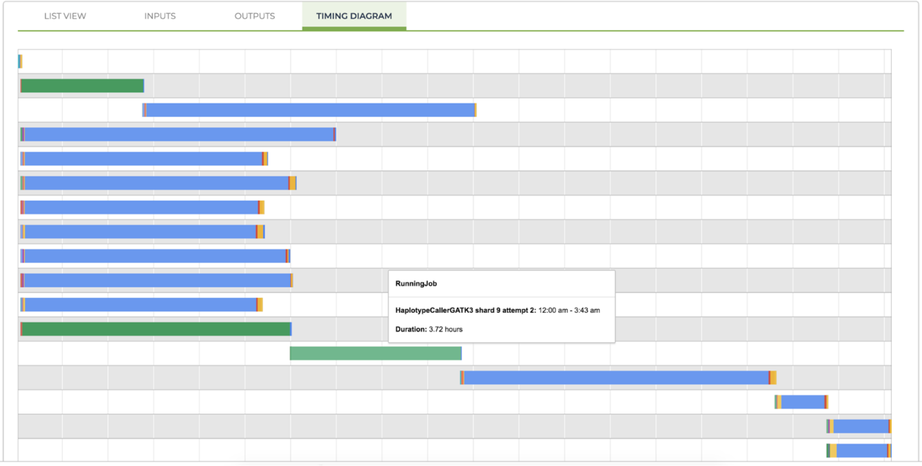


UserAction

HaplotypeCallerGVCF shard 2 attempt 1: 3.911s - 25.037s

Duration: 21.126 seconds

Figure 11.29: A timing diagram showing preempted calls (green bars, at lines 2, 12, and 13 from the top).

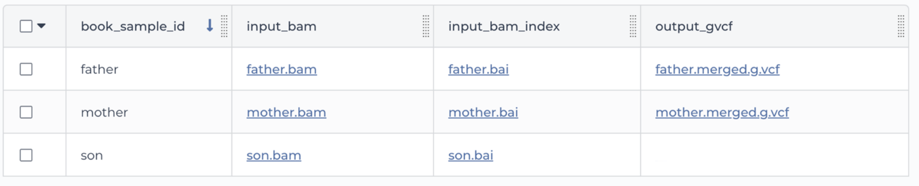


RunningJob

HaplotypeCallerGATK3 shard 9 attempt 2: 12:00 am - 3:43 am

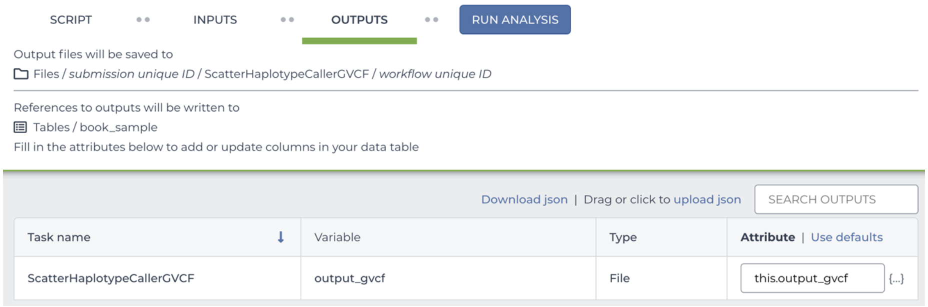
Duration: 3.72 hours

Figure 11.30: The data table showing the newly generated output$_gvcfcolumn$.



□ ▾	book_sample_id ↓	input_bam	input_bam_index	output_gvcf
□	father	father.bam	father.bai	father.merged.g.vcf
□	mother	mother.bam	mother.bai	mother.merged.g.vcf
□	son	son.bam	son.bai	

Figure 11.31: The workflow outputs configuration panel.

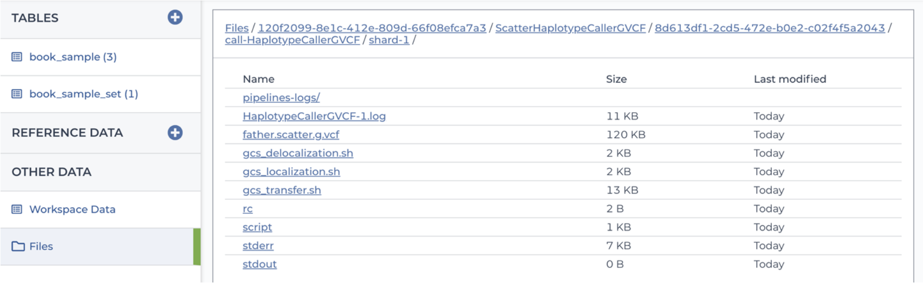


Output files will be saved to
Files / *submission unique ID* / ScatterHaplotypeCallerGVCF / *workflow unique ID*

References to outputs will be written to
Tables / book_sample
Fill in the attributes below to add or update columns in your data table

Task name ↓	Variable	Type	**Attribute** | Use defaults
ScatterHaplotypeCallerGVCF	output_gvcf	File	this.output_gvcf {...}

Figure 11.32: The file browser interface showing workflow outputs in the workspace bucket.



TABLES

- book_sample (3)
- book_sample_set (1)

REFERENCE DATA

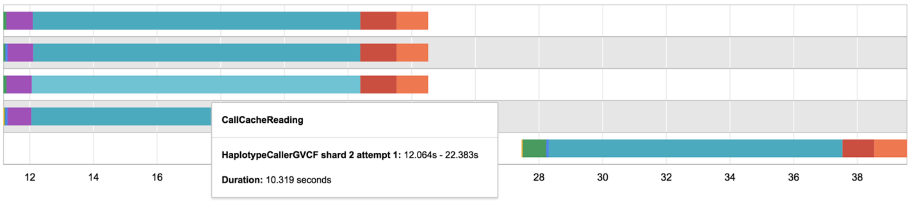
OTHER DATA

- Workspace Data
- Files

Files / 120f2099-8e1c-412e-809d-66f08efca7a3 / ScatterHaplotypeCallerGVCF / 8d613df1-2cd5-472e-b0e2-c02f4f5a2043 / call-HaplotypeCallerGVCF / shard-1 /

Name	Size	Last modified
pipelines-logs/		
HaplotypeCallerGVCF-1.log	11 KB	Today
father.scatter.g.vcf	120 KB	Today
gcs_delocalization.sh	2 KB	Today
gcs_localization.sh	2 KB	Today
gcs_transfer.sh	13 KB	Today
rc	2 B	Today
script	1 KB	Today
stderr	7 KB	Today
stdout	0 B	Today

Figure 11.33: A timing diagram showing CallCacheReading stage run time.

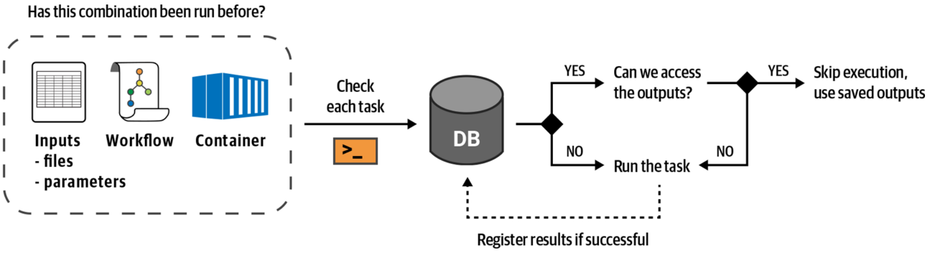


CallCacheReading

HaplotypeCallerGVCF shard 2 attempt 1: 12.064s - 22.383s

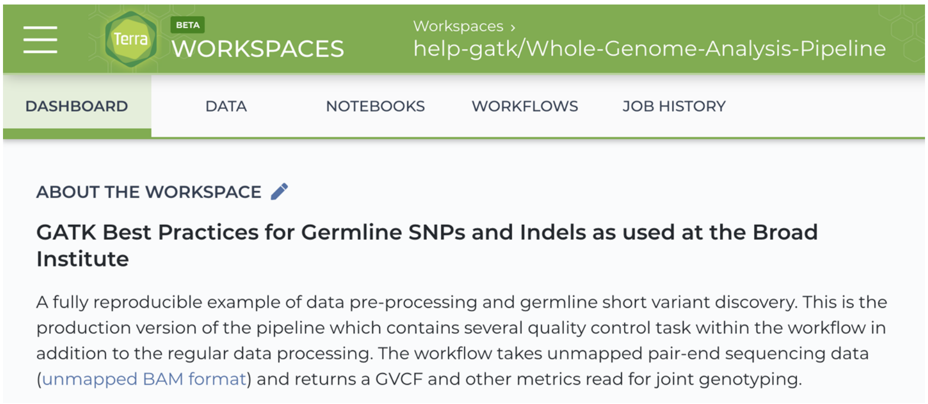
Duration: 10.319 seconds

Figure 11.34: Overview of Cromwell's call caching mechanism..



Has this combination been run before?
Inputs
- files
- parameters
Workflow
Container
Check each task
>_
DB
YES
Can we access the outputs?
YES
Skip execution, use saved outputs
NO
Run the task
NO
Register results if successful

Figure 11.35: Summary information for the Whole-Genome-Analysis-Pipeline workspace.

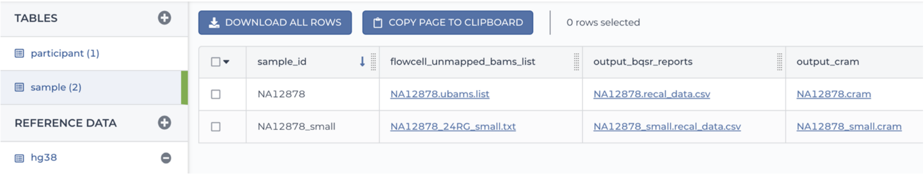


ABOUT THE WORKSPACE

GATK Best Practices for Germline SNPs and Indels as used at the Broad Institute

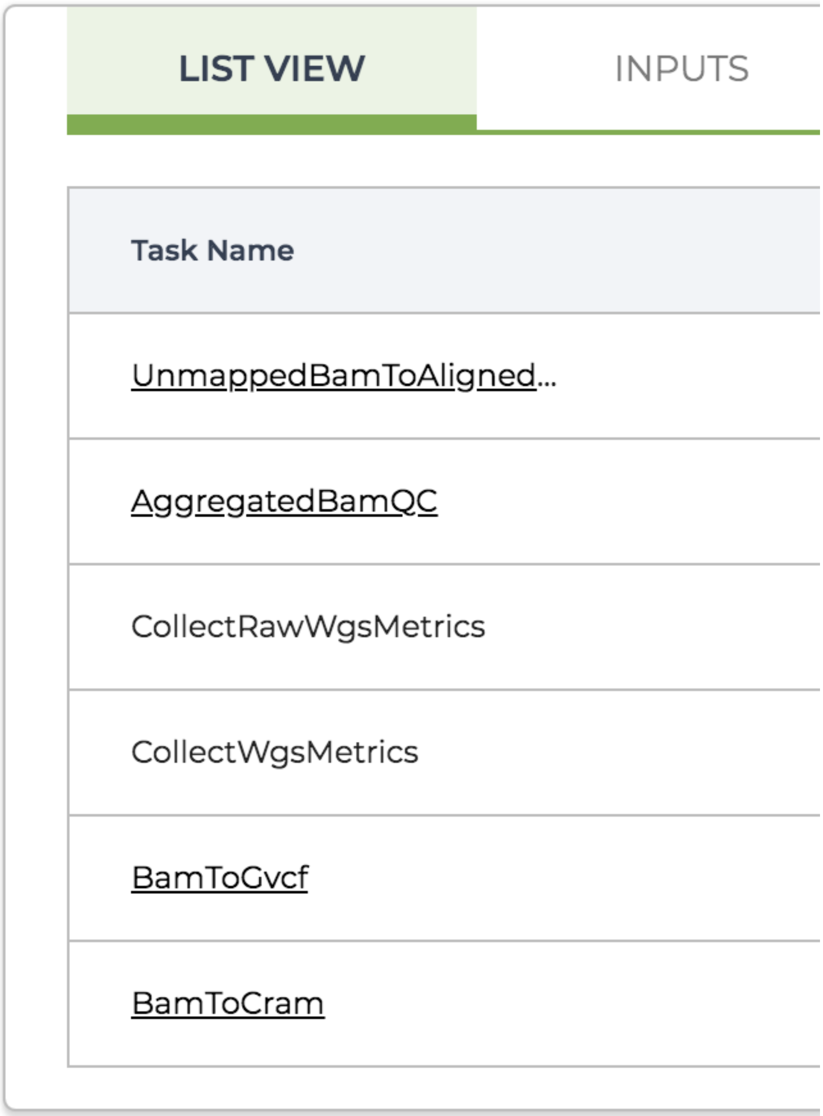
A fully reproducible example of data pre-processing and germline short variant discovery. This is the production version of the pipeline which contains several quality control task within the workflow in addition to the regular data processing. The workflow takes unmapped pair-end sequencing data (unmapped BAM format) and returns a GVCF and other metrics read for joint genotyping.

Figure 11.36: A list of tables and detailed view of the sample table.



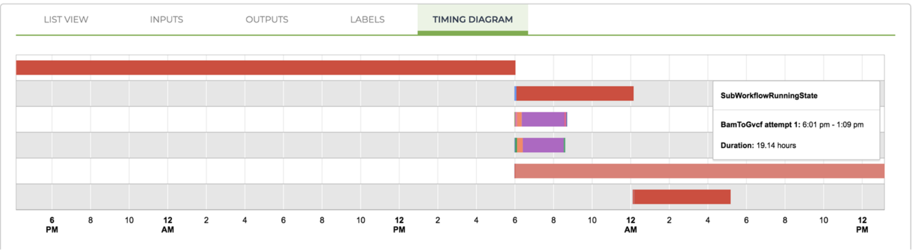
□	sample_id	flowcell_unmapped_bams_list	output_bqsr_reports	output_cram
□	NA12878	NA12878.ubams.list	NA12878.recal_data.csv	NA12878.cram
□	NA12878_small	NA12878_24RG_small.txt	NA12878_small.recal_data.csv	NA12878_small.cram

Figure 11.37: The List View of the task calls in the master workflow.



Task Name
UnmappedBamToAligned...
AggregatedBamQC
CollectRawWgsMetrics
CollectWgsMetrics
BamToGvcf
BamToCram

Figure 11.38: The timing diagram for the master workflow showing subworkflows (solid red bars) and individual tasks that are not bundled into subworkflows (multicolor bars).

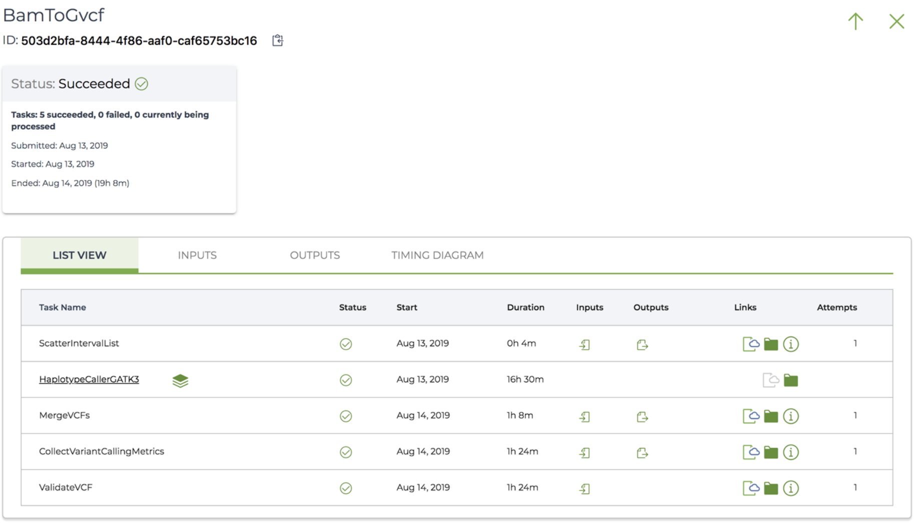


SubWorkflowRunningState

BamToGvcf attempt 1: 6:01 pm - 1:09 pm

Duration: 19.14 hours

Figure 11.39: The workflow details page for the BamToGvcf subworkflow.



BamToGvcf

ID: 503d2bfa-8444-4f86-aaf0-caf65753bc16

Status: Succeeded ✓

Tasks: 5 succeeded, 0 failed, 0 currently being processed

Submitted: Aug 13, 2019

Started: Aug 13, 2019

Ended: Aug 14, 2019 (19h 8m)

Task Name	Status	Start	Duration	Inputs	Outputs	Links	Attempts
ScatterIntervalList	✓	Aug 13, 2019	0h 4m				1
HaplotypeCallerGATK3	✓	Aug 13, 2019	16h 30m				
MergeVCFs	✓	Aug 14, 2019	1h 8m				1
CollectVariantCallingMetrics	✓	Aug 14, 2019	1h 24m				1
ValidateVCF	✓	Aug 14, 2019	1h 24m				1

Figure 11.40: File download windows showing A) the list of unmapped BAM files, and B) the final GVCF output.

Chapter 12 Interactive Analysis in Jupyter Notebook

Circling back to the GATK work from earlier chapters, we examine what that would all look like done in Jupyter Notebooks instead of the terminal shell. Between embedded IGV and ggplots galore, it looks good!

12.1 Introduction to Jupyter in Terra

12.1.1 Jupyter Notebooks in General

12.1.2 How Jupyter Notebooks Work in Terra

12.2 Getting Started with Jupyter in Terra

12.2.1 Inspecting and Customizing the Notebook Runtime Configuration

12.2.2 Opening Notebook in Edit Mode and Checking the Kernel

12.2.3 Running the Hello World Cells

12.2.4 Using gsutil to Interact with Google Cloud Storage Buckets

12.2.5 Setting Up a Variable Pointing to the Germline Data in the Book Bucket

12.2.6 Setting Up a Sandbox and Saving Output Files to the Workspace Bucket

12.3 Visualizing Genomic Data in an Embedded IGV Window

12.3.1 Setting Up the Embedded IGV Browser

12.3.2 Adding Data to the IGV Browser

12.3.3 Setting Up an Access Token to View Private Data

12.4 Running GATK Commands to Learn, Test, or Troubleshoot

12.4.1 Running a Basic GATK Command: HaplotypeCaller

12.4.2 Loading the Data (BAM and VCF) into IGV

12.4.3 Troubleshooting a Questionable Variant Call in the Embedded IGV Browser

12.4.4 Visualizing Variant Context Annotation Data

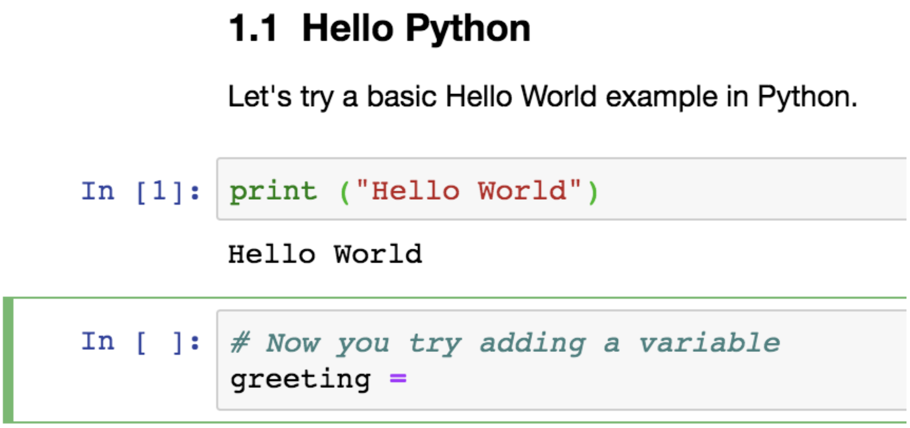
12.4.5 Exporting Annotations of Interest with VariantsToTable

12.4.6 Loading R Script to Make Plotting Functions Available

12.4.7 Making Density Plots for QUAL by Using makeDensityPlot

12.4.8 Making a Scatter Plot of QUAL Versus DP

12.4.9 Making a Scatter Plot Flanked by Marginal Density Plots



1.1 Hello Python

Let's try a basic Hello World example in Python.

```
In [1]: print ("Hello World")

        Hello World
```

```
In [ ]: # Now you try adding a variable
        greeting =
```

Figure 12.1: Doc text, code cell, and execution output in a Jupyter notebook.

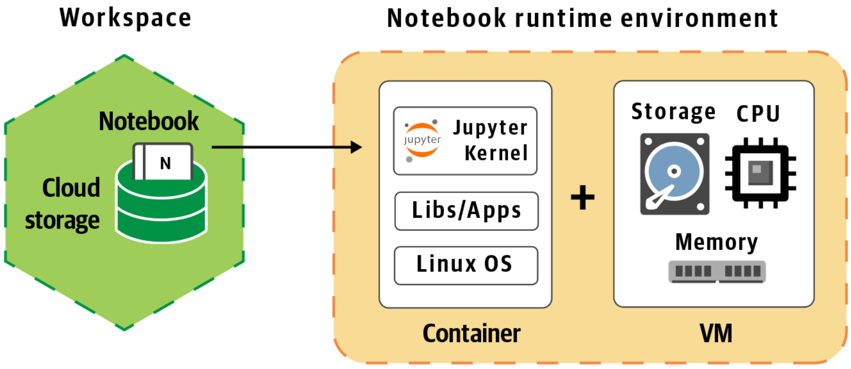


Figure 12.2: An overview of the Jupyter service in Terra.

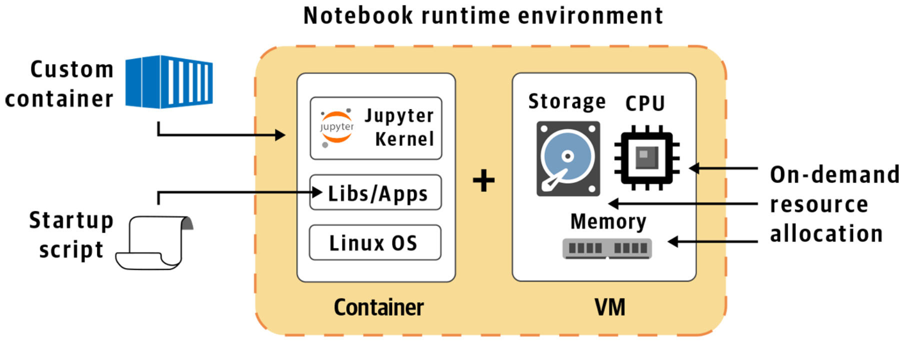


Figure 12.3: Options for customizing the software installed in the notebook runtime.

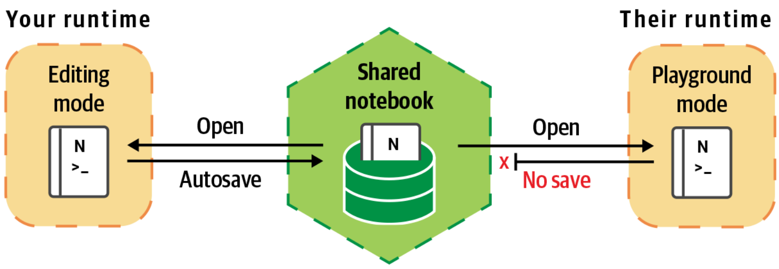
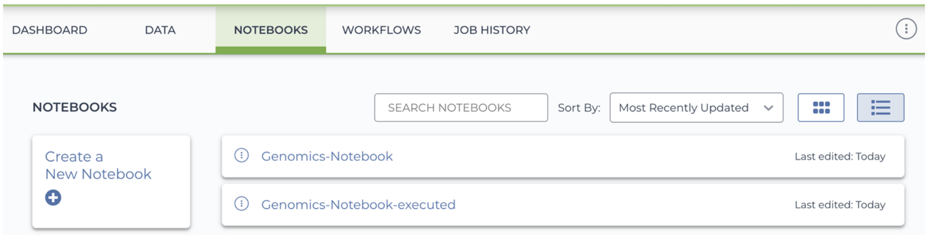


Figure 12.4: Notebooks in shared workspaces are protected from overwriting when two people open them concurrently.



Genomics-Notebook Last edited: Today

Genomics-Notebook-executed Last edited: Today

Figure 12.5: The Notebooks tab showing two copies of the notebook: one already executed and another without any previous results.

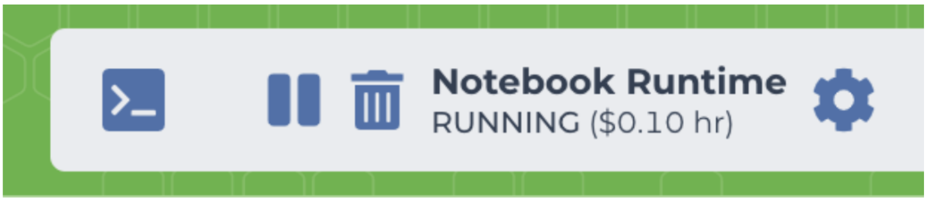
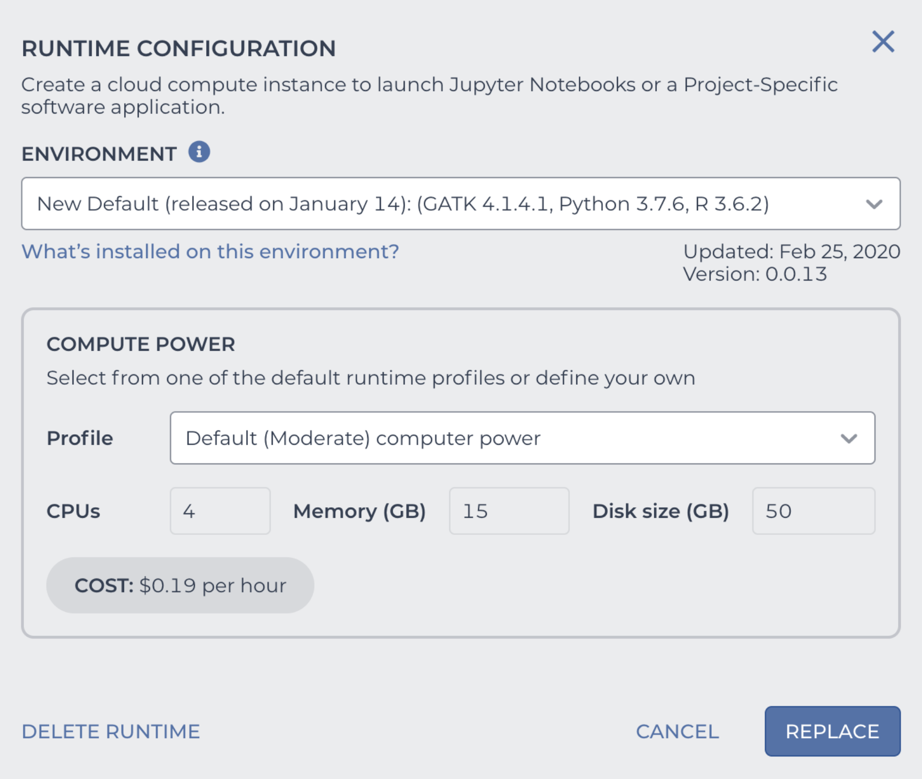


Figure 12.6: The Notebook Runtime status widget.



RUNTIME CONFIGURATION

Create a cloud compute instance to launch Jupyter Notebooks or a Project-Specific software application.

ENVIRONMENT

New Default (released on January 14): (GATK 4.1.4.1, Python 3.7.6, R 3.6.2)

What's installed on this environment?

Updated: Feb 25, 2020
Version: 0.0.13

COMPUTE POWER

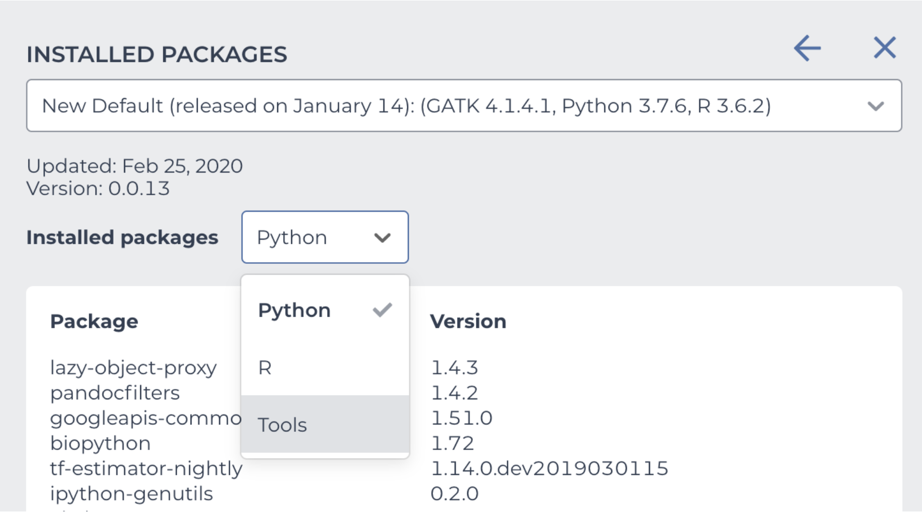
Select from one of the default runtime profiles or define your own

Profile: Default (Moderate) computer power

CPUs: 4 Memory (GB): 15 Disk size (GB): 50

COST: $0.19 per hour

Figure 12.7: The default Notebook Runtime configuration settings.



INSTALLED PACKAGES

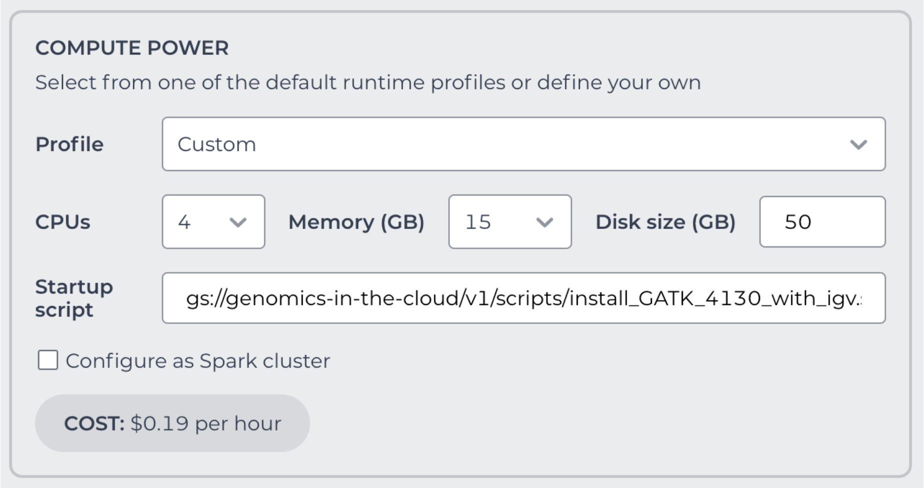
New Default (released on January 14): (GATK 4.1.4.1, Python 3.7.6, R 3.6.2)

Updated: Feb 25, 2020
Version: 0.0.13

Installed packages: Python

Package	Version
lazy-object-proxy	1.4.3
pandocfilters	1.4.2
googleapis-commo	1.51.0
biopython	1.72
tf-estimator-nightly	1.14.0.dev2019030115
ipython-genutils	0.2.0

Figure 12.8: Detailed view of the packages installed on the default runtime environment.



COMPUTE POWER

Select from one of the default runtime profiles or define your own

Profile: Custom

CPUs: 4 Memory (GB): 15 Disk size (GB): 50

Startup script: gs://genomics-in-the-cloud/v1/scripts/install_GATK_4130_with_igv.

□ Configure as Spark cluster

COST: $0.19 per hour

Figure 12.9: The Compute Power section allows you to specify a startup script if you choose the Custom profile.

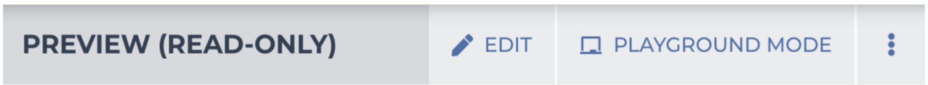


Figure 12.10: Menu on the notebook preview page displaying the main options: Preview, Edit, and Playground Mode.

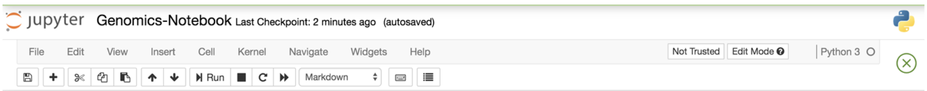


Figure 12.11: The standard Jupyter menu bar.

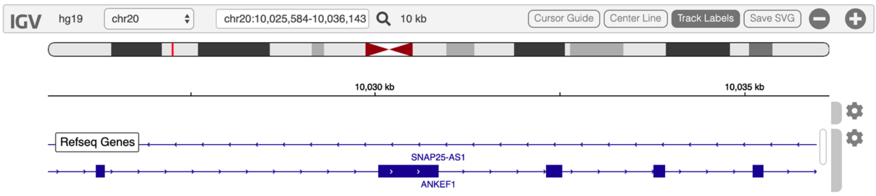


Figure 12.12: A newly created IGV browser.

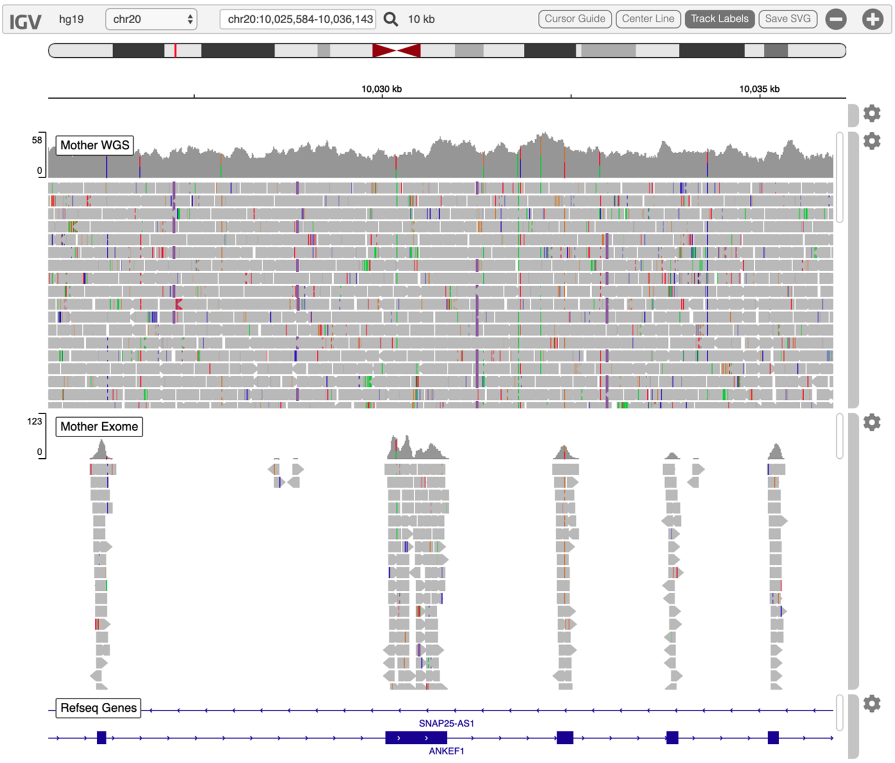


Figure 12.13: The IGV browser showing the two sequence data tracks.

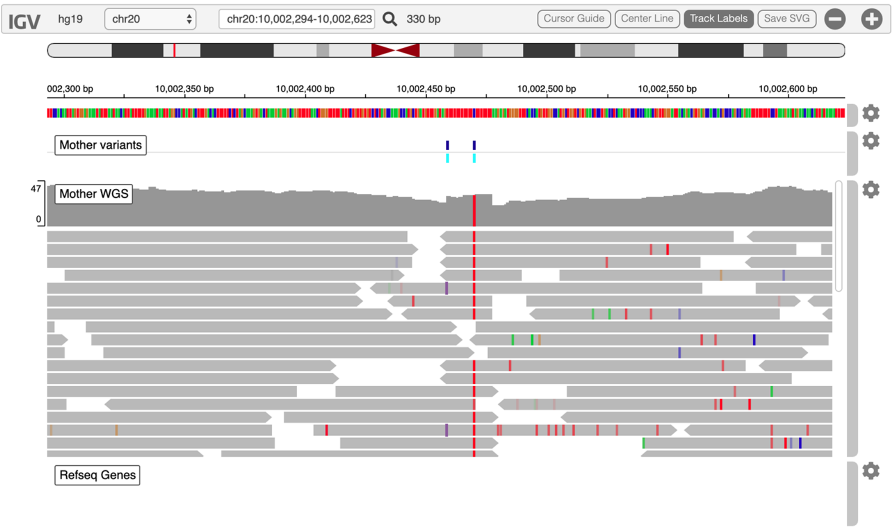


Figure 12.14: IGV.js rendering of the sequencing data ("Mother WGS" track) and output variants produced by HaplotypeCaller ("Mother variants" track).

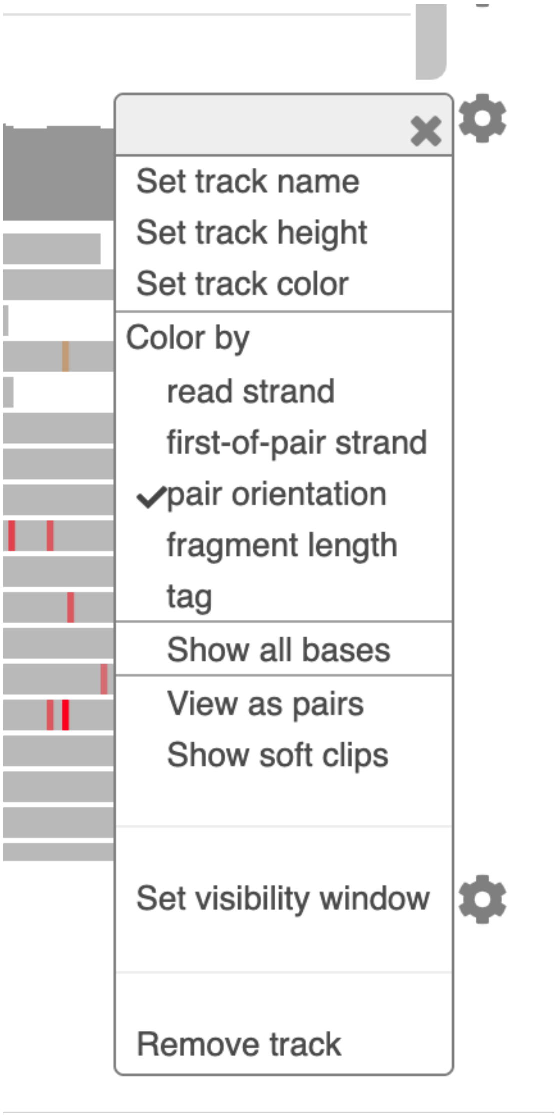


Figure 12.15: Menu of display options for the Mother WGS sequence data track.

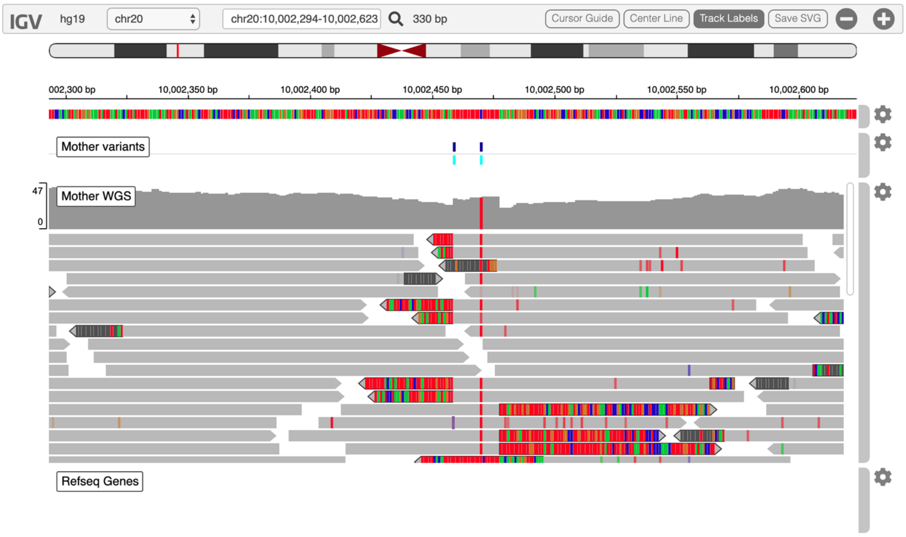


Figure 12.16: Display of soft clips.

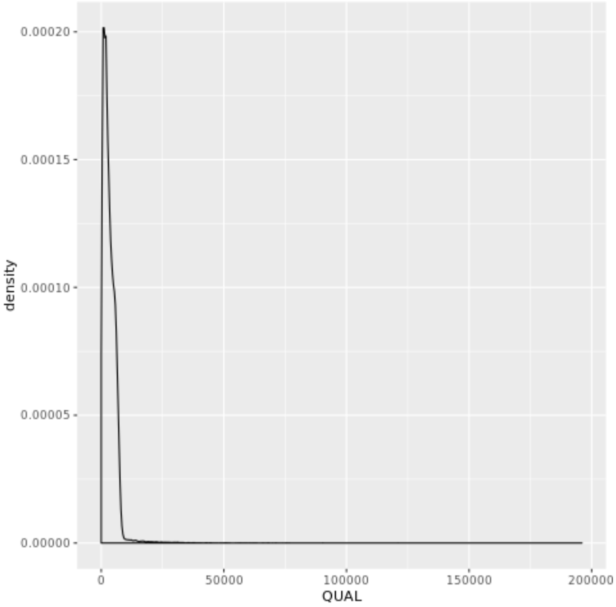


Figure 12.17: QUAL distribution.

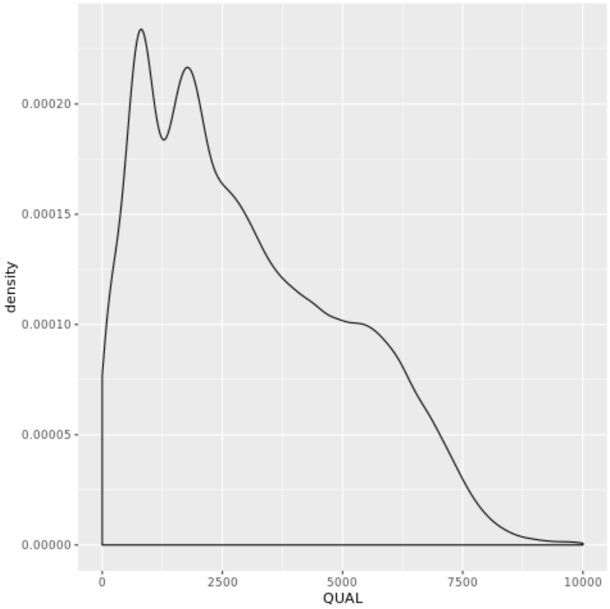


Figure 12.18: QUAL density plot.

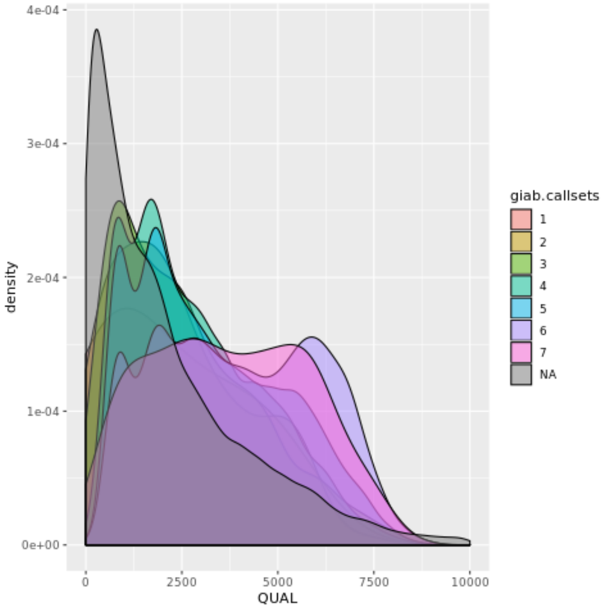


Figure 12.19: QUAL density plots by callsets from GiaB.

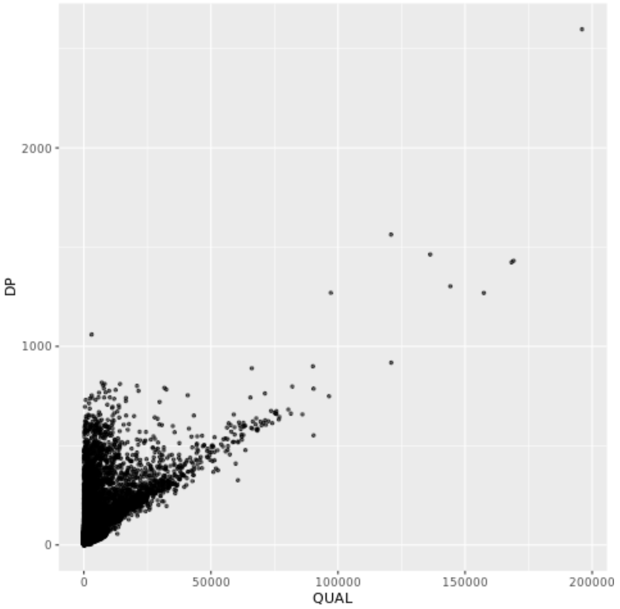


Figure 12.20: Scatter plot QUAL versus DP.

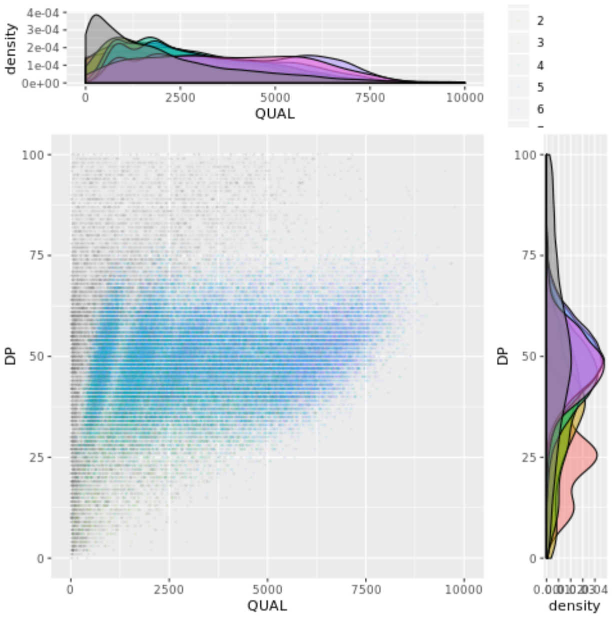


Figure 12.21: A scatter plot along with density plots.

Chapter 13 Assembling Your Own Workspace in Terra

Crossing the bridge from canned examples to importing your own data and methods into Terra in a few different scenarios. Draws on other services in the ecosystem including Dockstore and data repositories.

13.1 Managing Data Inside and Outside of Workspaces

13.1.1 The Workspace Bucket as Data Repository

13.1.2 Accessing Private Data That You Manage Outside of Terra

13.1.3 Accessing Data in the Terra Data Library

13.2 Re-Creating the Tutorial Workspace from Base Components

13.2.1 Creating a New Workspace

13.2.2 Adding the Workflow to the Methods Repository and Importing It into the Workspace

13.2.3 Creating a Configuration Quickly with a JSON File

13.2.4 Adding the Data Table

13.2.5 Filling in the Workspace Resource Data Table

13.2.6 Creating a Workflow Configuration That Uses the Data Tables

13.2.7 Adding the Notebook and Checking the Runtime Environment

13.2.8 Documenting Your Workspace and Sharing It

13.3 Starting from a GATK Best Practices Workspace

13.3.1 Cloning a GATK Best Practices Workspace

13.3.2 Examining GATK Workspace Data Tables to Understand How the Data Is Structured

13.3.3 Getting to Know the 1000 Genomes High Coverage Dataset

13.3.4 Copying Data Tables from the 1000 Genomes Workspace

13.3.5 Using TSV Load Files to Import Data from the 1000 Genomes Workspace

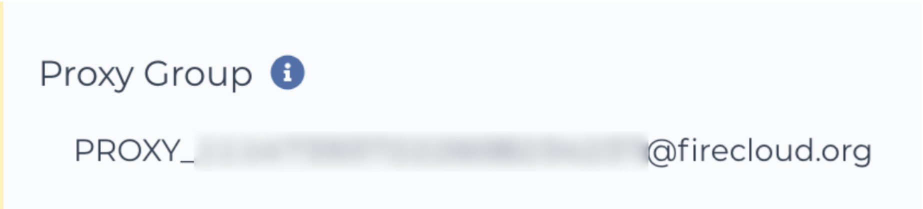
13.3.6 Running a Joint-Calling Analysis on the Federated Dataset

13.4 Building a Workspace Around a Dataset

13.4.1 Cloning the 1000 Genomes Data Workspace

13.4.2 Importing a Workflow from Dockstore

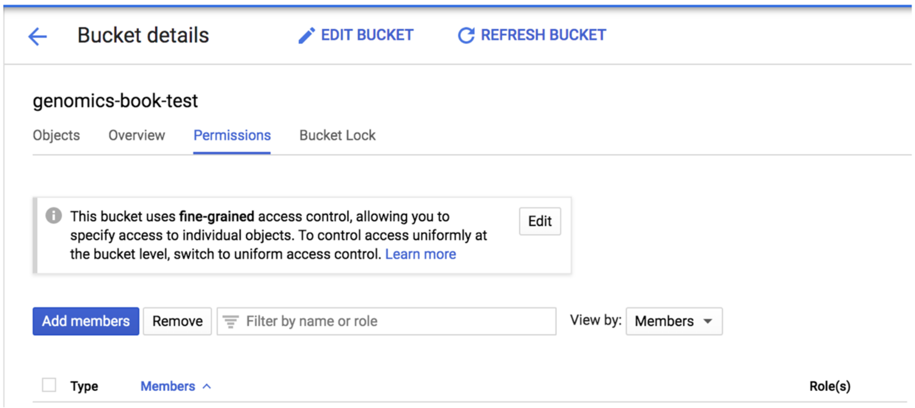
13.4.3 Configuring the Workflow to Use the Data Tables



Proxy Group

PROXY_[redacted]@firecloud.org

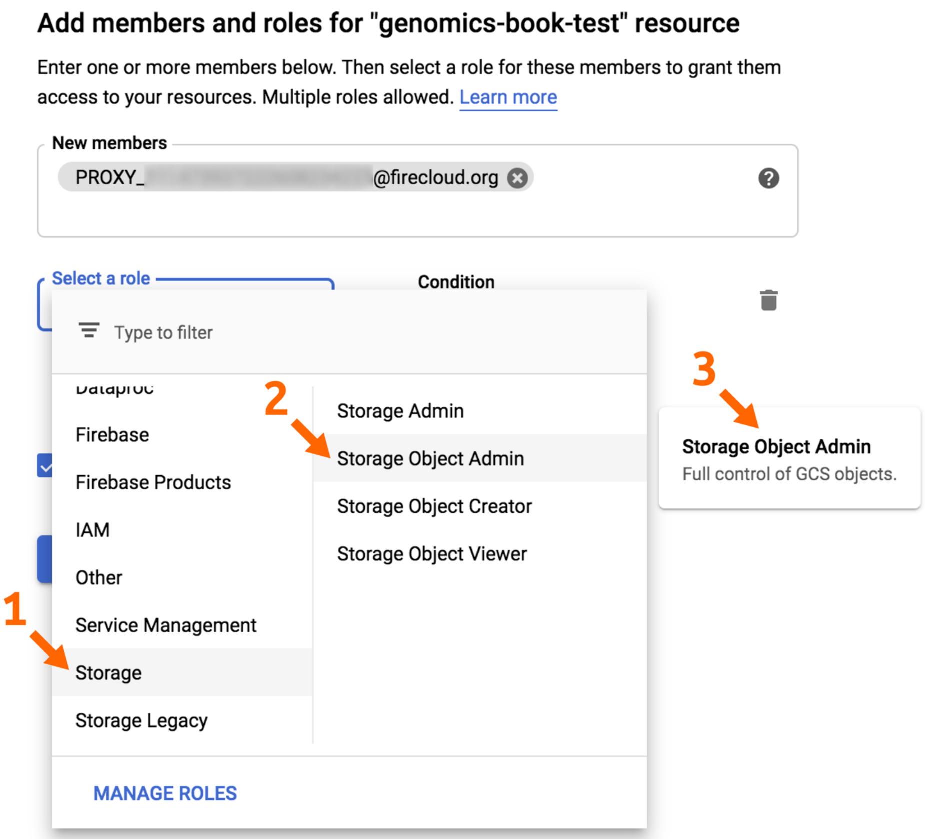
Figure 13.1: The proxy group identifier displayed in the user profile.



genomics-book-test

This bucket uses **fine-grained** access control, allowing you to specify access to individual objects. To control access uniformly at the bucket level, switch to uniform access control. Learn more

Figure 13.2: The bucket permissions panel showing accounts with access to the bucket.



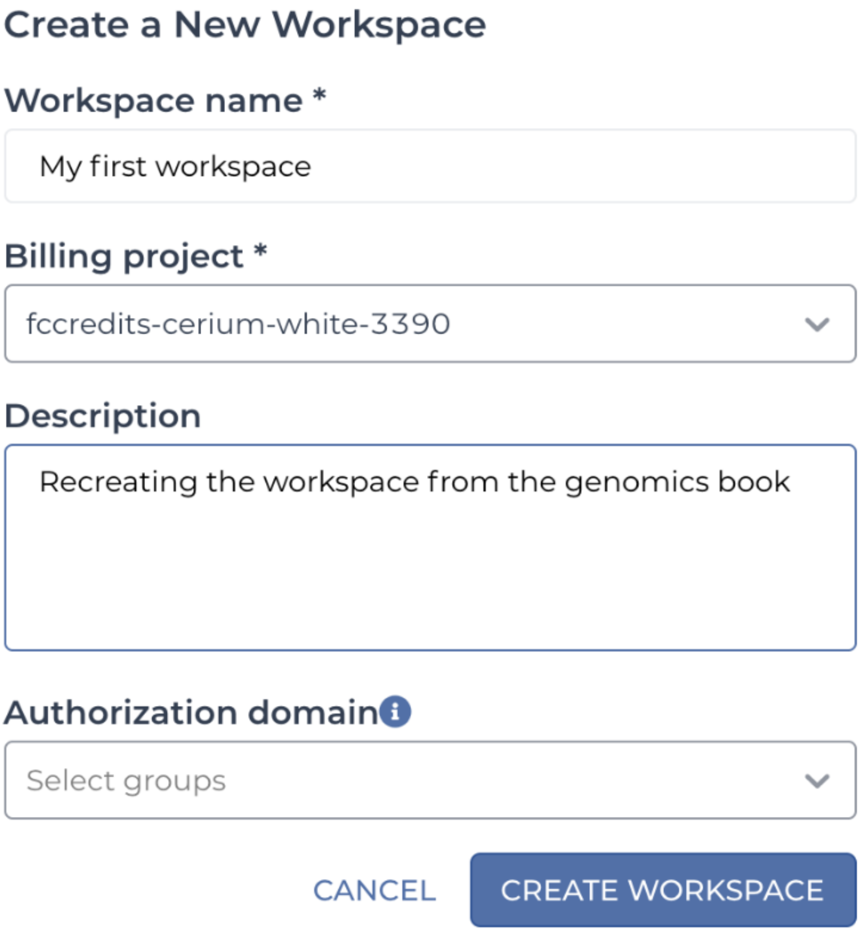
Add members and roles for "genomics-book-test" resource

Enter one or more members below. Then select a role for these members to grant them access to your resources. Multiple roles allowed. Learn more

New members: PROXY_[redacted]@firecloud.org

Storage Object Admin
Full control of GCS objects.

Figure 13.3: Granting access to a bucket to a new member.



Create a New Workspace

Workspace name *

My first workspace

Billing project *

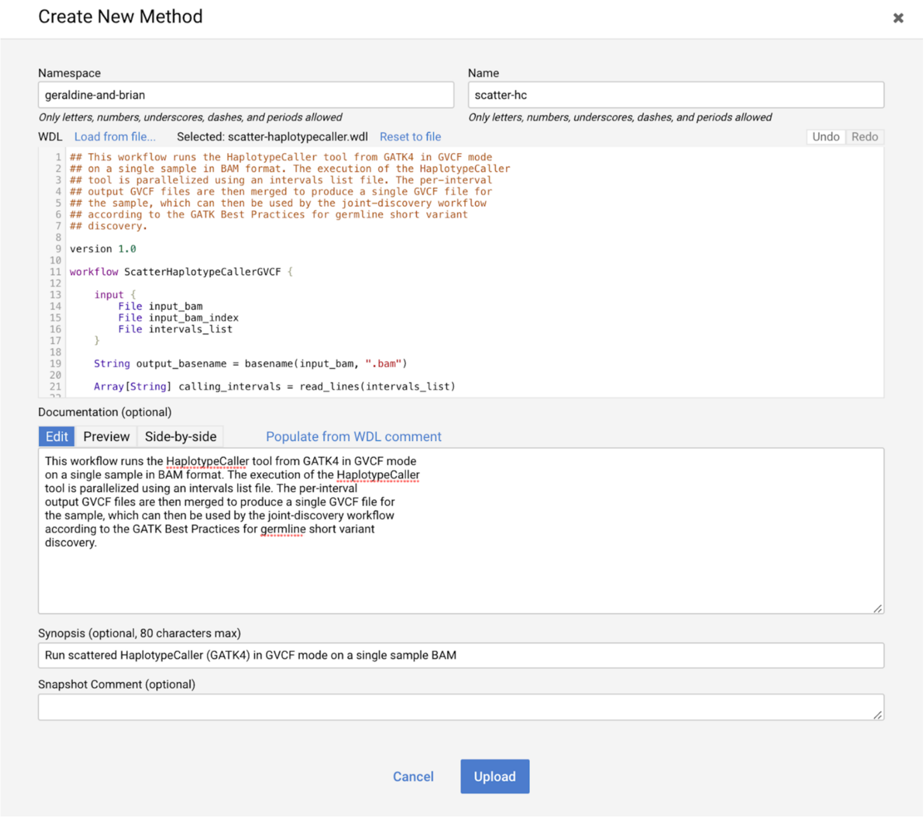
fccredits-cerium-white-3390

Description

Recreating the workspace from the genomics book

Authorization domain

Figure 13.4: The Create a New Workspace dialog box.



Create New Method

Namespace: geraldine-and-brian

Only letters, numbers, underscores, dashes, and periods allowed

Name: scatter-hc

Only letters, numbers, underscores, dashes, and periods allowed

WDL Load from file... Selected: scatter-haplotypecaller.wdl Reset to file

```
## This workflow runs the HaplotypeCaller tool from GATK4 in GVCF mode
## on a single sample in BAM format. The execution of the HaplotypeCaller
## tool is parallelized using an intervals list file. The per-interval
## output GVCF files are then merged to produce a single GVCF file for
## the sample, which can then be used by the joint-discovery workflow
## according to the GATK Best Practices for germline short variant
## discovery.

version 1.0

workflow ScatterHaplotypeCallerGVCF {

    input {
        File input_bam
        File input_bam_index
        File intervals_list
    }

    String output_basename = basename(input_bam, ".bam")

    Array[String] calling_intervals = read_lines(intervals_list)
```

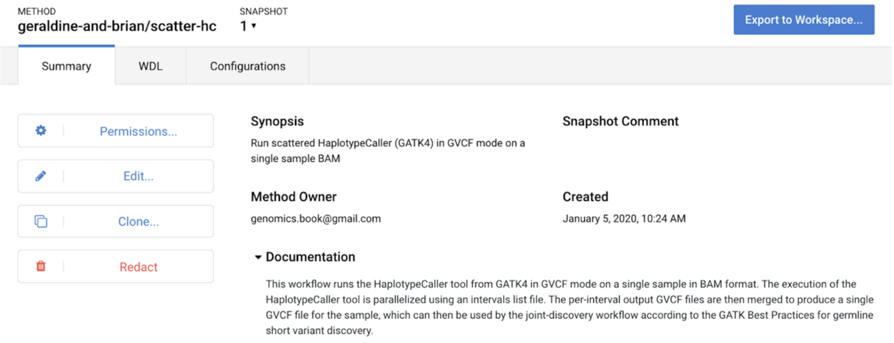
Documentation (optional)

This workflow runs the HaplotypeCaller tool from GATK4 in GVCF mode
on a single sample in BAM format. The execution of the HaplotypeCaller
tool is parallelized using an intervals list file. The per-interval
output GVCF files are then merged to produce a single GVCF file for
the sample, which can then be used by the joint-discovery workflow
according to the GATK Best Practices for germline short variant
discovery.

Synopsis (optional, 80 characters max): Run scattered HaplotypeCaller (GATK4) in GVCF mode on a single sample BAM

Snapshot Comment (optional)

Figure 13.5: The Create New Method page in the Broad Methods Repository.



METHOD: geraldine-and-brian/scatter-hc

SNAPSHOT: 1

Synopsis

Run scattered HaplotypeCaller (GATK4) in GVCF mode on a single sample BAM

Snapshot Comment

Method Owner

genomics.book@gmail.com

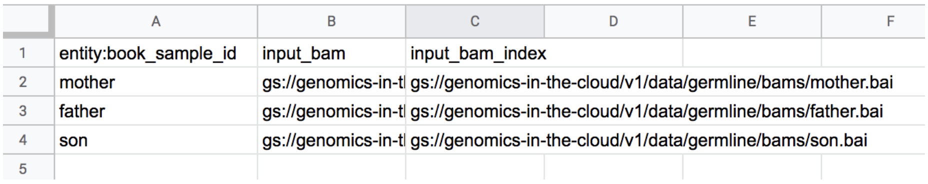
Created

January 5, 2020, 10:24 AM

Documentation

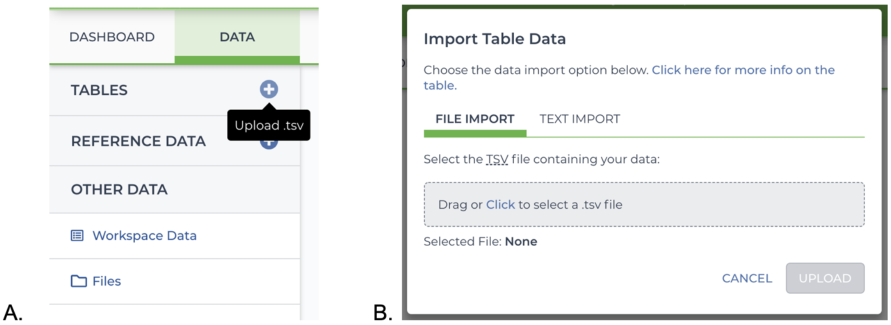
This workflow runs the HaplotypeCaller tool from GATK4 in GVCF mode on a single sample in BAM format. The execution of the HaplotypeCaller tool is parallelized using an intervals list file. The per-interval output GVCF files are then merged to produce a single GVCF file for the sample, which can then be used by the joint-discovery workflow according to the GATK Best Practices for germline short variant discovery.

Figure 13.6: Summary page for the newly created workflow.



	A	B	C	D	E	F
1	entity:book_sample_id	input_bam	input_bam_index			
2	mother	gs://genomics-in-t	gs://genomics-in-the-cloud/v1/data/germline/bams/mother.bai			
3	father	gs://genomics-in-t	gs://genomics-in-the-cloud/v1/data/germline/bams/father.bai			
4	son	gs://genomics-in-t	gs://genomics-in-the-cloud/v1/data/germline/bams/son.bai			
5						

Figure 13.7: A sample data table from the tutorial workspace, viewed in Google Sheets.



Import Table Data

Choose the data import option below. Click here for more info on the table.

Select the TSV file containing your data:

Drag or Click to select a .tsv file

Selected File: **None**

Figure 13.8: TSV load file import A) button, and B) dialog.

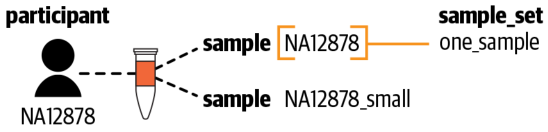
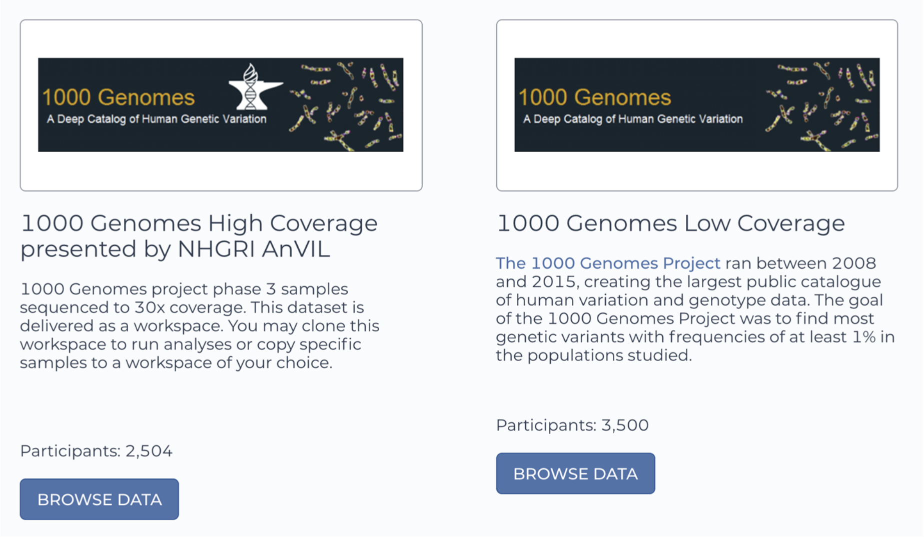


Figure 13.9: The data model—the structure of the example dataset.



1000 Genomes High Coverage presented by NHGRI AnVIL

1000 Genomes project phase 3 samples sequenced to 30x coverage. This dataset is delivered as a workspace. You may clone this workspace to run analyses or copy specific samples to a workspace of your choice.

Participants: 2,504

1000 Genomes Low Coverage

The 1000 Genomes Project ran between 2008 and 2015, creating the largest public catalogue of human variation and genotype data. The goal of the 1000 Genomes Project was to find most genetic variants with frequencies of at least 1% in the populations studied.

Participants: 3,500

Figure 13.10: The Terra Data Library contains two repositories of data from the 1000 Genomes Project.

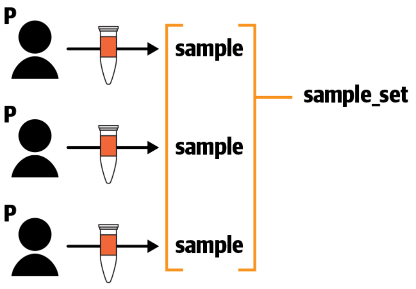
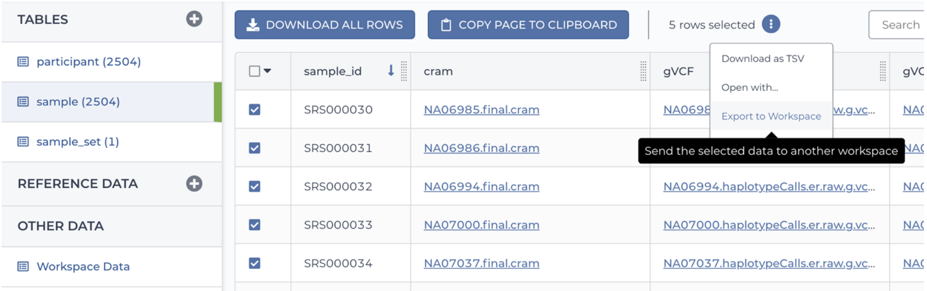


Figure 13.11: The data model for the 1000 Genomes High Coverage dataset.



TABLES
- participant (2504)
- sample (2504)
- sample_set (1)

REFERENCE DATA

OTHER DATA
- Workspace Data

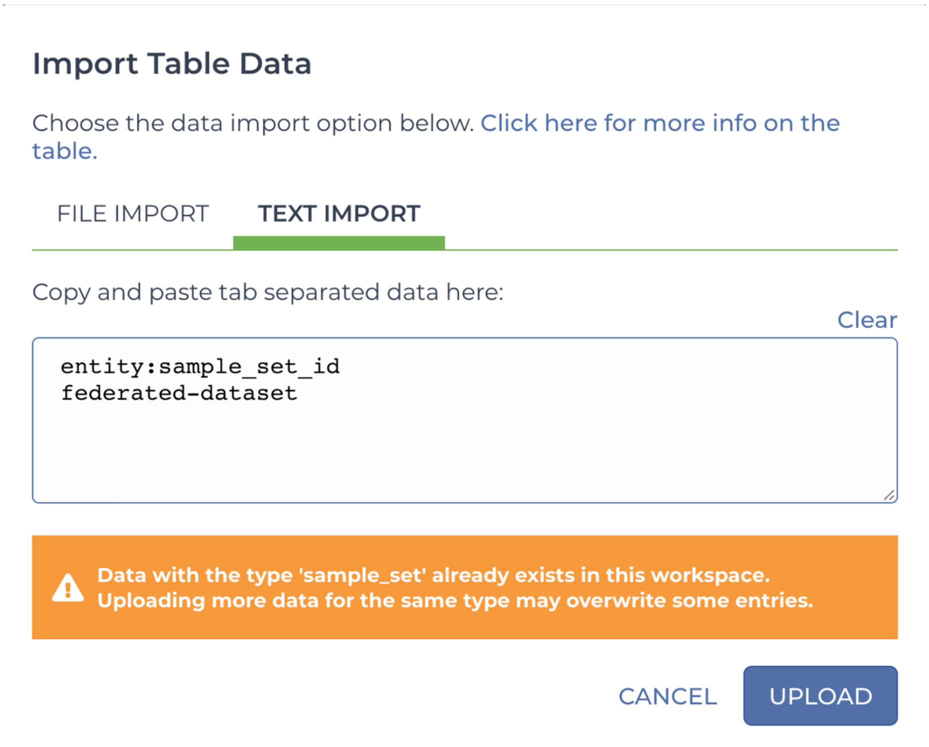
DOWNLOAD ALL ROWS | COPY PAGE TO CLIPBOARD | 5 rows selected | Search

Download as TSV
Open with...
Export to Workspace

Send the selected data to another workspace

☐	sample_id	cram	gVCF	gVC
☑	SRS000030	NA06985.final.cram	NA0698 ... w.g.vc...	NA(
☑	SRS000031	NA06986.final.cram		NA(
☑	SRS000032	NA06994.final.cram	NA06994.haplotypeCalls.er.raw.g.vc...	NA(
☑	SRS000033	NA07000.final.cram	NA07000.haplotypeCalls.er.raw.g.vc...	NA(
☑	SRS000034	NA07037.final.cram	NA07037.haplotypeCalls.er.raw.g.vc...	NA(

Figure 13.12: The Copy Data to Workspace dialog box.



Import Table Data

Choose the data import option below. Click here for more info on the table.

FILE IMPORT | TEXT IMPORT

Copy and paste tab separated data here:

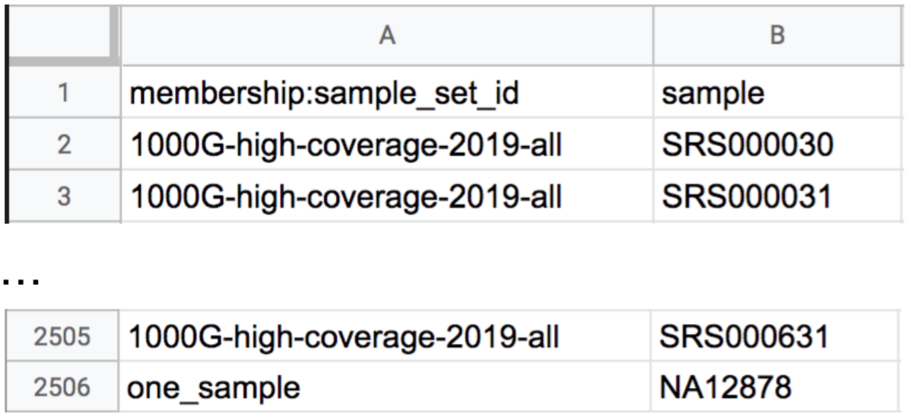
Clear

```
entity:sample_set_id
federated-dataset
```

Data with the type 'sample_set' already exists in this workspace. Uploading more data for the same type may overwrite some entries.

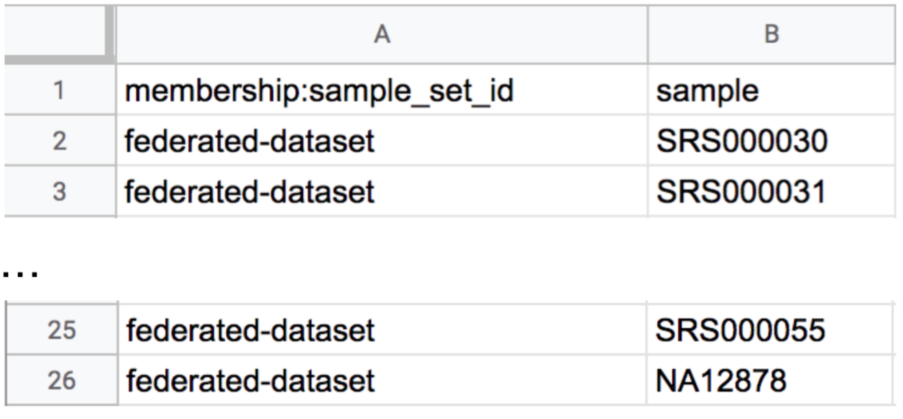
CANCEL | UPLOAD

Figure 13.13: Direct text import of TSV-formatted data table content.



	A	B
1	membership:sample_set_id	sample
2	1000G-high-coverage-2019-all	SRS000030
3	1000G-high-coverage-2019-all	SRS000031
...		
2505	1000G-high-coverage-2019-all	SRS000631
2506	one_sample	NA12878

Figure 13.14: Start and end rows of the membership load file sample$_{s}et_{m}embership.tsv$.

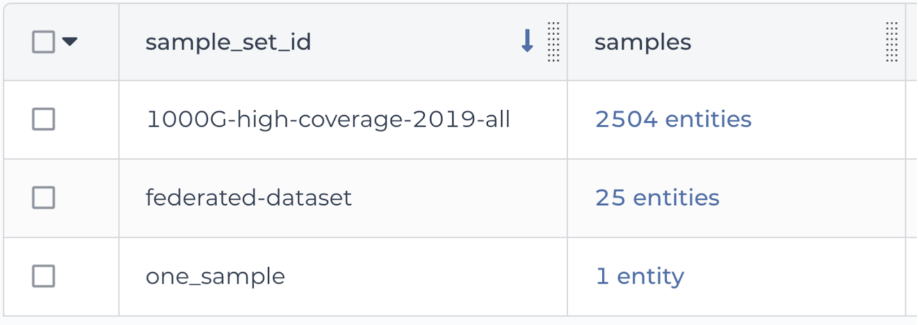


	A	B
1	membership:sample_set_id	sample
2	federated-dataset	SRS000030
3	federated-dataset	SRS000031

...

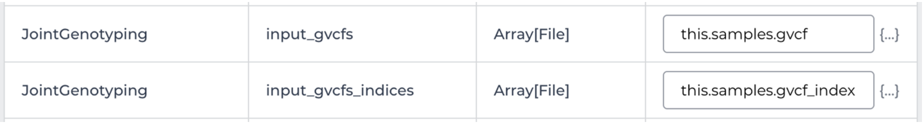
25	federated-dataset	SRS000055
26	federated-dataset	NA12878

Figure 13.15: Updated membership load file sample$_{s}et_{m}embership.tsvassigning25samplestothefederated-datasetsampleset.$



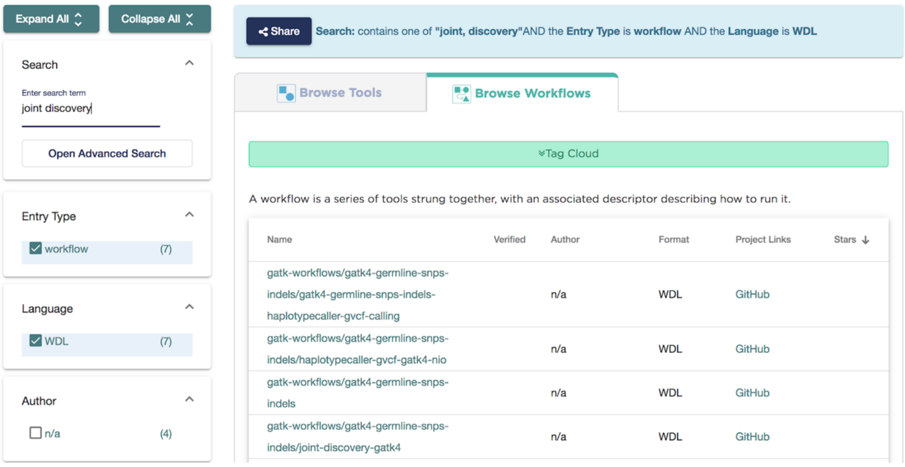
□	sample_set_id	samples
□	1000G-high-coverage-2019-all	2504 entities
□	federated-dataset	25 entities
□	one_sample	1 entity

Figure 13.16: The sample$_{s}ettableshowingthethreesamplesets.$



JointGenotyping	input_gvcfs	Array[File]	this.samples.gvcf {...}
JointGenotyping	input_gvcfs_indices	Array[File]	this.samples.gvcf_index {...}

Figure 13.17: Input configuration details for the input$_{g}vcfsandinput_{g}vcfs_{i}ndicesvariables.$



Search: contains one of "joint, discovery"AND the Entry Type is workflow AND the Language is WDL

Search

Enter search term: joint discovery

Entry Type

☑ workflow (7)

Language

☑ WDL (7)

Author

☐ n/a (4)

A workflow is a series of tools strung together, with an associated descriptor describing how to run it.

Name	Verified	Author	Format	Project Links	Stars
gatk-workflows/gatk4-germline-snps-indels/gatk4-germline-snps-indels-haplotypecaller-gvcf-calling		n/a	WDL	GitHub	
gatk-workflows/gatk4-germline-snps-indels/haplotypecaller-gvcf-gatk4-nio		n/a	WDL	GitHub	
gatk-workflows/gatk4-germline-snps-indels		n/a	WDL	GitHub	
gatk-workflows/gatk4-germline-snps-indels/joint-discovery-gatk4		n/a	WDL	GitHub	

Figure 13.18: Search results for "joint discovery" in Dockstore.

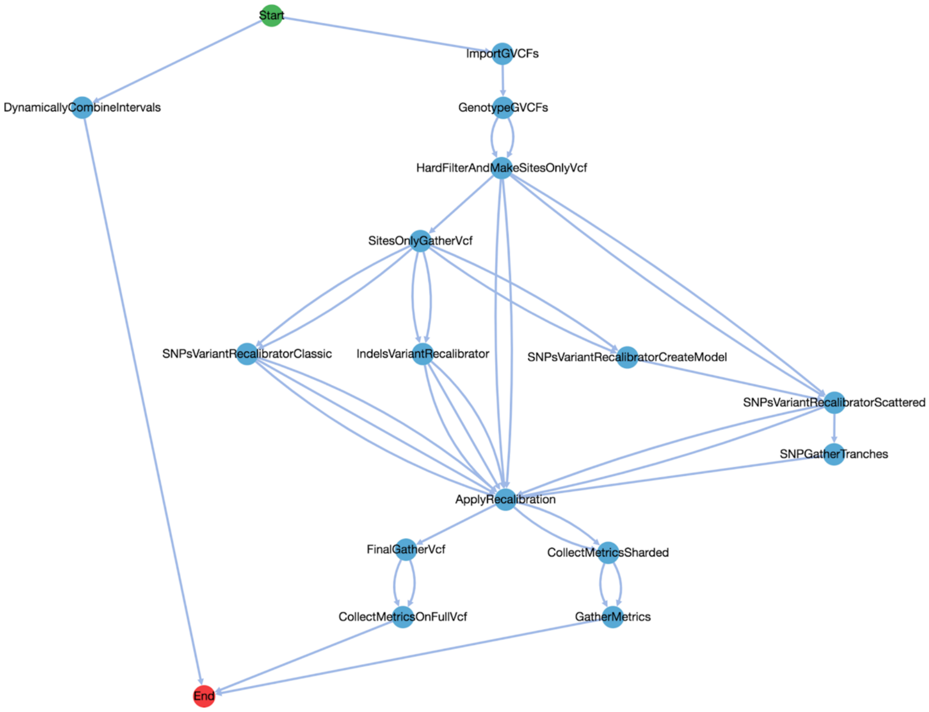

Start
ImportGVCFs
DynamicallyCombineIntervals
GenotypeGVCFs
HardFilterAndMakeSitesOnlyVcf
SitesOnlyGatherVcf
SNPsVariantRecalibratorClassic
IndelsVariantRecalibrator
SNPsVariantRecalibratorCreateModel
SNPsVariantRecalibratorScattered
SNPGatherTranches
ApplyRecalibration
FinalGatherVcf
CollectMetricsSharded
CollectMetricsOnFullVcf
GatherMetrics
End

Figure 13.19: The Joint Discovery workflow provided in the DAG tab in Dockstore.

Chapter 14 Making a Fully Reproducible Paper

Capstone case study on computational reproducibility involving synthetic data creation, GATK, downstream analysis and real biological findings by Dr. Matthieu Miossec et al.

14.1 Overview of the Case Study

14.1.1 Computational Reproducibility and the FAIR Framework

14.1.2 Original Research Study and History of the Case Study

14.1.3 Assessing the Available Information and Key Challenges

14.1.4 Designing a Reproducible Implementation

14.2 Generating a Synthetic Dataset as a Stand-In for the Private Data

14.2.1 Overall Methodology

14.2.2 Retrieving the Variant Data from 1000 Genomes Participants

14.2.3 Creating Fake Exomes Based on Real People

14.2.4 Mutating the Fake Exomes

14.2.5 Generating the Definitive Dataset

14.3 Re-Creating the Data Processing and Analysis Methodology

14.3.1 Mapping and Variant Discovery

14.3.2 Variant Effect Prediction, Prioritization, and Variant Load Analysis

14.3.3 Analytical Performance of the New Implementation

14.4 The Long, Winding Road to FAIRness

14.5 Final Conclusions

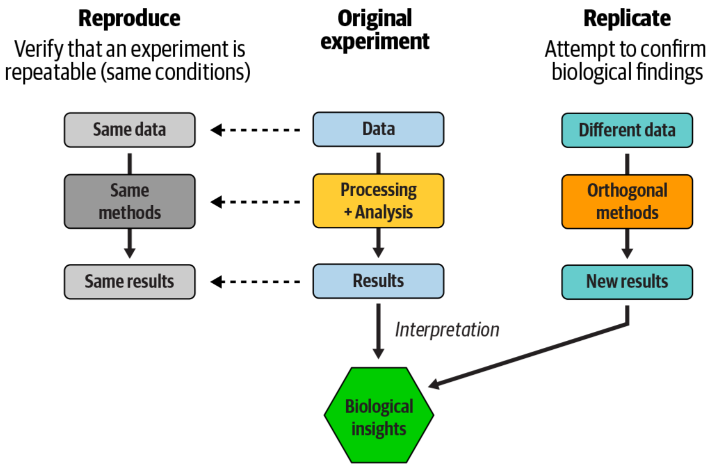


Figure 14.1: Reproducibility of an analysis versus replicability of study findings.

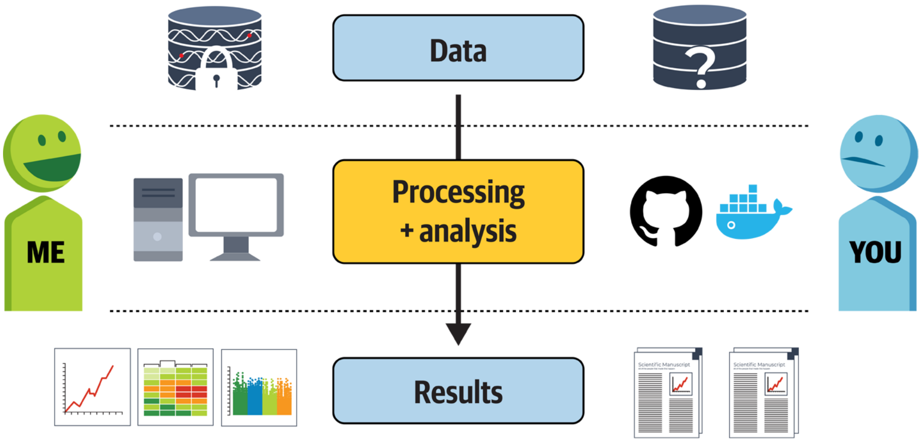


Figure 14.2: Typical asymmetry in the availability of information between author and reader.

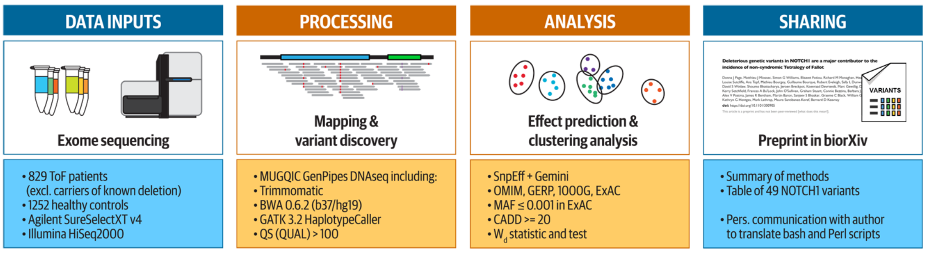

DATA INPUTS
PROCESSING
ANALYSIS
SHARING
Exome sequencing
Mapping & variant discovery
Effect prediction & clustering analysis
Deleterious genetic variants in NOTCH1 are a major contributor to the incidence of non-syndromic Tetralogy of Fallot
VARIANTS
Preprint in biorXiv
• 829 ToF patients (excl. carriers of known deletion)
• 1252 healthy controls
• Agilent SureSelectXT v4
• Illumina HiSeq2000
• MUGQIC GenPipes DNAseq including:
• Trimmomatic
• BWA 0.6.2 (b37/hg19)
• GATK 3.2 HaplotypeCaller
• QS (QUAL) > 100
• SnpEff + Gemini
• OMIM, GERP, 1000G, ExAC
• MAF ≤ 0.001 in ExAC
• CADD >= 20
• W_d statistic and test
• Summary of methods
• Table of 49 NOTCH1 variants
• Pers. communication with author to translate bash and Perl scripts

Figure 14.3: Summary of the information provided in the original preprint of the Tetralogy of Fallot paper.

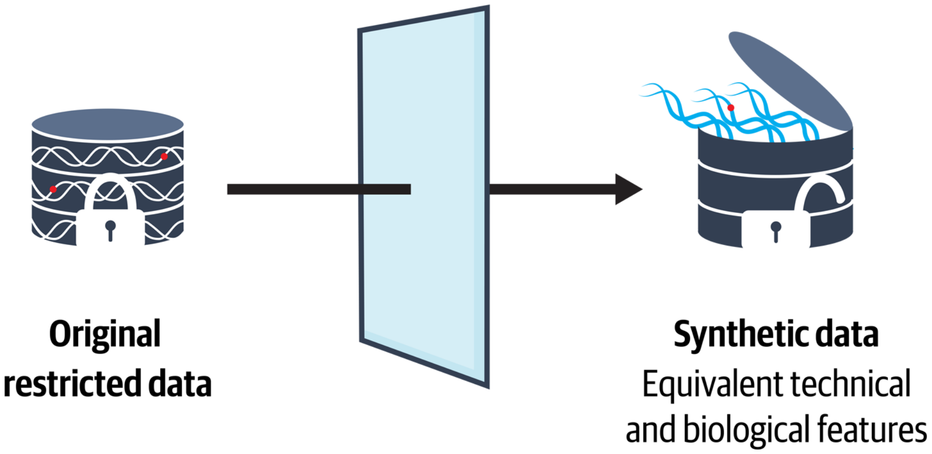
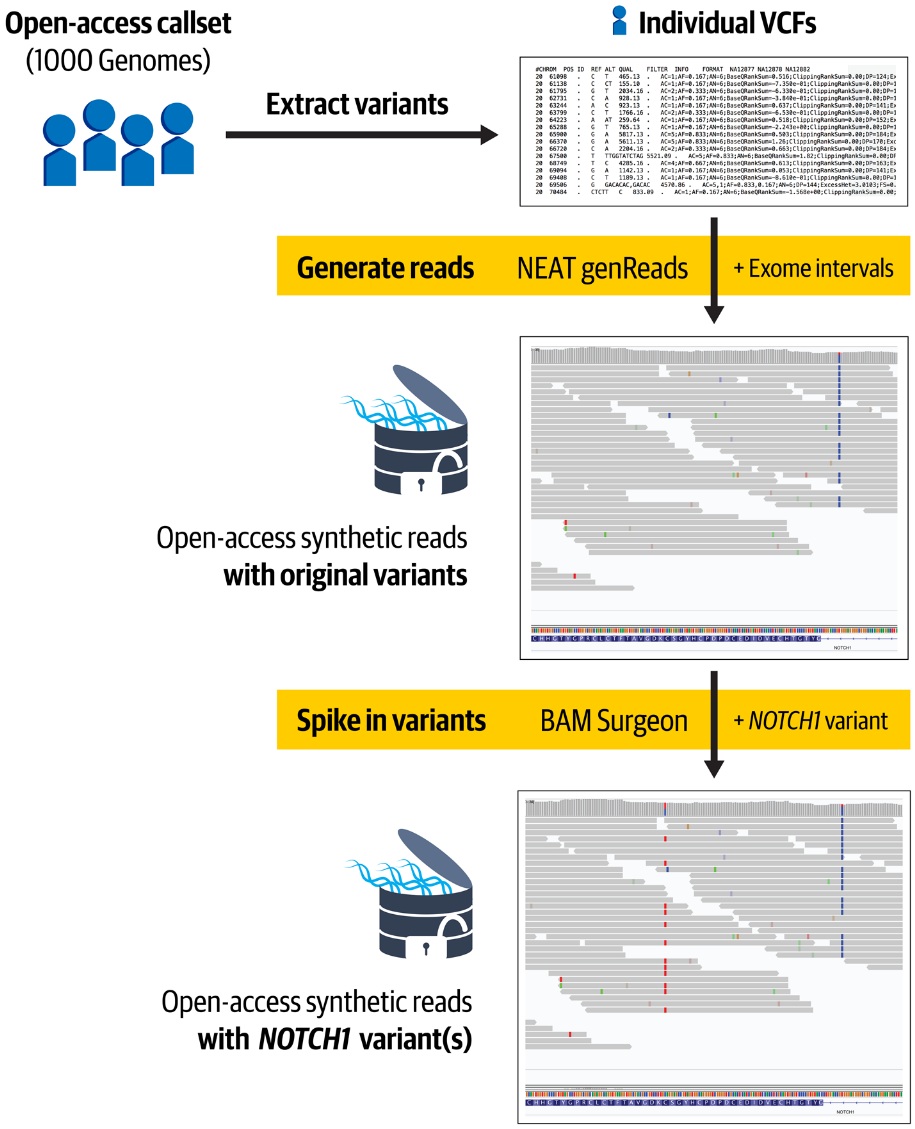


Figure 14.4: Replacing a real dataset that cannot be distributed with a synthetic dataset that mimics the original data's characteristics.



```
#CHROM  POS ID  REF ALT QUAL    FILTER  INFO    FORMAT  NA12877 NA12878 NA12882
20  61098   .   C   T   465.13  .   AC=1;AF=0.167;AN=6;BaseQRankSum=0.516;ClippingRankSum=0.00;DP=124;Ex
20  61138   .   C   CT  155.10  .   AC=1;AF=0.167;AN=6;BaseQRankSum=-7.350e-01;ClippingRankSum=0.00;DP=1
20  61795   .   G   T   2034.16 .   AC=2;AF=0.333;AN=6;BaseQRankSum=-6.330e-01;ClippingRankSum=0.00;DP=1
20  62731   .   C   A   928.13  .   AC=1;AF=0.167;AN=6;BaseQRankSum=-3.840e-01;ClippingRankSum=0.00;DP=1
20  63244   .   A   C   923.13  .   AC=1;AF=0.167;AN=6;BaseQRankSum=0.637;ClippingRankSum=0.00;DP=141;Ex
20  63799   .   C   T   1766.16 .   AC=2;AF=0.333;AN=6;BaseQRankSum=-6.530e-01;ClippingRankSum=0.00;DP=1
20  64223   .   A   AT  259.64  .   AC=1;AF=0.167;AN=6;BaseQRankSum=0.518;ClippingRankSum=0.00;DP=152;Ex
20  65288   .   G   T   765.13  .   AC=1;AF=0.167;AN=6;BaseQRankSum=-2.243e+00;ClippingRankSum=0.00;DP=1
20  65900   .   G   A   5817.13 .   AC=5;AF=0.833;AN=6;BaseQRankSum=0.503;ClippingRankSum=0.00;DP=184;Ex
20  66370   .   G   A   5611.13 .   AC=5;AF=0.833;AN=6;BaseQRankSum=1.26;ClippingRankSum=0.00;DP=170;Exc
20  66720   .   C   A   2204.16 .   AC=2;AF=0.333;AN=6;BaseQRankSum=0.663;ClippingRankSum=0.00;DP=184;Ex
20  67500   .   T   TTGGTATCTAG 5521.09 .   AC=5;AF=0.833;AN=6;BaseQRankSum=1.82;ClippingRankSum=0.00;DF
20  68749   .   T   C   4285.16 .   AC=4;AF=0.667;AN=6;BaseQRankSum=0.613;ClippingRankSum=0.00;DP=163;Ex
20  69094   .   G   A   1142.13 .   AC=1;AF=0.167;AN=6;BaseQRankSum=0.053;ClippingRankSum=0.00;DP=141;Ex
20  69408   .   C   T   1189.13 .   AC=1;AF=0.167;AN=6;BaseQRankSum=-8.610e-01;ClippingRankSum=0.00;DP=1
20  69506   .   G   GACACAC,GACAC   4570.86 .   AC=5,1;AF=0.833,0.167;AN=6;DP=144;ExcessHet=3.0103;FS=0.
20  70484   .   CTCTT   C   833.09  .   AC=1;AF=0.167;AN=6;BaseQRankSum=-1.568e+00;ClippingRankSum=0.00;
```

Figure 14.5: Overview of our implementation for generating appropriate synthetic data.

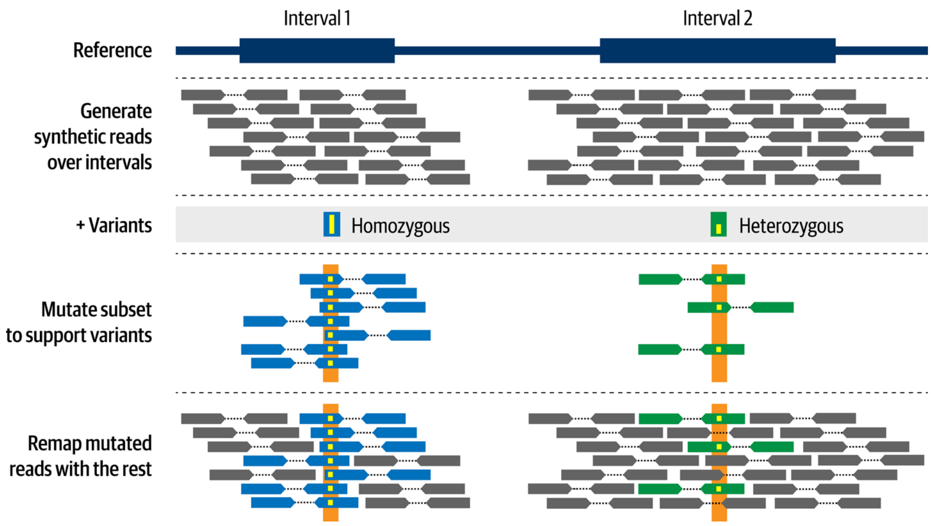

Interval 1
Interval 2
Reference
Generate synthetic reads over intervals
+ Variants
Homozygous
Heterozygous
Mutate subset to support variants
Remap mutated reads with the rest

Figure 14.6: NEAT-genReads creates simulated read data based on a reference genome and list of variants.

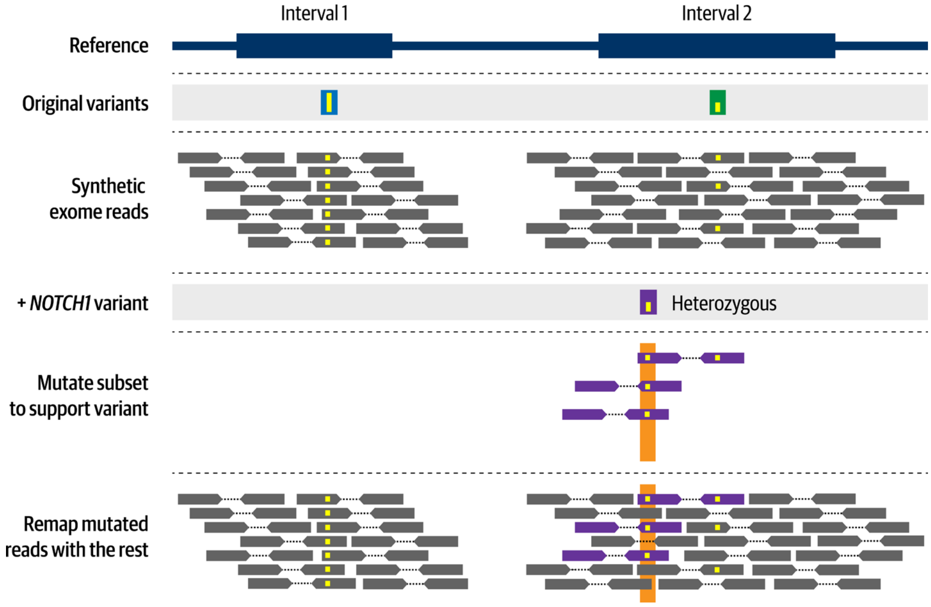


Figure 14.7: BAMSurgeon introduces mutations in read data.

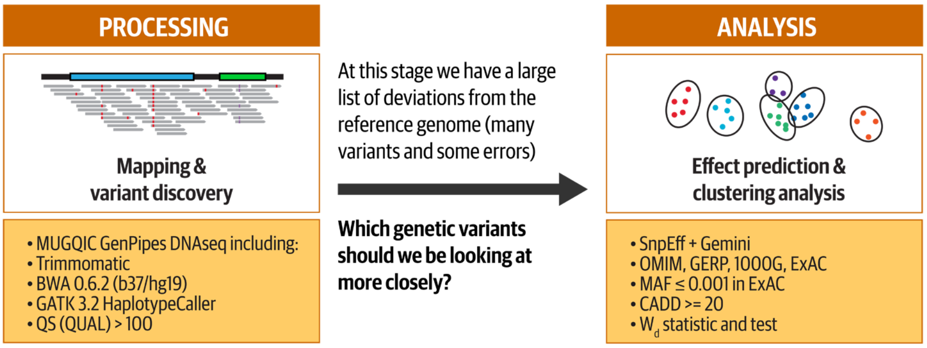

PROCESSING
Mapping & variant discovery
• MUGQIC GenPipes DNAseq including:
• Trimmomatic
• BWA 0.6.2 (b37/hg19)
• GATK 3.2 HaplotypeCaller
• QS (QUAL) > 100
At this stage we have a large list of deviations from the reference genome (many variants and some errors)
Which genetic variants should we be looking at more closely?
ANALYSIS
Effect prediction & clustering analysis
• SnpEff + Gemini
• OMIM, GERP, 1000G, ExAC
• MAF ≤ 0.001 in ExAC
• CADD >= 20
• W_d statistic and test

Figure 14.8: Summary of the two phases of the study: Processing and Analysis.

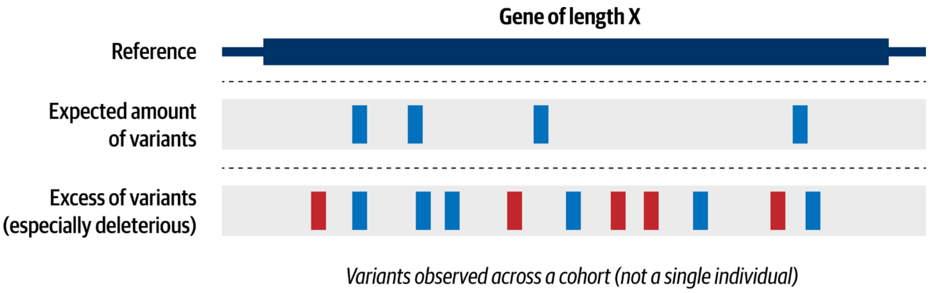

Gene of length X
Reference
Expected amount of variants
Excess of variants (especially deleterious)
Variants observed across a cohort (not a single individual)

Figure 14.9: Comparing variant load in a gene across multiple samples.

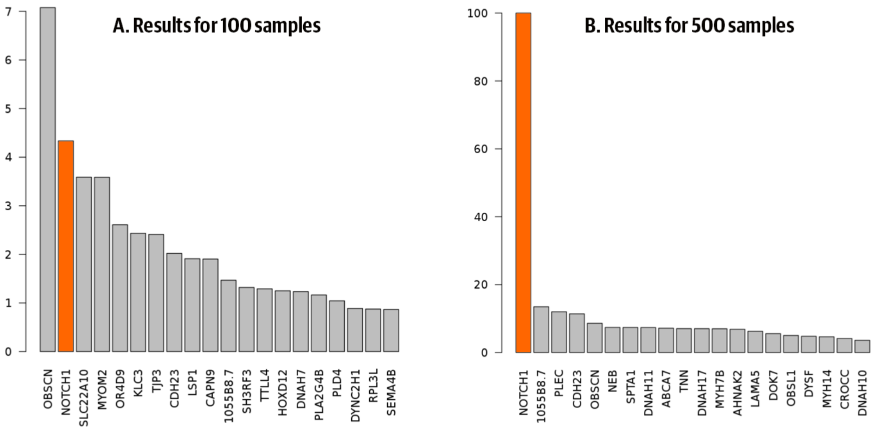


Figure 14.10: Ranking from the clustering test for A) 100-participant set, and B) 500-participant set.

End notes

2020 has been a rough year.

Let's all work together to make 2021 more safe, equitable and enjoyable for all.

Best wishes and don't hesitate to ask for help!