**AWS Bioinformatics**

|  |
| --- |
|  |
| |  | | --- | | **Bioconductor in the cloud**  <https://www.bioconductor.org/help/bioconductor-cloud-ami/> | | Bioconductor community maintained Public AMI image (R Packages, RStudio Server, Git, …)  **Excellent Instructions**  How to step-by-step instructions, starting with AWS Free Tier account. (These instructions will have you configure an AMI which will cost.) However, less than a dollar for hours of fun learning how to work with AWS EC2. |      |  | | --- | | **RStudio Server Amazon Machine Image (AMI)**  AMI specifically targeted at R and RStudio Server  <https://www.louisaslett.com/RStudio_AMI/> | | To use the AMIs described on this page, you simply click your chosen AMI ID  which will take you through to the Amazon web interface and preselect the correct region and AMI.  Simply ensure that your ‘security group’ settings allow incoming HTTP (port 80) traffic and then  copy-and-paste the ‘Public DNS’ for your running instance to a web browser address bar to bring up the login page.  Video 00:04:03  <https://www.louisaslett.com/RStudio_AMI/video_guide.html>  **US East, Virginia ami-0f2290fdad793f863**  <https://signin.aws.amazon.com/signin?redirect_uri=https%3A%2F%2Fconsole.aws.amazon.com%2Fec2%2Fhome%3Fregion%3Dus-east-1%26state%3DhashArgs%2523launchAmi%253Dami-0f2290fdad793f863%26isauthcode%3Dtrue&client_id=arn%3Aaws%3Aiam%3A%3A015428540659%3Auser%2Fec2&forceMobileApp=0&code_challenge=bEkiXYttLxSfqW-GZo7rCoYMLO0XpaFOtXitDHZkEwI&code_challenge_method=SHA-256> | | **Includes:**   * 30GB EBS storage — compact, but enables storage of more sizeable datasets.   + Defaults to fast SSD storage (faster, zero IO costs, only $1 per month in most regions) * Dropbox integration to up/down-load files and data.   + Setup can be completed entirely through RStudio in the web browser by running a single function.   + Selective syncing supported so that large Dropbox accounts don’t sync everything.   + Unlink and relink to new account supported. * Full LaTeX support enabling [R Markdown](http://rmarkdown.rstudio.com/), Sweave and regular document compiles within RStudio. * Java 8 JRE enabling full support for [H2O](https://www.h2o.ai/products/h2o/) and [Spark](https://spark.rstudio.com/). * GDAL dependencies for GIS packages. * [GSL](http://www.gnu.org/software/gsl/) and CURL libraries. * Database support:   + ODBC drivers installed.   + RMySQL package precompiled and installed. * Git support out of the box. * MCMC samplers:   + [Stan](http://mc-stan.org/rstan.html) (RStan) installed and ready to use for Hamiltonian Monte Carlo sampling.   + [JAGS](http://mcmc-jags.sourceforge.net) (and rjags) installed and ready to use for Gibbs sampling.   + [Greta](https://greta-stats.org) supported for GPU MCMC sampling via Tensorflow support. * CUDA and cuDNN.   + Enables immediate use of GPU instances (e.g. p2.\* instances) without any setup.   + [Tensorflow](https://www.tensorflow.org/) and [Keras](https://keras.io/) deep learning libraries can be accelerated using nVidia GPUs on the GPU compute instances, just 3 lines of R code each to setup:   + # Tensorflow   + install.packages("tensorflow")   + library("tensorflow")   + install\_tensorflow(version = "gpu")   + # Keras   + install.packages("keras")   + library("keras")   + install\_keras(tensorflow = "gpu")   You will then be prompted to install Miniconda: say "Yes" to this option.  See the [RStudio Keras page](https://keras.rstudio.com/) and [RStudio Tensorflow page](https://tensorflow.rstudio.com/) for details on using from R.   * + Also preinstalled Magma GPU linear algebra libraries for accelerated matrix decompositions. * Swap space for compiling of large packages on constrained memory instances (such as rugarch). * Arbitrary precision arithmetic and number theory libraries supported out of the box:   + [GMP](http://gmplib.org/)   + [MPFR](http://www.mpfr.org/)   + [FLINT](http://flintlib.org/) * Optimised BLAS for automatically faster matrix operations than base R libraries ([OpenBLAS](http://www.openblas.net/)). |   **AWS EC2 AMI Bioconductor.org Version:**  <https://www.bioconductor.org/help/bioconductor-cloud-ami/#ami_ids>  **Bioconductor.org Docker Image: RStudio Server, Bioconductor Packages, Git** <https://hub.docker.com/r/bioconductor/bioconductor_docker>  **Creating RStudio projects from GitHub Repositories**  <https://www.youtube.com/watch?v=YxZ8J2rqhEM>  **Git with RStudio R Projects**  <https://www.infoworld.com/video/97367/r-tip-how-to-use-git-and-github-with-r-projects>  **RNA-seq workflow: gene-level exploratory analysis and differential expression**  <http://master.bioconductor.org/packages/release/workflows/vignettes/rnaseqGene/inst/doc/rnaseqGene.html>  **Development of a cloud-based Bioinformatics Training Platform**  Briefings in Bioinformatics, Volume 18, Issue 3, May 2017, Pages 537–544  <https://academic.oup.com/bib/article/18/3/537/2453288>   |  | | --- | | **Running R on AWS**  **23 JUL 2015**  [**https://aws.amazon.com/blogs/big-data/running-r-on-aws/**](https://aws.amazon.com/blogs/big-data/running-r-on-aws/)  (October 2017 Update)  [**Aaron Friedman**](https://aws.amazon.com/blogs/big-data/author/ajfriedm/) is a Healthcare and Life Sciences Partner Solutions Architect with AWS.  [**https://aws.amazon.com/blogs/big-data/author/ajfriedm/**](https://aws.amazon.com/blogs/big-data/author/ajfriedm/) | | * Choosing an Amazon Machine Image * Choosing an instance type * Configuring instance details: EC2 user data * Configuring instance details: IAM roles * Configuring a security group * Loading data into your R-based environment on AWS * Storing data in S3 * Configuring Shiny Server * Automating deployment |      |  | | --- | | **Development of a cloud-based Bioinformatics Training Platform**  Briefings in Bioinformatics, Volume 18, Issue 3, May 2017, Pages 537–544  <https://academic.oup.com/bib/article/18/3/537/2453288> | | The 3 day NGS workshop has been organized into seven training modules:  introduction to the command line,  quality control of the NGS data and Alignment,  ChIPSeq, RNASeq, De novo Assembly and Post Workshop.  All the presentations, tools, data sets and tutorials for each training module are accessible and maintained on  GitHub <https://github.com/BPA-CSIRO-Workshops>  BTP has been given a sufficient allocation for running  50 instances with specifications of two CPUs and 8 GB of RAM each.    Specialist bioinformatics software tools for analysis of data are essential for any bioinformatics hands-on workshop. These software tools are packaged and installed into the BTP image on creation. To improve the software tool installation process onto the BTP virtual machine images, we have used a tool packaging system.  Table 2. Current list of analysis tools included and maintained on the Bioinformatics Training Platform. These tools are automatically configured and installed on the BTP images and instances   | **Tool** | **Function** | **Link** | | --- | --- | --- | | AMOS Hawkeye | Genome data visualization | <http://sourceforge.net/projects/amos/> | | BEDTools | Genome data manipulation | <http://bedtools.readthedocs.org/en/latest/> | | BLAT | Sequence location lookup in the genome | <https://genome.ucsc.edu/FAQ/FAQblat.html> | | Bowtie | Read Alignment | <http://bowtie-bio.sourceforge.net/index.shtml> | | CummeRbund | RNA-Seq analysis using R | <http://bioconductor.org/packages/release/bioc/html/cummeRbund.html> | | Cufflinks | RNA-Seq analysis | <http://cole-trapnell-lab.github.io/cufflinks/> | | DESeq2 | Differential gene expression analysis using R | <https://bioconductor.org/packages/release/bioc/html/DESeq2.html> | | edgeR | Empirical gene expression analysis using R | <https://bioconductor.org/packages/release/bioc/html/edgeR.html> | | FastQC | FastQC | <http://www.bioinformatics.babraham.ac.uk/projects/fastqc/> | | FASTX | Toolkit for short reads preprocessing | <http://hannonlab.cshl.edu/fastx_toolkit/> | | IGV | Interactive exploration of genomic data | <https://www.broadinstitute.org/igv/> | | igvtools | For preprocessing data before loading to IGV | <https://www.broadinstitute.org/igv/igvtools> | | MACS | ChIP-Seq analysis | <http://liulab.dfci.harvard.edu/MACS/> | | MUMmer | Rapid genome alignment, a dependency for AMOS | <http://mummer.sourceforge.net/> | | PeakAnalyzer | Multi-peak data analysis | <http://www.bioinformatics.org/peakanalyzer/wiki/> | | Picard | Sequence data analysis | <http://broadinstitute.github.io/picard/> | | SAMtools | For manipulating alignments in the SAM format | <http://samtools.sourceforge.net/> | | Skewer | Adapter trimmer for paired-end reads | <https://github.com/relipmoc/skewer> | |      |  | | --- | | **NGS Graph Generator on AWS Free Tier**  <https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&cad=rja&uact=8&ved=2ahUKEwjdn5_JvqrtAhUP2VkKHWdfAM0QFjABegQIBBAC&url=https%3A%2F%2Fwww.biorxiv.org%2Fhighwire%2Ffilestream%2F123738%2Ffield_highwire_adjunct_files%2F0%2F409573-1.docx&usg=AOvVaw1ctSQUOrm8NEQy_lD8FV1p> | | <https://github.com/systems-immunology-roslin-institute/ngs-graph-generator>  Choose an Instance Type page  This page lists all of the available instances type varying combinations of CPU, memory, storage and capacity.  In order to use this pipeline, it needs a small memory and low storage,  thus a user can select for t2.micro (free tier eligible) (1), then click ‘Review and Launch’ button (2).  BioLayout Express3D or Miru | |

|  |
| --- |
| **AWS EC2 Storage** |
| **Practical guide for managing large-scale human genome data in research** Oct 2020  <https://www.nature.com/articles/s10038-020-00862-1>  What kind of large-scale human genome projects are underway and available from data sharing?  How to store and analyze human genome data efficiently?  What kind of computational platforms are used to store and analyze human genome data?  How to maintain reproducibility, portability, and scalability in genome data analysis, and why is it important?  **Bringing data to life**  **Data management for the biomolecular sciences**  <https://www.ebi.ac.uk/training-beta/online/courses/bringing-data-life-data-management-biomolecular-sciences/what-is-data-management/> |

|  |
| --- |
| **AWS Data** |
| What type of data transfer is free for Amazon s3?  Pick the right AWS region for your S3 bucket.  The main benefit of having S3 and EC2 in the same region is the performance and lower transfer cost.   Data transfer is free  between EC2 and S3 in the same region.  Downloading file  from another AWS region will cost $0.02/GB.  May 18, 2019 |
| **New AWS public datasets available from the National Cancer Institute, Massachusetts Institute of Technology, Amazon, the National Renewable Energy Laboratory, and others**  Posted On: Jul 15, 2020  <https://aws.amazon.com/about-aws/whats-new/2020/07/new-aws-public-datasets/>  Twenty-three new or updated Amazon Web Services (AWS) public datasets from the National Center for Bioinformatics, Johns Hopkins University, University of Texas at Southwestern, National Oceanic and Atmospheric Administration (NOAA), the National Cancer Institute, National Herbarium of New South Wales, and others are now available in the following categories:  COVID-19 response:   * [COVID-19 Molecular Structure and Therapeutics Hub](https://registry.opendata.aws/molssi-covid19-hub/) from the Molecular Sciences Software Institute * [COVID-19 Genome Sequence Dataset](https://registry.opendata.aws/ncbi-covid-19) from the National Center for Biotechnology Information Life sciences: * [Cloud Genomic Indexes](https://registry.opendata.aws/jhu-indexes/) from Johns Hopkins University and the University of Texas at Southwestern * [Refgenie Genomic Assets](https://registry.opendata.aws/refgenie/) from University of Virginia * [Gabriella Miller Kids First Pediatric Research Program](https://registry.opendata.aws/kids-first/) from the National Cancer Institute * [The Cancer Genome Atlas](https://registry.opendata.aws/tcga/) from the National Cancer Institute * [Basic Local Alignment Sequence Tool (BLAST) Databases](https://registry.opendata.aws/ncbi-blast-databases/) from the National Library of Medicine * [National Herbarium of New South Wales](https://registry.opendata.aws/nsw-herbarium/) from the Royal Botanic Gardens and Domain Trust |
| **Access to Genomic Data for free**  In **S3**: <http://s3.amazonaws.com/1000genomes>  **IGSR: The International Genome Sample Resource**  Open human variation data  <https://www.internationalgenome.org/home>  **1000 Genomes Project FAQ**  If you have any other questions you can’t find the answer to please email [info@1000genomes.org](mailto:info@1000genomes.org)  <https://www.internationalgenome.org/faq> |

|  |
| --- |
| **Google Searches:** |
| |  | | --- | | **Google:**   <https://www.google.com/search?as_q=AWS+AMI+EC2+S3+&as_epq=open+data&as_oq=bioinformatics+bioconductor&as_eq=&as_nlo=&as_nhi=&lr=&cr=&as_qdr=all&as_sitesearch=&as_occt=any&safe=images&as_filetype=&tbs=> | |  |  |  | | --- | | **Google:**   <https://www.google.com/search?as_q=AWS+AMI+bioinformatics+&as_epq=&as_oq=%22instance+store-backed%22+%22EBS-backed%22&as_eq=&as_nlo=&as_nhi=&lr=&cr=&as_qdr=all&as_sitesearch=&as_occt=any&safe=images&as_filetype=&tbs=> | |  |  |  | | --- | | **Google: aws genomics tutorial**  844,000 results  <https://www.google.com/search?as_q=aws+genomics+tutorial&as_epq=&as_oq=&as_eq=&as_nlo=&as_nhi=&lr=&cr=&as_qdr=all&as_sitesearch=&as_occt=any&safe=images&as_filetype=&tbs=> **Google: aws genomics tutorial**  **Videos**  22,900 results  <https://www.google.com/search?q=aws+genomics+tutorial&lr=&source=lnms&tbm=vid&sa=X&ved=2ahUKEwjOt_zvpvTtAhWJuVkKHXHoDLwQ_AUoAnoECAYQBA> | | **Big Data Analysis in the Cloud**  Oct 24, 2016  <https://www.youtube.com/watch?v=LdPj5UKWtx0>  **How to Access SRA Data with Amazon Web Services**  <https://www.youtube.com/watch?v=rjjrHnZfymU&list=PLH-TjWpFfWrt5MNqU7Jvsk73QefO3ADwD>  **Accelerate Genomics Discovery with SRA in the Cloud**  <https://www.youtube.com/watch?v=PoseTXa8LU8&list=PLH-TjWpFfWrt5MNqU7Jvsk73QefO3ADwD&index=3>  **Accelerating Life Sciences with HPC on AWS - AWS Online Tech Talks**  <https://www.youtube.com/watch?v=fscMSO_vQ7o>  <https://www.youtube.com/watch?v=YfCgK1bmCjw> |  |  | | --- | | **Google:**   <https://www.google.com/search?as_q=AWS+Bioinformatics+NGS+&as_epq=&as_oq=BMC+bioRxio+&as_eq=&as_nlo=&as_nhi=&lr=&cr=&as_qdr=all&as_sitesearch=&as_occt=any&safe=images&as_filetype=&tbs=> | |  | |

|  |
| --- |
| **Google: aws genomics guide**  <https://www.google.com/search?q=aws+genomics+guide&lr=&as_qdr=all&ei=6LbrX8LrOMez5gKb6aXYDQ&start=20&sa=N&ved=2ahUKEwjC_-69p_TtAhXHmVkKHZt0Cds4ChDy0wN6BAgGEDg&biw=1094&bih=569> |
| |  | | --- | | **AWS genomics Publications Non-Profit** | | **Biostars**  **Tutorial: Introduction to AWS Cloud Computing**  <https://www.biostars.org/p/137477/>  **Broad Institute.org**  **Genome STRiP AWS cluster as described in Running Genome STRiP on the Amazon cloud**  <http://software.broadinstitute.org/software/genomestrip/cookbook-genotyping-novel-site-1000-genomes-phase-1-using-aws>  **IGSR: International Genome Sample Resource**  **open human variation data**  **Using 1000 Genomes Data in the Amazon Web Service Cloud**  <https://www.internationalgenome.org/using-1000-genomes-data-amazon-web-service-cloud/>  **Bioinformatics UC Davis**  **Introduction to Amazon Web Services EC2 - lecture** <https://bioshare.bioinformatics.ucdavis.edu/bioshare/download/kveirzo6fvkl2nb/build/Monday-AWS-Intro.html>  **Practical guide for managing large-scale human genome data in research** Oct 2020  <https://www.nature.com/articles/s10038-020-00862-1> |  |  | | --- | | **AWS genomics Publications Amazon** | | **Building High-Throughput Genomics Batch Workflows on AWS: Introduction (Part 1 of 4)** MAY 2017  <https://aws.amazon.com/blogs/compute/building-high-throughput-genomics-batch-workflows-on-aws-introduction-part-1-of-4/>  **Genomics Workflows on AWS**  <https://docs.opendata.aws/genomics-workflows/>  **GitHub: aws-samples / aws-genomics-workflows**  <https://github.com/aws-samples/aws-genomics-workflows>  **DRAGEN reanalysis of the 1000 Genomes Dataset now available on the Registry of Open Data** OCT 2020  <https://aws.amazon.com/blogs/industries/dragen-reanalysis-of-the-1000-genomes-dataset-now-available-on-the-registry-of-open-data/>  The release of a comprehensive reanalysis of 3,202 deeply-sequenced samples from the 1000 Genomes Project(1kGP)  using the Illumina DRAGEN (Dynamic Read Analysis for GENomics) Bio-IT platform.  This seminal dataset will be freely available for researchers across the world to use as part of their genomics analysis.  The alignment files small variant, copy-number, structural, and short-tandem repeat variant calls  are now available at no cost through the Registry of Open Data on AWS.  **1000 Genomes Project and AWS**  <https://docs.opendata.aws/1000genomes/readme.html>  The 1000 Genomes Project is an international collaboration which has established the most detailed catalogue of human genetic variation,  including SNPs, structural variants, and their haplotype context. The final phase of the project sequenced more than 2500 individuals  from 26 different populations around the world and produced an integrated set of phased haplotypes with more than 80 million variants  for these individuals.  The Amazon mirror contains the complete data set from the project and the data can be found in the  s3://1000genomes bucket in the us-east-1 AWS region.  For more information <http://www.1000genomes.org>    **AWS Labs**  <https://github.com/awslabs>  **AWS Labs Machine Learning**  <https://github.com/awslabs/dgl-ke> | |

|  |
| --- |
| **Amazon EC2**  Amazon Elastic Compute Cloud  Secure and resizable compute capacity to support virtually any workload  <https://aws.amazon.com/ec2/>  Is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. Amazon EC2’s simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon’s proven computing environment.  Amazon EC2 offers the broadest and deepest compute platform with choice of processor, storage, networking, operating system, and purchase model. We offer the fastest processors in the cloud and we are the only cloud with 400 Gbps ethernet networking. |
| **AWS Free Tier**  <https://aws.amazon.com/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=tier%2312monthsfree&awsf.Free%20Tier%20Categories=categories%23compute%7Ccategories%23analytics>  Gain free, hands-on experience with the AWS platform, products, and services  Explore more than 85 products and start building on AWS using the free tier. Three different types of free offers are available depending on the product used. See below for details on each product. |
| **AWS Pricing Calculator**  <https://calculator.aws/#/addService> |
| **Amazon Elastic Block Store (EBS)**  <https://aws.amazon.com/ebs/>  allows you to create persistent block storage volumes and attach them to Amazon EC2 instances.  Amazon Elastic Block Store (EBS) is an easy to use, high performance block storage service designed for use with Amazon Elastic Compute Cloud (EC2) for both throughput and transaction intensive workloads at any scale. A broad range of workloads, such as relational  and non-relational databases, enterprise applications, containerized applications, big data analytics engines, file systems, and media workflows are widely deployed on Amazon EBS.  You can choose from six different volume types to balance optimal price and performance. You can achieve single digit-millisecond latency for high performance database workloads such as SAP HANA or gigabyte per second throughput for large, sequential workloads such  as Hadoop. You can change volume types, tune performance, or increase volume size without disrupting your critical applications, so you have cost-effective storage when you need it.  Designed for mission-critical systems, EBS volumes are replicated within an Availability Zone (AZ) and can easily scale to petabytes of data. Also, you can use EBS Snapshots with automated lifecycle policies to back up your volumes in Amazon S3, while ensuring  geographic protection of your data and business continuity. |
| **Introducing new Amazon EBS general purpose volumes, gp3** Posted On: Dec 1, 2020 Today AWS announced the availability of gp3, the next-generation general purpose SSD volumes for Amazon Elastic Block Store (Amazon EBS) that enable customers to  provision performance independent of storage capacity and provides up to 20% lower price-point per GB than existing gp2 volumes.  With gp3 volumes, customers can scale IOPS (input/output operations per second) and throughput without needing to provision additional block storage capacity, and pay only for the resources they need.  General purpose SSD volumes make it easy and cost effective for customers to meet the IOPS and throughput requirements for transaction-intensive workloads, such as virtual desktops, test and development environments, low-latency interactive applications, and boot volumes. With existing general-purpose SSD (gp2) volumes, performance is tied to storage capacity, enabling customers to get higher IOPS and throughput for their applications by provisioning a larger storage volume size. But customers want to scale performance and throughput without paying for storage that they don’t need.  Next generation gp3 volumes offer the ability to independently provision IOPS and throughput, separate from storage capacity. This enables customers to scale performance for transaction-intensive workloads without needing to provision more capacity, so they only pay for the resources they need. The new gp3 volumes also deliver a baseline performance of 3,000 IOPS and 125MB/s at any volume size. For use cases, where your application needs more performance than the baseline, customers can scale up to 16,000 IOPS and 1,000 MB/s for an additional fee. This makes the new gp3 volumes ideal for a wide variety of applications that require high performance at low cost, including MySQL, Cassandra, virtual desktops, and Hadoop analytics clusters.  Customers can easily migrate gp2 volumes to gp3 volumes using Elastic Volumes, which is an existing feature of Amazon EBS.  Elastic Volumes allow customers to modify the volume type, IOPS, or throughput of their existing EBS volumes without interrupting their Amazon EC2 instances. gp3 volumes are available in all AWS commercial and gov cloud regions. For more information, please see the gp3 announcement on the AWS News blog, and documentation. |
| **Volume Gateway**  Hybrid cloud block storage with local caching  <https://aws.amazon.com/storagegateway/volume/>  Volume Gateway presents cloud-backed iSCSI block storage volumes to your on-premises applications. Volume Gateway stores and manages on-premises data in Amazon S3 on your behalf and operates in either cache mode or stored mode. In the cached Volume Gateway mode, your primary data is stored in Amazon S3, while retaining your frequently accessed data locally in the cache for low latency access. In the stored Volume Gateway mode, your primary data is stored locally and your entire dataset is available for low latency access on  premises while also asynchronously getting backed up to Amazon S3. In either mode, you can take point-in-time copies of your volumes using AWS Backup, which are stored in AWS as Amazon EBS snapshots. Using Amazon EBS Snapshots enables you to make space-  efficient versioned copies of your volumes for data protection, recovery, migration, and various other copy data needs.  **Amazon Simple Storage Service (S3)**  Object storage built to store and retrieve any amount of data from anywhere  <https://aws.amazon.com/s3/>  (Amazon S3) is storage for the internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web.  **Amazon ElastiCache**  offers fully managed Redis and Memcached. Seamlessly deploy, run, and scale popular open source compatible in-memory data stores. Build data-intensive apps or improve the performance of your existing apps by retrieving data from high throughput and low latency in-memory data stores.  **AWS Storage Gateway**  <https://aws.amazon.com/storagegateway/>  AWS Storage Gateway is a hybrid storage service that enables your on-premises applications to seamlessly use AWS cloud storage. You can use the service for backup and archiving, disaster recovery, cloud data processing, storage tiering, and migration.  **Amazon Elastic Container Registry**  (ECR) is a fully-managed Docker container registry that makes it easy for developers to store, manage, and deploy Docker container images. Amazon ECR eliminates the need to operate your own container repositories or worry about scaling the underlying infrastructure.  With Amazon ECR, there are no upfront fees or commitments. You pay only for the amount of data you store in your repositories and data transferred to the Internet.  **Amazon RDS for PostgreSQL**  Amazon RDS makes it easy to set up, operate, and scale PostgreSQL deployments in the cloud. With Amazon RDS, you can deploy scalable PostgreSQL deployments in minutes with cost-efficient and resizable hardware capacity. |

|  |
| --- |
|  |
| |  | | --- | | **Start AWS Computational Genomics Image**  Video  Aug 25, 2018  <https://www.youtube.com/watch?v=AiSvnLamE5Y> | |  |  |  | | --- | | **Exploratory data analysis of genomic datasets using ADAM and Mango with Apache Spark on Amazon EMR**  13 JUL 2018  Amazon EMR, AWS Big Data  <https://aws.amazon.com/blogs/big-data/exploratory-data-analysis-of-genomic-datasets-using-adam-and-mango-with-apache-spark-on-amazon-emr/> | | [**ADAM**](https://github.com/bigdatagenomics/adam)  <https://github.com/bigdatagenomics/adam>  and  [**Mango**](https://github.com/bigdatagenomics/mango)  <https://github.com/bigdatagenomics/mango>  provide a unified environment for processing, filtering, and visualizing large genomic datasets on Apache Spark. ADAM allows users to programmatically load, process, and select raw genomic and variation data using SparkSQL, an SQL interface for aggregating and selecting data in Apache Spark. Mango supports visualization of both raw and aggregated genomic data in a Jupyter notebook environment, allowing users to draw conclusions from large datasets at multiple resolutions. This combined power of ADAM and Mango allows users to load, query and explore datasets in a unified environment, allowing users to interactively explore genomic data at a scale previously impossible using single node bioinformatics tools. **Configuring ADAM and Mango on Amazon EMR** **Loading data from the 1000 Genomes Project** Now that we have a working environment, lets use ADAM and Mango to discover interesting variants in the child from the genome sequencing data of a trio (data from a mother, father, and child). These data are available from the  [1000 Genomes Project AWS Public Dataset](https://aws.amazon.com/1000genomes/).  <https://registry.opendata.aws/1000-genomes/>  **Resources on AWS**  Description  <http://www.internationalgenome.org/formats>  Resource type  S3 Bucket  Amazon Resource Name (ARN)  arn:aws:s3:::1000genomes  AWS Region  us-east-1  AWS CLI Access (No AWS account required)  aws s3  # Import ADAM modules  # Import Mango modules  # Import Spark modules  # Create ADAM Context  **Variant analysis with Spark SQL** # repartition genotypes to balance the load across memory  # cache genotypes and show the schema  # trio IDs  # Filter by individuals in the trio  # Add ReferenceRegion column and group by referenceRegion  # Register df with Spark SQL  # filter by alleles. This is a list of variant names that have an alternate allele for the child  # get parent records and filter by only REF locations for variant names that were found in the child with an ALT  # collect snp names as a list  **Working with alignment data** # load in NA19685 exome from s3a  # cache child RDD and count records  # takes about 2 minutes, on 4 c3.4xlarge worker nodes  # Count reads in the child  # Output should be 95634679  **Quality control of alignment data** # Calculate read coverage  # Takes 2-3 minutes  # Output should be 51252612  # Calculate coverage distribution  # You can check the progress in the SparkUI by navigating to  # :8088 and clicking on the currently running Spark application.      **Viewing sites with missense variants in the proband** # missense variant at GNA13: 63052580-63052581 (SNP rs201316886)  # define alignment summary for child reads  # Takes about 2 minutes to collect data from workers    # define alignment summary for parent reads  # view missense variant at GNA13: 63052580-63052581 in parent 1   **Summary** To summarize, this post demonstrated how to set up and run ADAM and Mango in Amazon EMR. We demonstrated how to use these tools in an interactive notebook environment to explore the 1000 Genomes dataset, a publicly available dataset on Amazon S3.  We used these tools inspect 1000 Genomes data quality, query for interesting variants in the genome, and validate results through the visualization of raw data. |  |  | | --- | | **Orchestrating analytics jobs by running Amazon EMR Notebooks programmatically**  23 NOV 2020 | Amazon EMR, AWS Big Data  <https://aws.amazon.com/blogs/big-data/orchestrating-analytics-jobs-by-running-amazon-emr-notebooks-programmatically/> | | [Amazon EMR](http://aws.amazon.com/emr) is a big data service offered by AWS to run Apache Spark and other open-source applications on AWS in a cost-effective manner. [Amazon EMR Notebooks](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-managed-notebooks.html) is a managed environment based on [Jupyter Notebook](https://en.wikipedia.org/wiki/Project_Jupyter#Jupyter_Notebook) that allows data scientists, analysts, and developers to prepare and visualize data, collaborate with peers, build applications, and perform interactive analysis using EMR clusters.  EMR notebook APIs are available on Amazon EMR release version 5.18.0 or later and can be used to run EMR notebooks via a script or command line. The ability to start, stop, list, and describe EMR notebook runs without the Amazon EMR console enables you to programmatically control running an EMR notebook. Using a parameterized notebook cell allows you to pass different parameter values to a notebook without having to create a copy of the notebook for each new set of parameter values. With this feature, you can schedule running EMR notebooks with cron scripts, chain multiple EMR notebooks, and use orchestration services such as [AWS Step Functions](https://aws.amazon.com/step-functions/) or Apache Airflow to build pipelines. If you want to use EMR notebooks in a non-interactive manner, this enables you to run ETL workloads, especially in production.  In this post, we show how to orchestrate analytics jobs by running EMR Notebooks programmatically with the following two use cases:   * Scheduling an EMR notebook run via crontab and the [AWS Command Line Interface](http://aws.amazon.com/cli) (AWS CLI) * Chaining your notebooks with Step Functions triggered by [Amazon CloudWatch Events](https://docs.aws.amazon.com/AmazonCloudWatch/latest/events/WhatIsCloudWatchEvents.html)   For our data source, we use the open-source, real-time COVID-19 US daily case reports provided by Johns Hopkins University CSSE in the following [GitHub repo](https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_daily_reports_us). Prerequisites Before getting started, you must have the following prerequisites:   * An AWS account that provides access to the following AWS services at least:   + [AWS CloudFormation](http://aws.amazon.com/cloudformation)   + Amazon CloudWatch   + [Amazon Elastic Compute Cloud](http://aws.amazon.com/ec2) (Amazon EC2)   + Amazon EMR   + [Amazon EventBridge](https://aws.amazon.com/eventbridge/)   + [AWS Identity and Access Management](http://aws.amazon.com/iam) (IAM)   + [AWS Lambda](http://aws.amazon.com/lambda)   + [Amazon Simple Storage Service](http://aws.amazon.com/s3) (Amazon S3)   + AWS Step Functions * AWS CLI Version 1.18.128 or later installed on your work station. * [Jupyter](https://jupyter.org/install) installed on your work station (this is used for the output visualization part for this post only). * An EMR cluster running Amazon EMR release 5.18.0 or later, with Hadoop, Spark, and Livy installed. Record the value of the cluster ID (for example, <j-\*\*\*\*\*\*\*\*\*\*\*\*\*>); you use this for the examples later. * An EMR notebook created on the Amazon EMR console, using the following two input notebook files:   + [demo\_pyspark.pynb](https://aws-bigdata-blog.s3.amazonaws.com/artifacts/aws-blog-runnable-notebook/demo/notebook/demo_pyspark.ipynb) – Used for both use cases in this post.   + [trailing\_N day.ipynb](https://aws-bigdata-blog.s3.amazonaws.com/artifacts/aws-blog-runnable-notebook/demo/notebook/trailing_N_day.ipynb) – Used for the second use case.   Record the notebook ID (for example, <e-\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*>); you use this later for our examples later. Organize the notebook files in the Jupyter UI as follows:   * /demo\_pyspark.ipynb * /experiment/trailing\_N\_day.ipynb   See [Creating a Notebook](https://docs.aws.amazon.com/emr/latest/ManagementGuide/emr-managed-notebooks-create.html) for more information on how to create an EMR notebook. |  |  | | --- | | **Interactive Analysis of Genomic Datasets Using Amazon Athena**  by Aaron Friedman, PhD | on 07 DEC 2016 | in Amazon Athena, AWS Big Data  [**https://aws.amazon.com/blogs/big-data/interactive-analysis-of-genomic-datasets-using-amazon-athena/**](https://aws.amazon.com/blogs/big-data/interactive-analysis-of-genomic-datasets-using-amazon-athena/) | | Aaron Friedman is a Healthcare and Life Sciences Solutions Architect with Amazon Web Services  The genomics industry is in the midst of a data explosion. Due to the rapid drop in the cost to sequence genomes, genomics is now central to many medical advances. When your genome is sequenced and analyzed, raw sequencing files are processed in a multi-step workflow to identify where your genome differs from a standard reference. Your variations are stored in a Variant Call Format (VCF) file, which is then combined with other individuals to enable population-scale analyses. Many of these datasets are publicly available, and an increasing number are hosted on AWS as part of our [Open Data](https://aws.amazon.com/public-data-sets/) project.  To mine genomic data for new discoveries, researchers in both industry and academia build complex models to analyze populations at scale. When building models, they first explore the datasets-of-interest to understand what questions the data might answer. In this step, interactivity is key, as it allows them to move easily from one question to the next.  Recently, we launched [Amazon Athena](https://aws.amazon.com/athena/) as an interactive query service to analyze data on [Amazon S3](https://aws.amazon.com/s3/). With Amazon Athena there are no clusters to manage and tune, no infrastructure to setup or manage, and customers pay only for the queries they run. Athena is able to query many file types straight from S3. This flexibility gives you the ability to interact easily with your datasets, whether they are in a raw text format (CSV/JSON) or specialized formats (e.g. Parquet). By being able to flexibly query different types of data sources, researchers can more rapidly progress through the data exploration phase for discovery. Additionally, researchers don’t have to know nuances of managing and running a big data system. This makes Athena an excellent complement to data warehousing on [Amazon Redshift](https://aws.amazon.com/redshift/) and big data analytics on [Amazon EMR](https://aws.amazon.com/emr/).   In this post, I discuss how to prepare genomic data for analysis with Amazon Athena as well as demonstrating how Athena is well-adapted to address common genomics query paradigms.  I use the [Thousand Genomes dataset](https://aws.amazon.com/1000genomes/) hosted on Amazon S3, a seminal genomics study, to demonstrate these approaches. All code that is used as part of this post is available in our GitHub [repository](https://github.com/awslabs/aws-big-data-blog/tree/master/aws-blog-athena-genomics/).  Although this post is focused on genomic analysis, similar approaches can be applied to any discipline where large-scale, interactive analysis is required.   Select, aggregate, annotate query pattern in genomics Genomics researchers may ask different questions of their dataset, such as:   * What variations in a genome may increase the risk of developing disease? * What positions in the genome have abnormal levels of variation, suggesting issues in quality of sequencing or errors in the genomic reference? * What variations in a genome influence how an individual may respond to a specific drug treatment? * Does a group of individuals contain a higher frequency of a genomic variant known to alter response to a drug relative to the general population?   All these questions, and more, can be generalized under a common query pattern I like to call “Select, Aggregate, Annotate”. Some of our genomics customers, such as Human Longevity, Inc., routinely use this query pattern [in their work](https://www.youtube.com/watch?v=CGbWEkszAlQ).  In each of the above queries, you execute the following steps:  **SELECT:** Specify the cohort of individuals meeting certain criteria (disease, drug response, age, BMI, entire population, etc.).  **AGGREGATE:** Generate summary statistics of genomic variants across the cohort that you selected.  **ANNOTATE:** Assign meaning to each of the variants by joining on known information about each variant. Dataset preparation Properly organizing your dataset is one of the most critical decisions for enabling fast, interactive analyses. Based on the query pattern I just described, the table representing your population needs to have the following information:   * A unique sample ID corresponding to each sample in your population * Information about each variant, specifically its location in the genome as well as the specific deviation from the reference * Information about how many times in a sample a variant occurs (0, 1, or 2 times) as well as if there are multiple variants in the same site. This is known as a genotype.   The extract, transform, load (ETL) process to generate the appropriate data representation has two main steps. First, you use ADAM, a genomics analysis platform built on top of Spark, to convert the variant information residing a VCF file to Parquet for easier downstream analytics, in a process similar to the one described in the [Will Spark Power the Data behind Precision Medicine?](https://blogs.aws.amazon.com/bigdata/post/Tx1GE3J0NATVJ39/Will-Spark-Power-the-Data-behind-Precision-Medicine) post. Then, you use custom Python code to massage the data and select only the appropriate fields that you need for analysis with Athena.  First, [spin up an EMR cluster](http://docs.aws.amazon.com/ElasticMapReduce/latest/ManagementGuide/emr-gs.html) (version 5.0.3) for the ETL process. I used a c4.xlarge for my master node and m4.4xlarges with 1 TB of scratch for my core nodes.  After you SSH into your master node, clone the [git repository](https://github.com/awslabs/aws-big-data-blog.git). You can also put this in as a bootstrap action when spinning up your cluster. |  |  | | --- | | **Building High-Throughput Genomics Batch Workflows on AWS: Introduction (Part 1 of 4)**  <https://aws.amazon.com/blogs/compute/building-high-throughput-genomics-batch-workflows-on-aws-introduction-part-1-of-4/>  by Andy Katz | on 30 MAY 2017 | | | **Genomics Research on AWS**  <https://github.com/aws-samples/aws-batch-genomics/tree/v1.0.0>  A tutorial on how to package and deploy a bioinformatics workflow on AWS using AWS Batch  This tutorial will cover the material presented within the "Genomics Workflows on AWS" blog post series  (Part 1, Part 2, Part 3, Part 4) that covers the basics of bootstrapping a bioinformatics analysis pipeline on AWS.  We break down the tutorial roughly as follows:  Setting up your AWS account (if you do not already have one)  Package a set of bioinformatics applications using Docker  Create a AWS Batch environment for analysis  Define and deploy AWS Step Functions to control the data processing steps  Initiate a workflow |  |  | | --- | | **Optimizing for cost, availability and throughput by selecting your AWS Batch allocation strategy**  by Bala Thekkedath | on 24 OCT 2019 | in Advanced (300), Amazon EC2, AWS Batch  <https://aws.amazon.com/blogs/compute/optimizing-for-cost-availability-and-throughput-by-selecting-your-aws-batch-allocation-strategy/> | | AWS offers a broad range of instances that are advantageous for batch workloads. The scale and provisioning speed of AWS’ compute instances allow you to get up and running at peak capacity in minutes without paying for downtime. Today, I’m pleased to introduce allocation strategies: a significant new capability in [AWS Batch](https://docs.aws.amazon.com/batch/latest/userguide/allocation-strategies.html) that makes provisioning compute resources flexible and simple. In this blog post, I explain how the AWS Batch allocation strategies work, when you should use them for your workload, and provide an example CloudFormation script. This blog helps you get started on building your personalized Compute Environment (CE) most appropriate to your workloads.  **Overview**  AWS Batch is a fully managed, cloud-native batch scheduler. It manages the queuing and scheduling of your batch jobs, and the resources required to run your jobs. One of AWS Batch’s great strengths is the ability to manage instance provisioning as your workload requirements and budget needs change. AWS Batch takes advantage of AWS’s broad base of compute types. For example, you can launch compute-based instances and memory instances that can handle different workload types, without having to worry about building a cluster to meet peak demand.  Previously, AWS Batch had a cost-controlling approach to manage compute instances for your workloads. The service chose an instance that was the best fit for your jobs based on vCPU, memory, and GPU requirements, at the lowest cost. Now, the newly added allocation strategies provide flexibility. They allow AWS Batch to consider capacity and throughput in addition to cost when provisioning your instances. This allows you to leverage different priorities when launching instances depending on your workloads’ needs, such as: controlling cost, maximizing throughput, or minimizing [Amazon EC2 Spot](https://aws.amazon.com/ec2/spot/) instances interruption rates.  There are now three instance allocation strategies from which to choose when creating an AWS Batch [Compute Environment](https://docs.aws.amazon.com/batch/latest/userguide/compute_environments.html) (CE). They are:  1.        Spot Capacity Optimized  2.        Best Fit Progressive  3.        Best Fit | |

|  |
| --- |
| **Registry of Open Data on AWS**  <https://registry.opendata.aws/>  **All of this is done with open data on S3 buckets** **currently 205 datasets** |
| **See**  [**Usage Examples/Tutorials for datasets listed in this registry**](https://registry.opendata.aws/usage-examples)   |  | | --- | | **Encyclopedia of DNA Elements (ENCODE)**  <https://registry.opendata.aws/encode-project/> | | The Encyclopedia of DNA Elements (ENCODE) Consortium is an international collaboration of research groups funded by the National Human Genome Research Institute (NHGRI). The goal of ENCODE is to build a comprehensive parts list of functional elements in the human genome, including elements that act at the protein and RNA levels, and regulatory elements that control cells and circumstances in which a gene is active. ENCODE investigators employ a variety of assays and methods to identify functional elements. The discovery and annotation of gene elements is accomplished primarily by sequencing a diverse range of RNA sources, comparative genomics, integrative bioinformatic methods, and human curation. Regulatory elements are typically investigated through DNA hypersensitivity assays, assays of DNA methylation, and immunoprecipitation (IP) of proteins that interact with DNA and RNA, i.e., modified histones, transcription factors, chromatin regulators, and RNA-binding proteins, followed by sequencing.  **Usage Examples/Tutorials**  **Exploring ENCODE data from EC2 with Jupyter notebook** Excellent Detailed instructions  [https://github.com/ENCODE-DCC/encode-data-usage-examples/blob/master/**mount\_s3\_bucket\_and\_run\_jupyter\_on\_ec2.ipynb**](https://github.com/ENCODE-DCC/encode-data-usage-examples/blob/master/mount_s3_bucket_and_run_jupyter_on_ec2.ipynb)  This notebook demonstrates how to mount s3://encode-public on an EC2 instance using Goofys.  Goofys makes an S3 bucket appear as a typical UNIX file system, and is useful for tools that expect a local file path.  Once the bucket is mounted we can launch a Jupyter notebook on the instance and connect to it remotely.  The benefit of using EC2 is that the compute is scalable to the analysis you would like to perform, and you don't have to download anything locally.  **Ingesting ENCODE data into TileDB with S3 backend**  [https://github.com/ENCODE-DCC/encode-data-usage-examples/blob/master/**ingest\_encode\_data\_tile\_db\_with\_s3\_backend.ipynb**](https://github.com/ENCODE-DCC/encode-data-usage-examples/blob/master/ingest_encode_data_tile_db_with_s3_backend.ipynb)  **Related Publication:**  **New developments on the Encyclopedia of DNA Elements (ENCODE) data portal**  <https://academic.oup.com/nar/article/48/D1/D882/5622708#190992715>  Nucleic Acids Research, Volume 48, Issue D1, 08 January 2020, Pages D882–D889  12 November 2019 |  |  | | --- | | **Genome Aggregation Database (gnomAD)**  <https://registry.opendata.aws/broad-gnomad/> | | **Description**  The Genome Aggregation Database (gnomAD) is a resource developed by an international coalition of investigators that aggregates and harmonizes both exome and genome data from a wide range of large-scale human sequencing projects. The summary data provided here are released for the benefit of the wider scientific community without restriction on use. The v2 data set (GRCh37) spans 125,748 exome sequences and 15,708 whole-genome sequences from unrelated individuals. The v3 data set (GRCh38) spans 71,702 genomes, selected as in v2. Sign up for the gnomAD mailing list here.  Resources on AWS  Description  gnomAD summary data aggregated from large-scale human genome and exome sequencing projects.  Resource type  S3 Bucket  **Amazon Resource Name (ARN)**  **arn:aws:s3:::gnomad-public-us-east-1**  AWS Region  us-east-1  AWS CLI Access (No AWS account required)  aws s3 ls s3://gnomad-public-us-east-1/ --no-sign-request | | **gnomAD browser**  Downloads  <https://gnomad.broadinstitute.org/downloads/>  gnomAD data is available for download through Google Cloud Public Datasets, the Registry of Open Data on AWS, and Azure Open Datasets. We recommend using Hail and our Hail utilities for gnomAD to work with the data.  In addition to the files listed below, gnomAD variants are available as a BigQuery dataset and Terra has a demo workspace for working with gnomAD data.  **Registry of Open Data on AWS**  Files can be browsed and downloaded using the AWS Command Line Interface.  **aws s3 ls s3://gnomad-public-us-east-1/release/** | | **Example use of the big data**  **A structural variation reference for medical and population genetics**  Nature volume 581, pages444–451(2020)  31 Citations  <https://www.nature.com/articles/s41586-020-2287-8> **Data availability** All gnomAD-SV site-frequency data for appropriately consented samples (*n* = 10,847) have been distributed in VCF and BED format via the gnomAD browser (<https://gnomad.broadinstitute.org/downloads/>), as well as from NCBI dbVar under accession nstd166. Furthermore, these SVs have been integrated directly into the gnomAD browser[8](https://www.nature.com/articles/s41586-020-2287-8#ref-CR8). The architecture of the gnomAD browser is described in the main gnomAD study[4](https://www.nature.com/articles/s41586-020-2287-8#ref-CR4), as well as instructions for how to access and query the data hosted therein.  **Code availability**  The gnomAD-SV discovery pipeline is publicly available via a series of methods configured for the FireCloud/Terra platform (<https://portal.firecloud.org/#methods>) under the methods namespace ‘Talkowski-SV’. The svtk software package used extensively in the gnomAD-SV discovery pipeline is publicly available via GitHub (<https://github.com/talkowski-lab/svtk>). Most custom scripts used in the production and/or analysis of the gnomAD-SV dataset are publicly available via GitHub (<https://github.com/talkowski-lab/gnomad-sv-pipeline>). All code is made available under the MIT license, unless stated otherwise. | | **gnomAD browser**  **Downloads**  <https://gnomad.broadinstitute.org/downloads/>  gnomAD data is available for download through Google Cloud Public Datasets, the Registry of Open Data on AWS, and Azure Open Datasets.  We recommend using Hail and our Hail utilities for gnomAD to work with the data.  In addition to the files listed below, gnomAD variants are available as a BigQuery dataset and Terra has a demo workspace for working with gnomAD data.  Registry of Open Data on AWS  Files can be browsed and downloaded using the AWS Command Line Interface.  **aws s3 ls s3://gnomad-public-us-east-1/release/**  **All chromosomes sites VCF**  Download Google / Amazon / Microsoft  **Exomes**  chr1 sites VCF 5.77 GiB  chr2 sites VCF 4.2 GiB  chr3 sites VCF 3.29 GiB  chr4 sites VCF 2.17 GiB  chr5 sites VCF 2.51 GiB  chr6 sites VCF 2.83 GiB  chr7 sites VCF 2.88 GiB  chr8 sites VCF 2.13 GiB  chr9 sites VCF 2.4 GiB  chr10 sites VCF 2.23 GiB  chr11 sites VCF 3.61 GiB  chr12 sites VCF 3.07 GiB  chr13 sites VCF 976.73 MiB  chr14 sites VCF 2.02 GiB  chr15 sites VCF 2.08 GiB  chr16 sites VCF 3.04 GiB  chr17 sites VCF 3.62 GiB  chr18 sites VCF 882.14 MiB  chr19 sites VCF 4.3 GiB  chr20 sites VCF 1.44 GiB  chr21 sites VCF 652.73 MiB  chr22 sites VCF 1.43 GiB  chrX sites VCF 1.33 GiB  chrY sites VCF 15.66 MiB  **Genomes**  -  -  - | | **Constraint**  <https://gnomad.broadinstitute.org/downloads/#v2-constraint>  The mutational constraint spectrum quantified from variation in 141,456 humans. Nature 581, 434–443 (2020).  **Multi-nucleotide variants (MNVs)**  <https://gnomad.broadinstitute.org/downloads/#v2-multi-nucleotide-variants>  Landscape of multi-nucleotide variants in 125,748 human exomes and 15,708 genomes. Nature Communications 11, 2539 (2020).  **Proportion expressed across transcripts (pext)**  <https://gnomad.broadinstitute.org/downloads/#v2-pext>  Transcript expression-aware annotation improves rare variant interpretation. Nature 581, 452–458 (2020) | | **Related Publications**  **The mutational constraint spectrum quantified from variation in 141,456 humans**  <https://www.nature.com/articles/s41586-020-2308-7>  Nature volume 581, pages434–443(2020)  300 Citations  **Transcript expression-aware annotation improves rare variant interpretation**  <https://www.nature.com/articles/s41586-020-2329-2>  **Thousands of human sequences provide deep insight into single genomes** 27 May 2020  A massive genome-sequencing and analysis effort has produced the most comprehensive sets of data and tools for understanding human genetic variation so far. The resource will be invaluable to biologists of every stripe.  **A structural variation reference for medical and population genetics**  <https://pubmed.ncbi.nlm.nih.gov/32461652/>  Nature volume 581, pages444–451(2020)  31 Citations | |

|  |
| --- |
| **Using Amazon EC2 to run large data analyses cheaply Spot instances on Amazon Elastic Compute Cloud (EC2)**  Jul 5, 2018  <https://gex.netlify.app/post/using-amazon-ec2-to-run-large-data-analysis-cheaply/> |
| **Spot instances on Amazon Elastic Compute Cloud (EC2)** allow researchers to do high performance computing at very low cost.  For example, a 64-core workstation with 256 GB of memory can be rented at about $0.7 per hour  Main topics covered this tutorial  Set up Key Pairs and Security Groups to enable SHH login,  Request spot instance (virtual machine),  SHH access to the instance via Putty and Filezilla,  Install Docker software,  Build a Docker image based on the Bioconductor Docker definition files,  Start R and compute from within the container,  Create a “volume” (a virtual hard drive) and attach it to running instances,  Take a snapshot of a volume, copy it across regions, and use it to create a volume,  Google Compute Engine set-up. |
| Start the Spot Advisor and enter your request.  **Change the amount required to 1. Otherwise, 20 instances will be requested**.  A 64-core instance with 256GB memory costs $0.67 per hour or about $16 per day.  Note that Amazon has web servers all across the world. You can switch regions from the top right of the screen.  Prices and availability of resources vary greatly across the region. |

|  |
| --- |
| **What is Packer?**  [**https://www.packer.io/**](https://www.packer.io/) |
| Packer is an open source tool for creating identical machine images for multiple platforms from a single source configuration.  Packer is lightweight, runs on every major operating system, and is highly performant, creating machine images for multiple platforms in parallel.  Packer does not replace configuration management like Chef or Puppet. In fact, when building images, Packer is able to use tools like Chef or Puppet to install software onto the image.  A machine image is a single static unit that contains a pre-configured operating system and installed software which is used to quickly create new running machines.  Machine image formats change for each platform.  Some examples include AMIs for EC2, VMDK/VMX files for VMware, OVF exports for VirtualBox, etc.  Cosine Similarity support in Amazon Elasticsearch Service  Posted On: Jul 29, 2020  Amazon Elasticsearch Service now supports cosine similarity distance metric with k-Nearest Neighbor (k-NN) to power your similarity search engine. Cosine similarity is used to measure similarities between two vectors, irrespective of their sizes and is most commonly used in information retrieval, image recognition, text similarity, bioinformatics and recommendation systems.  Reproducible and robust workshop materials  Workshops were authored using R Markdown, and compiled into a book (PDF and ePub) and website using Bookdown R package. Bookdown, in turn, uses the gitbook publishing system ( https://www.gitbook.com/) to produce a variety of formats from the same source material. R Markdown files intended to be part of a Bookdown project do not contain the required front matter of a typical stand-alone R Markdown document. To help authors use and test the correct format, we seeded each workshop document with the syllabus that had been submitted by that author, and successfully built the book of the submitted syllabi. Each workshop represented a chapter of a book compiled using the Bookdown software. This approach provided several advantages:  • R markdown syntax is already familiar to any developer of a Bioconductor package, since it is the standard approach to creating the package “vignette” or prose documentation.  • R markdown implements “literate programming” by including formatted text, runnable code, and output of the code  • Bookdown allows collating chapters as a clean, lightweight online book format, and pandoc additionally allows creation of PDF and ePub formats  • These formats can then be self-published with options to order paper copies through companies such as <https://leanpub.com> |

**bookdown: Authoring Books and Technical Documents with R Markdown**

<https://books.google.com/books?hl=en&lr=&id=_LrZDQAAQBAJ&oi=fnd&pg=PT12&dq=+Xie+Y+:+bookdown:+Authoring+Books+and+Technical+Documents+with+R+Markdown+%5BInternet%5D+.+Boca+Raton,+Florida:+Chapman+and+Hall/CRC%3B+2016++10.1201/9781315204963++&ots=tz1AhTAN96&sig=mmTuwDYYrl3sVm8SjtJD8tvnwd0#v=onepage&q&f=false>

|  |
| --- |
| **DevOps Pipeline Experience** |
| NGC Virtual envir intended for DevOps Pipeline  GitLab, Docker, Nexus, Jenkins --> Deployment Env  Git Local --> GitLab Remote -->  Docker  Docker Local --> Docker Hub --> Docker Hub Nexus  Dockerfile YAML  Jenkins  Jenkinsfile YAML |

**Google Cloud Computing**

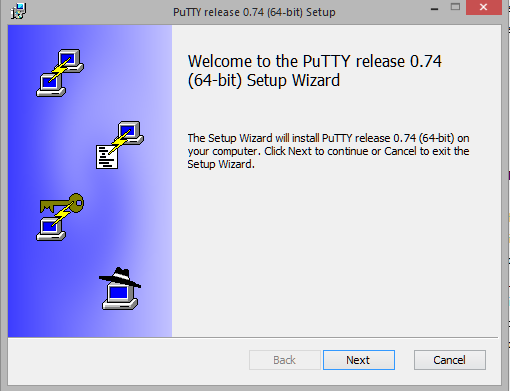
|  |
| --- |
| **Google Cloud Platform** |
| **Google Kubernetes Engine**  <https://cloud.google.com/kubernetes-engine/> Secured and managed Kubernetes service with four-way auto scaling and multi-cluster support.  New customers get $300 in free credits to spend on Google Cloud during the first 90 days.  All customers get one zonal cluster per month for free, not charged against your credits.  **Types of clusters**  <https://cloud.google.com/kubernetes-engine/docs/concepts/types-of-clusters> |

|  |
| --- |
| **Google Domain Name** |
| A domain name through a domain name registrar.  You can register a domain name through Google Domains or another domain registrar of your choice.  An IP address to point the A record of your zone. |

**PuTTy**

<https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

putty-64bit-0.74-installer.msi



**C:\Program Files\PuTTY\ putty.exe**

**puttygen.exe**

**AWS EC2 Free-Tier**

