Human Chromosome Cloud Compute & Interactive Exploratory Analysis (Progress Report)

Suma Venugopal
UI-Urbana Champaign
sumav2@illinois.edu

Emil Varghese
UI-Urbana Champaign
emilv2@illinois.edu

Bahman Sheikh
UI-Urbana Champaign
bahmans2@illinois.edu

Nagraj Naidu
UI-Urbana Champaign
nnaidu2@illinois.edu

I. ABSTRACT

Next-generation sequencing (NGS) of human genetic data and related ongoing researches to find genomics variants of genetic disorders for diagnostic purposes are bringing about groundbreaking advances in the healthcare, medical and pharmaceutical fields. Publicly accessible and constantly updated variant databases that host genetic variant data are often the primary source of pathogenic variant data used by researchers for clinical diagnosis.

In the past, it was not easy for users of genetic data to analyze and derive insights from their data sets, thanks to the enormous size of the genome sequences and complexity of the DNA data sets that they need to analyze. In the 21st century, distributed computing and high performance multi variant cloud computing capabilities and services offered by sophisticated cloud platforms like Amazon Web Services(AWS) have opened up a whole new realm for such researchers who deal with complex genetic data sets on a day to day basis.

II. INTRODUCTION

The proposal is to implement a genetic analysis and visualization pipeline that leverages cloud technologies offered by Amazon Web Services(AWS). to load, process and do interactive exploratory analysis. Our intention is to use Chromosome 22 data from the 1000 genomes project along with annotations from the ClinVar data set. The 1000 genomes project is a public project where human genetic samples are collected from populations around the world and made available to the public for genetic variant analysis. ClinVar is a public data set of human genetic variants - most of those pathogenic - aggregated by variant and clinical condition; with free public access for research and analysis.ClinVar data is maintained at the National Center for Biotechnology Information (NCBI), within the National Library of Medicine (NLM) at the National Institutes of Health (NIH).

The proposed system is expected to load the entire clinical variant data for select the chromosomes and reported genetic variants from ClinVar into AWS and use multiple AWS services to process and derive insights from this data. Such visual insights could help researchers planning to use ClinVar and 1000 genome project data get a better understanding of what these data-sets hold, in terms of genome sequencing, pathogenic genetic variants, disease causing alleles etc and a lot more in just a few minutes, thus helping them make critical decisions pertaining to their research.

We will extend on this publication by investigating additional genetic variant insights. We will also perform Principle Component Analysis on the data.

III. DATA SETS

The following are the data sets relevant to our paper.

- The Human Chromosome Data Sets from 1000 genomics contain variant genetic data. We will specifically focus on Chromosome 22. This is one of the smaller chromosomes and suitable for our analysis and requires relatively smaller compute. This data is available on the 1000 genome project website. It is also available as a public data set on Amazon Web Services. This data set is in the Variant Call Format (VCF) and must be converted to a suitable format like Parquet before our analysis.
- Annotation Data set from ClinVar has the relationships among human genetic variations and phenotype, with supporting evidence. This data is available on Amazon Web Services as a public data set.
- We intended to produce our own ”full column” Chromosome data from 1000 genomics using a variety of Cloud Compute Components but our limited monetary resources hampered us from running long cluster intensive jobs. So we used ”reduced column” Chromosome 22 data in Parquet format that is available on Amazon Web Services.
- Population lookup dataset that maps the 1000 genome samples to the respective population and super population.

IV. TECHNOLOGIES AND TOOLS

We intend deploying the following cloud components in a processing pipeline as follows.

- ADAM is a genomics analysis platform with specialized file formats built using Apache Avro, Apache Spark and Parquet. We will deploy this as an EC2 instance of a docker container to convert VCF format genetic variant files to parquet files suitable for use on Amazon Web Services. We will use the readily available docker image ”gelog/adam” for this task. We will run this container on the VCF files either on bare metal machines or on the cloud in EC2 instances.

- The files produced by ADAM are nested parquet structure formatted files. This appears to be not suitable for use with Amazon Web Services in general and Athena in particular. We will need to select columns and create a flat parquet file. We will use Spark 2.0 for this task. Spark is capable of reading and writing parquet files. This will again be deployed as a docker container on bare metal or as an EC2 instance on the cloud. A Spark 2.0 docker container is readily available on docker hub ”mesosphere/spark”
- The generated parquet files (or the readily available Chromosome 22 data) will be uploaded to an AWS S3 bucket suitable for further use by downstream pipeline components.
- Athena is a web service that can query big data parquet files and provides a SQL HIVE interface including JOINS and other standard SQL syntax. We will use Athena to query Human Genetic Variant files and the ClinVar data set for analysis and insights.
- Amazon QuickSight is a fast, cloud-powered business intelligence service that makes it easy to deliver insights into the datasets created in Athena.
- For Principle Component Analysis of the data set we will deploy the appropriate software on an EC2 micro instance to query Athena.

V. ARCHITECTURE

1000 Genome Project data to be used for this implementation is part of the AWS public data set. The data set is in a VCF format which is a commonly used file format for representing genome data. The ClinVar variant data is available to be accessed by the users in VCF, XML and txt formats. This implementation uses the txt variation.

Below is the architectural flow diagram.

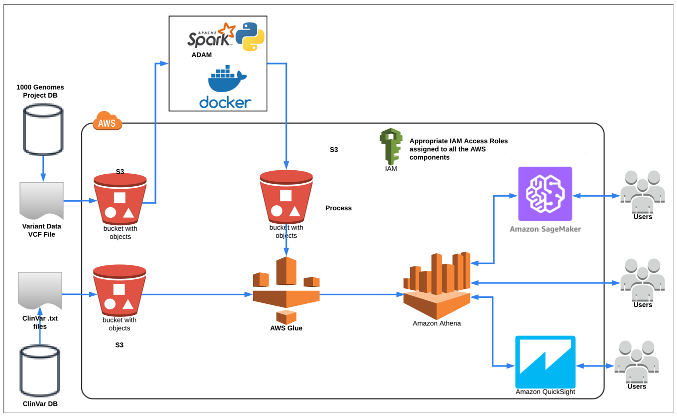


Fig. 1. Architecture Diagram

Genome variant data in the VCF format will be loaded to AWS S3. From there a genomics data processing engine ADAM installed on top of Apache Spark and Python Scripts within a Docker container virtualization layer will be used to pre-process these source files and convert them in to parquet format. These parquet files will be placed in an yet another output S3 folder. Both the Variant parquet files and ClinVar text files placed in another S3 source folder will be loaded to Amazon Athena using the metadata generated with the help of AWS Glue.

Once the data is available in Athena, users can directly interact with Athena to run queries or interface via Amazon Sagemaker or Amazon Quick Sight to do machine learning and exploratory data analysis among many other things.

VI. DESIGN

The following section gives the various components used in the processing pipeline. Snippets are included for better understanding.

- ADAM DOCKER CONTAINER: We used a readily available container from Docker Hub for this processing.
  - SPARK VERSION 1.4.1
  - ADAM VERSION 2.10
  - UBUNTU OS 14.04
  - JAVA VERSION 7

  The container was run on Bare Metal Windows 10 PC with 16GB RAM, i7 Processor to reduce Cloud Costs. We took the VCF format Chromosome 22 files and converted them to the commonly used Parquet format.

- AWS S3 BUCKET: We uploaded files to S3 for processing in the cloud. The following files were uploaded to S3
  - ADAM generated Parquet files
  - Population CSV Files
  - ClinVar Files
  - Population lookup Files

  Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance.

- AWS GLUE: The file schema for the Chromosome 22 Adam generated parquet is nested and complex. It is hard to infer the schema. Furthermore the file corpus is large. AWS Glue is an excellent tool for this and infers the meta data for the files.

  For an example of the Parquet meta data see Figure 2.

- ATHENA TABLES: We then created ATHENA Tables for each of the following
  - ClinVar (Annotations)
  - cr22_variant (Chr22 Variation File)
  - cr22population (Population File)

  Amazon Athena is an interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. Athena is server less, so there is no infrastructure to setup or manage.

Amazon Athena uses Presto with full standard SQL support and works with a variety of standard data formats including CSV and Parquet. Athena can handle complex analysis, including large joins, window functions, and arrays.

Because Amazon Athena uses Amazon S3 as the underlying data store, it is highly available and durable with data redundantly stored across multiple facilities and multiple devices in each facility.

For an example of the Create Table Dynamic Definition Language see Figure 3.

- ATHENA VIEWS:Views were created across tables to ease joins and have a reduced representation of the Data Sets

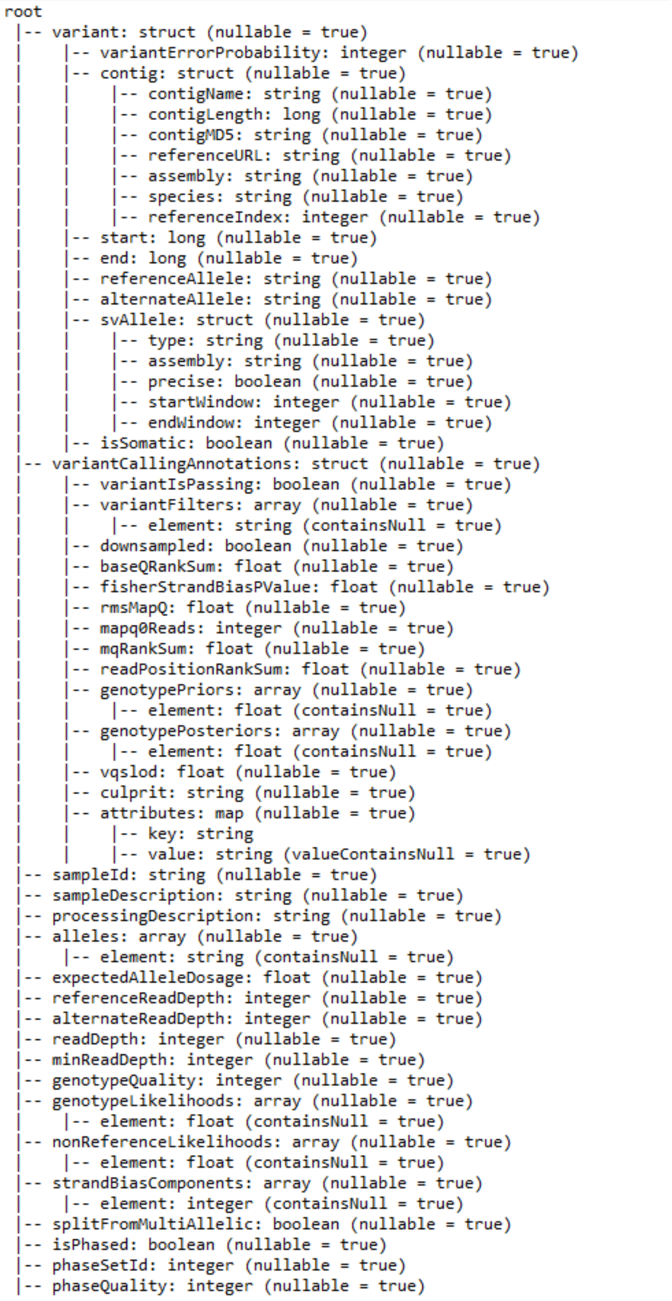
  For an example of the Create View Dynamic Definition Language see Figure 4.

- ATHENA QUERIES AND JOINS: We queried tables and views to get to the data from the Athena console.

  For an example of the Join SQL see Figure 5.

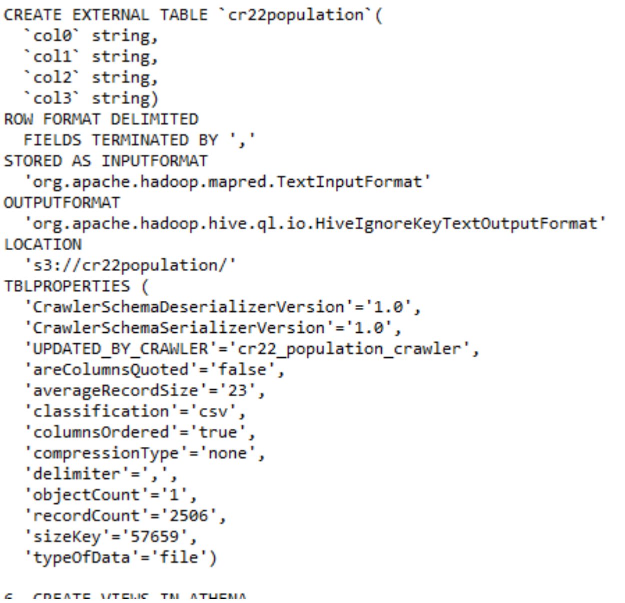
- AWS SAGEMAKER: This is used to query Athena from a Python environment and to run Machine Learning code on the genome data.The above Queries and Joins are reused here

  For an example of the Sage Maker Notebook see Figure 8.



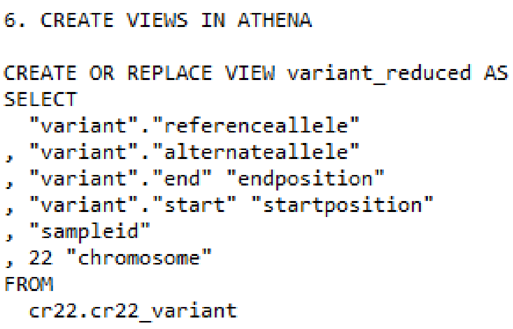
```
root
 |-- variant: struct (nullable = true)
 |    |-- variantErrorProbability: integer (nullable = true)
 |    |-- contig: struct (nullable = true)
 |    |    |-- contigName: string (nullable = true)
 |    |    |-- contigLength: long (nullable = true)
 |    |    |-- contigMD5: string (nullable = true)
 |    |    |-- referenceURL: string (nullable = true)
 |    |    |-- assembly: string (nullable = true)
 |    |    |-- species: string (nullable = true)
 |    |    |-- referenceIndex: integer (nullable = true)
 |    |-- start: long (nullable = true)
 |    |-- end: long (nullable = true)
 |    |-- referenceAllele: string (nullable = true)
 |    |-- alternateAllele: string (nullable = true)
 |    |-- svAllele: struct (nullable = true)
 |    |    |-- type: string (nullable = true)
 |    |    |-- assembly: string (nullable = true)
 |    |    |-- precise: boolean (nullable = true)
 |    |    |-- startWindow: integer (nullable = true)
 |    |    |-- endWindow: integer (nullable = true)
 |    |-- isSomatic: boolean (nullable = true)
 |-- variantCallingAnnotations: struct (nullable = true)
 |    |-- variantIsPassing: boolean (nullable = true)
 |    |-- variantFilters: array (nullable = true)
 |    |    |-- element: string (containsNull = true)
 |    |-- downsampled: boolean (nullable = true)
 |    |-- baseQRankSum: float (nullable = true)
 |    |-- fisherStrandBiasPValue: float (nullable = true)
 |    |-- rmsMapQ: float (nullable = true)
 |    |-- mapq0Reads: integer (nullable = true)
 |    |-- mqRankSum: float (nullable = true)
 |    |-- readPositionRankSum: float (nullable = true)
 |    |-- genotypePriors: array (nullable = true)
 |    |    |-- element: float (containsNull = true)
 |    |-- genotypePosteriors: array (nullable = true)
 |    |    |-- element: float (containsNull = true)
 |    |-- vqslod: float (nullable = true)
 |    |-- culprit: string (nullable = true)
 |    |-- attributes: map (nullable = true)
 |    |    |-- key: string
 |    |    |-- value: string (valueContainsNull = true)
 |-- sampleId: string (nullable = true)
 |-- sampleDescription: string (nullable = true)
 |-- processingDescription: string (nullable = true)
 |-- alleles: array (nullable = true)
 |    |-- element: string (containsNull = true)
 |-- expectedAlleleDosage: float (nullable = true)
 |-- referenceReadDepth: integer (nullable = true)
 |-- alternateReadDepth: integer (nullable = true)
 |-- readDepth: integer (nullable = true)
 |-- minReadDepth: integer (nullable = true)
 |-- genotypeQuality: integer (nullable = true)
 |-- genotypeLikelihoods: array (nullable = true)
 |    |-- element: float (containsNull = true)
 |-- nonReferenceLikelihoods: array (nullable = true)
 |    |-- element: float (containsNull = true)
 |-- strandBiasComponents: array (nullable = true)
 |    |-- element: integer (containsNull = true)
 |-- splitFromMultiAllelic: boolean (nullable = true)
 |-- isPhased: boolean (nullable = true)
 |-- phaseSetId: integer (nullable = true)
 |-- phaseQuality: integer (nullable = true)
```

Fig. 2. Parquet Meta Data for Variant File



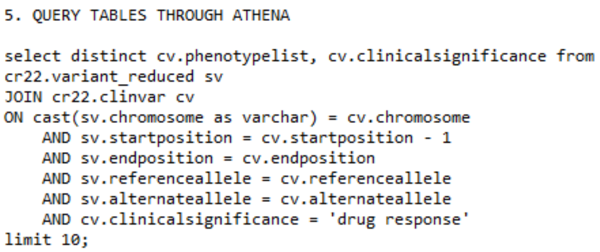
```
CREATE EXTERNAL TABLE `cr22population`(
  `col0` string,
  `col1` string,
  `col2` string,
  `col3` string)
ROW FORMAT DELIMITED
  FIELDS TERMINATED BY ','
STORED AS INPUTFORMAT
  'org.apache.hadoop.mapred.TextInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'
LOCATION
  's3://cr22population/'
TBLPROPERTIES (
  'CrawlerSchemaDeserializerVersion'='1.0',
  'CrawlerSchemaSerializerVersion'='1.0',
  'UPDATED_BY_CRAWLER'='cr22_population_crawler',
  'areColumnsQuoted'='false',
  'averageRecordSize'='23',
  'classification'='csv',
  'columnsOrdered'='true',
  'compressionType'='none',
  'delimiter'=',',
  'objectCount'='1',
  'recordCount'='2506',
  'sizeKey'='57659',
  'typeOfData'='file')
```

Fig. 3. Create Table Example of the Population File for Athena



```
6. CREATE VIEWS IN ATHENA

CREATE OR REPLACE VIEW variant_reduced AS
SELECT
  "variant"."referenceallele"
, "variant"."alternateallele"
, "variant"."end" "endposition"
, "variant"."start" "startposition"
, "sampleid"
, 22 "chromosome"
FROM
  cr22.cr22_variant
```

Fig. 4. Create View Example



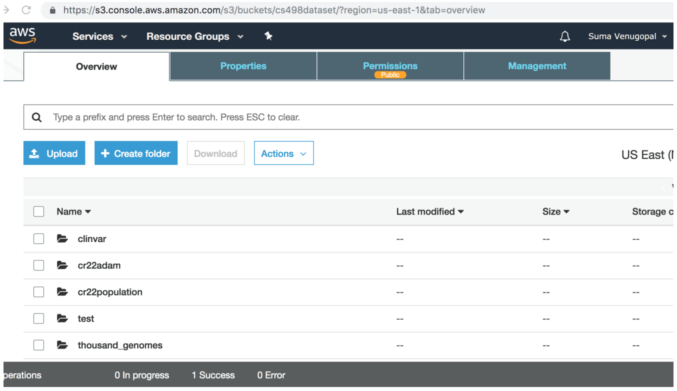
```
5. QUERY TABLES THROUGH ATHENA

select distinct cv.phenotypelist, cv.clinicalsignificance from
cr22.variant_reduced sv
JOIN cr22.clinvar cv
ON cast(sv.chromosome as varchar) = cv.chromosome
    AND sv.startposition = cv.startposition - 1
    AND sv.endposition = cv.endposition
    AND sv.referenceallele = cv.referenceallele
    AND sv.alternateallele = cv.alternateallele
    AND cv.clinicalsignificance = 'drug response'
limit 10;
```

Fig. 5. Athena Join Example

VII. EVALUATION / RESULTS

Below are the details of the progress made so far and the results.

- Trimmed parquet files creation for chromosome 22 variant data was completed successfully using docker, Apache Spark, ADAM and python scripts.

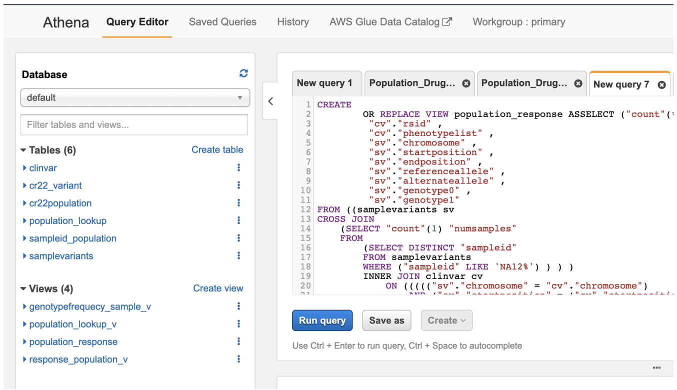


https://s3.console.aws.amazon.com/s3/buckets/cs498dataset/?region=us-east-1&tab=overview
aws
Services
Resource Groups
Suma Venugopal
Overview
Properties
Permissions
Public
Management
Type a prefix and press Enter to search. Press ESC to clear.
Upload
Create folder
Download
Actions
US East (

Name	Last modified	Size	Storage c
clinvar	--	--	--
cr22adam	--	--	--
cr22population	--	--	--
test	--	--	--
thousand_genomes	--	--	--

Fig. 6. Input S3 bucket view

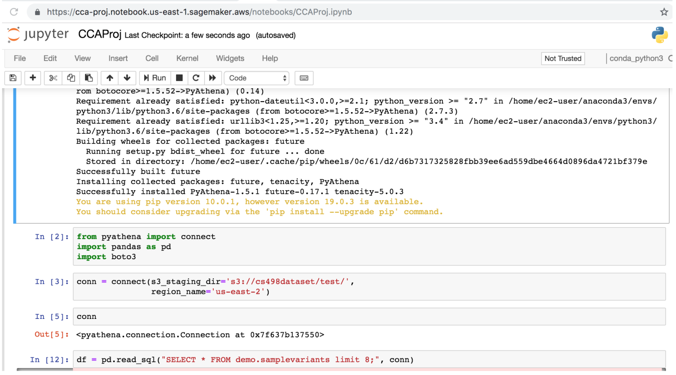
- Metadata for the parquet files were extracted using AWS Glue
- ClinVar text files and genome variant data present in parquet files were successfully loaded to Amazon Athena. Views were created in Athena combining ClinVar, sample variants and population data for meaningful analysis of data.



```
CREATE
        OR REPLACE VIEW population_response ASSELECT ("count"(
         "cv"."rsid" ,
         "cv"."phenotypelist" ,
         "sv"."chromosome" ,
         "sv"."startposition" ,
         "sv"."endposition" ,
         "sv"."referenceallele" ,
         "sv"."alternateallele" ,
         "sv"."genotype0" ,
         "sv"."genotype1"
FROM ((samplevariants sv
CROSS JOIN
    (SELECT "count"(1) "numsamples"
    FROM
        (SELECT DISTINCT "sampleid"
        FROM samplevariants
        WHERE ("sampleid" LIKE 'NA12%') ) ) )
        INNER JOIN clinvar cv
            ON (((((("sv"."chromosome" = "cv"."chromosome")
```

Fig. 7. Athena Tables

- Connection between Amazon Sagemaker and Athena was established and tested

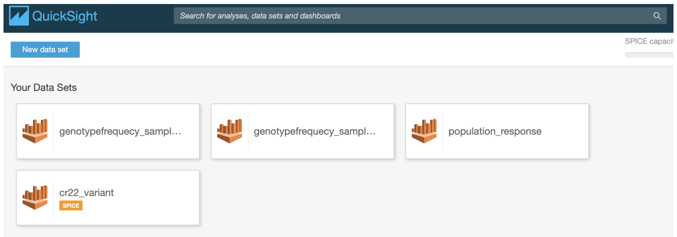


https://cca-proj.notebook.us-east-1.sagemaker.aws/notebooks/CCAProj.ipynb
jupyter
CCAProj Last Checkpoint: a few seconds ago (autosaved)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
conda_python3
Run
Code

```
rom botocore>=1.5.52->PyAthena) (0.14)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1; python_version >= "2.7" in /home/ec2-user/anaconda3/envs/
python3/lib/python3.6/site-packages (from botocore>=1.5.52->PyAthena) (2.7.3)
Requirement already satisfied: urllib3<1.25,>=1.20; python_version >= "3.4" in /home/ec2-user/anaconda3/envs/python3/
lib/python3.6/site-packages (from botocore>=1.5.52->PyAthena) (1.22)
Building wheels for collected packages: future
  Running setup.py bdist_wheel for future ... done
  Stored in directory: /home/ec2-user/.cache/pip/wheels/0c/61/d2/d6b7317325828fbb39ee6ad559dbe4664d0896da4721bf379e
Successfully built future
Installing collected packages: future, tenacity, PyAthena
Successfully installed PyAthena-1.5.1 future-0.17.1 tenacity-5.0.3
You are using pip version 10.0.1, however version 19.0.3 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.

In [2]: from pyathena import connect
        import pandas as pd
        import boto3

In [3]: conn = connect(s3_staging_dir='s3://cs498dataset/test/',
                       region_name='us-east-2')

In [5]: conn

Out[5]: <pyathena.connection.Connection at 0x7f637b137550>

In [12]: df = pd.read_sql("SELECT * FROM demo.samplevariants limit 8;", conn)
```

Fig. 8. Sagemaker Athena Connection

- Data sets were created in QuickSight using the Athena view to create analysis.



QuickSight

Search for analyses, data sets and dashboards

New data set

SPICE capaci

Your Data Sets

genotypefrequecy_sampl...

genotypefrequecy_sampl...

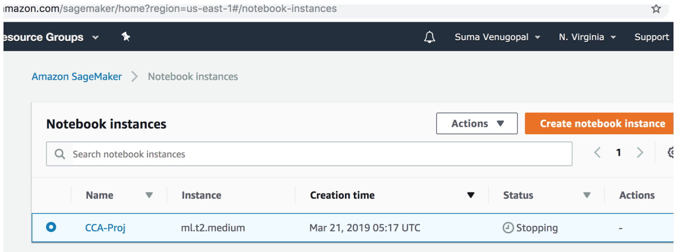
population_response

cr22_variant

SPICE

Fig. 9. QuickSight Datasets

- Necessary IAM (Identity Access Management) roles were provisioned to make sure that all the components had the permissions to access each other.
- As part of this implementation, Principal Component Analysis (PCA) was done on a subset of reduced Chromososome 22 data, with 1000000 rows. An ml.t2.medium instance provisioned by AWS running Sagemaker with Python Boto 3 was used for this purpose. Below diagrams show the details of how the Sagemaker was set up and also various 2D and 3D plots (created using Python Matplotlib) showing intermediate steps and results in the PCA process.



mazon.com/sagemaker/home?region=us-east-1#/notebook-instances

Amazon SageMaker > Notebook instances

Notebook instances

Search notebook instances

	Name	Instance	Creation time	Status	Actions
●	CCA-Proj	ml.t2.medium	Mar 21, 2019 05:17 UTC	Stopping	-

Fig. 10. A snapshot of Sagemaker instance used

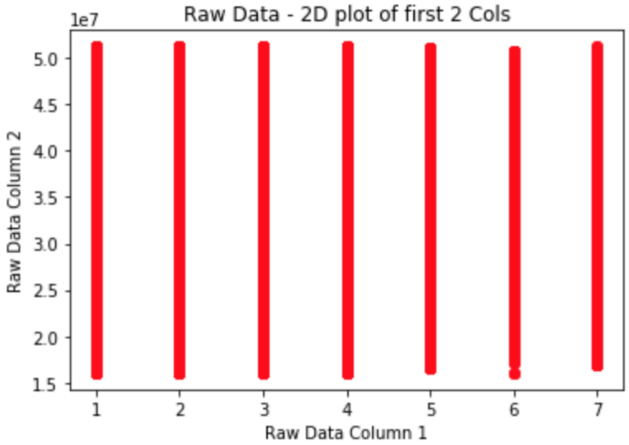


Fig. 11. PCA Initial Plot - 2 Attributes of Chromosome 22 Raw Data

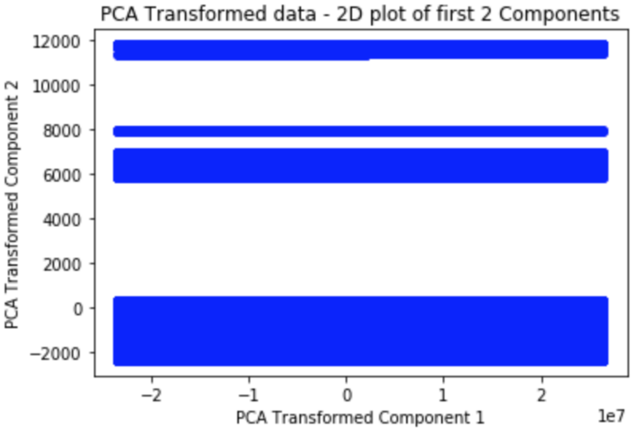


Fig. 12. Chromosome 22 - 2 Components After PCA

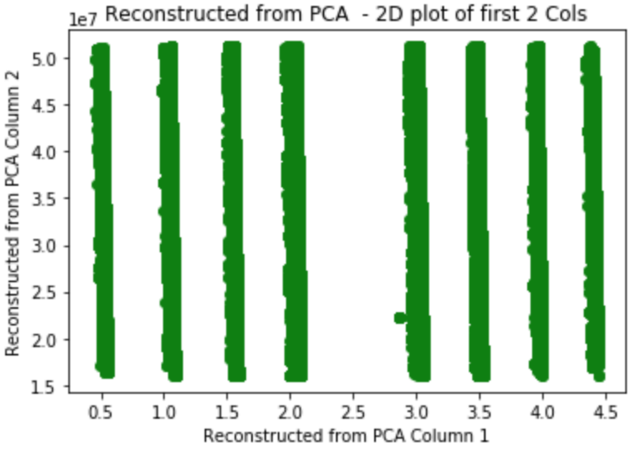


Fig. 13. Chromosome 22 - 2 Components Reconstructed from PCA

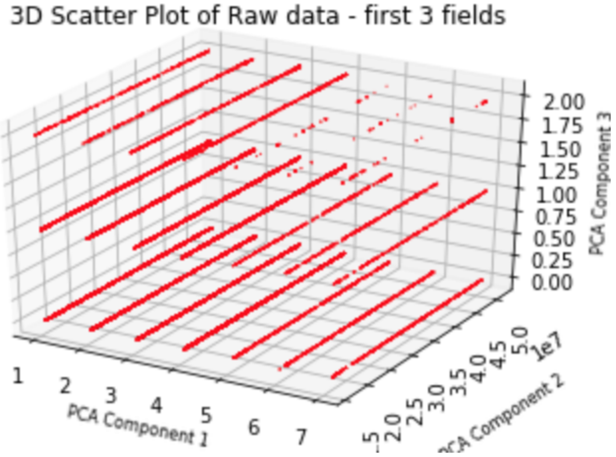


Fig. 14. Chromosome 22 - Raw Data Before PCA 3D Scatter Plot

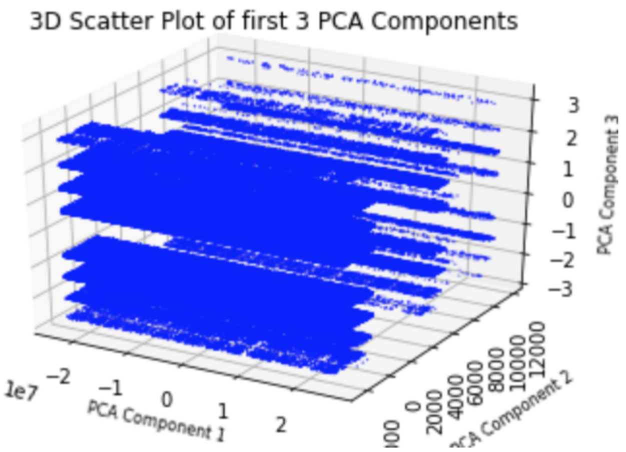


Fig. 15. Chromosome 22 - Post PCA 3D Scatter Plot

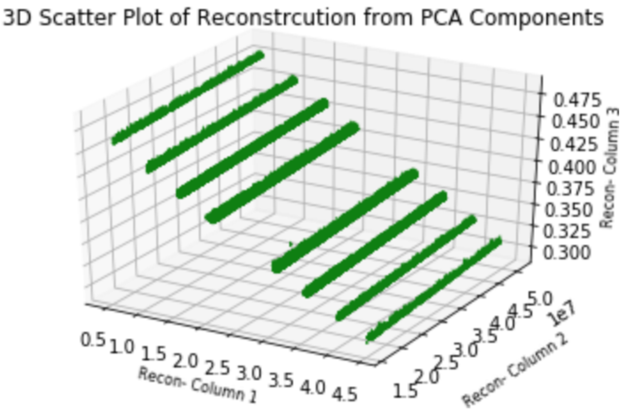


Fig. 16. Chromosome 22 - Post PCA Reconstruction 3D Scatter Plot

- Various QuickSight visualizations were also created from the Athena Human genome, ClinVar and Population tables that we loaded as part of this implementation.Those proved to be very useful in gathering a lot of hidden information about the Chromosome22 data sample variants and also about the ClinVar data as well as details of the various populations that contributed to the Human Genome project and ClinVar database.
Some of the visualizations are given below for reference.

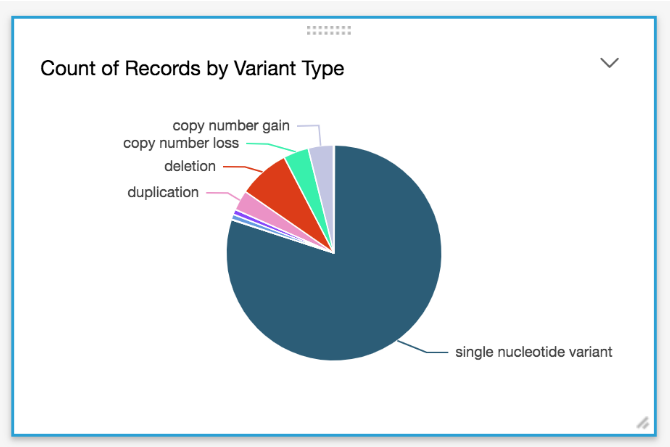


Fig. 17. A QuickSight Pie Chart Showing Prominant Variant Types in ClinVar

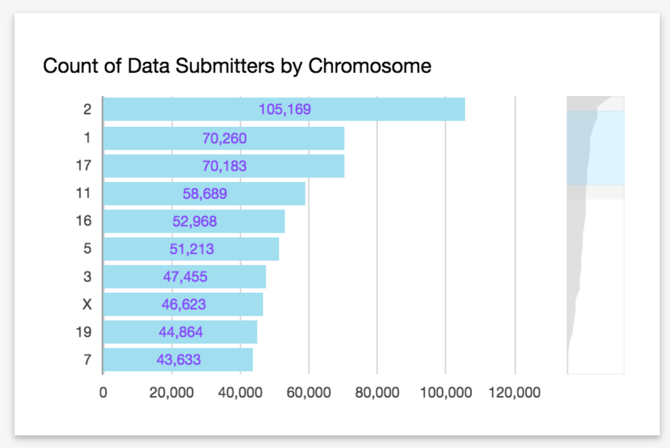


Fig. 18. A QuickSight Bar Chart Showing Which Chromosomes Receive Max Variant Submissions in ClinVar

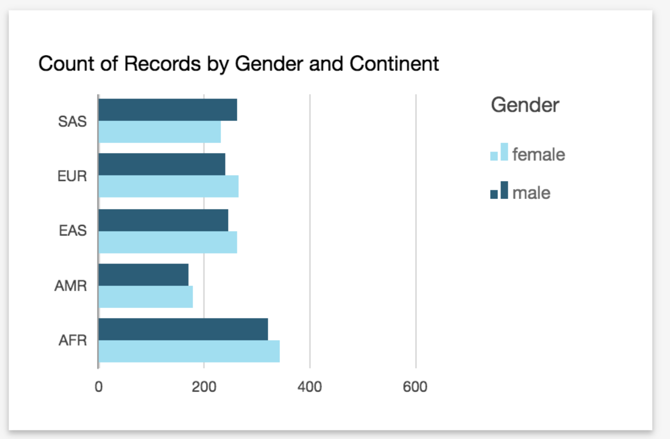


Fig. 19. A QuickSight Bar Chart Showing the Origin of Human Genome Project Contributors separated by Gender)

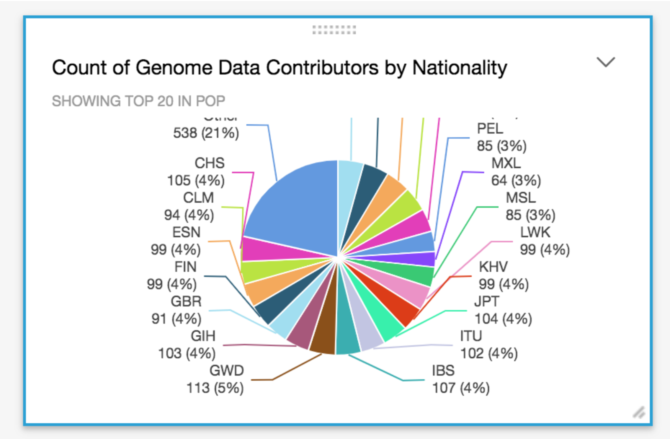


Fig. 20. A QuickSight Pie Chart Showing Percentage of Human Genome Project Contributors by Country)

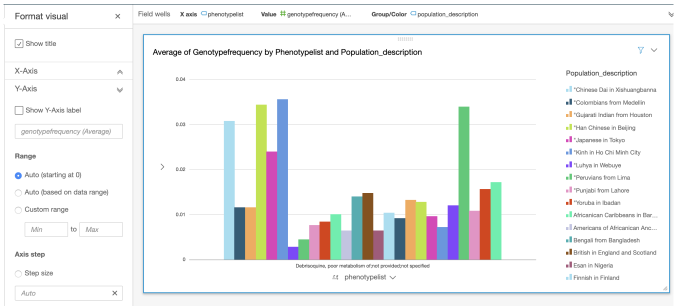
VIII. DISCUSSION

As mentioned in the initial proposal, considering the huge size of the Genome data and the monetary /time resources that it demands, the scope of the project will be limited to chromosome 22 data although the data for all the other chromosomes are also available as part of the 1000 Genome Project.

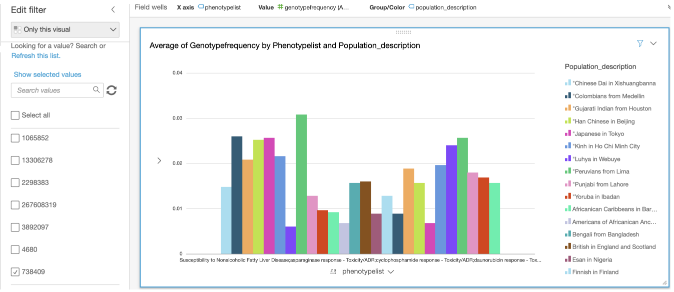
Based on the work done and progress made so far, there might be a need to do further data pre-processing to create an ideal input for Amazon Sagemaker Principle Component Analysis (PCA). There could be multiple ways to achieve this , like creating the needed query results from Athena or by creating and storing the necessary data fields in a separate S3 and accessing it from there. This part is still under analysis. (This has been completed and PCA done successfully)

The Sample variants data set provides insights to chromosome variations among subpopulation of individuals and joining this with the population lookup and ClinVar data for clinical significance opens up interesting insights. Interactive data analysis based on sub population groups were done and below QuickSight visualizations demonstrate some of the key findings.

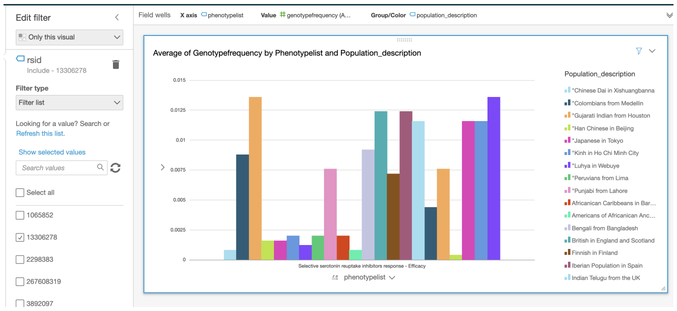
- Population drug response among population subgroups



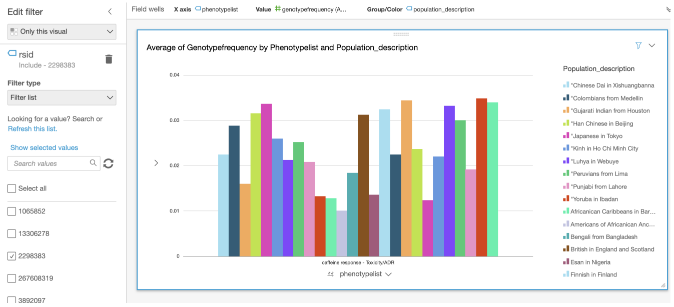
- Susceptibility to nonalcoholic fatty liver disease


Edit filter
Only this visual
Looking for a value? Search or Refresh this list.
Show selected values
Search values
□ Select all
□ 1065852
□ 13306278
□ 2298383
□ 267608319
□ 3892097
□ 4680
☑ 738409
Field wells
X axis
phenotypelist
Value
genotypefrequency (A...
Group/Color
population_description
Average of Genotypefrequency by Phenotypelist and Population_description
0.04
0.03
0.02
0.01
0
Susceptibility to Nonalcoholic Fatty Liver Disease;asparaginase response - Toxicity/ADR;cyclophosphamide response - Toxicity/ADR;daunorubicin response - Tox...
phenotypelist
Population_description
"Chinese Dai in Xishuangbanna
"Colombians from Medellin
"Gujarati Indian from Houston
"Han Chinese in Beijing
"Japanese in Tokyo
"Kinh in Ho Chi Minh City
"Luhya in Webuye
"Peruvians from Lima
"Punjabi from Lahore
"Yoruba in Ibadan
Africanican Caribbeans in Bar...
Americans of Africanican Anc...
Bengali from Bangladesh
British in England and Scotland
Esan in Nigeria
Finnish in Finland

- Selective serotonin re uptake inhibitors



- Caffeine Response



It has also been noticed that while QuickSight can create visualization using data from a single Athena table, doing exploratory data analysis using data from multiple tables would involve additional steps. Options to achieve this including writing SQL queries using the custom SQL option in QuickSight are being explored. (This has been completed and QuickSight visualizations created successfully)

IX. RELATED WORK

This work will extend on the following Amazon Web Services published article "Interactive Analysis of Genomics Data Set using Amazon Athena" This implementation is also an extension of the AWS blog 'Run SQL queries from your SageMaker notebooks using Amazon Athena', in that in addition to running Athena queries, we have also successfully implemented PCA using data from Athena tables.

X. FUTURE WORK

Below are the activities that will be covered as part of the future work:

- Data pre-processing for Sage-maker input.
- Principal Component Analysis and inferences using the results of Principal Component Analysis using Sagemaker, Jupyter notebooks and Python and R
- Work on SQL queries for Quick Sight input
- Exploratory data analysis and dashboards using Amazon QuickSight
- Documenting and demonstrating specific results of Principle Components Analysis and Exploratory Data Analysis in the final report
- These are some of the insights we plan to obtain using Athena queries and the sourced parquet files.
  - List of phenotype for the chromosome.
  - List of genes for the chromosome.
  - Clinical significance of the gene including drug response, pathogenic and benign.
  - Frequencies of drug responses.

  (All the above items have been completed successfully)
- Below are some possible future extensions of this implementation:
  - Extend the implementation to Chromosomes other than Chromosome22.
  - Extend on the PCA and perform additional Machine Learning and predictions based on Genome Data.
  - If funding is available to handle data at scale, analyze genome data on AWS using super powerful EC2 instances to unearth allele level variations that could be the possible causes of various diseases like diabetes, autism or even cancer

XI. DIVISION OF WORK

- Suma: Glue, QuickSight, Reporting.
- Bahman: PCA, Machine Learning, Security IAM Roles.
- Emil: Architecture, Athena, Queries.
- Nagraj: Adam Processing, Spark.

XII. CONCLUSION

As part of this project, we intended to use cloud components, especially those provided by the AWS platform to do exploratory data analysis and machine learning on publicly available Human Genome Data set. As we conclude the project, we have achieved all the end results that we envisioned at the beginning of the project including PCA and QuickSight visualizations of interpretations from Genome Data, as demonstrated in the evaluation results and screenshots above.

REFERENCES

[1] https://www.ncbi.nlm.nih.gov/clinvar/intro/

[2] https://docs.aws.amazon.com/

[3] https://github.com/bigdatagenomics/adam

[4] https://aws.amazon.com/blogs/big-data/interactive-analysis-of-genomic-datasets-using-amazon-athena/

[5] https://hub.docker.com/r/mesosphere/spark

[6] https://hub.docker.com/r/gelog/adam

[7] https://aws.amazon.com/blogs/big-data/running-r-on-amazon-athena/

[8] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4690620/

[9] https://academic.oup.com/bfg/article/16/3/163/2555377

[10] http://grch37.ensembl.org/info/docs/tools/vcftoped/index.html

[11] https://aws.amazon.com/1000genomes/