**Internship Report**

Data Analysis and Software Development for Food Security

**Michael Hall**

MSc Statistics and Data Science (2021), University of Texas at San Antonio, TX, USA

BS Applied Statistics (2018), University of Houston-Downtown, TX, USA

BBA in Economics (2010), Sam Houston State University, TX, USA

Data Science Intern at PBGL: **October 2021 – September 2022**

Supervisor: **Norman Warthmann, Molecular Geneticist, FAO/IAEA-PBGL**

**A picture containing histogram

Description automatically generated**

Table of Contents

**Executive Summary**3

**Background**4

**Software Tools for Genomic Data Analysis**……………………………………………………………………………………….5

**Install Open Source Package Management Software**……..………..………………………………………………………6

**Literature Reading**…………………………………………………………………………………………………………………………….9

**Molecular Biology Techniques**………………………………………………………………………………………………………..10

**pCloud**…………………………………………………………………………………………………………………………………………….14

**Certificate in Genomic Data Science John Hopkins University**…….………………………………………………….15

**IAEA Certifications**………………………………………………………………………………………………………..…………………16

**Read The Docs**…………………………………………………………………………………………………………………………………17

**QTL analysis in Rice Cold Tolerance a Snakemake Variant Caller Workflow Example**…….………………18

**Executive Summary**

*“Software’s are becoming the new cargo ships and freight trucks. Digital files are becoming the new core commodities. The formers won't eliminate the latter, but a restructuring is happening.”*

**― Hendrith Vanlon Smith Jr, CEO of Mayflower-Plymouth**

Agrarian societies have existed for thousands of years, a society in which production of crops are of economic importance. From hunter gather societies transitioning into agricultural societies and agricultural societies transitioning into industrial, technologically driven societies. The development of superior crop varieties has in some sense always existed even if only as an idea.

A revolution in science occurred when a pioneer discovered a method to sequencing genomes which led to a very expensive project. This project was known as “The Human Genome Project” which started in 1990 and which was completed in 2003. However, the first plant was not sequenced until the year 2000, which was part of a five-year effort and was published in Nature as a research article. And as technology improved the cost to sequence genomes has dramatically decreased making it affordable for everyone to do it. Low-cost genome sequencing of plant material allows scientists to investigate fundamental questions of nature. At PBGL most of the research covers mutations induced in a particular genome by gamma irradiation and its consequences.

For me it is not enough to identify a mutation for mutations are everywhere in a sense on the genome. It much more difficult to say this mutation caused this phenomenon in what I see or do not see and that there exists a valid justification to sufficiently explain it. Developing software tools to better analyse and manage big data generated from these experiments is necessary to continue in genetics research. I would hypothesize the growth of this data is exponential and finding ways to manage it better is perhaps more important than just analysis.

At PBGL I was tasked with training in multiple programming languages like Python, R, Snakemake, Bash, all while working on a Linux Operating System. All these languages have their points of integration and in my opinion not one is any better than the other. In my internship I have found numerous coding scripts all of which work as intended, albeit not without an occasional bug. However, adapting it to the laboratories main objectives requires a bit of finesse. Overall, my biggest contribution was discovering existing data analysis tools and adapting them to the work here at PBGL. I wanted to make a workflow that anybody can understand and use, even someone who does not have a background in Bioinformatics.

**BACKGROUND**

*“Open source is a development methodology; free software is a social movement.”*

**Richard Stallman**

Computer Sciencerules and related data workflows:

1. Clone a repository from Github
2. Create an environment of prerequisite software packages
3. Populate configuration files with its necessary specifications
4. Import external data sources such as import files etc.
5. Run the software

**Software and Documentation**

To find all software I worked on please visit my github page. Not every repository is complete, however, I used it as a data science journal, and you can reference it too.

[GithubPageSofwareDocumentation](https://github.com/PBGLMichaelHall)

**Software Tools for Genomic Data Analysis**

A bug in software is common so having ability to trouble shoot an issue is also important. Not everything works as expected.

*“Never allow the same bug to bite you twice”*

***Steve Maguire***

* [Oxford Nanopore Quality Control Workflow](https://minionqc.readthedocs.io/en/latest/MinION.html#figures-plots)(FAST5)
* [VCFHunter](https://github.com/PBGLMichaelHall/VCFHunter)(VCF)
* [Copy Number Variants Sequence Analysis](https://cnvseq.readthedocs.io/en/latest/CNV.html#standard-output-clumpify-python)(FASTQ)
* [QTL\_BSA\_in\_Python](https://py-qtl-parser.readthedocs.io/en/latest/Rice_BSA.html)(VCF)
* [QTLAlleleFrequenciesPlot](https://github.com/PBGLMichaelHall/QTLAlleleFrequenciesPlot)(VCF)
* [QTL\_BSA-Sorghum](https://qtl-bsa.readthedocs.io/en/latest/)(VCF)

Most of these software tools are used in downstream analysis of genomic data. What I mean by this is after aligning your sample sequences (FASTQ) with a reference genome (FASTA) a VCF file is made. Which tool you use depends on the question you want to answer.

*“It is a capital mistake to theorize before one has data.”*

***Sherlock Holmes***

Full Stream Raw Data

Downstream Analysis

Quality Control

**Install Management Software**

*“The best thing about a Boolean is even if you are wrong, you are only off by a bit.”*

**-Anonymous**

**R, Mamba, and Miniconda3:**

Website: [Miniconda3](https://docs.conda.io/en/latest/miniconda.html)

How many bits 32 or 64?

Ubuntu: $uname -m

X86\_64

Choose correct Platform and software Name Linux 64-bit

[Miniconda3 Software Download Link](https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh)

Run bash script from download in Linux Terminal.

$bash Miniconda3-latest-Linux-x86\_64.sh

$conda update --all

$conda upgrade --all

$conda install mamba --yes

$mamba install git

Website: [Comprehensive R Network](https://cran.r-project.org/)

*# update indices*

sudo apt update -qq

*# install two helper packages we need*

sudo apt install --no-install-recommends software-properties-common dirmngr

*# add the signing key (by Michael Rutter) for these repos*

*# To verify key, run gpg --show-keys /etc/apt/trusted.gpg.d/cran\_ubuntu\_key.asc*

*# Fingerprint: E298A3A825C0D65DFD57CBB651716619E084DAB9*

wget -qO- https://cloud.r-project.org/bin/linux/ubuntu/marutter\_pubkey.asc **|** sudo tee -a /etc/apt/trusted.gpg.d/cran\_ubuntu\_key.asc

*# add the R 4.0 repo from CRAN -- adjust 'focal' to 'groovy' or 'bionic' as needed*

sudo add-apt-repository "deb https://cloud.r-project.org/bin/linux/ubuntu $(lsb\_release -cs)-cran40/"

sudo apt install --no-install-recommends r-base

**Oxford Nanopore Quality Control Workflow**

**Clone github repository:**

$git clone <https://github.com/PBGLMichaelHall/MinionQC.git>

$cd MinionQC/

$mamba env create --file env/MinionQCenv.yaml

**Activate New Environment:**

$conda activate MinionQC

**Run Quality Control Script:**

$Rscript Minion\_R\_QC\_Script/MinIONQC.R -h

-i INPUT Full path to **sequencing\_summary**.**txt** file.

-q QSCORE\_CUTOFF, The cut off value for the quality score (default is 7).

$Rscript Minion\_R\_QC\_Script/MinIONQC.R -i summary/sequencing\_summary.txt -q 0

Chart, histogram

Description automatically generated

**VCF HUNTER**

Download Prerequisite Software:

***#Clone Github Repository*:**

$git clone <https://github.com/PBGLMichaelHall/VCFHunter.git>

$cd VCFHunter/

***#Create Environment:***

$mamba env create --file env/vcf2allPropandCov.yaml

$conda activate vcf2allPropandCov

***#Make Prerequisite Input file:***

$head -n VCF/freebayes\_D2.filtered.vcf | grep "#CHROM" | sed 's/\t/\n/g' | tail -n +10 > sorghum\_all\_names.tab

***#Filter vcf file:***

$python PythonScripts/vcfFilter.1.0.py --vcf freebayes\_D2.filtered.vcf --names sorghum\_all\_names.tab

--MinCov 10 --MaxCov 300 --MinAl 3 --nMiss 1 --RmAlAlt 1:3:4:5:6 --prefix DNAseq\_Filtered -g y

***#Separate VCF by Chromosome:***

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr01 --recode --out data/Sorghumvcf/Chr01\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr02 --recode --out data/Sorghumvcf/Chr02\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr03 --recode --out data/Sorghumvcf/Chr03\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr04 --recode --out data/Sorghumvcf/Chr04\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr05 --recode --out data/Sorghumvcf/Chr05\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr06 --recode --out data/Sorghumvcf/Chr06\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr07 --recode --out data/Sorghumvcf/Chr07\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr08 --recode --out data/Sorghumvcf/Chr08\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr09 --recode --out data/Sorghumvcf/Chr09\_DNAseq\_Filtered\_filt.vcf.gz

vcftools --gzvcf DNAseq\_Filtered\_filt.vcf.gz --chr Chr10 --recode --out data/Sorghumvcf/Chr10\_DNAseq\_Filtered\_filt.vcf.gz

**#Run python script on all configuration files**

$python PythonScripts/vcf2allPropAndCov.py --conf SorghumVcf.conf --origin SorghumOrigin.tab --acc D2\_F2\_tt --ploidy 2 --dcurve y --col SorghumColor.conf

**A picture containing table

Description automatically generated**

**Copy Number Variants**

*BANANA All Chromosomes*

Chart, scatter chart

Description automatically generated

*SORGHUM Chromosome 9*

*Chart, scatter chart

Description automatically generated*

**Literature Reading, mostly Research Papers:**

* Rapid re-identification of human samples using portable DNA sequencing
* Mapping of Quantitative Trait Loci Underlying Cold Tolerance in Rice Seedlings via High-Throughput Sequencing of Pooled Extremes
* QTLseqr: An R Package for Bulk Segregant Analysis with Next-Generation Sequencing
* The Statistics of Bulk Segregant Analysis Using Next Generation Sequencing
* A Low-Cost Genotyping Protocol and Kit for Marker-Assisted Selection of Orange Lemma (rob1.a), a Feed Quality Trait in Barley (Hordeum vulgare L.)
* Mapping immature fruit colour-related genes via bulked segregant analysis combined with whole-genome re-sequencing in pepper (Capsicum annuumm)
* High-accuracy long-read amplicon sequences using unique molecular identifiers with Nanopore or PacBio sequencing
* DNeasy Plant Mini Kit Protocol
* The Sun, The Genome, and The Internet
* Full Planet, Empty Plates The New Geopolitics of Food Security
* Plant immunity: towards an integrated view of plant-pathogen interactions
* Plant immunity: Danger Perception and Signalling
* Efficient Screening Techniques to Identify Mutants with TR4 Resistance in Banana
* Identity by Descent: Variation in Meiosis, Across Genomes, and in Populations
* Using next generation sequencing to isolate mutant genes from forward genetic screens
* CNV-seq, a new method to detect copy number variation using high-throughput sequencing
* Big Data: Astronomical or Genomical?
* Lonely Planet Europe
* Living in Vienna
* Statistical Analysis of Next Generation Sequencing Data-Springer
* Bioinformatics Data Skills
* Bioinformatics\_With\_Python\_Cookbook
* Computational Exome and Genome Analysis
* Copy Number Variants
* Variant Calling: Methods and Protocols

**Molecular Biology Techniques and a Training Course for Scientific Visitors**

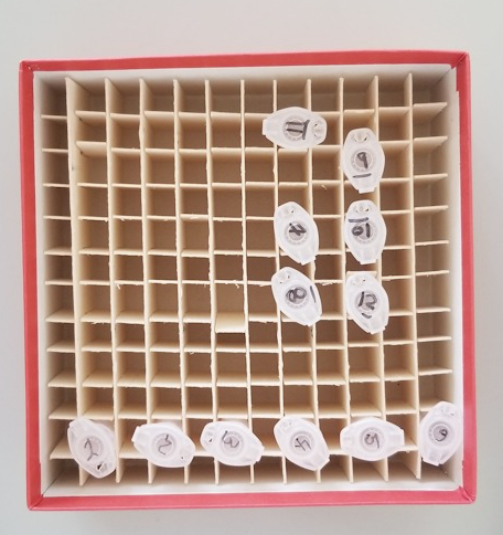
During the dates from June 13th through June 2022 5 Scientific visitors came to Seibersdorf from Qatar, Syria, Kuwait, and Iran. I also participated in this intensive two-week training course. The first week was designed around mutation breeding, forward and reverse genetics, next generation sequencing, and finally a demonstration in R of a Quantitative Trait Locus mapping example. The software is available on PBGL github webpage with accompanying documentation.

Week two covered marked assisted selection, genotyping assay [gel based] and [KASP] assay. To put it all in to practice we extracted DNA from Sorghum and barley plants and used these molecular biology techniques to genotype them. This allows us to have a level of confidence in predicting what phenotype/characteristics the plant will show as it matures. This knowledge is cost effective and accelerates breeding times which saves money. Therefore, we do it.

A picture containing text

Description automatically generated

In this photograph, a group of visiting scientists successfully used PBGL’s low-cost genotyping protocol which correctly identified one Barley sample as possessing the (rob1.a) gene mutation resulting in Orange Lemma.



I also participated in my first trial of extracting DNA samples from 12 Sorghum Samples.

**Graphical user interface, table, Excel

Description automatically generated**

And used nanodrop to determine DNA purity etc.

**pCloud**

I was given a yellow Kingston 8 GB flash drive with a true capacity of 7.25 GB from which all data files were stored. However, what would happen if you or I lost the stick? Of course, this is a rhetorical question. That is why I recommend backing up your data with the European version of Dropbox, P Cloud. It is Switzerland based free cloud storage solution with over 10 million users and comes with a 6 GB account after registering online. The benefit is you can access this genomic data from anywhere with an internet connection and you do not need a physical USB flash drive. So, live on the cloud.

[PCloud](https://www.pcloud.com/eu)

Certificate in Genomic Data Science

At the beginning of the internship, I pursued a certification in Genomic Data Science on a massive open online course. It covered such as Genomic Technologies, Python, Algorithms for DNA Sequencing, Command Line Tools, Bioconductor, and Statistics. In the end it was issued by John Hopkins University to verify specialization status in the field.

Text, timeline

Description automatically generated with medium confidence

**IAEA Certifications**

 The IAEA offers certification online.

**Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated**

**Read-the-Docs**

[**QTL\_BSA\_Python**](https://py-qtl-parser.readthedocs.io/en/latest/)

[**VCFHunter**](https://vcfhunter.readthedocs.io/en/latest/)

[**MinIonQC**](https://minionqc.readthedocs.io/en/latest/)

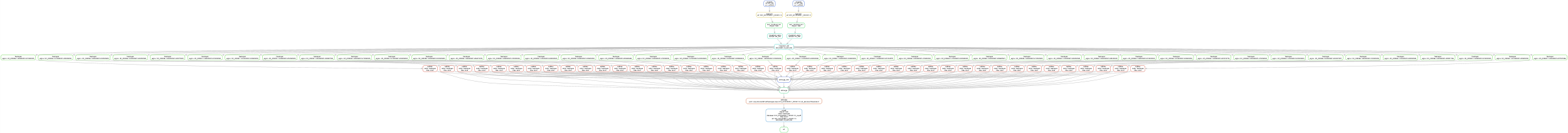
[**VCFstat**](https://vcfstat-r-package.readthedocs.io/en/latest/)

[**CNVSeq**](https://cnvseq.readthedocs.io/en/latest/)

[**QTL\_BSA\_R**](https://qtl-bsa.readthedocs.io/en/latest/)

**QTL analysis in Rice Cold Tolerance a Snakemake Variant Caller Workflow Example**

[**Oryza\_Satvia Asian Cultivated Rice PBGL dna-proto variant calling workflow with Quantitative Trait Locus and Bulk Segregant Analysis**](https://oryza-satvia-snakemake.readthedocs.io/en/latest/)



Chart

Description automatically generated with medium confidence