Cover page feather background with text block



**Abstract**

VCF files containing human genomic data are of great interest to biologists. With the data in these files, a biologist can infer vital information regarding a selected population. Specifically, anomalies in these genetic files can be indicative of many outcomes, such as mutation or sickness. However, analyzing files such as these can be an overwhelming task for a biologist due to their extreme size. A proposed solution to this problem is a web page that allows the biologist to upload their VCF file and receive various information regarding the file as output. Specifically, the output will show areas of interest through visualizations of how the data changes throughout the file. Using this output, the biologist will be alerted to specific anomalous areas which they could then analyze on a deeper level to understand why the anomalies exist.

**Table of Contents**

1. Introduction ……………………………………….…….……………….....
   1. Background ………….………………………….
   2. Statement of the Problem …………………………….
2. Scope of Work ……………….………………………………….…..…….
   1. Overview …………………………………..
   2. Literature Review …………………………………………...
   3. Alternative Solutions ……………………………………………
   4. Evaluation ……………………………………………
   5. Decision ……………………………………………
3. Implementation Details …………………………………………………….………
   1. System Specifications and Functionalities …………………………………..
   2. Overall System Design with Block Diagram ……………………………………..
   3. Circuit diagrams, Flowcharts and/or Use Case Diagrams …………………..
   4. Testing and calibration results ……………………………………….
   5. Discussions on Lessons Learned ………………………………………..
4. Conclusion ………………………….………..
5. References …..……………………………….
6. Personnel/Organization Chart ………………………………………………...........
7. Appendices ……………………………..

**1 Introduction**

This report describes a solution to locating anomalous genetic data in files containing millions of data points. The solution will be an interface where a biologist will upload their data file to an anomaly detection algorithm that will output the location of anomalous regions.

(Author: John Callaghan)

**1.1** **Background**

Due to the rapid progression of technology over the past few decades, our understanding of the human genome has expanded greatly. With the completion of the Human Genome Project, we now have a great deal of biological information regarding human DNA. Due to the new abundance of genetic information, the work which biologists do has become even more complex. Analyzing alleles of specific chromosomes can lead to files that contain millions of data points. It is not feasible for a biologist to conduct genetic research on a population without the assistance of a machine learning program. Though examples of this type of analysis exist, it hasn’t yet been made easily accessible to the average user. Ideally, a user would be able to conduct anomaly detection research on a given data file without prior coding or programming knowledge.

(Author: John Callaghan)

**1.2** **Statement of the Problem**

Anomalies in human genomic data could be indicative of many things. For one, they could show the process of mutation beginning in a given population. Mutations can range anywhere from unimportant to extremely impactful. Either way, a biologist who is studying the genomic data of a large number of individuals must be aware of any possible mutations. Another possible cause for anomalies in the data samples is sickness. Determining whether the outliers exist due to an illness could prove pivotal to protecting a given population’s health.

Anomalous data could also represent errors in the data gathering. If the data indicates values that are not possible or unreasonable, then the biologist should be made aware that the data as a whole may be unreliable. Overall, when dealing with human genomic data, it is vital for the biologist to be cognizant of any outliers present. Current methods for locating anomalies in human genomic data require the biologist to have a basic understanding of coding to parse through the data themselves. This can be a difficult process for the biologist and would ideally be streamlined. In order to make this process easier, the user interface should be simple and easy to use. Also, the algorithm should run without any prior understanding of coding. Finally, the output should be clear and indicate where in the data file areas of interest may exist.

(Author: Oasis Husband)

**2. Scope of Work**

**2.1 Overview**

The goal of this project is to create an algorithm that can take an input file containing human genomic data and inform the biologist whether anomalies exist in the data. Ideally, this algorithm would be simple enough to use so that the biologist would not need prior knowledge of coding. Also, the visualizations returned by the algorithm would be clear and concise, as well as show the locations of the anomalies.

In order to achieve the goal of this project, the main objectives have been broken down into general tasks. These tasks may have many different sub-tasks, which will be discussed further in the following sections. The team member responsible is listed with each task.

● Design and implement an end-user interface that allows the user to upload a VCF file and receive analysis output (Michael Hollinger / Avigail Martinez)

● Design and implement an anomaly detection method that is efficient and reliable (John Callaghan / Kamal Shrouder)

● Design and implement a visualization component of the algorithm which will be output to the biologist (Oasis Husband)

Essentially, the team will be split into three different sub-groups, each focusing on a key part of the project. These three key parts are user interface, anomaly detection, and data visualization.

(Author: John Callaghan)

**2.2 Literature Review**

In preparation for this project, it is important to perform patent searching for similar designs. Though finding a patent that is the same as our project is unlikely, there could still be other designs regarding anomaly detection algorithms that can be used as an example of this kind of analysis. In the following section, each team member will introduce a patent that is somehow related to our project. These patents were all chosen due to having some connection to the anomaly detection methods we will be using or the biological nature of the data we will analyze.

**Patent 1**

**A.** **Team member reviewing patent:** *John Callaghan*

**B.** **Pertinent patent information:**

Number: CN107657288B

Title: Power dispatching flow data anomaly detection method based on isolated forest algorithm

Inventor(s): 宁文元高欣郭子明贾庆轩李新鹏马龙穆永铮李军良徐建航

Assignee: Beijing University of Posts and Telecommunications Beijing Kedong Electric Power Control System Co Ltd State Grid Jibei Electric Power Co Ltd

Date filed: 10/26/17

Date granted: 07/03/2020

Number of claims: 3

**C.** **Summary of patent**

This patent seeks to improve power dispatching flow by using an anomaly detection algorithm known as isolation forest. This type of algorithm involves detecting anomalies by determining how far a data point is from the rest of the data. To do this, the patent describes splitting the data into n-samples, or sub-forests. The sub-forests are tested using a base forest abnormal detector to help identify where anomalies exist. The input of the algorithm in the patent is various data regarding the power dispatching flow, such as CPU occupancy rate and memory occupancy rate. The output of the algorithm will be a numerical range. It is with this numerical range that the algorithm will determine whether there is an anomaly. If the range reaches a certain threshold, then it will be classified as an anomaly. A key part of this algorithm is determining what is an anomalous score. To do this, the algorithm is trained using historical data. Finding anomalies in the data will help increase efficiency and prevent failures.

**D.** **How the patent related to our project**

This patent seeks to improve power dispatching flow by using an anomaly detection algorithm known as isolation forest. This type of algorithm involves detecting anomalies by determining how far a data point is from the rest of the data. To do this, the patent describes splitting the data into n-samples, or sub-forests. The sub-forests are tested using a base forest abnormal detector to help identify where anomalies exist. The input of the algorithm in the patent is various data regarding the power dispatching flow, such as CPU occupancy rate and memory occupancy rate. The output of the algorithm will be a numerical range. It is with this numerical range that the algorithm will determine whether there is an anomaly. If the range reaches a certain threshold, then it will be classified as an anomaly. A key part of this algorithm is determining what is an anomalous score. To do this, the algorithm is trained using historical data. Finding anomalies in the data will help increase efficiency and prevent failures.

**Patent 2**

**A.** **Team member reviewing patent:** *Oasis Husband*

**B.** **Pertinent patent information:**

Number: CN107657288B

Title: Power dispatching flow data anomaly detection method based on isolated forest algorithm

Inventor(s): 宁文元高欣郭子明贾庆轩李新鹏马龙穆永铮李军良徐建航

Assignee:

Date filed: 10/26/17

Date granted: 07/03/2020

Number of claims: 3

**C.** **Summary of patent**

This patent seeks to improve power dispatching flow by using an anomaly detection algorithm known as isolation forest. This type of algorithm involves detecting anomalies by determining how far a data point is from the rest of the data. To do this, the patent describes splitting the data into n-samples, or sub-forests. The sub-forests are tested using a base forest abnormal detector to help identify where anomalies exist. The input of the algorithm in the patent is various data regarding the power dispatching flow, such as CPU occupancy rate and memory occupancy rate. The output of the algorithm will be a numerical range. It is with this numerical range that the algorithm will determine whether there is an anomaly. If the range reaches a certain threshold, then it will be classified as an anomaly. A key part of this algorithm is determining what is an anomalous score. To do this, the algorithm is trained using historical data. Finding anomalies in the data will help increase efficiency and prevent failures.

**D.** **How the patent related to our project**

This patent seeks to improve power dispatching flow by using an anomaly detection algorithm known as isolation forest. This type of algorithm involves detecting anomalies by determining how far a data point is from the rest of the data. To do this, the patent describes splitting the data into n-samples, or sub-forests. The sub-forests are tested using a base forest abnormal detector to help identify where anomalies exist. The input of the algorithm in the patent is various data regarding the power dispatching flow, such as CPU occupancy rate and memory occupancy rate. The output of the algorithm will be a numerical range. It is with this numerical range that the algorithm will determine whether there is an anomaly. If the range reaches a certain threshold, then it will be classified as an anomaly. A key part of this algorithm is determining what is an anomalous score. To do this, the algorithm is trained using historical data. Finding anomalies in the data will help increase efficiency and prevent failures.

**Patent 3**

**A.** **Team member reviewing patent:** *Michael Hollinger*

**B.** **Pertinent patent information:**

Number: US10662474B2

Title: Identification of polymorphic sequences in mixtures of genomic DNA by whole genome sequencing

Inventor(s): Richard P. Rava

Assignee: Verinata Health Inc

Date filed: 1/19/2011

Date granted: 05/26/2020

Number of claims: 10

**C.** **Summary of patent**

This patent proposes via its claims a method for identifying numerous polymorphisms within the genome of a human fetus using blood samples obtained from pregnant women. These polymorphisms are comprised of cfDNA from both a fetal and maternal genome. The patent seeks to perform whole genome sequencing on various portions of this mixture in order to identify polymorphisms that are said to be unenriched as a form of prenatal screening and diagnosis.

**D.** **How the patent related to our project**

This patent shows obvious parallels to the project proposed by group 9, which seeks to identify anomalies in similar genome sequences comprised of RNA. Classification C12Q1/6827 proposes hybridization assays for the detection of mutations and polymorphisms within the genome sequence, which is not entirely dissimilar to the proposed solution of group 9. Similarly, classification C12Q2535/122 proposes a solution using massive parallel sequencing, which would be one of the prerequisites to group 9’s project proposal. Similarly, the patent and its respective invention use similar data visualization techniques using MATLAB/Numpy to plot the polymorphic sequences on a 2D axis.

**Patent 4**

**A.** **Team member reviewing patent:** *Avigail Martinez*

**B.** **Pertinent patent information:**

Number: US20160370961A1

Title: Organization, visualization and utilization of genomic data on electronic devices

Inventor(s): Patrick Merel

Assignee: Portabale Genomics Inc.

Date filed: 06/21/16

Number of claims: 29

**C.** **Summary of patent**

Here the methods, devices as well as systems for simple organization, visualization, and use of genome data on electronic devices have been described. In some of the embodiments, the data are organized and visualized as per digital music contents. This concept defines a new procedure for genomic data organization. This also facilitates the development of genomic data visualization tools. The methods that have been described here can certainly be implemented with some consumer-oriented software on electronic devices, computers as well as portable devices. One embodiment that has been included provides a graphical user interface (GUI) for the purpose of displaying genomic information on a mobile device. This GUI comprises a listing of phenotypic traits, diseases or combinations, a listing of genes, and much more. This has been reviewed because it is highly related to the topic under investigation. Visualization and Analysis of Genomic Data need that the idea about genomic data is known. The concept as already mentioned allows for a new procedure for the organization of genomic data and this can further enhance our thoughts on detecting anomalies in genomic data.

**D.** **How the patent related to our project**

The literature review conducted suggests many different methods in which genomic data can be visualized and analyzed but the method we propose with Python will be better in many ways. Firstly, with PyOD being the Python Outlier Detection Model, a variety of outlier detection models can be made use of. The model thus developed can be trained by defining the ID of the same. The fraction parameter is indicative of the number of outliers that are present in the dataset. The default value for this is 0.05 which is indicative of the fact that the dataset has 5% of outliers. The development of the program can be easier as the existing models can serve as a guide as to what to add or what to discard. The advantage of using Python for the development of the program is that this can be well integrated into the existing solutions for instance there exists Python API for Xena Hub. Thus, in this way a compact solution can be developed that can prove to be better than the existing ones. The focus will be on researching the issues and the shortcomings of the present solutions so that a more robust program can be developed. There is no doubt that with the use of Python productivity will be enhanced and the support of the vast libraries will make it even better.

**Patent 5**

**A.** **Team member reviewing patent:** *Kamal Shrouder*

**B.** **Pertinent patent information:**

Number: CN107657288B

Title: Power dispatching flow data anomaly detection method based on isolated forest algorithm

Inventor(s): 宁文元高欣郭子明贾庆轩李新鹏马龙穆永铮李军良徐建航

Assignee: Beijing University of Posts and Telecommunications Beijing Kedong Electric

Power Control System Co Ltd State Grid Jibei Electric Power Co Ltd

Date filed: 10/26/17

Date granted: 07/03/2020

Number of claims: 3

**C.** **Summary of patent**

This patent seeks to improve power dispatching flow by using an anomaly detection algorithm known as isolation forest. This type of algorithm involves detecting anomalies by determining how far a data point is from the rest of the data. To do this, the patent describes splitting the data into n-samples, or sub-forests. The sub-forests are tested using a base forest abnormal detector to help identify where anomalies exist. The input of the algorithm in the patent is various data regarding the power dispatching flow, such as CPU occupancy rate and memory occupancy rate. The output of the algorithm will be a numerical range. It is with this numerical range that the algorithm will determine whether there is an anomaly. If the range reaches a certain threshold, then it will be classified as an anomaly. A key part of this algorithm is determining what is an anomalous score. To do this, the algorithm is trained using historical data. Finding anomalies in the data will help increase efficiency and prevent failures.

**D.** **How the patent related to our project**

This patent seeks to improve power dispatching flow by using an anomaly detection algorithm known as isolation forest. This type of algorithm involves detecting anomalies by determining how far a data point is from the rest of the data. To do this, the patent describes splitting the data into n-samples, or sub-forests. The sub-forests are tested using a base forest abnormal detector to help identify where anomalies exist. The input of the algorithm in the patent is various data regarding the power dispatching flow, such as CPU occupancy rate and memory occupancy rate. The output of the algorithm will be a numerical range. It is with this numerical range that the algorithm will determine whether there is an anomaly. If the range reaches a certain threshold, then it will be classified as an anomaly. A key part of this algorithm is determining what is an anomalous score. To do this, the algorithm is trained using historical data. Finding anomalies in the data will help increase efficiency and prevent failures.

(Authors: John Callaghan, Avigail Martinez, Michael Hollinger, Kamal Shroud, Oasis Husband)

**2.3 Alternative Solutions**

In order to find the anomalies within a given VCF file, it is required that an algorithm is made. For this project specifically, python programming was used but can also be converted to other programming languages such as C++ or Java. With all the algorithms out there that are available for Python specific coding, the question wasn’t, “What algorithm are we going to use?” The question was which algorithm would be better suited for what we are doing? So, after doing some research on the different anomaly detection algorithms, three algorithms jumped out as being very similar and better suited for this project. These three algorithms are Isolation Forest, SVM(Support Vector Machine), and the Local Outlier Factor Algorithms. The main algorithm that was agreed upon for this project was the Isolation Forest Algorithm. However, the more algorithms that we have in place, the better results that we have to compare with each other and provide more precise and accurate results.

(Author: Kamal Shrouder)

**2.4 Evaluation**

Isolation Forest: This algorithm is based on decision trees and because there are no predefined labels in the algorithm, it is considered an unsupervised model. In the eyes of this algorithm, an anomaly is a point of data that is considered “few and different” in respect to other data points. All the sample data that travels deeper into the tree are less likely to be an anomaly since they would require more cuts in order to isolate them as an anomaly.

SVM: The objective of the SVM algorithm is to create a hyperplane that separates the data points into two categories, anomalies and non-anomalies. This hyperplane is created based off of support vectors that are created based off of the data points.

Local Outlier Factor: In the eyes of this algorithm, when a point is considered as an outlier, it is based on its local neighborhood. Thus, it is considered a local outlier as opposed to an anomaly. Based on the density of the neighborhood, it will identify the outlier. This algorithm performs better when the density throughout the neighborhood is not the same throughout the whole dataset.

(Author: Kamal Shrouder)

**2.5 Decision**

After conducting research without and with our sample data files, we have determined that while the local outlier factor was a good candidate for the project, it yielded similar results to the isolation forest algorithm. After seeing this, the decision was to keep the isolation forest and the SVM algorithm due to their different style of results , but still very useful.

(Author: Kamal Shrouder)

# 3. Implementation Details

**3.1 System Specifications and Functionalities**

The system should be able to run on any type of operating system, provided the computing power of the system is sufficient. Upon benchmarking the project on numerous systems, it is recommended to use at least an Intel i5 processor or equivalent with 16 gigabytes of DDR4 RAM. Failure to meet these hardware expectations may result in a significantly-delayed processing time for the analysis of the VCF file.  
 Additionally, it is recommended to use a hard-wired ethernet connection when using the web application. A stable connection is integral to ensuring that the server does not time out while the program attempts to analyze a VCF. Such errors have been known to occur when benchmarking the program on various systems, and this precautionary detail is integral to the functionality of said program.

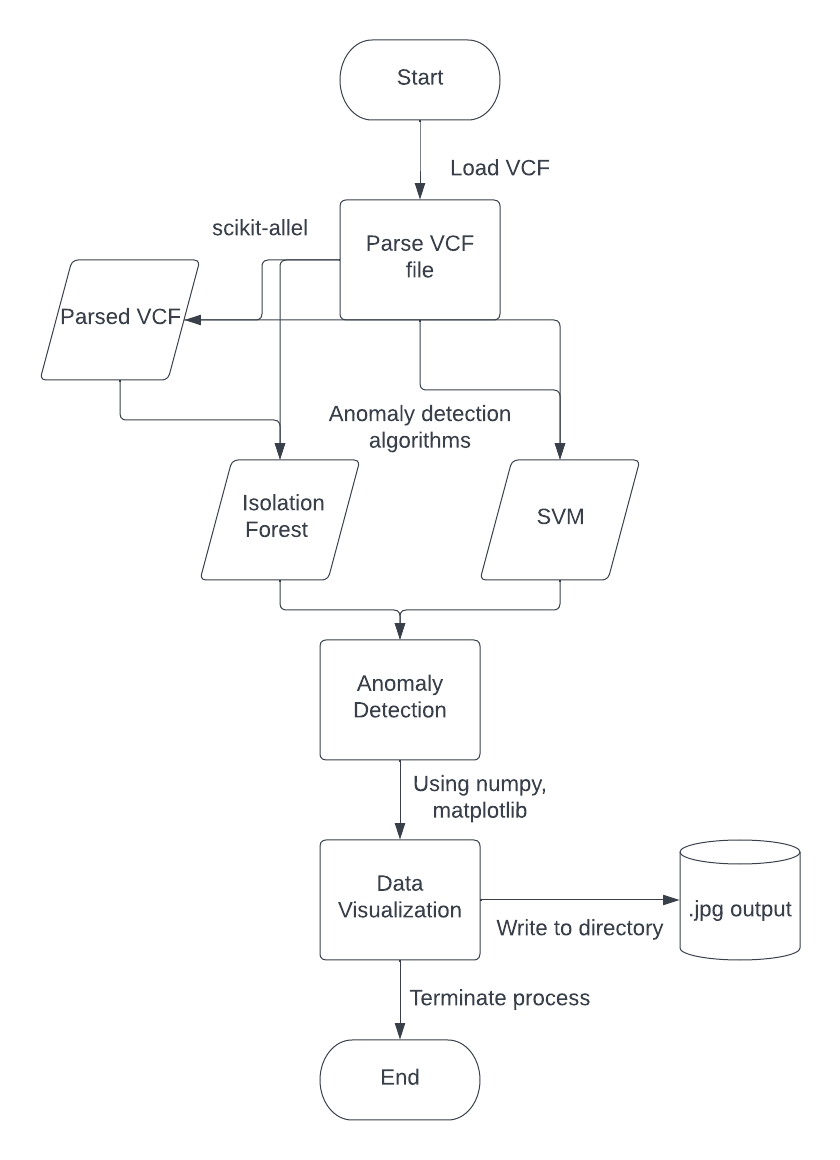
(Author: Michael Hollinger)

## 3.2 Overall System Design with Block Diagram

The design and implementation are split into three different pieces of the algorithm which will make up the software. The program will use the Python coding language. Notable packages used include NumPy, scikit-allel, Matplotlib, and pandas. To start, the program will import each of these packages. Then, the program must load the given VCF file into the environment. Doing this will require functions from the scikit-allel package. Once the VCF file has been properly loaded, the anomaly detection can begin. The anomaly detection algorithm and its accompanying visualizations will be implemented into an interactive web page where the end-user can upload their VCF file. Avigail Martinez and Michael Hollinger will be responsible for creating this user interface.

The anomaly detection algorithm focuses on two separate models; Isolation Forest and Support Vector Machine. Using two different models will help ensure accuracy in the outputs. The models each require different summary statistics to calculate an "anomaly score." The summary statistics used are mean, variance, kurtosis, and skewness. In order to calculate these summary statistics, a sliding window is used to cut the data into sections of 1000 data points each until the entirety of the data file is parsed. After the gathering of the summary statistics, the calculations are input into the anomaly detection models to find each section’s respective anomaly score. This anomaly score will be tested against different regions of the data file to determine if any regions contain clusters of anomalous sections. The key to anomaly detection will be balancing runtime and accuracy. Given the large size of the given input file, we must adhere to previously described runtime requirements. Once the anomaly detection portion of the code is complete, it will be output to the biologist through various visualizations which can then be downloaded. John Callaghan and Kamal Shrouder will be responsible for anomaly detection implementation.

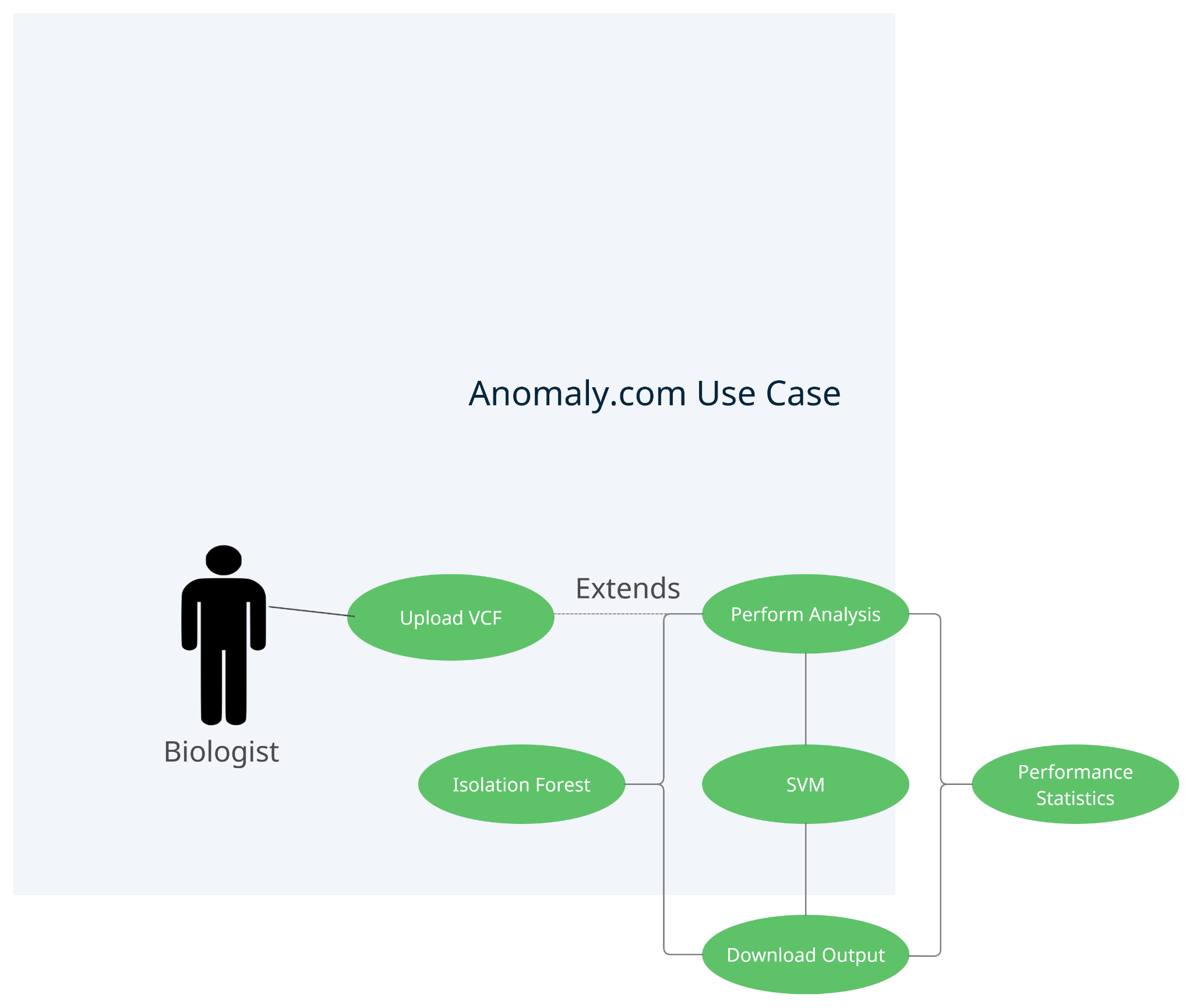
Data visualizations will comprise most of the output of the program. The key package used is Matplotlib. Each visualization returned will indicate areas of interest to the biologist. A baseline level of biological knowledge will be assumed from the user, so there is no need to explain terms such as alleles and chromosomes. The output is shown to the user through the web page itself, as well as through a downloadable option. Oasis Husband is the team leader with regard to data visualizations.



(Author: Michael Hollinger)

## 3.3 Circuit diagrams, Flowcharts and/or Use Case Diagrams

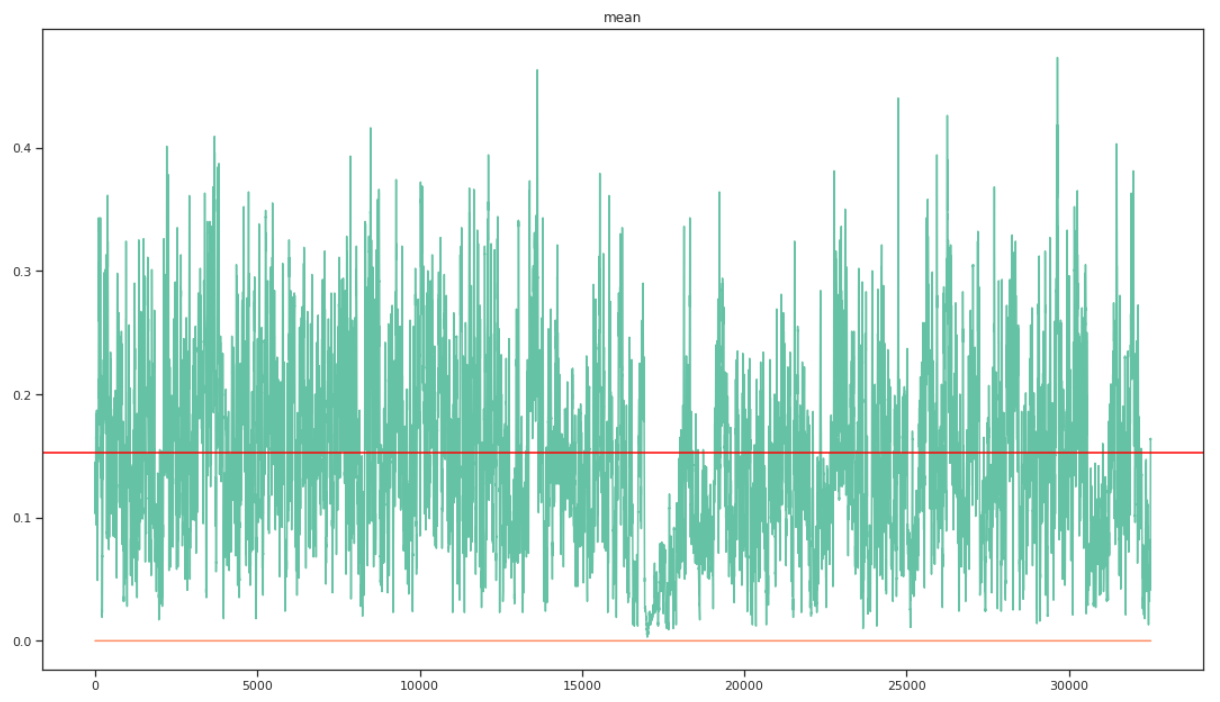
## Due to this design being primarily an algorithmic data analysis that is intentionally simple to use, the only use case scenario will be a user uploading their VCF file and downloading their output. The goal is to allow the end-user to receive their analysis without having to do any of the coding or parsing themselves, leading to a one-step use case diagram.

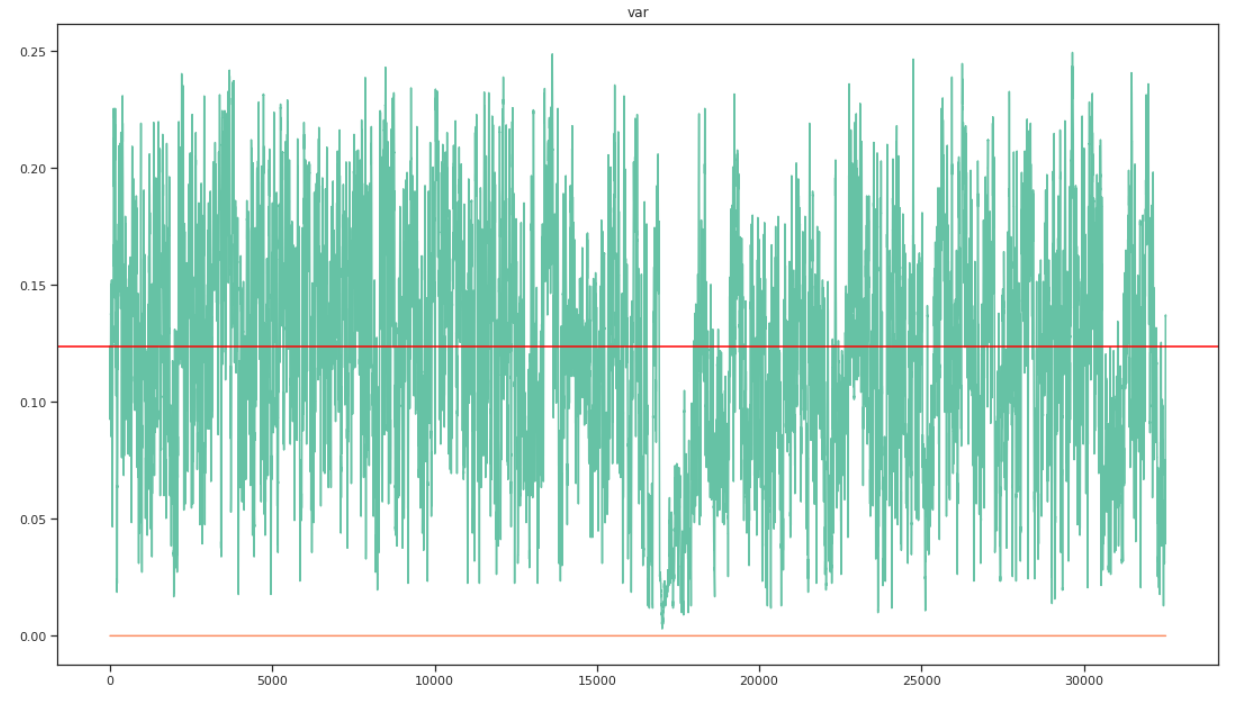


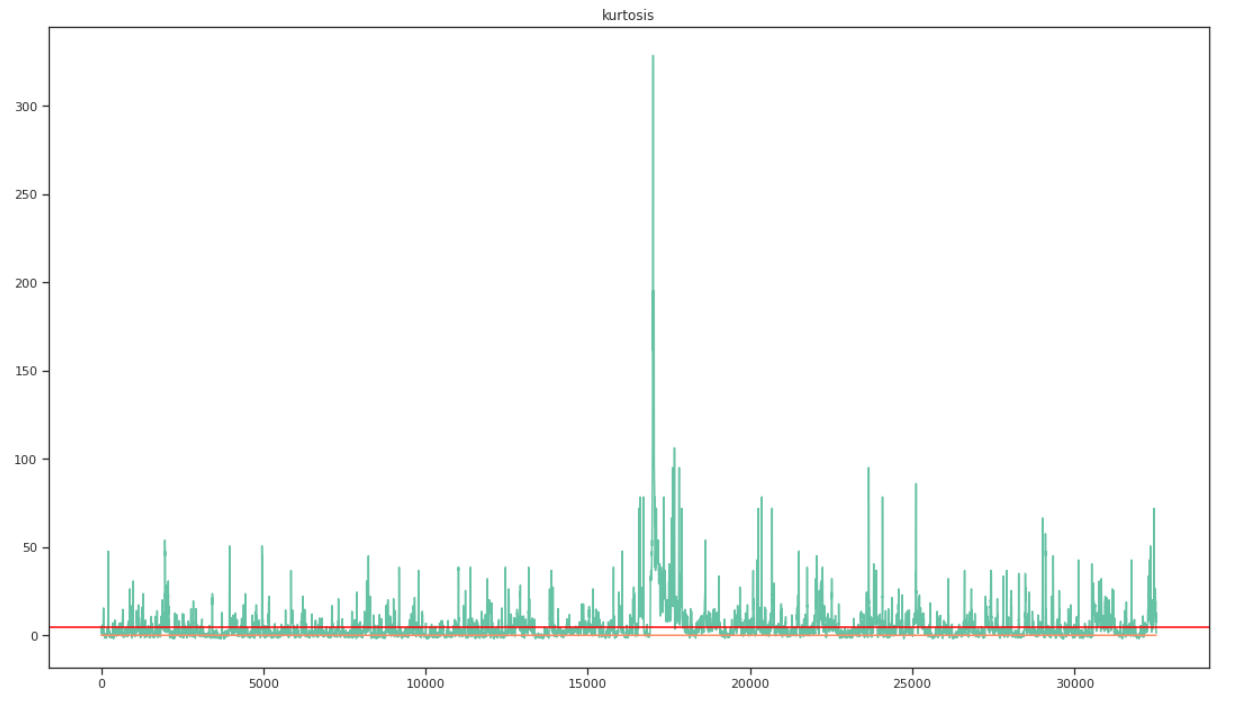
(Author: Michael Hollinger)

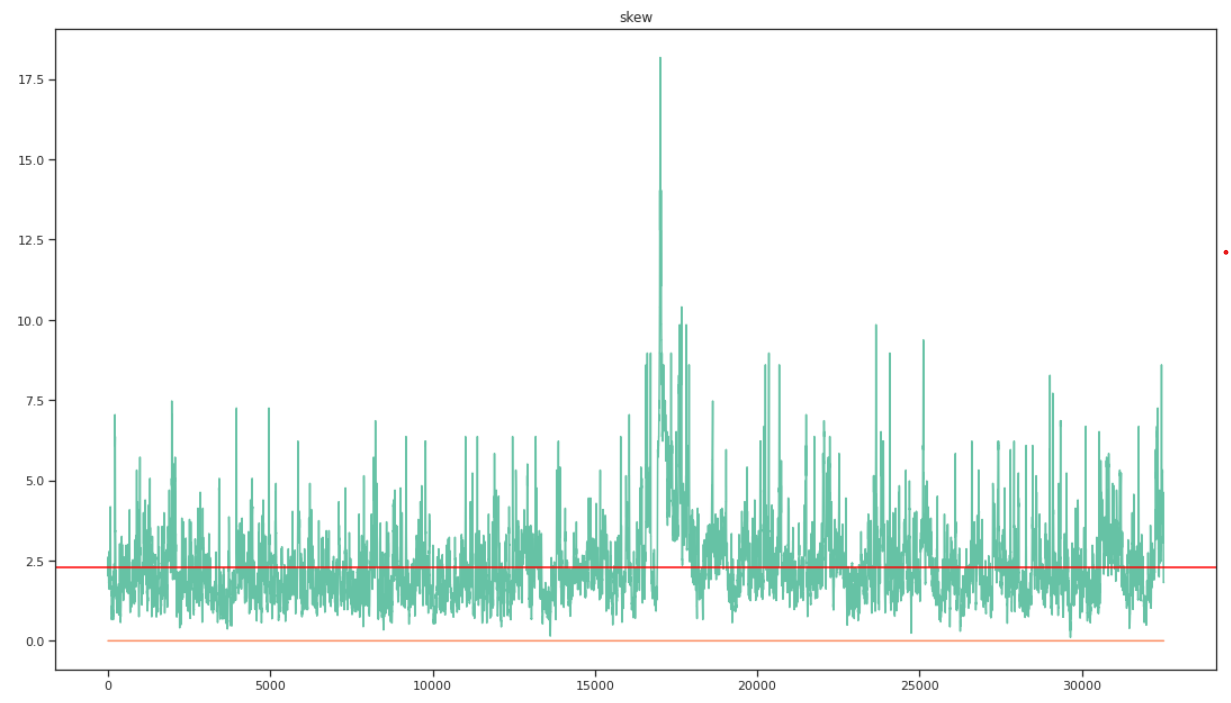
## 3.4 Testing and Calibration Results

Testing of the models is done using the “test\_sample” input file. In this file, there is a clear dip in the rate of heterozygosity in the center of the data. If our summary statistics and our models are accurate, they should return clear visualizations of this decrease. Examples of the summary statistic output for this test file can be seen below:

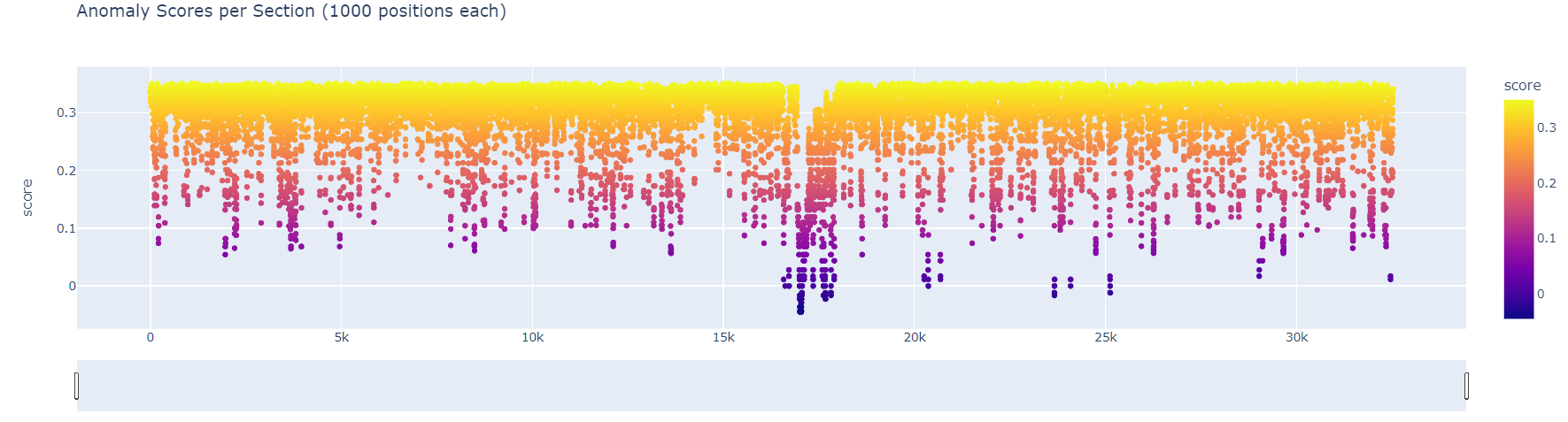








Clearly, there is an interesting area seen around the middle of the data file. This indicates that the summary statistics are able to find the desired anomaly. Once the summary statistics are deemed accurate, they are inserted into a data frame which is then input into the anomaly detection models. These models require certain variables, such as the expected anomaly rate, which must be optimized until the expected anomaly is returned. An ideal output of the anomaly detection model is shown below:



In this plot, it is clear to see where the cluster of anomalies exists. Also, it is possible to zoom in using the slider to examine the cluster more thoroughly. Finally, the anomaly score of each section is included in the plot to indicate which sections are the biggest outliers. Generally, the most accurate models were created with a low expected anomaly rate. This is because many of the chromosome files can have millions of data points. If a high number of data points are returned as anomalous, it can cause the true anomalies to be hidden. Only the clearest anomalous clusters should be returned to the biologist. Also, due to the number of data points, it is expected that at least a small number of sections will be deemed anomalous throughout the entire file. However, the most interesting areas in the data files are the regions that have a high concentration of anomalies clustered together. This would be the best indication of actual changes to a population, such as an evolutionary mutation.

(Author: John Callaghan)

**3.4 Discussions on Lessons Learned**

Throughout the implementation of this project, many of the difficulties encountered were due to the size and organization of the chromosome files. Extracting the changes in allele heterozygosity proved challenging because of the way VCF files are sorted. Numerous Python packages were tested to parse through the VCF files, though each had different sets of rules for and constraints for outputting the data needed for anomaly detection. Also, the size of the input files led to extremely long runtimes. The program taking up to an hour to run or even crashing was not uncommon. Trying to balance compilation time while maintaining accuracy was a key struggle throughout implementation.

One vital lesson learned was the importance of organization when designing and implementing an application such as this. Due to the constant testing and optimization of each part of the application, team members were forced to work separately despite the interrelation of each sub-system. These sub-systems include data parsing, anomaly detection, data visualization, and end-user interface. Constant communication with regard to the progress of each sub-system was necessary to the success of this project. Also, having strict deadlines for various key milestones was crucial to keeping up an ideal pace. Without these, completion of the analysis and implementation would have not been possible.

(Author: Avigail Martinez)

**4. Conclusions**

In the completion of this design, many milestones were achieved. One key achievement is the implementation of the isolation forest and SVM models. This required a full understanding of the biological information in the input files. It also required fully implementing a sliding window to analyze the files and gather the needed summary statistics. The models themselves also had to be constantly changed and optimized in order to output accurate and useful visualizations. Extensive research was done to not only determine applicable anomaly detection methods but also to understand the way they worked. Further research was needed to ensure the information being output would be useful to a biologist. This required constant collaboration between the team and our sponsor.

Another achievement in this design is the implementation of an end-user interface. Originally, the proposed design had just a simple command line interaction with the end-user. This was changed due to the belief that having a robust end-user interface would be a clear improvement to the usability of the algorithm. Despite challenges such as combining the front-end implementation with the anomaly detection algorithm and allowing the end-user to upload such a large file, the web page is now fully functional.

Despite adhering to the proposed goals, some improvements could still be made if the design were to be continued or replicated. The biggest improvement would be making the x-axis of the output plots the singular genomic position rather than each section. This would make the returned anomalies have more specific coordinates. The issue with this is anomaly scores are only found for each individual section and not for each singular data point. Changing this would lead to discrepancies in the size of the x-axis and the y-axis. Any solution would require altering the way the summary statistics are gathered, likely leading to extreme increases in compilation time. Much more testing would then be needed to ensure the end-user does not wait for multiple hours to receive their analysis.

(Author: John Callaghan)

**5. References**

Goldman MJ, Craft B, Hastie M, Repečka K, McDade F, Kamath A, Banerjee A, Luo Y, Rogers D, Brooks AN, Zhu J. Visualizing and interpreting cancer genomics data via the Xena platform. Nature biotechnology. 2020 Jun;38(6):675-8.

· Goldman M, Craft B, Hastie M, Repečka K, McDade F, Kamath A, Banerjee A, Luo Y, Rogers D, Brooks AN, Zhu J. The UCSC Xena platform for public and private cancer genomics data visualization and interpretation. biorxiv. 2019 Jan 1:326470.

· Buels R, Yao E, Diesh CM, Hayes RD, Munoz-Torres M, Helt G, Goodstein DM, Elsik CG, Lewis SE, Stein L, Holmes IH. JBrowse: a dynamic web platform for genome visualization and analysis. Genome biology. 2016 Dec;17(1):1-2.

· Hofmeister BT, Schmitz RJ. Enhanced JBrowse plugins for epigenomics data visualization. BMC bioinformatics. 2018 Dec;19(1):1-6.

· Gu Z, Eils R, Schlesner M. Complex heatmaps reveal patterns and correlations in multidimensional genomic data. Bioinformatics. 2016 Sep 15;32(18):2847-9.

· Gu Z, Eils R, Schlesner M, Ishaque N. EnrichedHeatmap: an R/Bioconductor package for comprehensive visualization of genomic signal associations. BMC genomics. 2018 Dec;19(1):1-7.

· Merel P. Organization, visualization and utilization of genomic data on electronic devices. Pub. No. US 2016/0370961 Dec. 22, 2016.

· Ethically Aligned Design | standards.ieee.org Ethically Aligned Design - IEEE SA - the IEEE Standards Association - Home. 2019. https://standards.ieee.org/wpcontent/uploads/import/documents/other/ead\_v2.pdf.

· Rosencrance, Linda. “What It Ops Needs to KNOW ABOUT ANOMALY DETECTION: Better Security and Ops.” TechBeacon, TechBeacon, 22 Jan. 2019, https://techbeacon.com/enterprise-it/what-it-ops-needs-know-about-anomaly-detection-better-security-ops#:~:text=The%20benefits%20of%20anomaly%20detection,when%20using%20traditional %20security%20methods.

**6. Personnel/Organization Chart**

| **Role** | **Personnel** |
| --- | --- |
| Project Sponsor | Dr. Michael DeGiorgio |
| Head Supervisor | Dr. Hanqi Zhuang |
| End-User Interface | Michael Hollinger, Avigail Martinez |
| Anomaly Detection Algorithm | Kamal Shrouder, John Callaghan |
| Data Visualization | Oasis Husband |

**End-User Interface** - Design and implement a web page that allows the end-user to upload VCF files and receive output from the anomaly detection algorithm.

* Web Page must be aesthetically pleasing
* Must act as a connection between each individual part of the design (visualizations and anomaly detection)
* Allow users to download output directly to their device.
* Must indicate an error message to the user if a non-VCF file is uploaded

**Anomaly Detection Algorithm -** Design and implement an algorithm that locates anomalous regions in input data files.

* Implement a sliding window to analyze input file fully
* Accurately calculate summary statistics using Heterozygousity rate
* Create summary statistic data frame to initialize isolation forest and SVM models
* Return anomalous genomic data points and sections

**Data Visualizations -** Design and create multiple different visualizations to aid in the understanding of anomalous regions.

* Visualizations should be understandable to a theoretical biologist
* Determine what information is vital to show in visualizations
* Plot anomaly score for each individual section
* Allow user to zoom into interesting regions

(Authors: Oasis Husband, Avigail Martinez)