Software Development for Genome Sequence Analysis

David Farr & Dr. Alison Scoville

Department of Biological Sciences, Central Washington University
Ellensburg, WA

ABSTRACT

The cost of genome sequencing has decreased rapidly, expanding availability for many biological applications (Muir 2016). For example, researchers can now obtain genome sequences from multiple populations under different types of selection. Comparison of these sequences allows for identification of chromosome regions and specific genes associated with adaptive evolution (Kelly 2013). As an increasing number of researchers engage in this type of inquiry, many have created in-house computer scripts to analyze the raw sequence data (e.g., Kelly 2013), creating a gap in both continuity and standardization.

Using a test dataset and preliminary results from an ongoing artificial selection experiment in *Mimulus guttatus* (Yellow Monkeyflower), I translated, verified, and expanded five software programs representing stages of a single analysis into one software package written in the C# programming language. This program is helping researchers to streamline their analysis and increase precision, while remaining dynamic enough that it can be expanded to any like-set of data, regardless of species.

OBJECTIVES

1. Consolidate and update five individual genetic analysis programs into a single software package.
2. Determine the feasibility of development in the C# software language for genetic sequence analysis.
3. Use the single software package to analyze variant-call format pooled genomic data from populations of *Mimulus guttatus* (Yellow Monkeyflower) under selection for high and low trichome production and identify chromosome regions involved in formation of this trait.

Figure 1 Photograph of the Yellow Monkeyflower.

METHODS

Objective 1: Five separate programs were selected that in successive order process raw variant-call format (VCF) data sourced from genome sequencing of populations experiencing different selection regimes. The major programs include:

1. A Python implementation of VCF parsing and analysis using the statistics proposed in Figure 2 and Figure 3 (Kelly 2013).
2. An R script implementing the GenWin package to identify optimal breakpoints from using spline analysis of the Python output.
3. An R script to identify statistically significant genomic regions from the Python output, based on a chosen False Discovery Rate.
4. A Java program to compare output of the previous programs with annotated genomic sequence data.

Objective 2: Microsoft's C# using .Net framework 4.7 was used initially in a Mac Xamarin environment followed by Visual Studio 2015 Community Edition on Microsoft Windows 10 due to Mac complications with 64-bit development in .Net. No C# equivalent of the GenWin package currently exists. To utilize GenWin, the latest version of RdotNet packagage for C# was used to synchronously pipeline the program through an existing R installation and finish in the C# environment.

Objective 3: A data set consisting of a VCF file with genomic BAM data from pooled, replicate control and treatment populations was run through the software package. The software obtained B values for each SNP, used spline analysis in GenWin to determine the optimal median SNP window size, and then used this window size to obtain B* and P-values based on comparison to a chi-square distribution. Genomic windows that were significantly associated with selection for high trichome production were identified using an FDR of 0.05. An updated version of the Java program served as an adjunct to C# to compare the results to annotated genome sequence data. These results were reported to the principal investigators in order to advance ongoing research.

Figure 2 The formula for the B test statistic for divergent selection, where *I* represents the index individual SNP, *S* represents the number of SNPs in the window, and *d* represents standardized divergence in allele frequencies between populations.

$$B = \sum_{i=1}^{S} d_i^2$$

Figure 3 The formula for the B* test statistic, which has a tractable distribution that can be used to determine p-value. B* is calculated using *delta* degrees of freedom using a chi-square distribution with a calculated Bowley skew, *sigma* standard deviation derived from the interquartile range, B from Figure 2, and *S*, the number of SNPs in the window.

$$B^* = \delta + \sqrt{2\delta}\left(\frac{B-S}{\sigma(B)}\right)$$

RESULTS

The combination of the C# program and the Java adjunct for the specialized QTL analysis comparison were completed. Window size for final B* analysis reflected the median window size obtained through GenWin spline analysis across the genome.

The individual values for calculation of test statistics in Figure 2 and Figure 3 were compared to the original values reported from the Python implementation and were accurate to four significant figures of the given test statistics. The resulting data in comma-separated values (CSV) format were provided to researchers at Central Washington University for their ongoing use and interpretation.

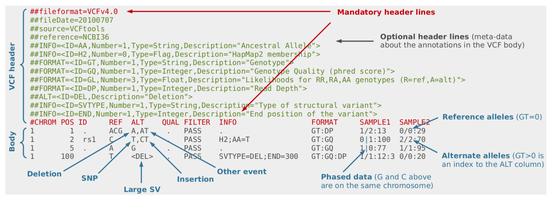
Development barriers with .Net framework 64-bit vs. 32-bit limitations for Mac and Linux environment may inhibit analysis of very large (> 4GB) VCF files outside of the Windows environment. (Xamarin 2017)

This project is accessible at https://github.com/davidfarr/mg-gap (not currently packaged for public release).

DISCUSSION & CONCLUSIONS

- **This software provides a novel implementation of five individual programs that can be used to identify genetic regions of interest based upon differences among populations subject to control and treatment selection.**
- **C# is a viable environment for genome sequence analysis; however, it is not ideal for data-heavy usage in Mac or Linux environments due to processor architecture limitations.**
- **Analysis using a single median SNP window based on GenWin spline results eliminates arbitrary selection for window size.**
- **Median window size may not provide ideal breaks compared to using sliding window sizes, as proposed in the original Beissinger implementation. However, it allows for calculation of p-values for formal hypothesis testing for each genome segment.**

Figure 5 Annotated example of VCF file design and interpretation.



DISCUSSION & CONCLUSIONS (CONT.)

The software developed here provides an informative and useful tool for detecting regions of chromosomes that may be responsible for phenotypic traits. Although this individual implementation uses project-specific RNASeq and QTL data, the overall package is designed to be expanded to implement the comparison of significant SNP variants against a reference genome.

As the users of sequence analysis products grow and process increasingly frequent and complex data sets, the need to develop and maintain software that not only is adaptable enough to handle entirely different types of organisms as well as standard enough to provide statistically sound results has become imperative.

Future development of this method will include capability to compare against reference genomic databases and handle the complex challenge of calculating P-values for any user-selected comparison of populations. The current method proposes use of a median window that eliminates random or investigator-biased selection of window size and uses the most appropriate window size in terms of all the population SNP variance.

REFERENCES

- Beissinger, Timothy M., Rosa, Guilherme, JM., Kaeppler, Shawn M., Gianola, Daniel, de Leon, Natalia. "Defining window-boundaries for genomic analyses using smoothing spline techniques". 2015. *Genetics Selection Evolution* (2015) 47:30.
- Kelly, John K., Koseva, Boryana, Mojica, Julius P. "The Genomic Signal of Partial Sweeps in *Mimulus guttatus*". 2013. *Genome Biology and Evolution* 5(8): 1457-1469.
- Benjamini, Yoav, and Yekutieli, Daniel. "Quantitative Trait Loci Analysis Using the False Discovery Rate". 2005. *Genetics*. (171)2 783-790.
- Muir, Paul et. al. "The real cost of sequencing: scaling computation to keep pace with data generation". 2016. *Genome Biology.* (2016)17:53.
- Xamarin. "32/64 bit Platform Considerations". Accessed 5/8/2017. https://developer.xamarin.com/guides/cross-platform/macios/32-and-64/

MEDIA

- Gaither, James. "*Mimulus guttatus*, the yellow monkey flower." 2013. *Joint Genome Institute*. Accessed 5/8/2017. http://jgi.doe.gov/monkey-flower-see-monkey-flower-do-model-plants-legacy-highlights-gene-shuffling-hotspots/
- Universitat Politècnica de València. "SNP Calling". Image. Accessed 5/8/2017. https://bioinf.comav.upv.es/courses/sequence_analysis/snp_calling.html

ACKNOWLEDGMENTS

- Dr. Alison Scoville, who supported my research as a faculty mentor at Central Washington University Department of Biological Sciences.
- Dr. John Kelly, a critical source of information in translating his original Python work, as well as an excellent host at Kansas University Department of Ecology and Evolutionary Biology.
- Central Washington University College of the Sciences (COTS) for their financial support through a Research Grant used to develop the Python translation on-site at Kansas University.
- Central Washington University Office of Undergraduate Research (OUR) for their financial support through a Travel grant for use to attend and present at the Evolution Meetings 2017 conference in Portland, OR.