A bioinformatics workflow for detecting signatures of selection in genomic data

Murray Cadzow[1,2], James Boocock[1,2], Hoang Tan Nguyen[1,2],
Phillip Wilcox[2,3], Tony R Merriman[1] and Michael A Black[1]

[1]Department of Biochemistry, University of Otago
[2]Department of Mathematics and Statistics, University of Otago
[3]Scion Research, Rotorua, New Zealand

September 9, 2013

Contents

1 Introduction

Mik to fill this in

2 Getting Started

2.1 Prerequisites

The selection pipeline requires some basic libraries and tools are installed on your system. These are python3, Zlib and others.

2.2 Installation

To install the package standalone, requiring manual configuration of the config file, run this as root.

```
./install.sh --standalone
```

The rest of this section will be dedicated to the automatic installation. To perform an automatic installation of the selection pipeline run as root.

```
./install.sh
```

Installation creates a default config file located in the base directory of the pipeline. Installation adds a program called selection_pipeline to the system path. To test the program is installed correctly run the following command at a terminal prompt.

```
selection_pipeline -h
```

The pipeline uses VCF tools, VCF tools requires a perl library to work correctly. Assuming you are using bash as your default shell open your .bashrc file in your home directory and append to the PERL5LIB environment variable.

```
# Change the path to match the location of your selection pipeline bin folder.
export PERL5LIB=\${PERL5LIB}:/home/sfk/selection_pipeline/lib/perl5
```

To avoid having to change the default config file append the same folder to you PATH environment variable.

```
export PATH=\${PATH}:/home/sfk/selection_pipeline/bin
```

2.3 Genetic Maps and Impute Haplotypes

To use the phasing and imputation features of the pipeline requires both genetic map files and haplotype files. For humans these files that conform to the format required for shapeit and impute2 can be found here. Download and extract the reference files of your choice and extract them. To use the files with the selection pipeline requires setting a few options in the config file you will use these are as follows an example config file is avaliable in the base directory of the selection pipeline.

```
[shapeit]
genetic_map_dir = # genetic map file directory #
genetic_map_prefix = # full file name with ? where chromosome number changes #
[impute_executable]
impute_map_dir= # genetic map file directory location #
impute_reference_dir = # impute reference directory location
impute_map_prefix = # the full file name with a ? for where the chromosome number is #
impute_reference_prefix = # The file name with a ? for chromosome name and the extension
```

3 Tutorial

3.1 Selection Signatures at the Lactase Locus

3.1.1 Getting the Data

The lactase gene is located on Chromosome 2 between 136,545,410-136,594,750 positions. For the example we will use a 10 megabase region containing the Lactase gene and the CEU and YRI populations from the 1000 genomes. Usually iHS would be done chromosome wide but in order to demonstrate how to use the pipeline we will use the chromosome 2 region 130,000,000-140,000,000. To download the example dataset enter the command below.

```
wget http://tutorial_file_location.com
```

Extract the data into a new folder

3.2 Setting up Pipeline Run

To perform the analysis the first step will is to navigate to the folder you extracted the tutorial data.

3.3 Population Files

Population files are required for any cross population comparisions. The commands below will initiate the data generation step.

```
multipopultions -p POP1.txt -p POP2.txt -
```

The generated folders and current folder have all the data required to perform further selection analysis.

4 Command line Arguments

The selection pipeline contains

5 Multipopulation

6 Selection Pipeline

7 Configuration File

The selection pipeline requires a configuration file, by default the program looks in the current working directory for a file named defaults.cfg but you can point the program to any file using command line arguments. There are two main programs in selection pipeline namely *selection_pipeline* and *multi_population*. These programs share a config file but certain configuration parameters can be ommitted when using the *selection_pipeline* program exclusively. A clean install of the program generates an example configuration file containing default arguments for all the compulsory parameters.

7.1 Argument Description

8 Extra Features

8.1 Genome Wide Selection Scans

The script is designed to work on one chromosome starting with a VCF file containing one or more populations. The pipeline was designed to be applied genome wide in the genome_wide/ directory we have provided a script that can perform a genome wide analysis. Read the README in this folder for more information.

8.2 Ancestral Allele Annotation

Needs to be fixed to take any set of human fasta files as the ancestral allele combination.

8.3 Galaxy Intergration

The galaxy folder contains the scripts required to add the selection pipeline to your local galaxy installation. The pipeline is also avaliable on the galaxy toolshed at galaxy_url