# User Manual for GIGI-Pick v0.9

Author:
Charles Y K Cheung [cykc@uw.edu]
Department of Biostatistics
University of Washington

Last Modified on 2/6/2014

## Contents

## Introduction

GIGI-Pick is a computer program to evaluate sequencing choices and to suggest who to select for sequencing in pedigrees. We assume that a common goal is to collect sequence data on subjects in pedigrees to identify risk allele(s), and to accomplish this goal, we seek to sequence a subset of subjects in the pedigrees and analyze data using imputed genotypes. To select subjects, GIGI-Pick can leverage the location of the evidence of linkage and existing markers from linkage analysis to more effectively target subjects for sequencing. This situation usually arises when linkage analysis has already suggested a chromosomal region to prioritize. Alternatively, GIGI-Pick leverages the pedigree structure to suggest subjects for sequencing, which may be more suitable in situations when multiple phenotypes are of interest or when prior evidence of linkage is not available.

## Disclaimer and future directions

We are still actively performing quality checks on the program, editing the program code, and improving the documentation. We will post new, dated, versions of the documentation and example files soon, as the unfinished sections are completed. Also, for users who expect to be using inheritance-vector information to better optimize their subject selection, note that the current version of GIGI-Pick requires version 2.9 of MORGAN, which is not the most recent MORGAN release (currently 3.2). This will be fixed in the near future.

## Citing GIGI-Pick

Cheung CYK, Blue EM, Wijsman EM (2014) A statistical framework to guide sequencing choices in pedigrees. American Journal of Human Genetics 94:257-267.

## Software URL

GIGI-Pick is developed under the **linux** environment. The program is available for download at http://faculty.washington.edu/wijsman/software.shtml

## Files in GIGI-Pick software distribution

GIGI-Pick software code and its dependency files - the Mersenne random number generator

- example folder

## Installing GIGI-Pick

Simply unzip the files, navigate to the code directory, and type

```
make
```

If make does not work, go to the GIGI-Pick.cpp's directory and install the program by

```
g++ GIGI-Pick.cpp -o GIGI-Pick
```

## Running GIGI-Pick

GIGI-Pick accepts a parameter file. To run GIGI-Pick, type

```
./GIGI-Pick -param=<parameter file> <options>
```

To run the example file, go to the main GIGI-Pick program directory, and type

```
./GIGI-Pick
```

Then, type "Example" without the quotation and press enter.

Users can run GIGI-Pick using the "interactive" mode or the automatic subject selection "batch" mode. The interactive mode allows users to have extensive interaction with the program to evaluate different design choices. This mode is assumed by default. To run the batch mode instead, use the "-batch=#".

e.g.

```
./GIGI-Pick -param=selectionExample/param.txt   -batch=10
```

to ask GIGI-Pick to select 10 subjects for sequencing.

## Options

[The flags are case-sensitive.]

| Flag | Purpose |
|---|---|
| -param=FILE | To specify the path of the parameter file. |
| -outD=DIRECTORY | To specify the absolute directory path of where the output files will be created. If this flag is missing, the output files will be saved to the user's current directory. |
| -m=# | To specify the parameter value for the joint-prioritized selection algorithm. (default: m=3). If m=1, the joint-prioritized selection algorithm reduces to forward-stepwise selection. The GIGI-Pick paper refers to this parameter as γ. |
| -MAF=# | To specify the minor allele frequency between 0 and 0.5 used to calculate coverage. The GIGI-Pick paper refers to this parameter as α. |
| -batch=# | To run GIGI-Pick as the "batch" mode. # specifies the number of subjects to select |

## Two Use Cases

GIGI-Pick uses a metric called “coverage” to compare sequencing choices. The g*enome-wide coverage* measures the coverage at a random locus in the genome and the *local coverage* measures the coverage at the position of interest in a specified region.

A) Genome-wide coverage may be desired if prior information about a candidate chromosomal region is not available or if multiple trait phenotypes are collected on a pedigree. To compute the genome-wide coverage, GIGI-Pick only requires a pedigree structure file.

B) Local coverage may be desired if the user intends to prioritize the search of causal variants on a specific region on a chromosome. To compute the local coverage, GIGI-Pick needs the sampled inheritance vectors (IVs) file from **gl_auto** and other corresponding files.

## GIGI-Pick’s Parameter File

The parameter file tells GIGI-Pick where to look for the required files. For case A, GIGI-Pick only minimally requires one file that describes the pedigree structure. For case B, GIGI-Pick minimally requires a pedigree meiosis file that user prepares from the output of gl_auto, an IVs file (from gl_auto), and a file specifying the map positions of framework markers.
An **example** of the parameter file is found in the example directory under **selectionExample/param.txt**

In GIGI-Pick_v1.00, the parameter file is organized as follows:

**pedigree=**selectionExample/ped52.pedMeio
**frameworkIV=**selectionExample/frameworkPanel.IV
**numIV=**100
**frameworkMap=**selectionExample/framework.map
**samplingPoints=**selectionExample/dense_48_52cM.map
**otherPedigreeInformation=**selectionExample/ped52_can_need_select.txt

For Case A, only **pedigree** is required. **otherPedigreeInformation** is optional.
For Case B, **pedigree, frameworkIV, numIV, frameworkMap,** and **samplingPoints** are required. **otherPedigreeInformation** is optional.

Notes:
I suggest using the absolute paths of the filenames instead of relative paths. A relative path is relative to the directory containing the executable program. (The parameter file in the example folder is created using a relative path.)

# File Formats

***Examples of these files are provided in the selectionExample directory [refer to the param.txt for the filename of these files]***

*Case A.*

**pedigree:**

The pedigree file should either be the **ordered** MORGAN-compatible pedigree structure OR the pedigree meiosis file. Please refer to Case B for the description of the latter file.

An example of the MORGAN-compatible pedigree structure file:

**example:**

```
input pedigree size 6
input pedigree record names 3 integers 2
input pedigree record trait 1 integer 2
*****
101 0 0 1 0
102 0 0 2 0
201 101 102 1 0
202 101 102 2 0
2010 0 0 2 0
301 201 2010 1 0
```

The first 3 lines contains information describing the pedigree. The 4th line is simply five *.
In the remaining lines, the first column is the subject ID, the second column is the ID of the subject's father, the third column is the ID of the subject's mother, the fourth column is the sex (1=male, 2=female), and the fifth column is the trait (integer value – just put 0).

Note:

1. The pedigree data must begin on the 5th line.
2. The pedigree data **must be ordered** such that **ancestors must preceed descendants**. For example, subject 101 and 102 must be specified before 201 and 202. Subject 201 and 2010 must be specified before subject 301.

Please refer to
http://www.stat.washington.edu/thompson/Genepi/MORGAN/Morgan.shtml for more information.

*Case B.*

**pedigree:**

Here, the pedigree refers to the pedigree meiosis file obtained from gl_auto's console output. The pedigree meiosis specifies the pedigree structure and the index of meioses that GIGI-Pick needs to use to read in the corresponding Inheritance Vectors. GIGI-Pick imputes genotypes on subjects specified in this pedigree. Please refer to the **Inferring IVs using gl_auto** section for creation of this file.

**frameworkIV:**
The Inheritance Vectors file describes the descent pattern of chromosomes at the positions of the framework markers. It is the output file that **gl_auto** generates. The content of this file is generated by gl_auto. When you run gl_auto, you should instruct gl_auto to print Meiosis Indicators instead of Founder Genome Labels.

**numIV:**
This corresponds to the number of samples that the user actually prints to the the frameworkIV file.

**frameworkMap**: map positions of framework markers
The marker map positions of the framework markers file is a text-file which contains the map distance in centi-Morgans (cM) based on the Haldane map function. Markers must be ordered in ascending order and consistent with the order used in gl_auto. Each line contains the position of a marker.

```
position of Marker1
position of Marker2
position of Marker3
...
position of MarkerN
```

eg

```
1.0
2.0
3.0
4.0
...
```

**samplingPoints:**
Region on the chromosome to sample in Haldane map position in centi-Morgan. For example, if the bounding intervals of the region of interest are 48 and 52, the file would be a single line containing the two numbers:

e.g.

```
48 52
```

**otherPedigreeInformation:**
This file specifies additional details about the pedigree. This file contains three columns. The first column is the subject ID. The second column specifies who are available for sequencing in the pedigree (0=no, 1=yes). The third column specifies who have already been chosen for sequencing (0=not pre-chosen, 1=pre-chosen).

e.g.

```
 101 0 0
 102 1 0
 201 1 0
 202 1 1
 2010 0 0
 301 1 0
…
```

In this example, subject 102, 201, 202, and 301 are available for sequencing, and subject 202 has been pre-chosen for sequencing.

## Output files

**TO-BE-COMPLETED.**

## Inferring IVs using gl_auto for Case B

The first step to impute genotypes is to use framework markers to infer IVs. Framework markers are a relatively sparse set of markers that are used to infer IVs on a chromosome of interest. For this purpose, we use gl_auto, a program in the MORGAN package that is freely available at http://www.stat.washington.edu/thompson/Genepi/MORGAN/Morgan.shtml.

Please note that the version of MORGAN that GIGI-Pick currently supports is version 2.9, which is not the latest version of MORGAN. We will make GIGI-Pick compatible with MORGAN version 3.2 in the near future.

To infer IVs in gl_auto, we need to supply the required files in MORGAN format:

(1) Pedigree file
(2) Marker file: this is a composite file that contains the map positions of framework markers (in centiMorgans  assuming the Haldane map function), allele frequencies of framework markers , and genotype data of framework markers
(3) Parameter file used to run gl_auto

Refer to the documentation of MORGAN for guidance on setting up these files and on running gl_auto.

Example files used to infer IVs using gl_auto are included under the “example/gl_auto_example” directory.

**Overview**
From running gl_auto, we need to obtain 2 files:

(a) Framework IVs file: GIGI-Pick uses this file, which contains IVs at framework positions
(b) Pedigree-meiosis file: this file contains the information about the structure of the pedigree on which we want to im
(c) pute genotypes. This file is different from the pedigree file used in gl_auto. In addition to the pedigree structure, this file also contains information that GIGI-Pick needs to determine how the Inheritance Vectors (Meiosis indicators) are organized (i.e. the i$^{th}$ line of the meisosis indicator belongs to which subject in the pedigree and whether this meiosis indicator is this person’s maternal or paternal chromosome). We need to create this file from the console output of gl_auto.

## Obtaining the files

### (a) Framework IVs file

In gl_auto's parameter file, we must instruct gl_auto to display the output as Meiosis Indicators instead of Founder Genome Labels. Please make sure we use the option

**output meiosis indicators**

instead of "output founder genome labels" in the control/parameter file of gl_auto.

### (b) Pedigree-Meiosis file

We need to create the pedigree-meiosis file (b) from the console output of gl_auto. It is very easy to make this file.
When we run gl_auto, the program prints a huge amount of output to the console. This console output actually contains the content of the pedigree-meiosis file that we need to extract.

1. In order to extract this content, we first need to direct the console output to a file by using the ">" directive so we can subsequently extract the content from this file.
   ie. **./gl_auto gl_auto_parameter_file > glauto_console_output.txt**

2. Then, we extract the pedigree-meiosis content from the console output to a new file. To simplify the creation of this file, use the Perl script "extractPedMeiosis.pl"
   - **Usage: perl extractPedMeiosis.pl glauto_console_output.txt FILENAME_PED_MEIO**
     - We need to have Perl installed in linux.
     - assuming glauto_console_output.txt is in the same directory as extractPedMeiosis.pl
   - Alternatively, this file can also be easily created by the user. See Appendix B for the creation of this file manually and also for an example of how this file looks like.

## Other Information

1. The coverage metric does not differentiate between subjects who are vs. are not likely to carry a variant that affects the trait. Branches of a large pedigree that are thought to be uninteresting should be designated as unavailable for sequencing, in order to focus attention on the pedigree components that may carry variants of interest to the trait. It can also be useful to manually select at least one subject with an unusual phenotype of interest to start with.

2. For users who expect to be using inheritance-vector information to better optimize their subject selection, note that the current version of GIGI-Pick requires version 2.9 of MORGAN, which is not the most recent MORGAN release (currently 3.2). This will be fixed in the near future.

3. A future topic is likely to add other metrics for subject selection.

## Appendix A

Creating the Pedigree-Meiosis file manually from the console output of gl_auto.

Using a text editor, we open the console_output.txt and fetch the line that begins with
**“name      name.pa      name.ma Compnt pat.meio mat.meio”**
We copy this line and table below. We paste this table to another file and save it.
The file includes the header line and looks like this:

```
 name      name.pa      name.ma Compnt pat.meio mat.meio
        2100_6            0             0       1      0      0
       2100_21            0             0       1      0      0
       2100_25            0             0       1      0      0
       2100_29            0             0       1      0      0
       2100_31            0             0       1      0      0
       2100_39            0             0       1      0      0
…
      2100_907            0             0       1      0      0
      2100_908      2100_901      2100_902      1      2      1
      2100_909            0             0       1      0      0
      2100_910      2100_901      2100_902      1      4      3
      2100_911      2100_901      2100_902      1      6      5
      2100_915      2100_907      2100_908      1      8      7
```
**...(until the end of table)**

## ChangeLog

The ChangeLog describes the recent changes to the program. It is located in the root directory of the uncompressed file.

## License

GIGI-Pick is free software: you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.
This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

There is NO WARRANTY for the program, to the extent permitted by applicable law.   In no event unless required by applicable law will any GIGI-Pick copyright holder be liable to you for damages, including any general, special, incidental or consequential damages arising out of the use or inability to use the program (including but not limited to loss of data or data being rendered inaccurate or losses sustained by you or third parties or a failure of the program to operate with any other programs).

## Acknowledgement

We thank Elizabeth Thompson for providing valuable comments. This research was supported by funding from National Institutes of Health grants R37GM046255, P50AG05136, R01AG039700, R01MH094293, and R00AG040184.