

## NAME

genepool – analyze genotyping data from pooled genomic DNA

## SYNOPSIS

`gpcommand` [ **options** ]

`gpextract` [ **options** ]

`gpanalyze` [ **options** ]

## DESCRIPTION

GenePool is a software package that provides analysis tools for the detection of shifts in relative allele frequency between pooled genomic DNA from cases and controls using SNP-based genotyping microarrays. GenePool is currently Affymetrix-centric however development efforts are underway to add the ability to incorporate data from other platforms including Illumina.

The GenePool system consists of two executable programs, **gpextract**(1) and **gpanalyze**(1), and one perl script **gpcommand**(1):

**gpextract** uses the Affymetrix Fusion SDK library to extract intensity values from Affymetrix CEL files and write them to a customized, more compact binary file format. The new files have the same name as the original CEL file but with the string *.gpb* (GenePool Binary) appended to each filename.

**gpanalyze** takes the intensity values from the *.gpb* files and uses a variety of data analysis methods to assign a score to each SNP where the score indicates how significant is the observed difference in allele frequency between the hybridizations for the two DNA pools. Note that the scores are *not* p-values.

**gpcommand** is a perl script that helps users run basic pooling analyses by reading a configuration file and automatically invoking the other two GenePool programs. New GenePool users should almost certainly start with **gpcommand** and move on to direct use of **gpextract** and **gpanalyze** once they are confident that they understand the system.

### Dependencies

#### Affymetrix Fusion SDK.

The **gpextract** program uses the Affymetrix Fusion SDK library to read the native CEL and CDF files so GenePool users will have to get a copy of the Fusion SDK source code if they want to compile GenePool from source code. This adds a dependency to the GenePool system but saves us having to create and maintain code to read all of the different types and versions of Affymetrix files. The Fusion SDK is written in C++ which is why **gpextract** is written in C++ while **gpanalyze** is written in C which the GenePool developers are more comfortable with.

#### Apache Xerces XML Parser

The Affymetrix Fusion SDK relies on the "C" version of the Apache Xerces XML Parser so this is effectively also a dependency for GenePool. If in future we determine that GenePool is not using any Affymetrix code that uses Xerces, we may be able to remove Xerces as a dependency for GenePool.

## INSTALLATION

There are two distributions for GenePool - binary and source - and each has its own installation instructions. In general, if a binary distribution is available for your machine architecture and operating system (Intel x86 Linux, PowerPC Mac, SPARC Solaris etc) then your easiest option is to try the binary distribution first. If the binaries don't work for you then please contact us with the details so we can try to remedy the problem. If there is not a suitable binary distribution for you or you would like to get the best possible performance from GenePool then you should probably try installing from source code.

### Binary Installation

1. Obtain a copy of the GenePool binary distribution file that matches your operating system and architecture and uncompress it. A command something like:  
**gunzip -c GenePool-bin-linux-0.2.0.tar.gz | tar xvf -**  
should work on most unix machines.
2. Execute **./configure** to configure the local installation ready for installation. If you do not have permissions to install programs and man pages into the **/usr/local/** directory then you will need to

specify an alternative install location using a command of the form **./configure --prefix=/your/alternate/path**.

3. There is nothing to build so you can skip the usual "make" step and go straight to executing **make install** to install the executables and man pages.

You should now be ready to GenePool. Two caveats - to run the executables, wherever you installed the executables (**/usr/local/bin** by default) needs to be in your PATH environment variable; and to see the man-pages, wherever you installed the man pages (**/usr/local/man** by default) needs to be in your MANPATH. If you cant run the programs or see the man pages then you may need to have your systems administrator help you set up your PATH and MANPATH environment variables.

### Source Code Installation

To compile GenePool from source, you will need 3 source code distributions - GenePool, Affymetrix Fusion SDK, and Apache Xerces XML parser (C-version). Full instructions for obtaining each is given below. If you are not compiling on Linux, you may also need a copy of the GNU Autotools (autoconf and automake) so that you can construct a valid **configure** script (see step 3 below).

At some point during the Affymetrix download process, you will be required to log into the Affymetrix Developer Nextwork (ADN) which means you must register for an ADN account if you havent already. Registering for the ADN is free and is probably a good idea if you are regularly analyzing Affymetrix data since the ADN pages contain useful data files and software as well as forums where Affymetrix software developers will answer questions. You will also have to accept the license terms for the Affymetrix Fusion library or the download will be blocked.

1. Obtain a copy of the GenePool source code distribution file and uncompress it. A command something like:  
**gunzip -c GenePool-0.0.2.tar.gz | tar xvf -**  
should work on most unix machines.
2. Obtain and compile a copy of the Affymetrix Fusion SDK and the "C" version of the Apache xerces XML processor as detailed in the **Compiling the Affymetrix Fusion SDK** section below. The Fusion and xerces source code will be used to create a Fusion library file (**libfusion.a**) and we will need that library plus the original Fusion and xerces header files (.h) during compilation and linking of the GenePool executables **gpextract** and **gpanalyze**.
3. The GenePool source code distribution contains a **configure** script but it was built on a Linux box so if you are compiling on any other platform you may need to regenerate this script to have it correctly tailored to your platform. To do this you will need a copy of **autoconf** and **automake** which are part of the GNU Autotools. Assuming you have these tools installed, all you need to do is execute **autoreconf** which will read the configure.ac file and regenerate **configure**.
4. Execute **./configure** to configure the local installation ready for compilation. If you do not have permissions to install programs and man pages into the **/usr/local/** directory then you will need to specify an alternative install location using a command of the form **./configure --prefix=/your/alternate/path**.
5. Execute **make** to compile and link the source code.
6. Execute **make install** to install the executables and man pages.

You should now be ready to GenePool. Two caveats - to run the executables, wherever you installed the executables (**/usr/local/bin** by default) needs to be in your PATH environment variable; and to see the man-pages, wherever you installed the man pages (**/usr/local/man** by default) needs to be in your MANPATH. If you cant run the programs or see the man pages then you may need to have your systems administrator help you set up your PATH and MANPATH environment variables.

### Compiling the Affymetrix Fusion SDK

You should already have completed step 1 of the **Source Code Installation** section above so you should already have a directory containing the uncompressed source code for GenePool. To create a compiled Affymetrix Fusion SDK library ready for linking with the GenePool source code:

1. Go to the Affymetrix website (<http://www.affymetrix.com>), and click on the *Support* tab at the top of the page. On the next page, click on the *Developer Network* link from the menu of links on the left side of the page. This will take you to the home page of the Affymetrix Developer Network (ADN) where a link to the Fusion SDK is available. When you reach the download page, you'll want the "Full SDK".
2. Copy the Fusion SDK distribution file (usually called something like **affy-fusion-release-107.zip**) *inside* the GenePool source code directory and unzip it. This should create a directory called **affy/** in which case you can safely skip step 3. If unzipping the fusion distribution creates a directory called **cvs-head** or any name other than **affy/** then you will need to do step 3.
3. Edit the `SDK_DIR` variable in **Makefile.FusionSDK** so that it points to the "root" of the Affymetrix Fusion code that was uncompressed in step 2. The "root" directory is called **sdk/** and it should contain a heap of subdirectories including **calvin\_files/**, **files/**, and **file\_formats/**. You may have to browse through the fusion distribution to find the **sdk/** directory.
4. Go to the Apache Xerces XML parser website (<http://xerces.apache.org>), and click on the *Xerces C* link in the menu on the left margin of the page. On the next page, click on the *Download* link in the menu on the left margin of the page. You should now be on the Download page for the C version of the Xerces XML parser so scroll down until you find a section titled *Current Source Releases of Xerces-C*. You can download the *.zip* or *.tar.gz* file but we'll assume you took the *.tar.gz* version.
5. Place the Xerces distribution file (usually called **xerces-c-current.tar.gz**) *inside* the GenePool source code directory and uncompress it. A command something like:  
**gunzip -c xerces-c-current.tar.gz | tar xvf -**  
should work on most unix machines.
6. Edit the `XERCES_ROOT` variable in **Makefile.FusionSDK** so that it points to the "root" of the xerces-c code that was uncompressed in step 5. The xerces directory name usually incorporates the version number (for example **xerces-c-src\_2\_7\_0/**) so you are almost certainly going to have to edit the default xerces directory that appears in **Makefile.FusionSDK**.
7. Execute **make --file=Makefile.FusionSDK** which will compile and link the Fusion and xerces code and create a **libfusion.a** library that we can link **gpextract** and **gpanalyze** against. This process could take up to 10 minutes depending upon the power of your CPU.

## FILE FORMATS

The GenePool binaries **gpextract** and **gpanalyze** generate and process many different plain txt files. This section provides a brief outline of the role and format of each of these files. Unless specified otherwise, all plain text files should be tab-delimited and should have Unix-style line endings - a single "LineFeed" character.

### 1. Experiment.txt

This file is read by **gpanalyze** and contains a line for each platform in the analysis. A platform is a chip type so a pooling experiment run on the Affymetrix 10K platform would have a single line but an experiment run on the Affymetrix 100K platform would have 2 lines - one for the Hind chips and one for the Xba chips. An Affymetrix 500K experiment would also have 2 lines - one for the Sty chips and one for the Nsp chips. All current Illumina HumanHap platforms are single chips not chipsets so Illumina-based experiments will only have a single line in **Experiment.txt**.

Each line has 5 items:

```
CasesFile NumCasesFile ControlsFile NumControlsFile SNPNames
```

where the description of each item is:

```
CasesFile      - file containing a list of Cases datafiles
NumCasesFile   - number of files listed in CasesFile
ControlsFile    - file containing a list of Controls datafiles
```

NumControlsFile - number of files listed in ControlsFile  
 SNPNames - file containing IDs of the SNPs on the chip

For example, an **Experiment.txt** file for an Affymetrix 500K pooling experiment might look like:

```
CasesStyFiles.txt 6 ControlStyFiles.txt 6 StySnpNames.txt
CasesNspFiles.txt 6 ControlNspFiles.txt 6 NspSnpNames.txt
```

## 2. CasesFiles

This file contains a list of the names of **gpextract** processed datafiles for Cases. This file is required by **gpanalyze** and its name is placed in the first column of the **Experiment.txt** file detailed above. If the file is not in the current directory then the filename should contain an absolute pathname. Each filename should be placed on a separate line as shown in the example below:

```
CasesNsp1.cel.gpb
CasesNsp2.cel.gpb
CasesNsp3.cel.gpb
CasesNsp4.cel.gpb
CasesNsp5.cel.gpb
```

## 3. ControlFiles

This structure of this file is identical to the **CasesFiles** file detailed above but the contents are a list of the **gpextract** processed datafiles for Controls. This file is required by **gpanalyze** and its name is placed in the third column of the **Experiment.txt** file detailed above.

## 4. SNPNames

This file contains information about each SNP on the platform. This file is required by **gpanalyze** and its name is placed in the fifth column of the **Experiment.txt** file detailed above. This file is generated differently for Affymetrix and Illumina chips since the SNPs appearing on an Affymetrix chip are predetermined whereas each Illumina chip may contain a slightly different number of SNPs since some SNPs may be represented by too few beads to be considered a valid measurement.

In the case of both Affymetrix and Illumina platforms the file contains the same 3 columns of data with one line for each SNP:

```
SNPName SerialNo DefaultRank
```

SerialNo *MUST* be UNIQUE for every SNPName. It is used to keep track of the order in which SNPs were extracted from the raw image intensity files. It is a 9 digit integer which allows for arrays with up to 999 million SNPs. By default, the DefaultRank field is set to 1 for each SNP in the SNPNames file. This field is used in a multistage analysis where **gpanalyze** creates a new SNPNames file for each analysis stage and populates the DefaultRank field with the rank of the SNP in that stage which allows subsequent stages to filter based on rank.

### Affymetrix

The order in which the SNPs occur in this file is critical to a successful analysis - they must be in **EXACTLY** the same order as the SNPs occur in the datafiles output by **gpextract**. Since the datafiles produced by **gpextract** are in binary format there is no way for a user to work out the order of the SNPs within the file so the user cannot expect to create the **SNPNames** file manually. Every **gpextract** run produces a **SNPNames** file in addition to the datafile and within a platform, the SNP order produced by **gpextract** is always the same so for each platform the user just has to use one of these **SNPNames** files produced by **gpextract**. Typically the filename includes the Enzyme type, for example: NspSnpNames.txt, HindSnpNames.txt XbaSnpNames.txt, etc.

### Illumina

For Illumina data, a SNP could be missing on one or more chips so the SNPs must be ordered in increasing numerical order which matches the order in which **gpextract** writes Illumina intensities into the *.gpb* binary datafile.

## 5. Output.txt

This is the analysis file output by **gpanalyze**. It contains 5 columns:

```
SNPName Score CaseValues ControlValues SerialNo
```

where *SNPName* is the identifier for the SNP, *Score* shows the degree of separation computed using the chosen analysis algorithm, *CaseValues* and *ControlValues* indicate how many case/control points were available, and *SerialNo* is same as in the description provided above for the **SNPNames** file.

Note that for Illumina, the values for *CaseValues* and *ControlValues* will be equal to the total number of beads on the cases and controls chips for this SNP whereas for Affymetrix, the values are calculated as being `NumberOfChips*NumberOfQuartets`.

## 6. SortedOutput.txt

The first five columns of this file are the same as for **Output.txt** but it has a 6th column which contains the rank. This file is sorted in descending numerical order on the score field.

## 7. AnnotationFile

The user can optionally supply this file which contains annotation information about the SNPs on the arrays. If supplied, it allows **gpanalyze** to produce annotated versions of the **Output.txt** file. The file contains 4 columns:

```
SNPId dbSNPId Chromosome Base
```

## 8. ChromosomeSortedAnnotated.txt

This output file is sorted ascending on chromosome (primary key) and basepair location (secondary key). The file has 5 columns:

```
SNPId Rank dbSNP Chromosome Basepair
```

## 9. ScoreSortedAnnotated.txt

This file is identical to the **ChromosomeSortedAnnotated.txt** file except that it is sorted ascending on Rank.

## 10. SlidingWindow.txt

This file is an extension of **ChromosomeSortedAnnotated.txt** with the addition of a number of fields related to the sliding window calculations performed on the ranks of a user-specified number of adjacent SNPs. The number of additional fields is  $(2 + n)$  where  $n$  is the number of SNPs in the sliding window which is determined by examining the window minimum and maximum values specified with the **-w** and **-W** command line options to **gpanalyze**. The extra columns are in the order:

```
AvgRank_Min AvgRank_(i) ... AvgRank_Max
```

## 11. KCorrectFile

This file is only useful for Affymetrix analyses. It contains average allele frequencies for AA, AB and AA calls for every probe quartet on a given chip and allows for the calculation of quartet-specific k-correction factors. Because the number of quartets differs between platforms, this file will contain a variable number of columns however the general pattern is:

```
SNPId NoOfQuartets Q1_AA Q1_AB Q1_BB ... QN_AA QN_AB QN_BB
```

At the time of this writing, it appears that there is no definitive formula/algorithm for k-correction factor and only a single RAS calculation formula has been implemented to include k-correction factors:

$k \cdot A / (A + B)$ . This RAS calculation method can be specified with the **-r 1** option to **gpanalyze** and the name of the file of k-correction factors is specified using the **-K** option. Sometimes it is not possible to provide values for AA, AB and BB for every quartet. The current implementation will look at only AB correction factor. An **\_EXTREMELY\_** important issue to remember is that the order in which SNPs are stored in the k-correction file **MUST** match that in the **SNPNames** file.

## KNOWN ISSUES

### Compiler limitations

The target compilation environment is gcc and a number of gcc-specific features are used including the arpg commandline argument processing system so GenePool may not compile on non-gcc C/C++ compilers.

### Data clipping

The original intensity data for each chip feature is stored in the Affymetrix CEL files as a 4-byte floating point number. In the TGen binary data files produced by **gpextract**, each of these intensity values has been converted into a 2-byte unsigned integer meaning that the intensity has been rounded to a whole number and that values above 65535 cannot be stored.

### perl location

The perl binary location on the first line of the gpcommand script is hardcoded to be `/usr/bin/perl` so if your perl is installed in a different place, you may have to edit the first line of the gpcommand script.

## TO DO

As soon as version 1 of GenePool is released, work will commence on version 2. There are a number of substantial improvements already planned for version 2:

### Unified experiment file

**gpanalyze** currently relies on an uncomfortable number of files created by **gpextract**. There are files to list the Case files, to list the Control files, to list the names and order of the SNPs in each extracted data file, and an Experiment file that names all of the other files. It's all too easy to imagine a user accidentally deleting or modifying a critical file. The next version of **gpextract** will create a single experiment file that contains all of the information currently spread through the multiple files.

### INI-style configuration file

**gpcommand** is driven by a configuration file in Windows INI-style format. This format is relatively easy for users to manipulate and provides a one-stop-shop definition of the pooling experiment. **gpextract** should be using this same configuration file to drive its operation rather than relying on command-line options. In combination with the Unified experiment file (see first TO DO item) the configuration file will vastly simplify GenePool - **gpextract** will read a configuration file and output a single experiment file, and **gpanalyze** will read the experiment file and accept a small range of commandline options to specify the required analysis.

### Unit tests

The prototype perl scripts that GenePool is based on had unit tests to ensure that as we changed and extended the programs, the underlying calculations were not impacted. We have manually checked GenePool results against the output from the prototypes however to maintain our sanity, we need to roll the unit testing feature forward into the C/C++ version.

### Quality metrics

We need "chips and SNPs" quality scores and a multi-level method to selectively exclude some chips/SNPs/samples from a given analysis. All exclusions effectively just drop SNP scores but at different levels: a SNP-level exclusion would drop all scores for a given SNP across all samples; a Sample-level exclusion would drop all SNPs for all chips for the given sample; and a chip-level exclusion would drop all SNPs on a given chip (i.e. some SNPs for a given sample).

## AUTHORS

John Pearson <jpearson@tgen.org>

David Craig <dcraig@tgen.org>

Matt Huentelman <mhuentelman@tgen.org>

Waibhav Tembe <wtembe@tgen.org>

Nils Homer <nhomer@tgen.org>

**SEE ALSO**

**gpextract(1), gpanalyze(1), gpcommand(1).**

**COPYRIGHT**

GenePool is copyright 2006 by The Translational Genomics Research Institute. All rights reserved. This License is limited to, and you may use the Software solely for, your own internal and non-commercial use for academic and research purposes. Without limiting the foregoing, you may not use the Software as part of, or in any way in connection with the production, marketing, sale or support of any commercial product or service. For commercial use, please contact [licensing@tgen.org](mailto:licensing@tgen.org). By installing this Software you are agreeing to the terms of the LICENSE file distributed with this software.

In any work or product derived from the use of this Software, proper attribution of the authors as the source of the software or data must be made. The following URL should be cited:

*<http://bioinformatics.tgen.org/software/genepool/>*