Genvisis Techmanual

Genvisis avails the visualization of genetic markers (the intensity, Log R Ration, or Theta) or sample (allele frequency), and navigates through different markers and samples at a fast pace. It aims to be a convenience tool for visual analysis or data Quality Control, as a complementary to the numerical based software, such as plink.

Genvisis is different from plink in that it provides that visual analysis or;

Genvisis is different Excel in that Excel cannot handle data having more than 200,000 rows, and cannot display and switch among different markers easily.

Genvisis is different from R code in that it lets users annotate and comment;

The basic procedure of using Genvisis is:

1. Initialize a new project:

???;

1. Parse Illumina/Infimax/csv/:

Load and parse micro array data (Create a new project);

1. Transpose Data:

Save the data into a marker-based format;

1. ~~Slim Plots:~~

~~An optional step to speed up the scatter plot for mid-size data~~

1. Plots or Tools:

Select the kind of analysis to run.

While there are graphical software being so popular today, Genvisis provides some features that are not easily found elsewhere.

Genvisis is a software product that Dr. Nathan Pankratz has developed for years. He started the project because he couldn’t find the right tool for his genetic research.

# Convention:

* Format: indentation and etc;
* Naming of variables, methods, and classes;
* Documentation.

# Data structure for files:

Genvisis uses Java Random Access File format to save data in bytes.

## Sample based files (.sampRAF):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0 ~ 3 | 4 | 5 ~ 8 | 9 ~ 16 | 17 ~ |  |
| int: # of markers | byte: nullStatus | int: length of the byte[] of Out Of Range Hashtable | long: marker fingerprint | byte[] of reduced precision data | byte[]: Out Of Range Hashtable |

Reduced Precision Data segment:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2 bytes | 2 bytes | 2 bytes | 2 bytes | 3 bytes | 1 byte |
| GC[sample1] | X[sample1] | Y[sample1] | BAF[sample1] | LRR[sample1] | Genotype[sample1]: AB + Forward |
| GC[sample2] | X[sample2] | Y[sample2] | BAF[sample2] | LRR[sample2] | Genotype[sample2]: AB + Forward |
| GC[sample3] | X[sample3] | Y[sample3] | BAF[sample3] | LRR[sample3] | Genotype[sample3]: AB + Forward |

…

## Marker based files (Transposed files or .mdRAF):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 ~ 3 | 4 ~ 7 | 8 | 9 ~ 16 | 17 ~ 20 | 21 ~ |  |  |  |
| int: # of samples | int: # of markers | byte: nullStatus | Long: sample fingerprint | int: length of the byte[] of String[] markername | String[]: markernames | byte[] of reduced precision data | int: size of Out Of Range Hashtable | byte[]: Out Of Range Hashtable |

Suggestion:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 ~ 3 | 4 ~ 7 | 8 | 9 ~ 16 | 17 ~ 20 | 21 ~ 24 | 25 ~ |  |  |
| int: # of samples | int: # of markers | byte: nullStatus | Long: sample fingerprint | int: length of the byte[] of String[] markername | int: size of Out Of Range Hashtable | byte[] of reduced precision data | String[]: markernames | byte[]: Out Of Range Hashtable |

## nullStatus (bitmap of the 1 byte flag):

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | GC | X | Y | BAF | LRR | AB Type | Forward Type |

## OutOfRange Hashtable at the tail of each Sample based file (.sampRAF):

|  |  |
| --- | --- |
| String | float |
| marker index + “\tx” | value |
| marker index + “\ty” | value |
| marker index + “\tlrr” | value |

## OutOfRange Hashtable at the tail of each MarkerData based file (Transposed file or .mdRAF):

|  |  |
| --- | --- |
| String | float |
| marker index in current file + “\t” + sample index + “\tx” | value |
| marker index in current file + “\t” + sample index + “\ty” | value |
| marker index in current file + “\t” + sample index + “\tlrr” | value |

## OutOfRange Hashtable in standalone outlier file (outliers.ser):

|  |  |
| --- | --- |
| String | float |
| marker index + “\t” + sample name + “\tx” | value |
| marker index + “\t” + sample name + “\ty” | value |
| marker index + “\t” + sample name + “\tlrr” | value |

## Reduced Precision format (current version):

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Actual Range (GEDI\_EXOME) | Reduced Precision bytes | | Reduced Precision Range | Reduced Precision OutOfRange Flag | Reduced Precision NaN |
| X, Y | scheme 1 | 0.000 ~ 230.791 | 2 bytes | | 0.000 ~ 65.533 | 65.534 {255,254} | 65.535 {255,255} |
| scheme 2 | -6.272 ~ 230.791 | 2 bytes | | -32.765 ~ 32.767 | -32.676 {255,254} | -32.677 {255,255} |
| GC, BAF | | 0.0000 ~ 1.0000 | 2 bytes | | 0.0000 ~ 1.0000 | N/A | 1.0002 {39, 18} |
| LRR | | -19.1824 ~ 13.5227 | 3bytes(18bits) | | -13.1070 ~ 13.1071 | -13.1071 {2, 0, 1} | -13.1072 {2, 0, 0} |
| Genotype(AB) | | 0 ~ 3, or -1 ~2 | 1 byte | 3 bit | -1 ~ 3 |  | N/A |
| Genotype(Forward) | | 0 ~ 20 | 5 bit | 0 ~ 29 | N/A | N/A |

Notes:

For AB Genotypes, - 1 means missing.

For Forward Genotypes, 0 means missing.

## TransposeData.java design thoughts:

sampleFile.read(sampleDataReadBuffer

Byte[ ] sampleDataReadBuffer

Markers / BufferChunk

bytes / SampMark

markerFile.write(markerDataReadBuffer[ j ])

Byte[ ][ ] markerDataWriteBuffer

Chunks / WriteBuffer

bytes / Marker

Markers / WriteBuffer

maxMarkfileSize

allSamplesProj

heapSpaceSize

Integer.maxValue

Markers / markerFile

Calculation of heap space usage:

|  |  |  |
| --- | --- | --- |
| heapspace | markerDataWriteBuffer | sampleDataReadBuffer |
|  |  | 247,519 markers \* 12 b = 2.83 mb |
| 16 gb | (16 gb – 2.83 mb) / (12 b \* 7680 samples) = 186,381 markers | 186,381 markers \* 12 b = 2.13 mb |
| 6 gb | (6 gb – 2.83 mb) / (12 b \* 7680 samples) = 69,872 markers | 69,872 markers \* 12 b = 0.80 mb |
| 2 gb | (2 gb – 2.83 mb) / (12 b \* 7680 samples) = 23,269 markers | 23,269 markers \* 12 b = 0.27 mb |
| 1 gb | (1 gb – 2.83 mb) / (12 b \* 7680 samples) = 11,618 markers | 11,618 markers \* 12 b = 0.13 mb |

# Backup System of Annotations and Cluster Filters

1. Temporary file (when in Scatter Plot):
   1. projectDir / data / 39810043.tempAnnotation.ser
   2. projectDir / data / 39810043.tempClusterFilters.ser
2. Permanent file (after Scatter Plot is closed):
   1. projectDir / data / clusterFilters.ser
   2. projectDir / data / annotations.ser
3. ~~Versioning of Permanent file:~~
   1. ~~if last time of update is over 2 weeks: start a new version, in case it is a new task on the project~~
   2. ~~else start a new version every~~
4. Export of permanent file:
   1. projectDir / annotations.xln;

projectDir / annotation\_m\_null.xln;

projectDir / annotation\_u\_null.xln

# Appendix

## Older versions of data structure of file storage

### For version 7b:

Sample based data:

|  |  |
| --- | --- |
| int: # of markers | byte[] of reduced precision data |

Marker based data:

|  |  |  |
| --- | --- | --- |
| int: # of bytes of String: markernames | String: markernames | byte[] of reduced precision data |

### For version 8:

Sample based data:

|  |  |  |  |
| --- | --- | --- | --- |
| int: # of markers | byte[] of reduced precision data | int: # of Out Of Range values | int[]: Out Of Range Array |

Marker based data:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| int: # of samples | int: # of markers | int: # of bytes of String: markername | String: markernames | byte[] of reduced precision data | int: # of Out Of Range Values | int[]: Out Of Range Array |

### For version 9:

Sample based data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| int: # of markers | long: marker fingerprint | byte[] of reduced precision data | int: size of Out Of Range Hashtable | HashTable: Out Of Range Array |

Marker based data:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| int: # of samples | int: # of markers | int: # of bytes of String: markername | String: markernames | byte[] of reduced precision data | int: size of Out Of Range Hashtable | int[]: Out Of Range Hashtable |

### For version 9b:

Sample based data:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| int: # of markers | byte: nullStatus | long: marker fingerprint | byte[] of reduced precision data | int: size of Out Of Range Hashtable | HashTable: Out Of Range Array |

### Reduced Precision format (older version):

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Actual Range (GEDI\_EXOME) | Reduced Precision bytes | Reduced Precision Range | Reduced Precision OutOfRange Flag | Reduced Precision NaN |
| X, Y | 0.000 ~ 230.791 | 2 bytes | 0.000 ~ 65.533 | 65.534 {255,254} | 65.535 {255,255} |
| GC, BAF | 0.0000 ~ 1.0000 | 2 bytes | 0.0000 ~ 1.0000 | N/A | 1.0002 {39, 18} |
| LRR | -19.1824 ~ 13.5227 | 18 bits | -13.1070 ~ 13.1071 | -13.1071 {2, 0, 1} | -13.1072 {2, 0, 0} |
| AB Genotypes | 0 ~ 3, or -1 ~2 | 3 bit | -1 ~ 3 |  | -2 -> N/A |
| Forward Genotypes | 0 ~ 20 | 5 bit | 0 ~ 29 | 30 -> N/A | 31 -> N/A |

### OutOfRange Hashtable at the tail of each MarkerData based file (older version) (Transposed file or .mdRAF):

|  |  |
| --- | --- |
| String | float |
| marker index + “\t” + sample index + “\tx” | value |
| marker index + “\t” + sample index + “\ty” | value |
| marker index + “\t” + sample index + “\tlrr” | value |

### OutOfRange Hashtable in standalone outlier file (older version) (outliers.ser):

|  |  |
| --- | --- |
| String | float |
| marker index + “\t” + sample index + “\tx” | value |
| marker index + “\t” + sample index + “\ty” | value |
| marker index + “\t” + sample index + “\tlrr” | value |