Genvisis Techmanual

Genvisis is a software package that provides the convenience of visualizing many gene markers and samples at a time, and navigates through them fast. It also has embedded tools for CNV analysis.

The basic steps of using it are:

1. Parse Illumina/Infimax/csv/:

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

1. Extract Plots:

Convert the data into a format that is

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 |

## 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 | 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 |  | N/A |
| Forward Genotypes | 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

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 |

# 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 |