

Gene Score Algorithm
Tod Casasent and Rong Yao
2018-09-04-1536

Overview

Some files used for Batch Effects are in terms of probes, and need to be converted to a measurement or score in terms of genes using HG18 or HG19 mappings. Dr. Rehan Akbani provided a PERL script for converting Copy Number and similar data into gene scores. The target audience for this document is consumers of Standardized Data who have an interest in how we standardized the data, in order confirm the data contains what they think it contains.

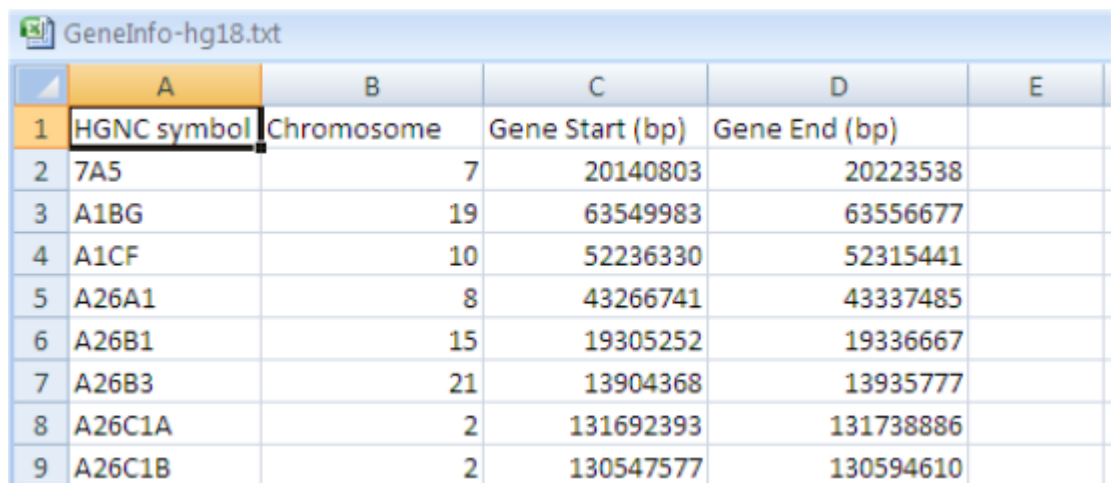
Data File Formats

HG18/HG19 Mapping File

This is a tab delimited file with four columns. The file has a header row containing:

- HGNC symbol
- Chromosome
- Gene start (bp)
- Gene end (bp)

The beginning of the file looks similar to this:



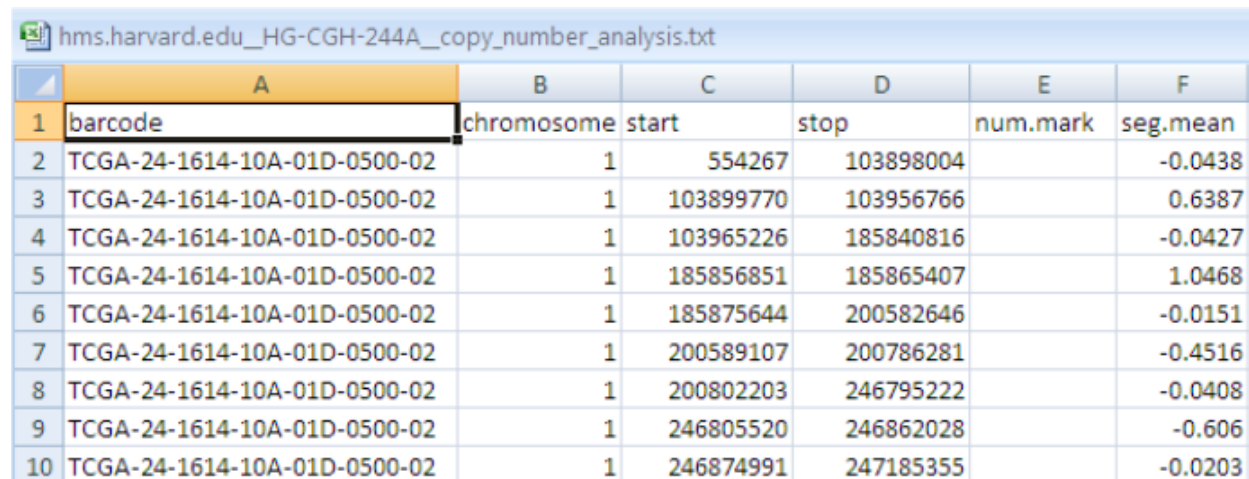
	A	B	C	D	E
1	HGNC symbol	Chromosome	Gene Start (bp)	Gene End (bp)	
2	7A5	7	20140803	20223538	
3	A1BG	19	63549983	63556677	
4	A1CF	10	52236330	52315441	
5	A26A1	8	43266741	43337485	
6	A26B1	15	19305252	19336667	
7	A26B3	21	13904368	13935777	
8	A26C1A	2	131692393	131738886	
9	A26C1B	2	130547577	130594610	

TCGA Data Access Matrix Copy Number (or Similar) File

This is a level 3 file downloaded from the Data Access Matrix and is tab delimited. The file has a header row containing:

- barcode
- chromosome
- start
- end
- num.mark
- segment mean

The beginning of the file looks similar to this:



	A	B	C	D	E	F
1	barcode	chromosome	start	stop	num.mark	seg.mean
2	TCGA-24-1614-10A-01D-0500-02	1	554267	103898004		-0.0438
3	TCGA-24-1614-10A-01D-0500-02	1	103899770	103956766		0.6387
4	TCGA-24-1614-10A-01D-0500-02	1	103965226	185840816		-0.0427
5	TCGA-24-1614-10A-01D-0500-02	1	185856851	185865407		1.0468
6	TCGA-24-1614-10A-01D-0500-02	1	185875644	200582646		-0.0151
7	TCGA-24-1614-10A-01D-0500-02	1	200589107	200786281		-0.4516
8	TCGA-24-1614-10A-01D-0500-02	1	200802203	246795222		-0.0408
9	TCGA-24-1614-10A-01D-0500-02	1	246805520	246862028		-0.606
10	TCGA-24-1614-10A-01D-0500-02	1	246874991	247185355		-0.0203

Gene Score (Output) File

The output file for Gene Scores is a tab delimited file. The first row has a column label “Barcode” and is followed by the gene ids. Subsequent rows have a sample id followed by gene scores.

The beginning of the file looks similar to this:

output											
	A	B	C	D	E	F	G	H	I	J	K
1	Barcode	OR4F5	SAMD11	NOC2L	KLHL17	PLEKHN1	HES4	ISG15	AGR1	C1orf159	TTL10
2	TCGA-01-0628-11A-01D-0360-02	0	-0.0869	-0.0869	-0.0869	-0.0869	-0.0869	-0.0869	-0.0869	-0.0869	-0.0869
3	TCGA-01-0630-11A-01D-0360-02	0	-0.1291	-0.1291	-0.1291	-0.1291	-0.1291	-0.1291	-0.1291	-0.1291	-0.1291
4	TCGA-01-0631-11A-01D-0360-02	0	-0.0834	-0.0834	-0.0834	-0.0834	-0.0834	-0.0834	-0.0834	-0.0834	-0.0834
5	TCGA-01-0633-11A-01D-0360-02	0	-0.0988	-0.0988	-0.0988	-0.0988	-0.0988	-0.0988	-0.0988	-0.0988	-0.0988
6	TCGA-01-0636-11A-01D-0360-02	0	-0.1072	-0.1072	-0.1072	-0.1072	-0.1072	-0.1072	-0.1072	-0.1072	-0.1072
7	TCGA-01-0637-11A-01D-0360-02	0	-0.0917	-0.0917	-0.0917	-0.0917	-0.0917	-0.0917	-0.0917	-0.0917	-0.0917
8	TCGA-01-0639-11A-01D-0360-02	0	-0.0981	-0.0981	-0.0981	-0.0981	-0.0981	-0.0981	-0.0981	-0.0981	-0.0981

Gene Score Algorithm

Read the mapping data from the HG18/HG19 Mapping File and sort the data by chromosome, and then by start location. (This is the @genes variable in the Perl file.)

Read the sample data from the Copy Number (or Similar) File and sort the data by sample id, then by chromosome, then by start location. (This is the @data variable in the Perl file.)

LOOP 1: While an index into the sorted sample data has not reached the end of the list, keep looping. (The index is represented by the \$d variable in the Perl file. Looping is done using \$d as an index into the sorted sample data.)

If the sample id for the index (\$d) is the same as the stored current sample id, go to the next row of sorted sample data. The sample id for the index will not match on the first pass, since the stored current sample id has not been set. (The index (\$d) may have been incremented during the previous pass through the loop.)

Set the current sample id.

LOOP 2: Loop through the sorted mapping data (which is a list of genes). (This is represented by the \$g variable in the Perl file, with the @gens variable containing the current row.)

Calculate the size of the gene from the mapping data. (This is gene end minus gene start plus 1.)

Get the current sample data row.

Set the score array to empty.

While

sample data row sample id is equal to current sample id AND
 chromosome for gene row is greater or equal to chromosome for
 sample data row AND
 (chromosome for gene row is not equal to chromosome for sample
 data row OR

gene start for gene row is greater than stop for sample row)
go to the next sample row

If (we ran out of data)
sample data row sample id is not equal to current sample id OR
we've gone through all the sample ids ($\$d \geq @data$) OR
the gene chromosome is less than the sample row chromosome
push the current score value into the score array, and set score value
to zero and go to next gene (LOOP 2).

If (chromosome is the same and gene ends before sample starts)
gene row gene end is less than sample start
push the current score value into the score array, and set score value
to zero and go to next gene (LOOP 2).

Now, we calculate the overlap and score for it.

The partsize is the sample stop minus the sample start plus 1.

The overlap is the sample stop minus the mapping gene start plus 1.

If the overlap is greater than the gene size, then the overlap is equal
to the gene size. (Overlap can't be bigger than gene being checked.)

If the overlap is greater than the partsize size, then the overlap is
equal to the partsize. (Overlap can't be bigger than the sample being
checked.)

If (gene actually starts in front of sample and ends during)
gene's gene start is less than sample start AND
gene's gene end is less than sample stop
overlap is gene's gene end minus sample start plus 1

~~The score is the score plus ((overlap divided by gene size) divided
by sample mean).~~

The score is the score plus ((overlap divided by gene size) multiplied
by sample mean).

If
gene's gene end is less than or equal to sample stop
push the current score value into the score array, and set score value
to zero

ELSE
decrement the gene index (we've gone one to far)
increment the sample index (we've checked the last one)

Go to next gene (LOOP 2)

Go to top of LOOP1 (probably next sample)