Tae-Min Kim, Loverace J. Luquette, Ruibin Xi, and Peter J Park, Jan/18/2011

Introduction to rSW-seq

rSW-seq is designed to identify CNVs between two genomes. Such a comparison can be made between tumor and matched normal genomes (i.e., conventional cancer genome analysis) or normal vs normal genomes (i.e., variation analysis). Unlike methods that use paired-end mapping such as BreakDancer (SVs in an individual genome of interest are identified with respect to the reference genome), rSW-seq uses 'read-depth signature' that does not required paired-end sequencing. rSW-seq directly identifies genomic segments with substantial copy number differences as measured by the ratio between the read counts between the two genomes without *a priori* determination of window/interval size. Genomic segments identified this way are candidates for copy number gain or loss. Another published read-depth signature-based algorithm, SegSeq is a point-centric algorithm in which the chromosomal breakpoints are identified first and used as the boundaries of copy-number segments (thus, the output is similar to that of Circular Binary Segmentation). For more details about algorithms and performance, please see the original paper (pubmed link).

Quick tutorial:

To use rSW-seq source code, the sequencing data of two genomes to be compared should be partitioned into separated files for individual chromosomes. For example, after mapping the sequencing reads onto reference genome using conventional alignment tool (e.g., Bowtie or BWA), the individual read information can be further separated according to their belonging chromosomes. If tumor and matched normal genomes are to be compared for chromosome 1 to X (human), there will be 46 chromosomal read files (e.g., tumor/normal × chr1 ~ X). The individual read files should contain a *single* data column - the chromosomal position of the individual reads (need not to be sorted *a prior*) without a header line. This is a simple structure (single data column), which is easy to generate by exporting 'chromosomal position' of mapped reads per chromosome. It is also possible to remove some reads mapped onto multiple genomic positions or those with poor sequencing quality; however, current rSW-seq code does not provide such preprocessing step.

To compile the source code:

```
gcc -msse3 sw-seq.c random.c vector.c audic.c -lm
```

The code should be executed per chromosome as following:

```
./a.out <tumor file> <normal file> [total tumor reads] [total normal reads]
[SW-score cutoff]

Ex) ./a.out tumor_chr1_reads.txt normal_chr1_reads.txt 10000000 10000000 100
```

- The <tumor file> and <normal files> are the tumor and normal read files of a chromosome, respectively.
- To run the algorithm, the [total tumor/normal read number] should be defined to consider the sequencing depth between two dataset. These values are simply the sum of total (genome-wide, not chromosomal sum) reads number of tumor and normal genomes. If the sequencing depth of two datasets is the same, it is possible to simply put '1 and 1' (similarly, '32 and 21' or '32000000 and 21000000' works the same if the total read numbers from the tumor and matched normal genome have 32 million and 21 million reads, respectively).

- SW-score cutoff is the minimum SW-score of segments to be identified. During the iteration, the algorithm reports the high-scoring segment first (each iteration gives a genomic segment as putative copy number alteration/variation with SW-score and significance). The iteration continues until the observed SW-score is below the predetermined cutoff. The high-scoring segments (i.e., large read counts discrepancy between two genomes) are potential candidates compared to those with low SW-score (likely to be false positives). The default SW-score cutoff is 80-100; however, it can be adjusted according to the sequencing depth of the dataset. For example, high-coverage data may require higher SW-score cutoff to minimize the false-positives.

Example usage of rSW-seq

The example dataset is available at:

http://compbio.med.harvard.edu/tkim/rsw-seq/testChr_tumor_1millionReads.txt

http://compbio.med.harvard.edu/tkim/rsw-seq/testChr_normal_1millionReads.txt

In 'tumor' dataset, human chromosome 1 (~250Mb) is simulated for 16 copy number alterations (8 single copy gains and 8 single copy losses with varying sizes of 10kb-1Mb) with 1 million sequencing reads. For details, see the Methods in the original article of rSW-seq.

For test run, try:

```
./a.out testChr_tumor_1millionReads.txt testChr_normal_1millionReads.txt 1 1 50
```

The results will be:

```
Using Smith-Waterman score cutoff: 50
Total tumor reads: 1, total normal reads: 1
Calculated gain threshold: 0.100000, loss threshold: 0.166667
gain   152900903     153919610     6477   4493   887.00 9.6009e-81
gain   169407843     169908134     3290   2155   590.50 4.6620e-54
gain   73241676      73540065      2166   1398   411.60 1.9017e-38
gain   5935871       6035305       743    470    151.70 1.7968e-15
gain   38841156      38921084      461    311    72.80  3.1215e-08
loss   223802171     224784587     4344   2138   1125.67       7.2612e-169
loss   167153690     167654851     2311   1111   629.67 9.2889e-96
loss   194810469     195109360     1119   598    234.83 4.4038e-37
loss   35715895      35810836      436    195    135.83 1.3956e-22
```

```
loss   229373550     229439920     324    167    75.17  4.6920e-13
loss   112530070     112557018     116    37     53.50  3.5823e-11
```

In this test run, the algorithm identified 5 gains and 6 losses (tumor-specific gain or loss with respect to normal). When compared to the simulation setting (Table below), single copy gains ≥ 50kb and single copy losses ≥ 30kb were all captured without false positives.

	Gain (single copy)		Loss (single copy)	
Size	Start	End	Start	End
1Mb	152900890	153900890	223799078	377699968
500Kb	169405397	169905397	167154513	337059910
300Kb	73239281	73539281	194810533	268349814
100Kb	5935984	6035984	35711015	41746999
50Kb	38856857	38906857	229369828	268276685
30Kb	81364556	81394556	112529785	193924341
20Kb	179630052	179650052	109652702	289302754
10Kb	236530721	236540721	2582215	239122936

The simulation setting for the test 'Tumor' chromosome is shown. The alterations identified by test run of rSW-seq are in yellow. Note that this example does not guarantee the actual performance of the algorithm (please see Figure5 in the original article of rSW-seq for detail).

In the results, the header section shows the used SW-score cutoff, total tumor/normal read number (as mentioned, '1 and 1' is used since we assume that the sequencing depth is equal for tumor and normal in this simulation test set), and gain/loss threshold used. The highest scoring tumor-specific gain is observed at 152.9-153.9Mb of the test chromosome, 6477/4493 is the number of tumor and normal reads in this segment (thus, tumor/normal read ratio = 1.4415 roughly corresponds to 1.5 = 3/2 of single copy gain of tumor genome as simulated). The last 2 columns are SW-score and significance of the segment. For detailed description of the parameters and output values, please see the original rSW-seq article. Since the SW-score cutoff was set to be 50, only 5 gain segments whose SW-score > 50 are reported. SW-score (and accompanying significance level) is a good measure to determine whether the gain/loss calls are true or not (however, the optimum level of SW-score or significance should be adjusted depending on the sequencing depth, or other genomic factors).

Among the 6 losses, the first segment (224.8 – 224.8Mb) has 4344 normal and 2138 tumor reads (tumor/normal read ratio = 0.49 ~ roughly corresponding to single copy loss of tumor genome). This read ratio discrepancy is very significant, i.e., SW-score (1125.67) and significance level (7.2E-169).

Due to reciprocal nature, the loss-segment of tumor genome is equal to relative-gain segment of normal genome. This is the case when one compares two normal genomes. It is notable that the 'loss' of rSW-seq is optimized to detect single copy loss of tumor genome, which assumes that normal genome is 'diploid'. This may not be true in the comparison between normal genomes. Thus, in case of normal variation analyses, it is recommended to swap the dataset and only consider 'gain' segments in two comparisons. For example, in comparison of two sequencing data from normal A and B genomes, 'the gain segments of B compared to A' can be more sensitive than 'the loss segments of A compared to B'.