User Manual for *csig* for Windows

Version 0.2.9

program implemented by

Robert Yu

theoretical methodology developed by

Jian Wang and Sanjay Shete

August 6, 2008

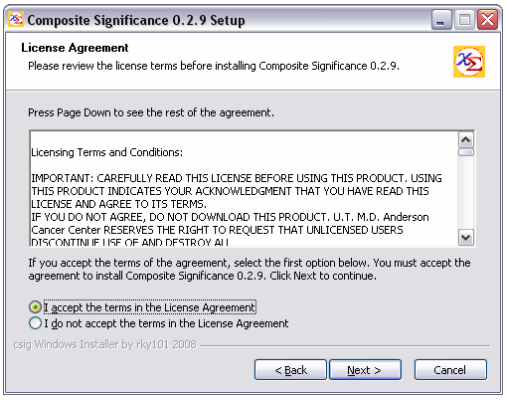
Table of Contents

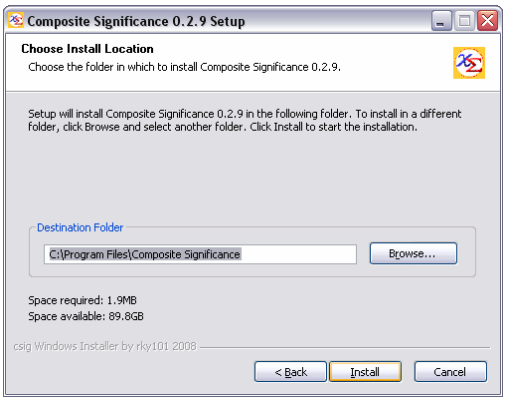
Installation and uninstallation

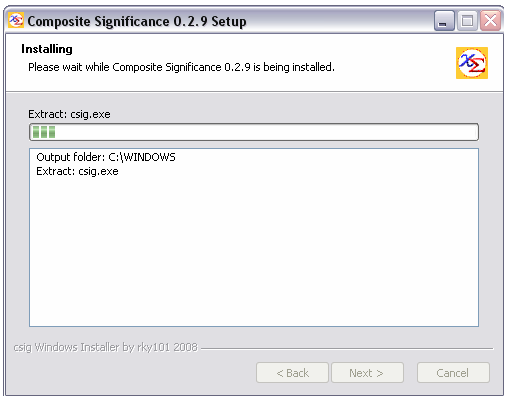
Installation

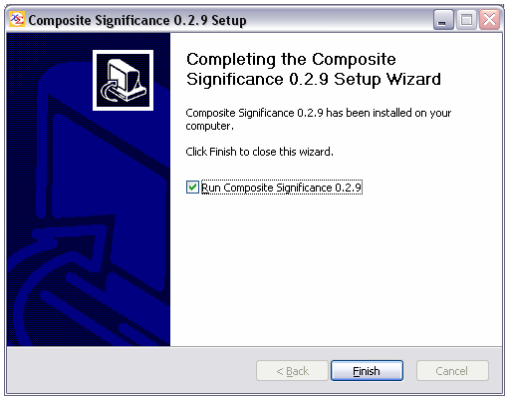
This version is limited to Windows (2000, XP and Vista). Installation can be done in the following steps:

Upon saving a copy of the program on your local machine, click on CSig029Installer.exe , and follow the on-screen instructions:

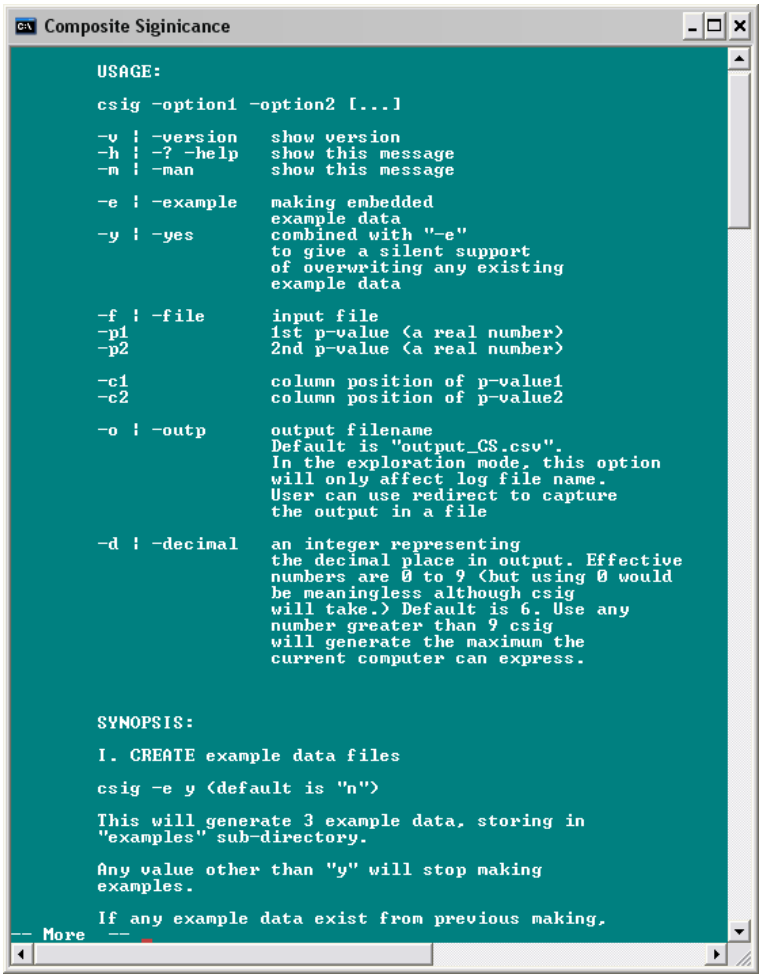








If the checkbox was checked on the last screen (above), a console window with help message will pop up.

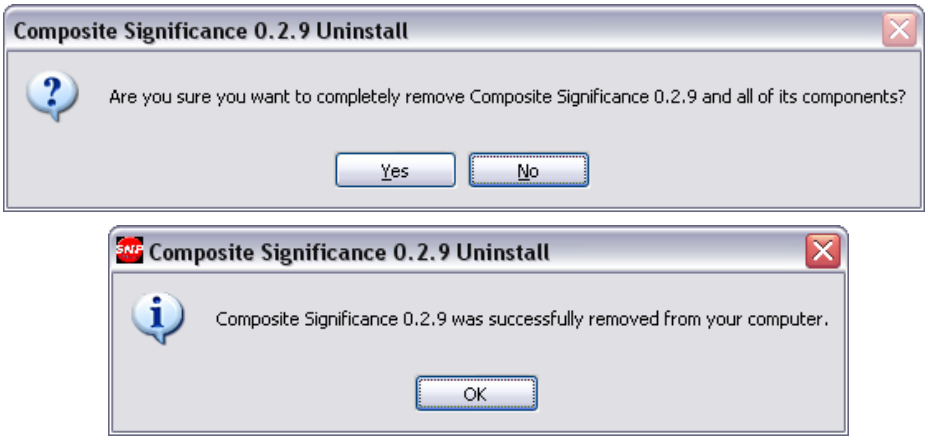


You may press “space” bar to go through the pages, or simply press “Q” to exit the message, and it should stay at “C:\”. You may start to explore, or choose to quit the console to view the start menu of the installation.

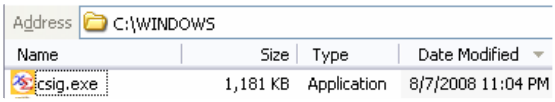
At the start menu, within “Composite Significance”, you have a “manual” folder containing “README.pdf” serving as user guide (this document), “Composite Significance” shortcut link to launch console window. “Uninstall” will allow you easily remove all install files without your looking for separately. A link to “M.D. Anderson Cancer Center” provides a convenience to visit its website.

Uninstallation

To remove the installation, go to “Start” → “Programs” → “Composite Significance” → “Uninstall”. It’ll delete all the files and programs.



Note: after installations, "`csig.exe`" is physically sitting in "`C:\WINDOWS`",

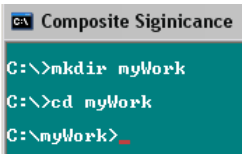


from where user can directly invoke it without setting it in the path. If you have your own working directory in path, such as "`c:\bin`", you may create a copy of `csig.exe` there. And you can use it without any links in the start menu. But if the installation was done through the installer, it'd be better not to change the file location otherwise it'll make later uninstallation problematic.

Get started

Exploration Mode

Before start, it might benefit to know how to configure console window in appendix.



1. "Start" → "Programs" → "Composite Significance" to invoke the console. Then make and change to a working folder, e.g. "`c:\myWork`",

2. To explore the program, type this

   `C:\myWork\csig -p1 your_p_value1 -p2 your_p_value2 [-decimal 3]`

   where `[-decimal 3]` is optional if user desires to have output values with 3 decimal places. And "*`your_p_value1`*" and "*`your_p_value2`*" are the pair of p-values from your logistic regression analysis and Hardy-Weinberg Equilibrium test (order doesn't matter). For example, to calculate composite significance results using p-value 1 = 0.03 and another 0.005, use the following command

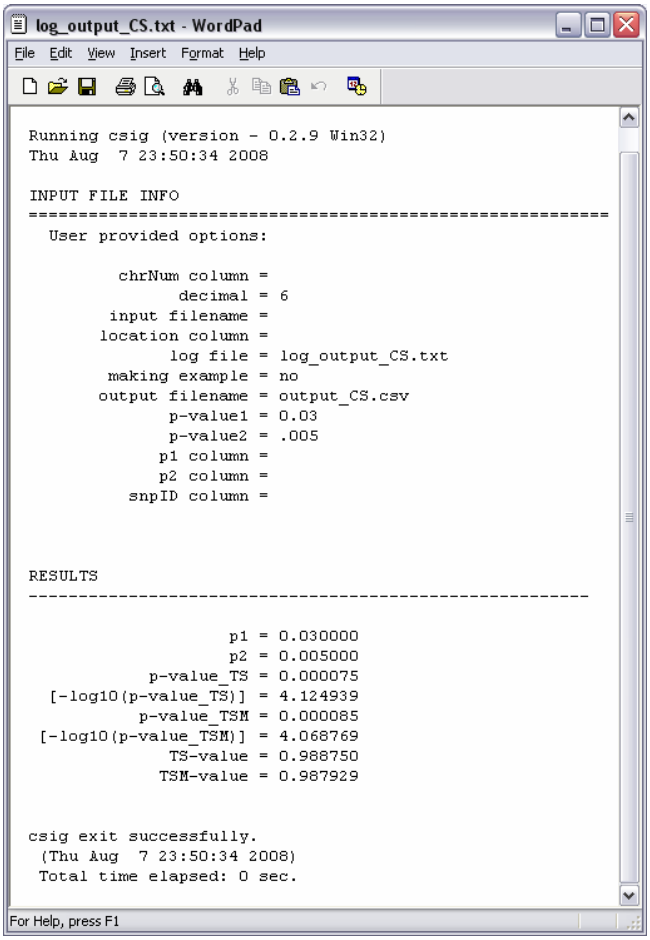
   ```
   C:\myWork>csig -p1 0.03 -p2 .005
   ```

   Upon pressing "Enter", the result will print to screen and be saved in a log file with default filename "`log_output_CS.txt`".

```
csig (version - 0.2.9 Win32) is running ...
with options:

        chrNum column =
              decimal = 6
       input filename =
     location column =
            log file = log_output_CS.txt
      making example = no
     output filename =
            p-value1 = 0.03
            p-value2 = .005
          p1 column =
          p2 column =
        snpID column =


RESULTS
------------------------------------------------------------

                  p1 = 0.030000
                  p2 = 0.005000
         p-value_TS = 0.000075
 [-log10(p-value_TS)] = 4.124939
        p-value_TSM = 0.000085
[-log10(p-value_TSM)] = 4.068769
           TS-value = 0.988750
          TSM-value = 0.987929
```

To view the log file, you can either type "`more`" or "`write`" command with the filename. The former will print the content on screen and the latter will launch Wordpad with the file content.

```
C:\myWork>write log_output_CS.txt
```



Note: using this exploration mode of `csig.exe` can't indicate an output file to save the result. You can use commandline redirect "`> outputfile`" to capture or "`>> outputfile`" to accumulate the result. Using commandline option "`-o result.txt`" here can allow user to make dedicated log filename, e.g. "`log_result.txt`" in this example.

```
csig (version - 0.2.9 Win32) is running ...
with options:

        chrNum column =
               decimal = 6
       input filename =
      location column =
              log file = log_output_CS.txt
       making example = y
      output filename =
             p-value1 =
             p-value2 =
            p1 column =
            p2 column =
          snpID column =



Could Not Find C:\myWork\examples

       MESSAGE (Making Sample Data) : Writing to testing.dat ....

       MESSAGE (Making Sample Data) : Writing to sample_data2.dat ... 1000 lines..

       MESSAGE (Making Sample Data) : Writing to sample_data2.csv ... 5000 lines..
 Volume in drive C has no label.
 Volume Serial Number is 3C9D-6646

 Directory of C:\myWork\examples

08/07/2008  11:59 PM               317 testing.dat
08/07/2008  11:59 PM            49,148 sample_data2.dat
08/07/2008  11:59 PM           238,655 sample_data2.csv
               3 File(s)        288,120 bytes
               0 Dir(s)  96,461,004,800 bytes free
```

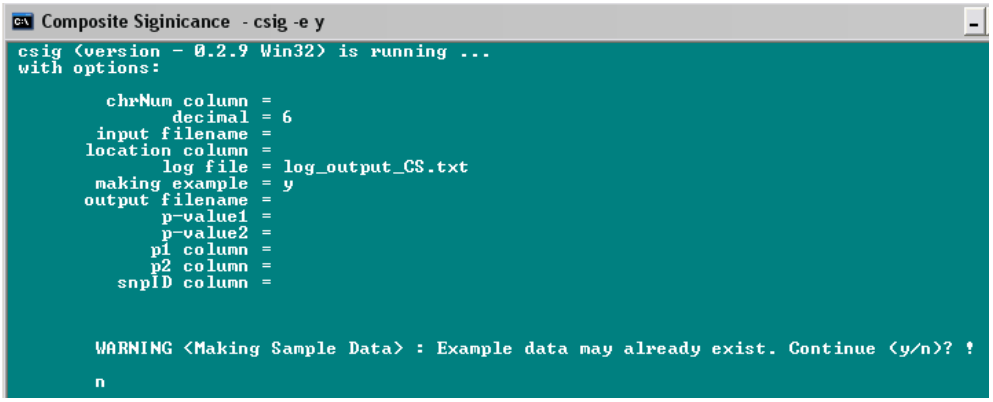
Batch Mode

1. First to get the embedded example data sets (total 3) by running csig with option "`-e y`" or "`-example yes`":

   ```
   C:\myWork\csig -e y
   ```

   Then a screen on the right will show up. The program terminates and lists the 3 example data sets sitting in a sub folder "`examples\`".

2. Now if you invoke this "`-e y`" option again or accidentally use it in future while you've already created the example data sets, the program will pause to ask if you want to replace the existing ones.



If "`y`" is chosen, new example data sets will be created. Except for "`testing.dat`", in the other two data sets the p-values are generated by random although sample sizes are fixed (see detail in the help manual by typing "`-h`" or "`-?`" or see below). Here is part of "`sample_data2.dat`":

```
p1    p2    snp
0.0745406337082386      0.0464276634156704      snpID2000
0.15244622528553  0.40436440333724  snpID2001
0.00894095934927464     0.390869622118771 snpID2004
0.0162267098203301      0.511203425936401 snpID2009
0.279701948165894 0.125540100038052 snpID2016
0.268928272649646 0.130892470479012 snpID2025
0.0720584439113736      0.0093390978872776      snpID2036
```

3. To avoid such a pause, this version of csig.exe supports a silent permission of overwrite (or abort) of re-creating example data sets. To overwrite, use this

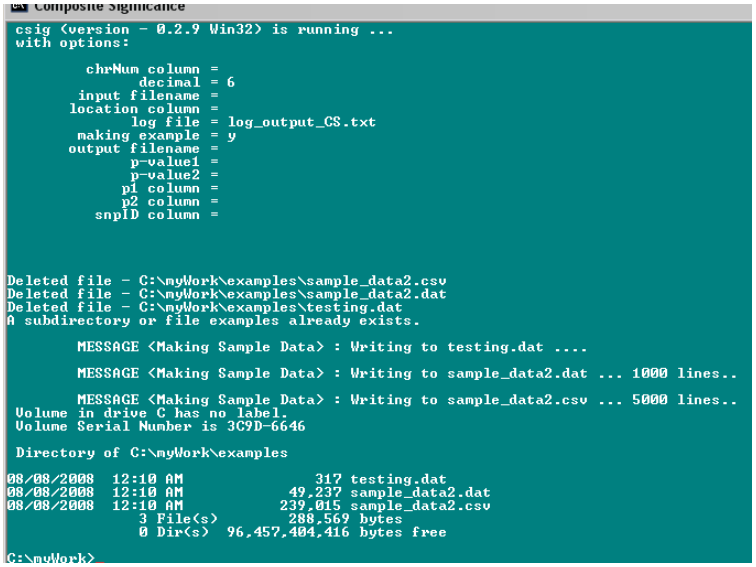
```
C:\myWork\csig -e y -y y
```

Or abort when previous make is existing

```
C:\myWork\csig -e y -y n
```

4. If using "`sample_data2.dat`" for testing, now at prompt,

```
C:\myWork\csig -f
examples\sample_data2.dat
-c1 1 -c2 2 -d 7
```



This requests `csig` to take a sample data with one set of p-values in column 1 (option `-c1 1`) and another in column 2 (option `-c2 2`), and the output result with 7 decimal places.

The screen will print out the one on the right.

And two files are generated, a log file `log_output_CS.txt` and `output_CS.csv`.

5. To view the result file, type this

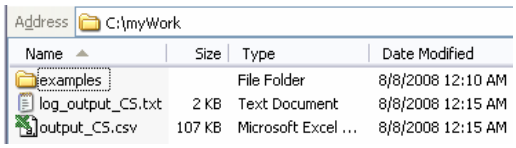
```
C:\myWork\write output_CS.csv
```

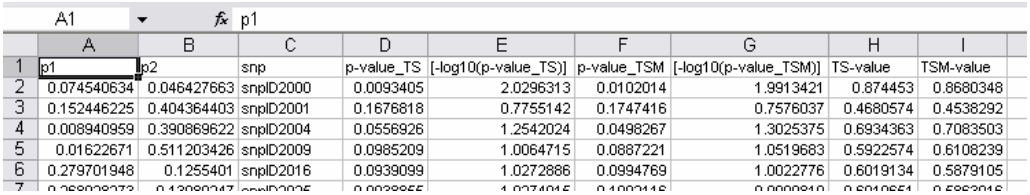
```
p1,p2,snp,p-value_TS,[-log10(p-value_TS)],p-value_TSM,[-log10(p-value_TSM)],TS-value,TSM-value
0.0745406337082386,0.0464276634156704,snpID2000,0.0093405,2.0296313,0.0102014,1.9913421,0.8744530,0.8680348
0.15244622528553,0.40436440333724,snpID2001,0.1676818,0.7755142,0.1747416,0.7576037,0.4680574,0.4538292
0.00894095934927464,0.390869622118771,snpID2004,0.0556926,1.2542024,0.0498267,1.3025375,0.6934363,0.7083503
0.0162267098203301,0.511203425936401,snpID2009,0.0985209,1.0064715,0.0887221,1.0519683,0.5922574,0.6108239
0.279701948165894,0.125540100038052,snpID2016,0.0939099,1.0272886,0.0994769,1.0022776,0.6019134,0.5879105
0.268928272649646,0.130892470479012,snpID2025,0.0938855,1.0274015,0.1002116,0.9990819,0.6019651,0.5863916
0.0720584439113736,0.0093390978872776,snpID2036,0.0027444,2.5615559,0.0026214,2.5814623,0.9319475,0.9331041
0.00106794945895672,0.00849222112447023,snpID2049,0.0000377,4.4242084,0.0000359,4.4449578,0.9920289,0.9921720
```

Another way to view the result is to open it up in MS Excel, where user may take advantage of tools Excel provides, e.g. sorting and charting, etc. To open it in Excel, just go to the working folder and directly click on `output_CS.csv` file. A quick way to go to the current working folder, user may type at the prompt the following

```
C:\myWork\start .
```

Notice a little dot after "`start`".





	A	B	C	D	E	F	G	H	I
1	p1	p2	snp	p-value_TS	[-log10(p-value_TS)]	p-value_TSM	[-log10(p-value_TSM)]	TS-value	TSM-value
2	0.074540634	0.046427663	snpID2000	0.0093405	2.0296313	0.0102014	1.9913421	0.874453	0.8680348
3	0.152446225	0.404364403	snpID2001	0.1676818	0.7755142	0.1747416	0.7576037	0.4680574	0.4538292
4	0.008940959	0.390869622	snpID2004	0.0556926	1.2542024	0.0498267	1.3025375	0.6934363	0.7083503
5	0.01622671	0.511203426	snpID2009	0.0985209	1.0064715	0.0887221	1.0519683	0.5922574	0.6108239
6	0.279701948	0.1255401	snpID2016	0.0939099	1.0272886	0.0994769	1.0022776	0.6019134	0.5879105

6. Now you may test how `csig` detects and treats errors in the input data set. Example data file "`testing.dat`" was provided just for this purpose. Here is what the data looks like:

```
SNPID,Chr,Loc,p1,p2
rs34221 1 23432 0.4332 0.001
rs34222 1 23442 3.3e-5 0.001
rs34223 1 23452 1.56332 p2
rs34224 1 23462 0.5674
rs34221, 4 43432 0.4332 0.005
rs674222 5, 33442 1.2e-15 -3.051
rs994223, 61, 73452, 1.4332, p2
rs764224, z 83462,,0.1332
rs764224, x 83462, ,0.1332
rs764224 , XY 83462 ,2,0.1332
```

The data has p-values pairs in columns 4 and 5. Some of them either containing one p-value or containing non-number. Also notice the mixture of the delimiters; not only mixing "," with spaces/tabs, but also some comma comes with space before and/or after it. But `csig` can detect and process them properly.

Run this following command,

```
csig -f examples\testing.dat -c2 5 -c1 4 -o result_of_problem_data.txt
```

where –o option indicates user defined output filename. Although it contains ".txt" extension, the program changes it to ".csv".

After executing the command, the screen and log file both report the finding of errors and how many error lines were dropped, and how many valid lines yield the results.

```
Composite Siginicance

csig (version - 0.2.9 Win32) is running ...
with options:

        chrNum column =
             decimal = 6
      input filename = examples\testing.dat
     location column =
            log file = log_output_CS.txt
      making example = no
     output filename = result_of_problem_data.txt
           p-value1 =
           p-value2 =
         p1 column = 4
         p2 column = 5
       snpID column =


csig (version - 0.2.9 Win32) is running with opts:
       -f = examples\testing.dat -o result_of_problem_data.csv;
       log file = log_result_of_problem_data.txt


Checking the basic file properties ...

       counting lines ...

              11


  BASIC INFORMATION
 ====================
       Variable                                Value
 ------------------------------------------------------
   Filename stem                                testing
   Filename extension                              .dat
   Path                                        examples\

-> File size (byte)                                317 <-
   File size                                317 Bytes

   Total number of lines                            11
-> Line size (byte)                              28.82 <-
   Line size                                28.82 Bytes

   Last access time           Fri Aug  8 00:10:37 200
   Last modify time           Fri Aug  8 00:10:37 200
   Last inode chage time      Thu Aug  7 23:59:15 200
   Mode (permission)                              666
 ------------------------------------------------------

 PRINT FIRST 3 LINES OF examples\testing.dat:
       SNPID,Chr,Loc,p1,p2
       rs34221 1 23432 0.4332 0.001
       rs34222 1 23442 3.3e-5 0.001

============================================================

                SUMMARY OF ERROR REPORTING
------------------------------------------------------------
        Total error found:                          7
       Total lines dropped:                         8
          Total lines kept:                         3
------------------------------------------------------------

                    LIST OF DROPPED LINES
                   -------------------------------
                                1 (header)
                                4
                                5
                                7
                                8
                                9
                                10
                                11
------------------------------------------------------------



C:\myWork>
```

Here is the list of resulting files in `C:\myWork` and the output from running "`more`" result file:

```
08/08/2008  12:15 AM             1,809 log_output_CS.txt
08/08/2008  12:21 AM             2,456 log_result_of_problem_data.txt
08/08/2008  12:15 AM           109,456 output_CS.csv
08/08/2008  12:21 AM               356 result_of_problem_data.csv
               4 File(s)        114,077 bytes
               0 Dir(s)  96,182,865,920 bytes free

C:\myWork>more result_of_problem_data.csv
SNPID,Chr,Loc,p1,p2,p-value_TS,[-log10(p-value_TS)],p-value_TSM,[-log10(p-value_TSM)],TS-value,TSM-value
rs34221,1,23432,0.4332,0.001,0.063133,1.199743,0.055579,1.255087,0.673600,0.691974
rs34222,1,23442,3.3e-5,0.001,0.000000,6.421607,0.000000,6.466709,0.999200,0.999237
rs34221,4,43432,0.4332,0.005,0.065475,1.183922,0.058071,1.236042,0.667600,0.685146
```

Description

This program is an implementation of the following methodology, which has been published:

> Wang, J. and Shete, S. A Test for Genetic Association that Incorporates Information about Deviation from Hardy-Weinberg Proportions in Cases. Volume 83, Issue 1, 53-63, 26 June 2008.

Methodology

For assessment of genetic association between single-nucleotide polymorphisms (SNPs) and disease status, the logistic regression model or generalized linear model is typically employed, and has been applied to detect a variety of disease-causing SNPs. However, the regression approaches do not integrate information that is available from other sources, such as departure from Hardy-Weinberg proportion in cases. The Hardy-Weinberg proportion is one of the most important principles in population genetics. Deviation from Hardy-Weinberg proportion among cases (patients) may provide additional evidence for the association between SNPs and diseases. Thus, testing for deviation from Hardy-Weinberg proportion could be another approach for genetic association studies.

We developed two approaches: (1) a mean-based tail strength (*TS*) measure and (2) a median-based tail strength (*TSM*) measure, to combine evidence from Hardy-Weinberg proportion and from regression approaches to perform the case-control association study. Both measures combined two very different hypothesis tests, the logistic-regression model and the test for deviation from Hardy-Weinberg proportion in cases, and allowed dependence between these two tests. The newly developed approaches are more powerful than the traditional association study approaches, achieving higher power than each individual test and maintaining good control over type I error probabilities. We derived the exact formulas for calculation of corresponding p-values. We also proposed an approach for estimating empirical p-values with the use of a resampling procedure. For the detail of the approaches, please refer to the study of Wang and Shete.

Note: According to the simulation studies in the paper, we can see that the empirical p-values from the permutation approach and the exact p-values from the exact formulas are very similar, and the exact formulas give slightly conservative p-values. Therefore, the exact p-values are considered satisfactory. The current version of this computer program only applies to calculation of the p-values using the exact formulas. The program for the permutation method as described in the paper is under construction, and will be available soon.

Detailed CSig Usage and Description

USAGE：

```
csig -option1 -option2 [...]

-v | -version   show version
-h | -? -help   show this message
-m | -man       show this message

-e | -example   making embedded
                example data
-y | -yes       combined with "-e"
                to give a silent support
                of overwriting any existing
                example data

-f | -file      input file
-p1             1st p-value (a real number)
-p2             2nd p-value (a real number)

-c1             column position of p-value1
```

```
-c2              column position of p-value2

-o | -outp       output filename
                 Default is "output_CS.csv".
                 In the exploration mode, this option
                 will only affect log file name.
                 User can use redirect to capture
                 the output in a file

-d | -decimal    an integer representing
                 the decimal place in output. Effective
                 numbers are 0 to 9 (but using 0 would
                 be meaningless although csig
                 will take.) Default is 6. Use any
                 number greater than 9 csig
                 will generate the maximum the
                 current computer can express.
```

SYNOPSIS:

I. CREATE example data files

csig -e y (default is "**n**")

This will generate 3 example data, storing in "examples" sub-directory.

Any value other than "y" will stop making examples.

If any example data exist from previous making, csig will pause a question for permission of replacement. A negative answer will abort the step and continue to use the current ones.

A silent permission can provide at initial starting of csig instead of waiting for response:

csig -e y -y y

II: EXPLORATION with a pair of p-values

csig -p1 0.002 -p2 1.2e-5

Note: the p-value in "dot-leading" format is also acceptable, e.g. ".002" for 0.002.

csig -p1 .002 -p2 0.03

or

csig -p1 0.002 -p2 1.2e-5 -d 3 > output.txt

III. BATCH MODE of list of p-value pairs in file,

csig -f sample_data.dat -o result.txt -c1 4 -c2 5 [-d 99]

Note: any value greater than 9 for option -d allows csig to generate result values in max decimal places that the working computer can.

EXPLANATION:

I. Making Example Data:

Command "**csig -e y**" will lead to create 3 example data. And if csig detects prexisting examples, it'll prompt for user's decision to generate new set replacing the old ones.

csig chooses to terminate upon this execution to allow user to choose examples for the next run.

Except for "testing.dat", the other two are generated using random numbers. Thus, each run may have different value sets.

How are the random samples created?

"sample_data2.dat", containing 1000 lines, uses double random for p-value 1 and p-value 2, i.e.

```
p-value1 = rand() * rand();
p-value2 = rand() * rand();
```

"sample_data2.csv", containing 5000 lines, uses the following for p-value 1 and p-value 2, i.e.

```
p-value1 = rand() * 0.02;
p-value2 = rand() * rand() * 0.1.
```

II. Data input:

1. For exploration, user just needs to give a pair of p-values, and/or output decimal place requirement, e.g.

csig -p1 0.023 -p2 1.05e-3 [-d 3]

If option "-d" is not provided, a default 6 will be used. Any number greater than 9 will direct csig to generate whatever the computer may be able to express.

2. For batch mode, user is required to provide the list of p-value pairs in a data file using either "," or space (including tab) or mixture of both as delimiter, and then tell csig where are the column positions for p-value1 and p-value2. One example is:

```
SNPID,Chr,Loc,p1,p2
rs34221 1 23432 0.4332 0.001
rs34222 1 23442 3.3e-5 0.001
rs34223 1 23452 1.56332 p2
rs34224 1 23462 0.5674
rs34221, 4 43432 0.4332 0.005
rs674222 5, 33442 1.2e-15 -3.051
rs994223, 61, 73452, 1.4332, p2
rs764224, z 83462,,0.1332
rs764224, x 83462, ,0.1332
rs764224 , XY 83462 ,2,0.1332
```

where p1 and p2 are in columns 4 and 5, and the first 3 columns are optional SNP info. But notice that a header line is always expected. This dataset was created on purpose with mixed types of delimiter, i.e. ", " and " , " and

" ". Also some "invalid" p-values were added
to test whether csig will detect. Upon detected,
the entire SNP line will be dropped but continues
to process the next line.

To run csig with this dataset (if saved as "data.dat"),
the commandline typing would look like this:

csig -f data.dat -c1 4 -c2 5 -o result.csv -d 5

III. Output

And two output files will be generated, one result
file "result.csv" and a log file "log_result.txt".
The log file will list total drops of 8 lines
(including the header lines) for analysis and keep
3 valid lines for output results. In "result.csv",
the result looks like this:

```
SNPID,Chr,Loc,p1,p2,p-value-TS,[-log10(p_TS)],p-value-TSM,[-log10(p_TSM)],TS-value,TSM-value
rs34221,1,23432,0.4332,0.001,0.06313,1.19974,0.05558,1.25509,0.67360,0.69197
rs34222,1,23442,3.3e-5,0.001,0.00000,6.42161,0.00000,6.46671,0.99920,0.99924
rs34221,4,43432,0.4332,0.005,0.06548,1.18392,0.05807,1.23604,0.66760,0.68515
```

Interpretation of output

Columns	Column head	Explanation
1	SNPID	SNP rs number (user provided)
2	Chr	Chromosome number (user provided)
3	Loc	SNP location (user provided)
4	p1	p-value 1 (user provided)
5	p2	p-value 2 (user provided)
6	p-value-TS	p-value-TS
7	[-log10(p-value-TS)]	negative log10 of p-value-TS
8	p-value-TSM	p-value-TSM
9	[-log10(p-value-TSM)]	negative log10 of p-value-TSM
10	TS-value	Tail Strength Mean value
11	TSM-value	Tail Strength Median value

Columns 6 and 8 are the composite p-values, and columns
7 and 9 are the log transformed version of the two
composite p-values.

Notice that this is a comma delimited file with "csv"
file extension, and thus user may click on it and let
it automatically opened up in Excel and have better view
or even take advantage of tools in Excel, e.g. basic sorting
and charting results.

In exploration mode, the results are output to screen
only but a log file will be generated, which also
contains result. Another way to collect several
runs of the exploration is to redirect the screen
output, e.g.

csig -p1 0.233 -p2 0.03 -d 3 >> result_explore.txt

DESCRIPTION:

This program csig is to calculate
the composite significance combining both
logistic regression p-value and HWE testing
p-value. Please refer to README.PDF for
rationale and detailed information of this
methodology, or email to authors below:

Dr. Sanjay Shete: *sshete@mdanderson.org*
Dr. Jian Wang: *jianwang@mdanderson.org*

Reference:

Wang, J. and Shete, S. A Test for Genetic Association that Incorporates Information about Deviation from Hardy-Weinberg Proportions in Cases. Volume 83, Issue 1, 53-63, 26 June 2008.

csig was designed to facilitate the calculation of this method for both exploration purpose and batch mode processing need.

Note: According to the simulation studies in the paper, we can see that the empirical p-values from the permutation approach and the exact p-values from the exact formulas are very similar, and the exact formulas give slightly conservative p-values. Therefore, the exact p-values are considered satisfactory. The current version of this computer program only applies to calculation of the p-values using the exact formulas. The program for the permutation method as described in the paper is under construction, and will be available soon.

csig (version 0.2.8 win32) was designed as a analysis module in our ongoing programming effort to create a SNP Analytics, which is a GUI-based software. That software will integrate other tools including LD measurements, exact HWE test, haplotype estimation, SNP simulation, association tests, etc.

Please send us your email address and we will update our new tools, or tell us your suggestions.

Version: 0.2.8 win32
Release: 08/04/2008

History:

08/07/2008 Modification
- Standardized prototyping of common subroutines
- Added another option, "-yes" to combine with "-example" for silent support of overwriting any existing example data

08/04/2008 Development
- Release for other platforms:
  * Linux (x86-32), tested on
    RedHat / Fedora 8
    SUSE 10 (AMD x86-64) using 32-bit format
  * Solaris - SPARC 32-bit, tested on
    SunOS 5.9 Generic_122300-24
    sun4u sparc SUNW, Sun-Fire-880
    SunOS 5.10 Generic_127111-09
    sun4u sparc SUNW, Sun-Fire-V490
    SunOS 5.10 Generic_120011-14
    sun4u sparc SUNW, Sun-Fire-480R
- Modified output messages
- Changed heading by using "[-log10 (...)]" instead of "neg_log10".
- Revised the help/manual message

```
07/31/2008      Fix
                - allow leading "." for a real number
                  in the input, e.g. ".001" for "0.001"
                - change the name "Pmean" and "Pmedian" to
                  "p-value-TS" and "p-value-TSM"
                - in batch mode, the default log file won't
                  take the same file extension as the one of
                  input data file (as it did in the previous
                  versions)
                - output file will always take ".csv" to
                  reflect its comma-delimited format and force
                  to take the advantage of easily being able
                  to open in MS Excel (since this is a Windows
                  version)
07/30/2008      Modifications
                - embed example data into csig
                  allowing user's choice to generate.
                  Two of the three examples are now
                  generated with randomness (see
                  details in manual).
                - Windows installer thus changes
                  accordingly; no link to example
                  files at start menu. Meanwhile,
                  improvement on invoking console
                  made.
                - Fixed some typos in PDF user
                  manual.
                - Changed heading by using "neg_log10"
                  instead of "-log10". The latter
                  would have to cause problem if user
                  open the csv file in MS Excel, which
                  has trouble to decipher the leading
                  "-" sign.
                - Added an alarm bell at the end of
                  csig. It'll ring only when it runs
                  over 2 seconds. This may be helpful
                  when longer input data file is fed.
07/29/2008      Modifications
                - allow flexible column positions
                  of p-values
                - allow user indication of output
                  decimal place
07/13/2008      Initial release.
                single module calculating composite
                significance.

-----------------------------------------------------
Further questions or reporting bugs, please email
Robert Yu at rkyu@mdanderson.org
```

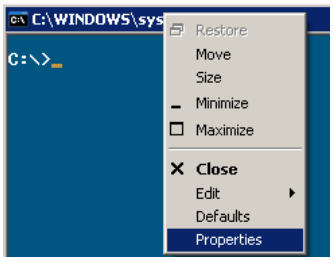
Appendix

How to configure console window

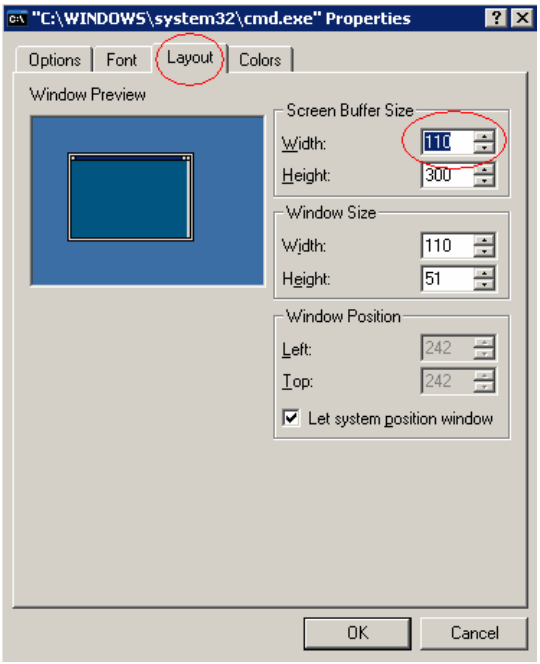
The default setting of the Windows console window may not be well fit your commandline program, e.g. lower screen buffer size, etc. The set it wider, for example, you can follow the steps below (and you may choose to adjust other properties, too).

Upon the console window is up running, e.g. Start→run→cmd, or through launching “Composite Significance” program, do the following:

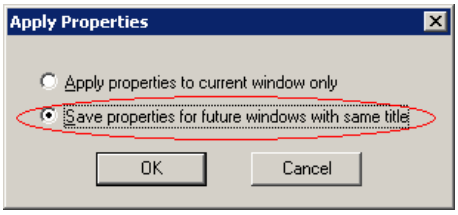
1. Right click the top title bar, and choose “Properties”



2. Select tab “Layout”, and change the value at “Width” in “Screen Buffer Size”. The default is 80.



3. Upon click “OK”, in the next dialog box, check “Save properties for future windows with the same title”. Then “OK”. Ready to go.



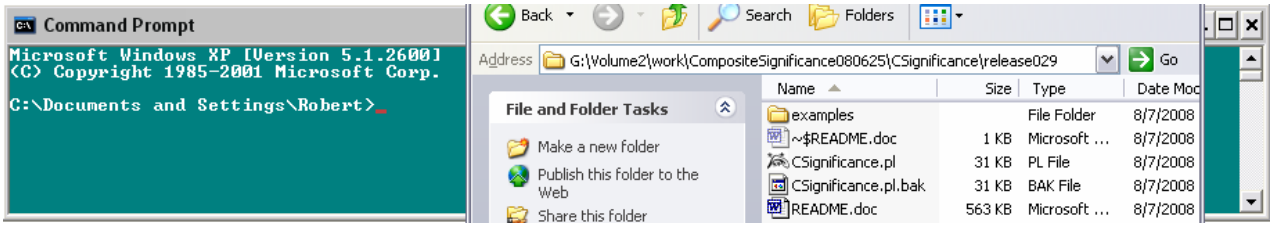
How to quickly change directory to a desired destination in console window

At console window, using tab-key for self completion has accelerated the typing for changing directory to a deep sit folder. However, it still takes quite a bit typing and tab-key pressings. In particular, when the desired folder sits in another drive, it may ask for one or two more steps.

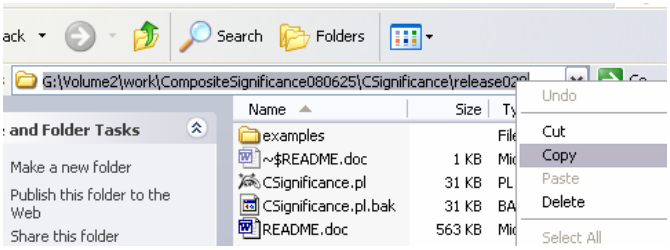
Here lists a simple way to accomplish the task.

If the initial directory is at C: drive when a console is just launched, and a desired folder deeply sits at G: drive, the following steps may make it "jump":

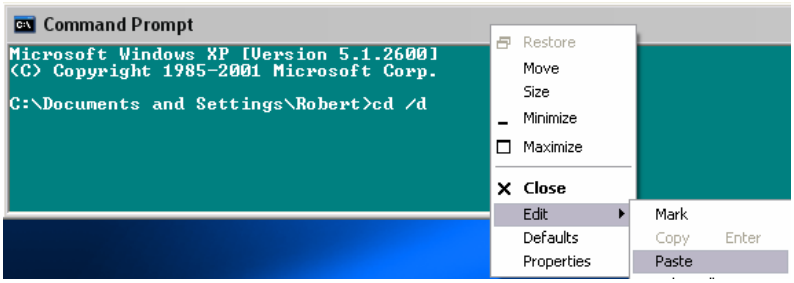
Step 1. With launched console and opened Windows Explore in your desired folder:



Step 2. Highlight and copy the path at the "Address" bar in Windows Explorer. Note, the "Copy" step can be done either using right click on the highlighted path or pressing Ctl+C.



Step 3. At console prompt, type "`cd /d `", and at the console title bar, right click and choose "Edit"→"Paste".



Notice that switch "/d" for "cd" is critical when a directory change is attempted from one drive to another.

Step 4. Press Enter. Done