Bayesian sample size determination for prevalence and diagnostic studies in the absence of a gold standard

Sample size calculations and asymptotic results

(Version 6.0, February 2019)

1. Introduction

The program **PropMisclassSampleSize** --- available for Windows only --- was developed to estimate sample size in the context of prevalence and diagnostic test studies, along the lines of our papers

Bayesian Sample Size Determination for Prevalence and Diagnostic Test Studies in the Absence of a Gold Standard Test
Nandini Dendukuri, Elham Rahme, Patrick Bélisle and Lawrence Joseph
Biometrics 60, pp. 388-397
June 2004

and

Bayesian sample size determination for diagnostic studies when tests are conditionally dependent
Zhuoyu Wang, Nandini Dendukuri, Patrick Bélisle and Lawrence Joseph
Biometrics XX, pp. XX-XX
XX 2018

We recommend that you read the above two papers carefully before using this software; these papers are available from http://www.medicine.mcgill.ca/epidemiology/Joseph/

PropMisclassSampleSize can also be used for asymptotic calculations, i.e., to find the maximum possible interval average coverage or minimum possible average interval length, if an infinite sample size is used.

You are free to use this program, for non-commercial purposes only, under two conditions:

- This note is not to be removed;
- Publications using **PropMisclassSampleSize** results should reference the manuscripts mentioned above;
- While we have done our best to ensure the program works as described in this manual, the user acknowledges that this program is not necessarily bug-free. We assume no liability for any errors or consequences that may arise from the use of this program. The use of this software is at the exclusive risk of the user.

If you have not installed **PropMisclassSampleSize** yet, please read the Installation Instructions (InstallInstructions.html) first.

The easiest way to open this program[1] is to use the shortcut found in Programs list from the Start menu. Once opened, you will be prompted by a graphical user interface (GUI) to describe the problem, that is:

- choose between sample size calculations or asymptotic calculations
- fill in your prior information about disease prevalence and diagnostic test(s) sensitivity and specificity

[1] You can start PropMisclassSampleSize by browsing through the User's Programs menu (available by clicking the Start button and then Programs) and selecting PropMisclassSampleSize. You can also start PropMisclassSampleSize by opening Windows Explorer, browsing to this package's location (c:\Users\user name\Documents\Bayesian Software\PropMisclassSampleSize or c:\Documents and Settings\user name\My Documents\ Bayesian Software\PropMisclassSampleSize by default, depending on your platform) and clicking on PropMisclassSampleSize.vbs.

- sample size criterion selection
- output file (where you want the results to be saved)
- and a few more technical questions (number of Gibbs iterations, starting sample size, etc.).

Once the GUI has collected all of the inputs required for the problem, it will be closed and the program will continue almost invisibly; the only thing that you will see on your screen is a WinBUGS window, which you can minimize.

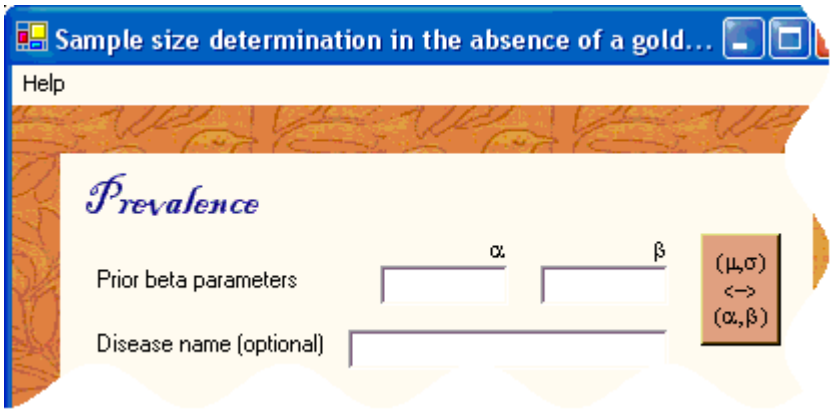
When the program has finished (running time can vary, and could be several hours if you are running sample size calculations) another GUI will appear announcing program completion and giving you the opportunity to view the output immediately. This GUI will not appear when **PropMisclassSampleSize** is called from a script (see section 2.1).

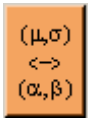
When started from the .vbs file (for example, when run from the Start menu), **PropMisclassSampleSize** will always run at low priority, allowing your system to use more CPU for higher priority tasks when needed. Thus, you can continue to work as this program runs in the background.

2. How to use PropMisclassSampleSize

The initial window (below) is used to choose between sample size calculations, estimation of average or percentile of HPD lengths or coverages for a series of predetermined sample sizes, or asymptotic property calculations.

The next two forms will be used to enter your prior information on the prevalence of the disease and diagnostic test(s)' sensitivity and specificity. Each prior is given a beta distribution with parameters (α,β), such that prior mean and variance are $\alpha/(\alpha+\beta$ and $\alpha\beta/(\alpha+\beta)^2(\alpha+\beta+1)$.



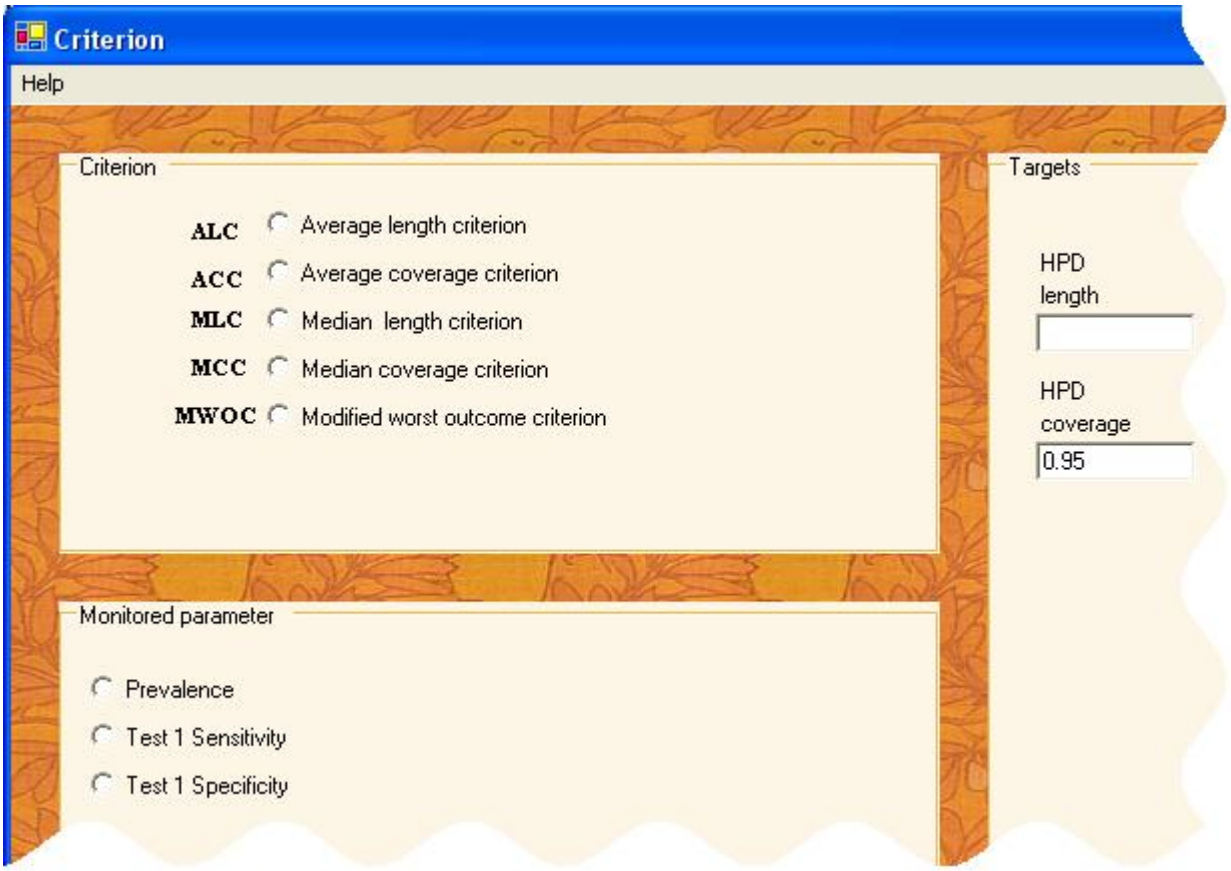


The orange button (μ,σ) <–> (α,β) allows you to specify your prior distributions in terms of prior moments (μ,σ) rather than in terms of (α,β).

Entering a label in the *Disease name* cell will save the corresponding prior values you just entered, making them only one click away the next time you run the program.

After prior parameters values for the prevalence have been entered, similar windows will appear for the sensitivity and specificity for each test used.

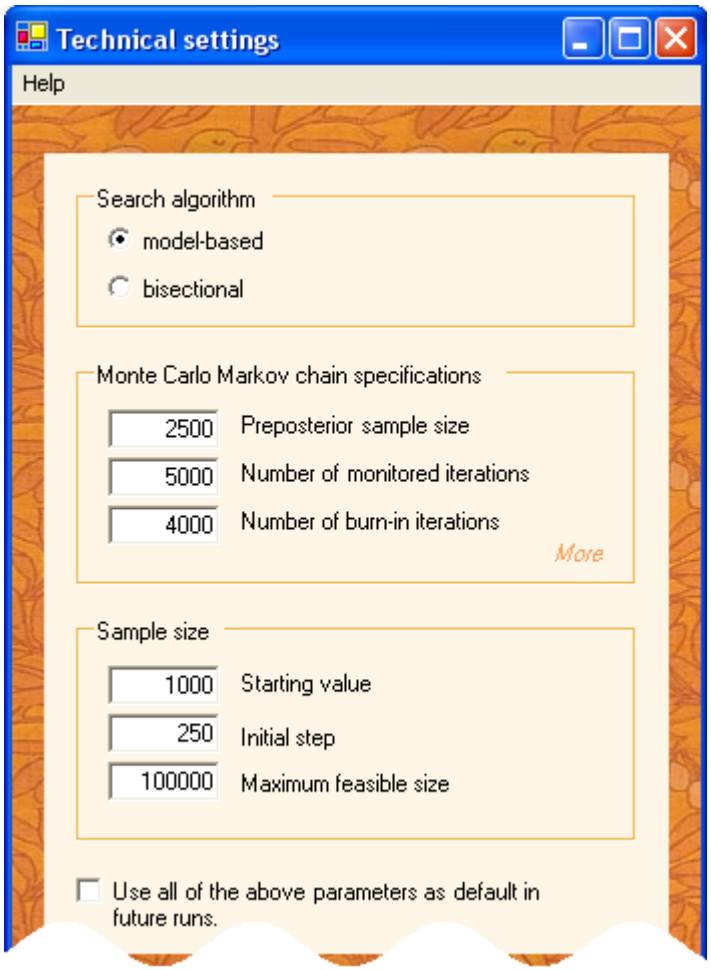
If sample size determination was selected in the initial window, the next form (below) allows selecting one of the five sample size criteria available. This window is also used to specify the fixed or target (depending on criterion selected) HPD length and coverage for the parameter of interest, also selected from this form.


Criterion
Help
Criterion
ALC
Average length criterion
ACC
Average coverage criterion
MLC
Median length criterion
MCC
Median coverage criterion
MWOC
Modified worst outcome criterion
Monitored parameter
Prevalence
Test 1 Sensitivity
Test 1 Specificity
Targets
HPD length
HPD coverage
0.95

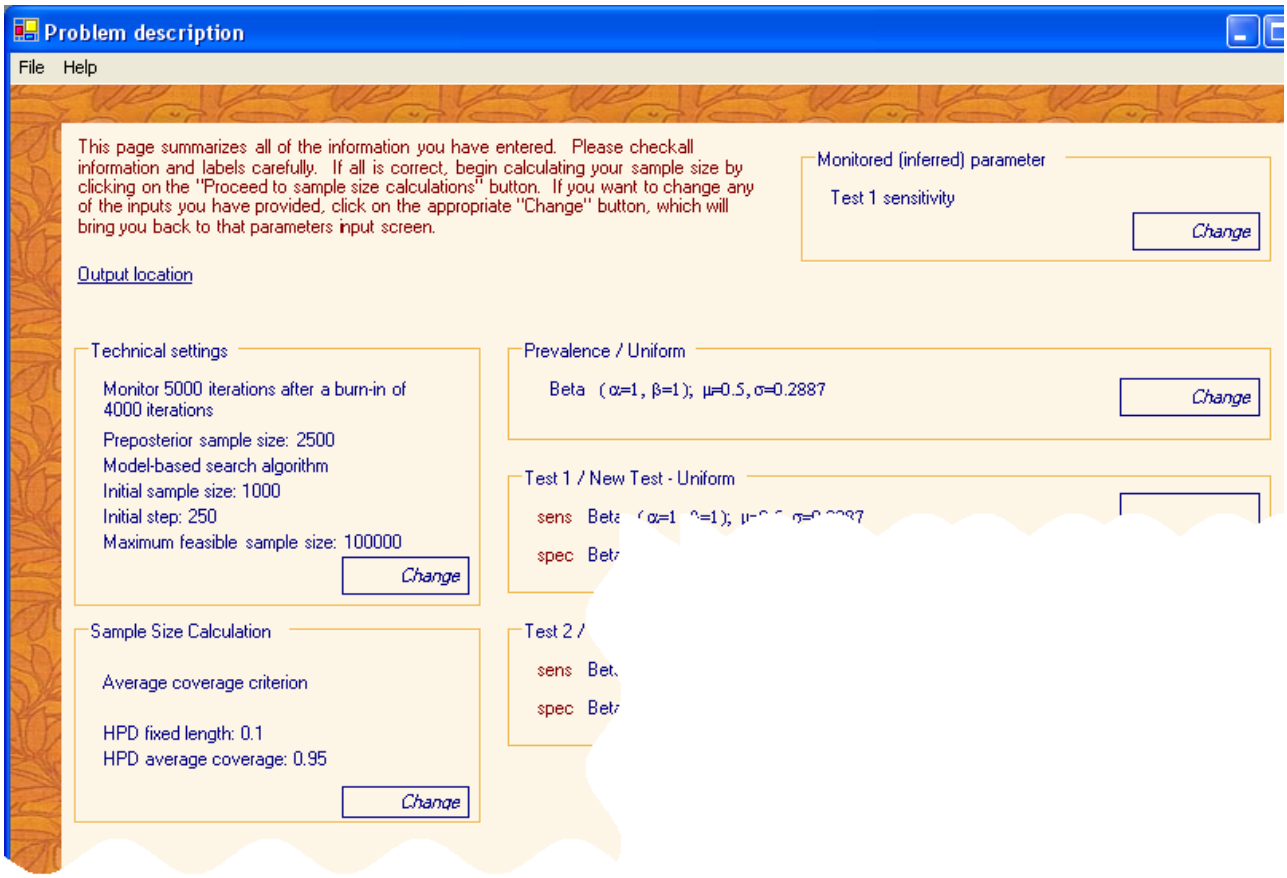
The next window allows the user to specify the preposterior sample size and the number of burn-in and monitored iterations of the Gibbs sampler algorithm that is used throughout. Changing these values is optional, the default values will usually provide reasonable estimates.

When **PropMisclassSampleSize** is used to calculate sample sizes, this form also allows the user to specify whether the optimal sample size should be found via a bisectional search or with a so called model-based algorithm. The latter, the default choice, usually converges faster to the optimal sample size neighbourhood.

In either case, the first three sample sizes for which the outcome of interest (e.g. average HPD length) will be measured are based on a bisectional search, after which this option comes into effect.



Finally, a Problem Reviewal form (below) allows the user to review each parameter entered through the different forms and modify any, if necessary, by clicking the appropriate *Change* button.



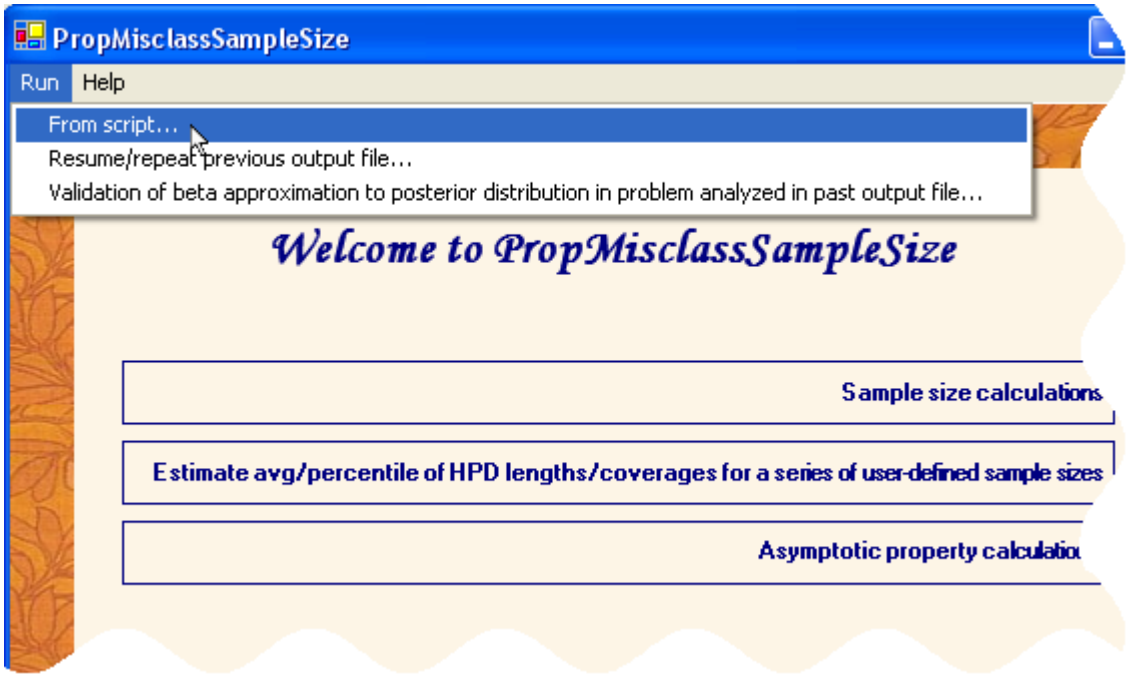
The above form is also used to select the output file location, either by selecting the top-left menu item *File/Save as…* or by clicking the Output location link in the upper left portion of the form.

2.1 Saving and running scripts

Once a problem is fully described by filling the appropriate forms, the actual computations can be launched right away by clicking the *[Run now] Proceed to Sample Size calculations* >> or saved for future submission by clicking the *[Run later] Register Problem Description* >> button, both found in the lower right corner of the Problem Reviewal form. Problems saved for future computation will be saved to *script* files and identified by a label entered by the user in the next form.

The initial form of **PropMisclassSampleSize** allows the user to run one or more previously registered scripts through the *Run/From script…* top-left menu item.

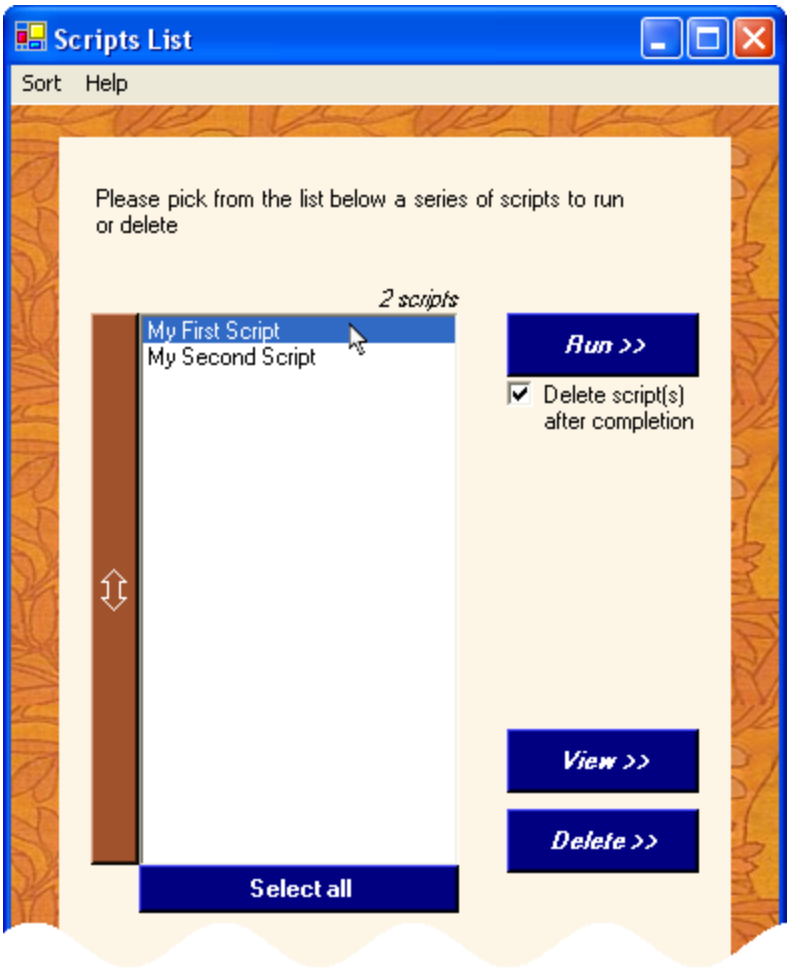
Script submission is useful when computing sample size in a number of variants (e.g., with different criteria or with different prior distributions) in that it eliminates delays between each run.



Select the scripts you wish to run now by clicking the appropriate script label(s) from the list and click the *Run>>* button.

Note the tick box below the *Run>>* button which can be ticked off if you do not wish to delete the script files when the calculation is completed

By default, the scripts are listed in order of entry date and time. Clicking the two-sided arrow button to the left of the list will reverse the order of the list.



3. Examples of running PropMisclassSampleSize

3.1 Sample size calculation

We will illustrate the use of **PropMisclassSampleSize** to estimate sample size requirements with a problem from Table 3 of our paper *Bayesian Sample Size Determination for Prevalence and Diagnostic Test Studies in the Absence of a Gold Standard Test,* cited in the introduction.

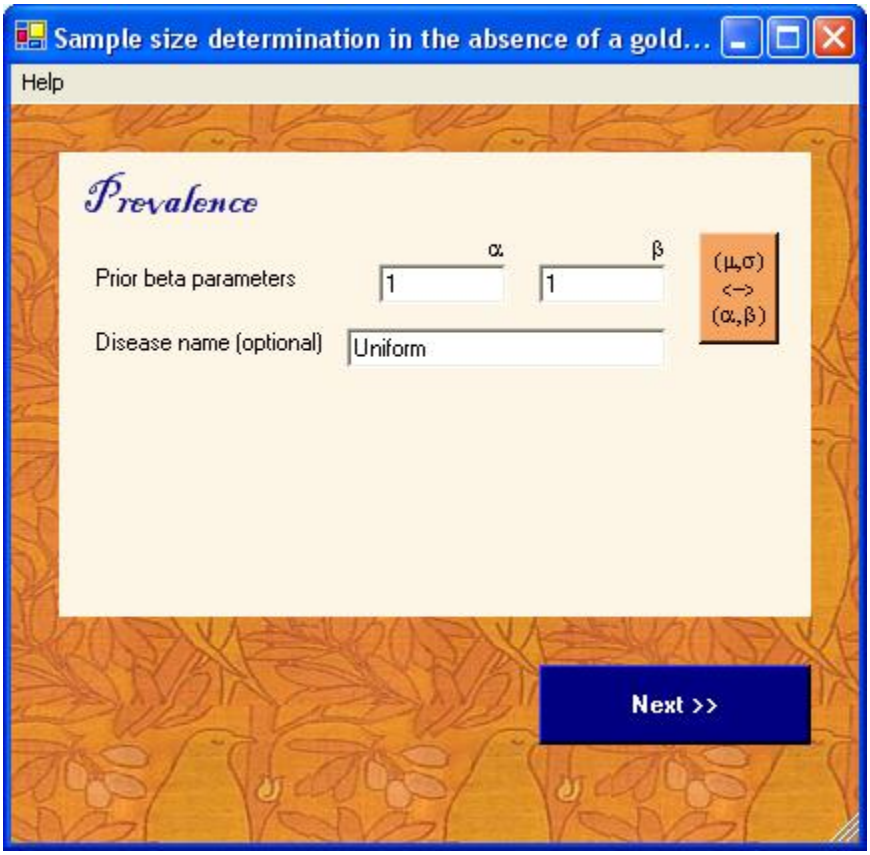
Suppose we want to calculate the sample size necessary to get an average coverage of 0.95 for an HPD region of fixed length 0.1for a new test for Chlamydia trachomatis. Suppose not much is known about this test nor about prevalence, and thus uniform prior distributions are used for prevalence and both sensitivity and specificity of the new test (test 1); a reference test – tissue culture-- is used concurrently, with well known properties, modeled through Beta prior distributions Beta(155.63, 66.15) for sensitivity and Beta(906.68, 3.63) for specificity (please refer to our paper for details on their elicitation).

Click the *Sample size calculations* item menu from the initial form.

As discussed above, we will assume little information is available about the disease prevalence. Thus, enter 1 in both α and β prior parameters cell boxes. Enter a label in the disease name text box: we have entered a label (Uniform) for the prior distribution used.

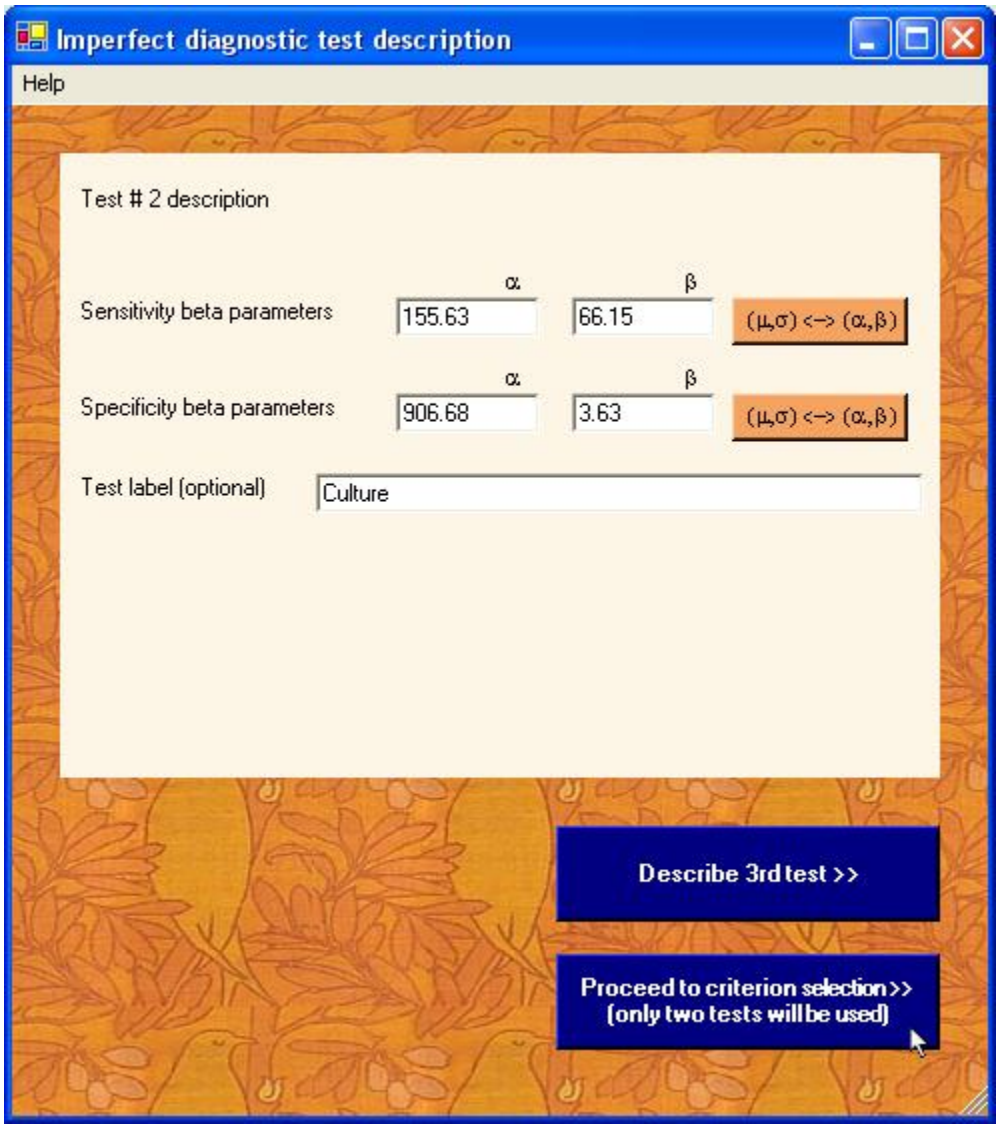
Doing so will make this uniform prior only one click away the next time you run this program.

Do similarly for the new test (test 1) sensitivity and specificity priors (not pictured here).



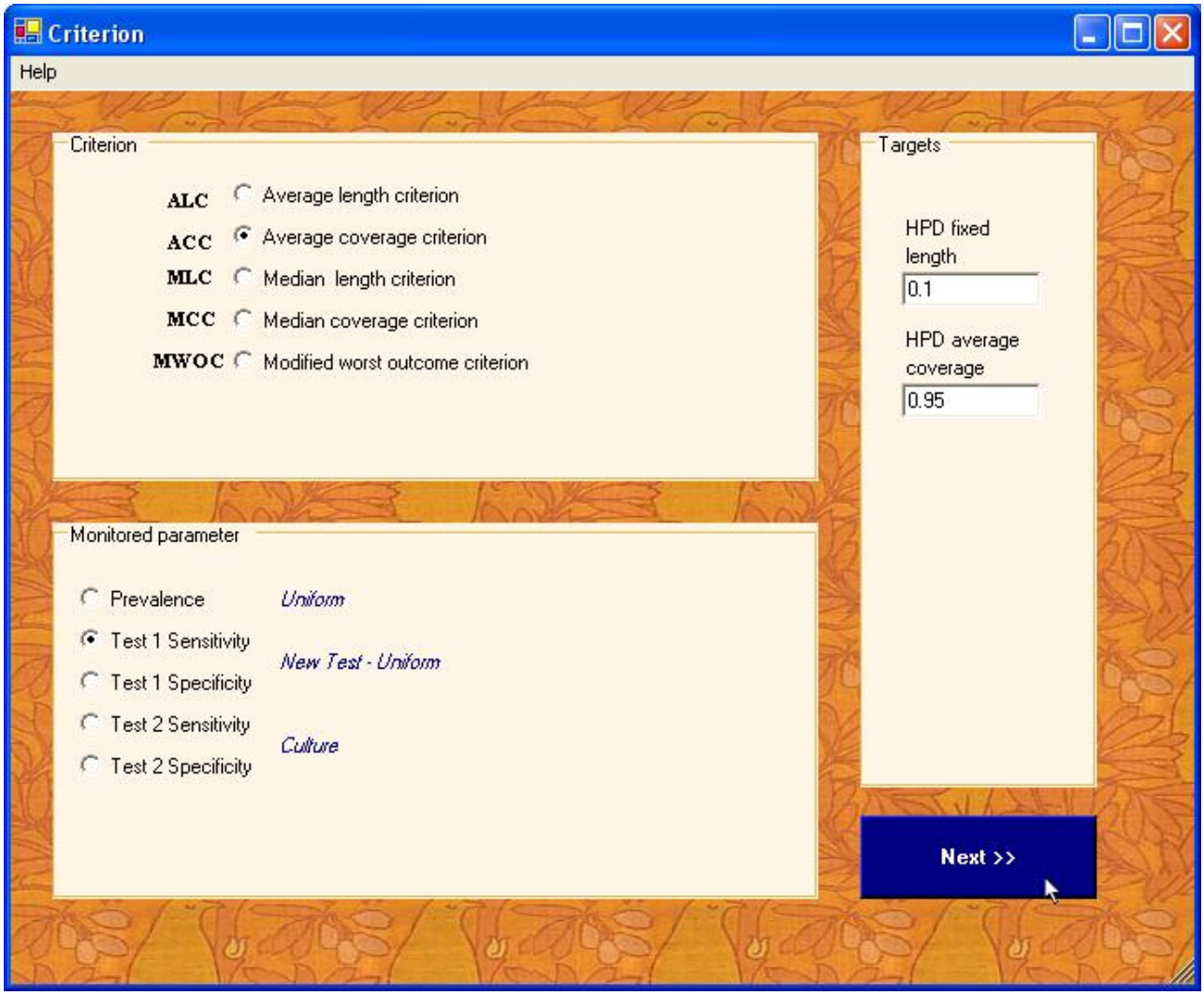
Enter the parameter values discussed above for the reference test (Culture).

Enter a test label and click the *Proceed to criterion selection>>* button.



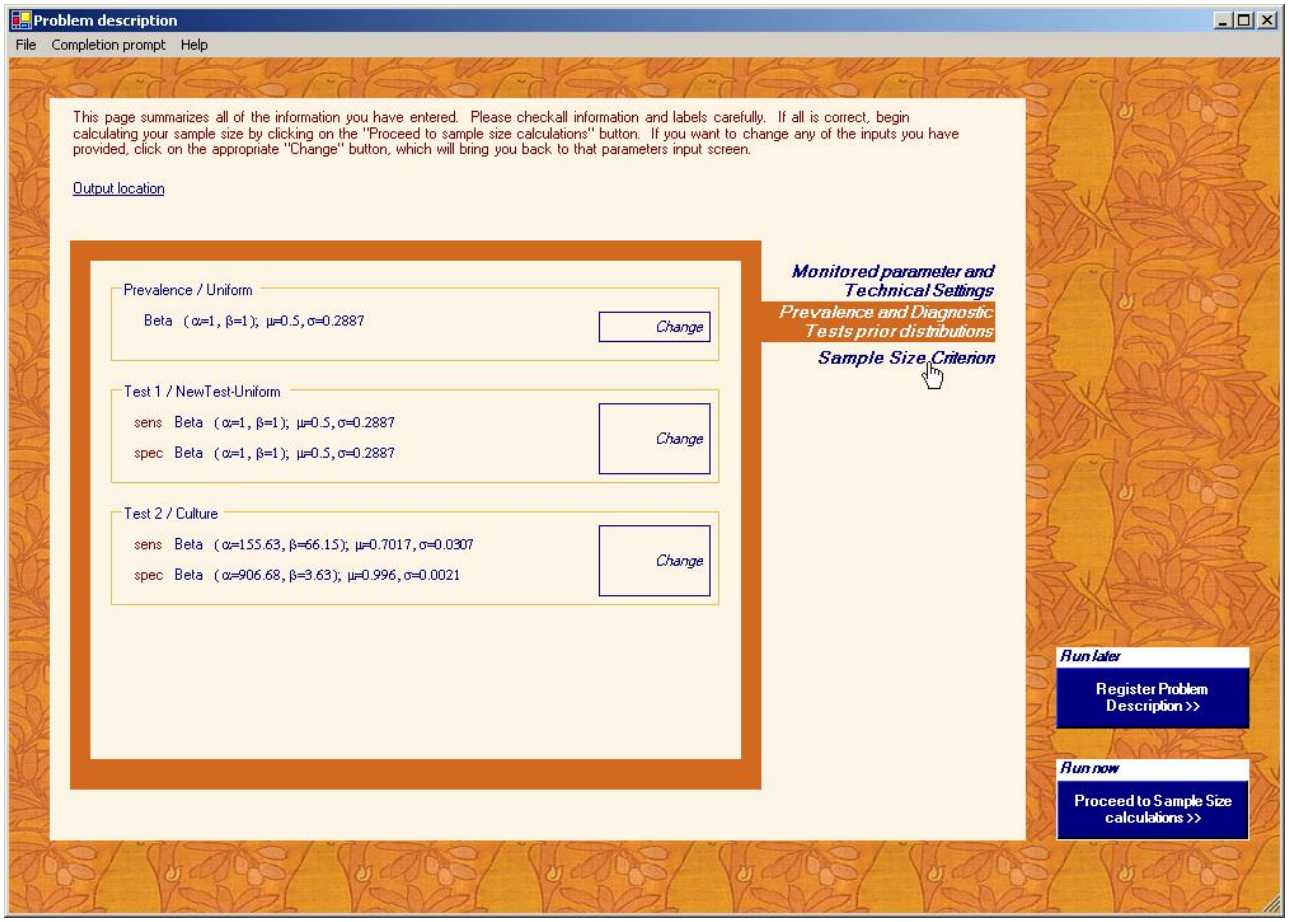
When two diagnostic tests are used, the next form offers the possibility to model conditionally dependent diagnostic tests, if necessary. The default is to assume conditionally independent diagnostic tests, which we will accept in this example by clicking the *Ok>>* button. The modeling of tests dependence is relegated to section 4.

In the next form (below), select the average coverage criterion (ACC), specify the HPD fixed length (0.1), pick *Test 1 Sensitivity* in the *Monitored parameter* group box and click the *Next>>* button.


Criterion
Help
Criterion
ALC
Average length criterion
ACC
Average coverage criterion
MLC
Median length criterion
MCC
Median coverage criterion
MWOC
Modified worst outcome criterion
Targets
HPD fixed length
0.1
HPD average coverage
0.95
Monitored parameter
Prevalence
Uniform
Test 1 Sensitivity
Test 1 Specificity
New Test - Uniform
Test 2 Sensitivity
Test 2 Specificity
Culture
Next >>

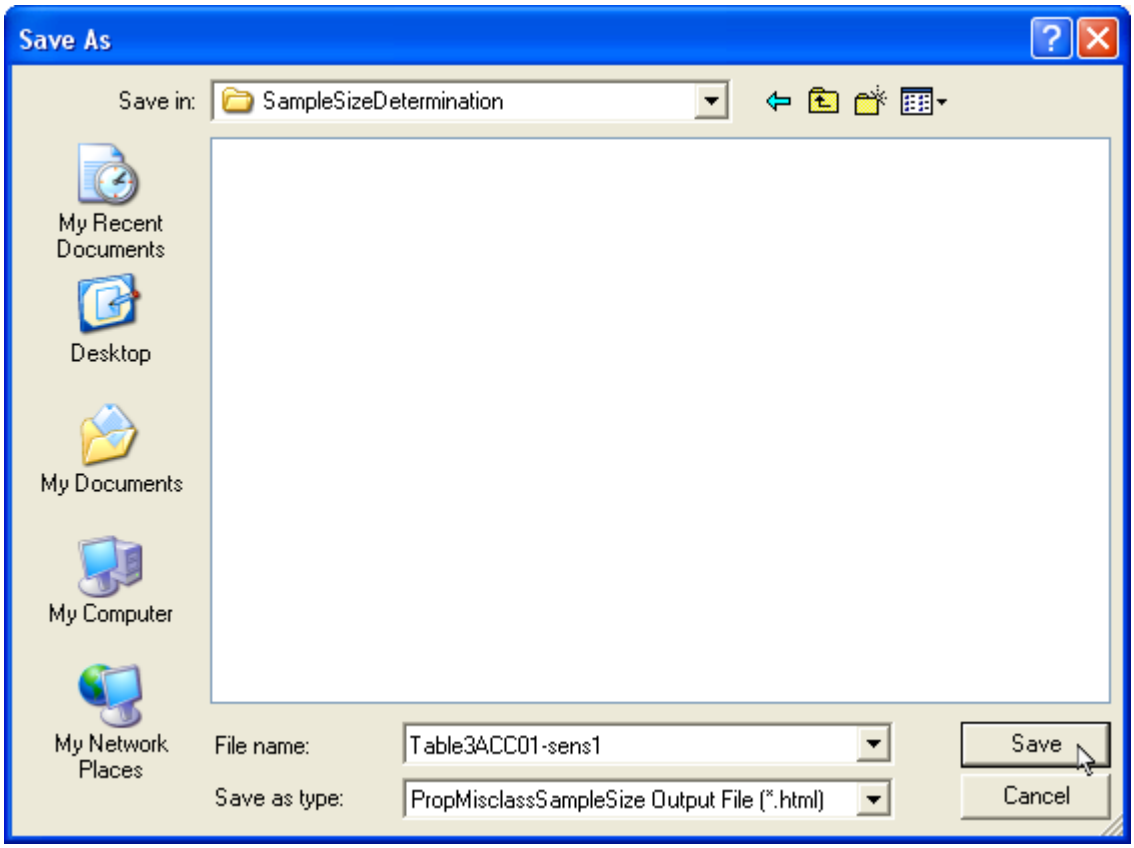
The next form (below) allows reviewing each parameter previously entered. Confirm that all is ok.

Click the item labels *Monitored parameter and Technical Settings*, *Prevalence and Diagnostic Tests prior distributions* and *Sample Size Criterion* to review the different pieces of information you have entered.



Click the Output location item and browse to the desired output file location: enter a file name and click *Save*.

Click the *[Run now] Proceed to Sample Size calculations>>* button on next form.



PropMisclassSampleSize will then start the sample size calculations, launching short R programs and some long WinBUGS scripts with different sample sizes, until it converges to a sample size N for which the selected criterion is met (while it is not met for sample size N-1); the march towards optimal sample size will also stop if, in a series of six consecutive sample sizes, the larger three satisfy the sample size criterion while the smaller three do not, and these six consecutive sample sizes do not span more than 2% of their midpoint value. (The latter stop criterion, while useful, may take effect later than the average user would stop the march towards optimal sample size: see section 3.1.1 for an example where the user may want to stop the sample size calculations manually before the algorithm officially finds convergence.)

Upon completion, the user will be prompted by a form which links to the main html output file and the secondary graphical output files. Note that the main html output file will also contain a section with links to the secondary output files for the ease of future reference.

The next page shows the main body of the html output file obtained for the problem illustrated above.
The first three sections describe the problem in terms of prior distributions parameters, sample size criterion used, fixed HPD length or coverage, target HPD coverage or length and Gibbs specifications (number of burn-in iterations, monitored iterations, etc.). The fourth section reports the outcome (Average HPD coverage, in this case) for each sample size visited along the march towards optimal sample size, where the optimal sample size is highlighted.

Bayesian sample size determination for prevalence and diagnostic studies
in the absence of a gold standard

Prior distributions

Disease prevalence	Beta(α = 1; β = 1)	*Uniform*
Test 1		
Sensitivity	Beta(α = 1; β = 1)	*New Test - Uniform*
Specificity	Beta(α = 1; β = 1)	
Test 2		
Sensitivity	Beta(α = 155.63; β = 66.15)	*Culture*
Specificity	Beta(α = 906.68; β = 3.63)	

Criterion

Criterion	ACC
Main parameter of interest	Test 1 sensitivity
HPD length	0.3
HPD coverage	0.95

Gibbs specifications & technical details

Number of Gibbs iterations to monitor	5000
Number of burn-in iterations	4000
Number of values sampled from preposterior	2500
Initial sample size	1000
Maximum feasible sample size	100,000
Initial step	250
March towards optimal sample size	model-based

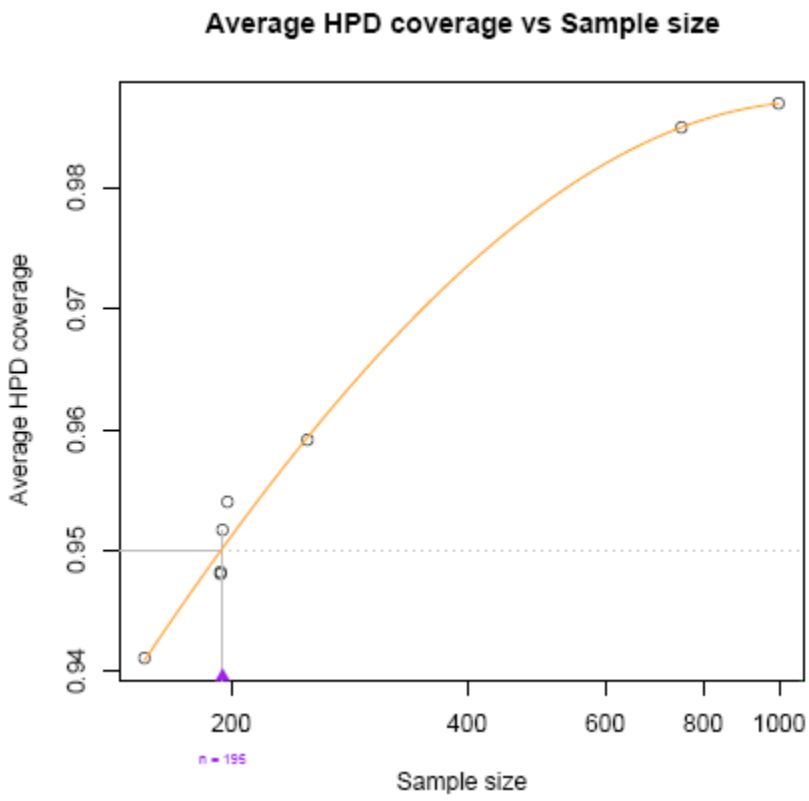
Results obtained along the march to optimal sample size

	Sample size	Average HPD coverage	Number of values sampled from preposterior (when > 2500)
	1000	0.986996	
	750	0.985006	
	250	0.959170	
	155	0.941034	
	194	0.947965	
	198	0.953921	
	194	0.948189	3500
Optimal sample size	195	0.951641	

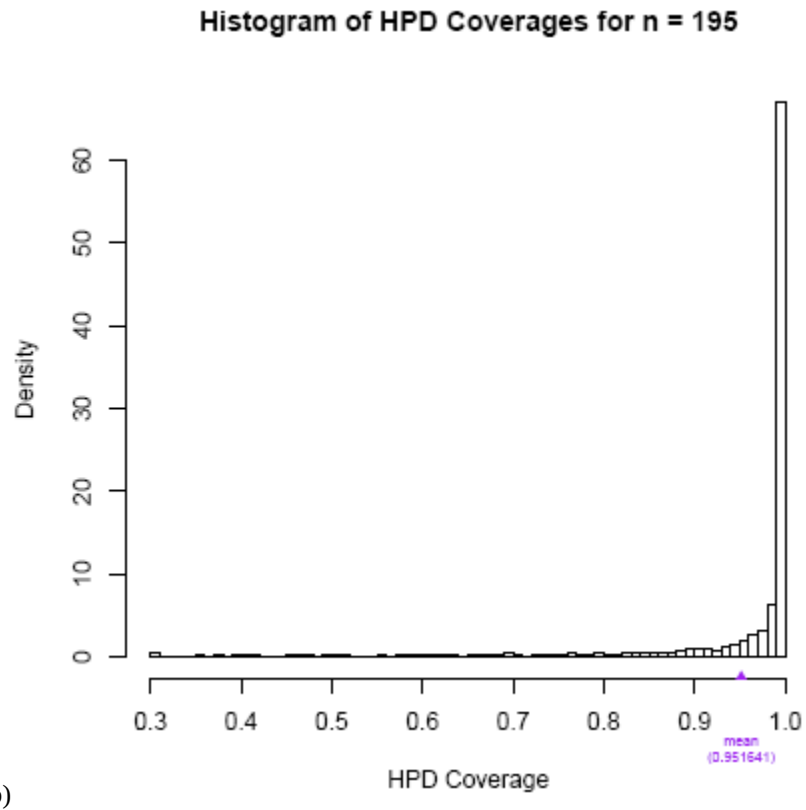
The final section (not shown) of the html output file gives a series of links to secondary graphical output files and the 95% HPD regions' average length for prevalence and each test sensitivity and specificity.

Secondary graphical output files are:

a) a scatter plot of the outcome (in this case the Average HPD coverage) vs sample size;



and b) a histogram of the HPD coverages obtained for each data point sampled from the preposterior at the estimated optimal sample size.



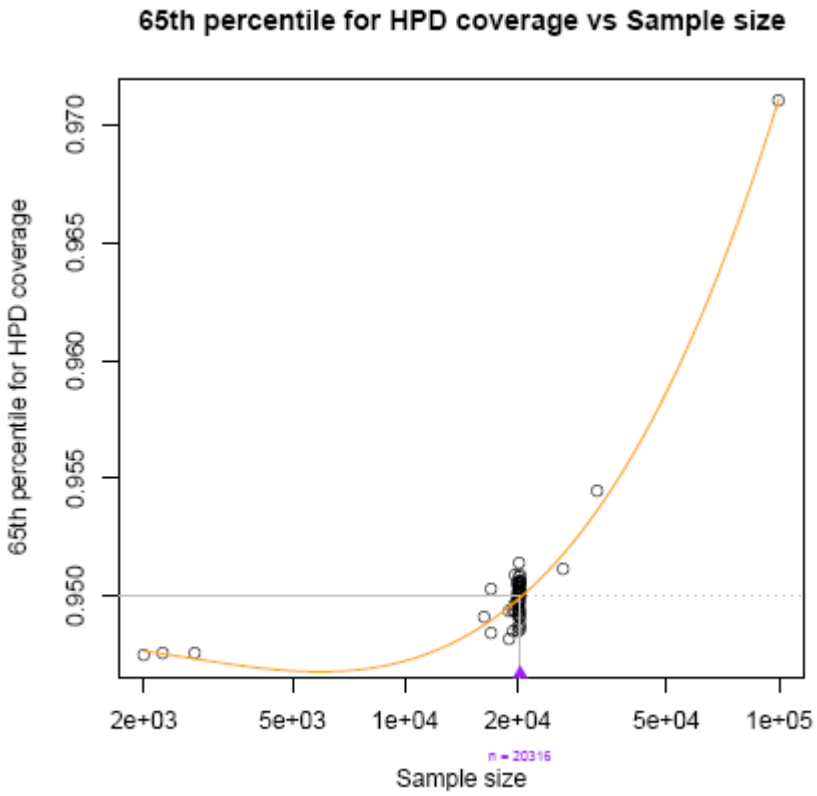
b)

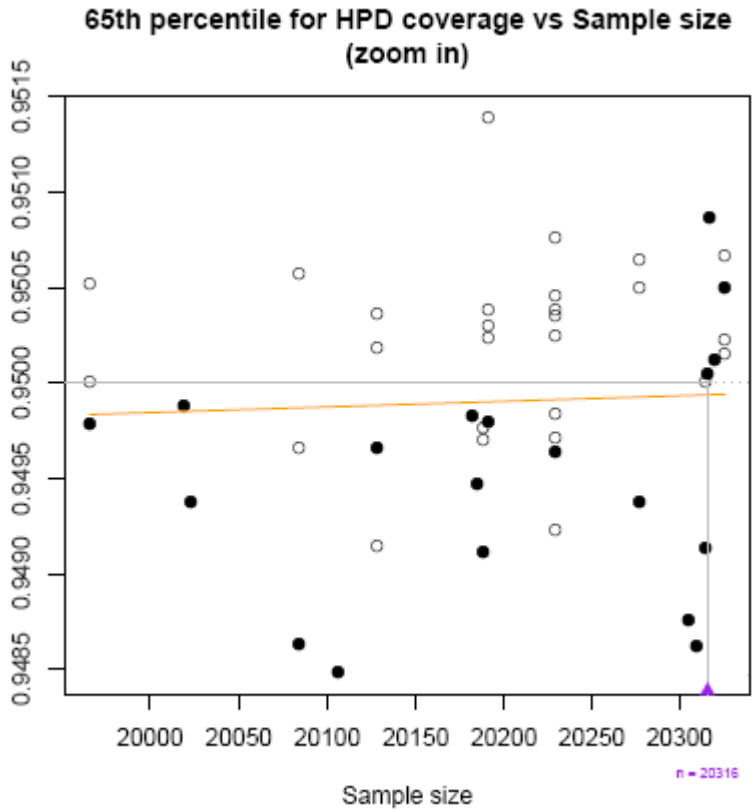
Figure shown in (a) gives an idea of the uncertainty around the optimal sample size found while figure shown in (b) gives an idea of the likelihood of HPD coverages lower than that assured *on average*.

The superimposed orange curve shows the estimated outcome as a function of sample size as estimated from a relationship that has proven to work well in every problem we have run with **PropMisclassSampleSize** thus far: however, we have not developed a theoretical background that would ensure its safe use in every possible problem. If you use the optional model-based approach suggested in **PropMisclassSampleSize**, this best-fitting line will be used to hopefully get to the optimal sample size in fewer steps (hence hopefully faster) than would a bisectional search. However, the end criterion is not based on that best-fitting equation, that is, the algorithm stops when it finds a sample size N that was estimated to be sufficient while N-1 was estimated not to be sufficient (by MCMC sampling in both cases, that is, we are not relying on a mathematical solution of the equation).

Note that the main html output file is created at the very beginning of the algorithm and updated often in the process: when used for a sample size calculation, for example, **PropMisclassSampleSize** will update the html output file after the estimation of the outcome for each sample size visited in the march towards the optimal sample size. Thus the user can sometimes have an idea of the final sample size even before the program actually completes, by looking at this evolving file while the program is still running.

Sometimes it is difficult to view the points near the optimal sample size, as in the example to the right. In these cases, the output file includes a second page with a zoom-in of the region around the optimal sample size. When there are one or more estimates at a given sample size, the last one is represented by a dark dot. Since each new outcome estimate for a given sample size is done with higher preposterior sample, dark dots therefore present the most accurate estimate. See second graph below for an example of a zoomed-in graph.





3.1.1 Assessing convergence

As discussed in the above section, the main html output file is regularly updated while **PropMisclassSampleSize** is running. Consulting the main html output file in the process may give the reader enough information in his search for an optimal sample size before the algorithm converges, especially when the final sample size is expected to be large.

The html output file excerpt below displays the intermediate results in a sample size calculation: in this example, we are searching for the minimal sample size such that HPD average length is 0.205. The upper part of this output file section lists each sample size visited in the process thus far, along with the estimated HPD average length; entries are listed in sequential order, according to when they were run in the program.

When a sample size appears more than once, the last reported represents an average over all runs for the averaged criteria, and represents an overall percentile for the MWOC.

Sample size N	Average HPD length	Number of values sampled from preposterior (when ≠ 1000)
1000	0.2496273	
1250	0.2477043	
1750	0.2362303	
2738	0.2310551	
100,000	0.1717847	
9798	0.2160742	
24,434	0.2029393	
22,122	0.2015906	
20,131	0.2018398	
18,950	0.2045491	
18,824	0.2033153	
17,662	0.2066576	
18,100	0.199496	
17,662	0.2081477	2000
17,916	0.2079913	
18,100	0.202457	2000
17,916	0.2068952	2000
18,100	0.2032281	3000
17,916	0.2046232	3000
17,662	0.2072427	3000
17,916	0.2048624	4000
17,662	0.2067979	4000
17,916	0.2053905	5000
18,100	0.204038	4000
17,916	0.2049111	6000
17,662	0.2054954	5000
17,916	Now running	

The same numbers are repeated in a subsequent block (below) in a different order, sorted by sample size; only the last estimate for each sample size is reported.

sorted by sample size:

1000	0.2496273		
1250	0.2477043		
1750	0.2362303		
2738	0.2310551		
9798	0.2160742		
17,662	0.2054954	5000	
17,916	0.2049111	6000	Now re-running
18,100	0.2040380	4000	
18,824	0.2033153		
18,950	0.2045491		
20,131	0.2018398		

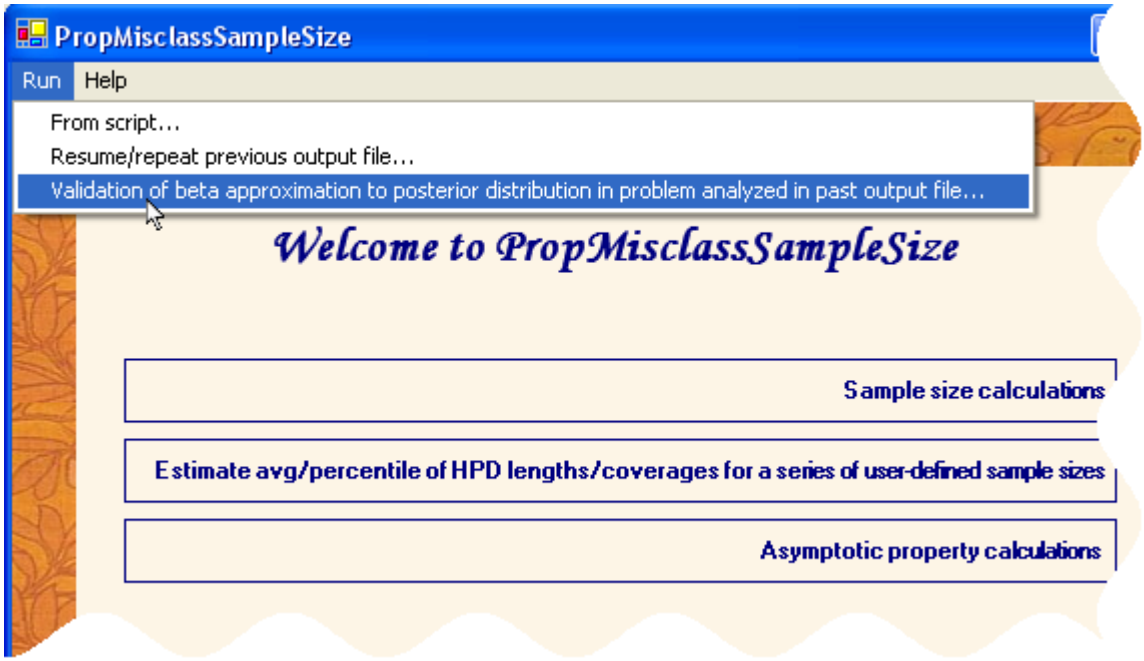
The algorithm has – for technical reasons – re-estimated the average length for sample sizes 17,662, 17,916 and 18,100. While the stopping criterion has not yet been met (and **PropMisclassSampleSize** is running the sample size 17,916 again), the average user may be satisfied with the knowledge already gained in the process and stop **PropMisclassSampleSize** at this stage. Indeed, the numbers above already give a strong indication than a sample size close to n = 17,900 or n = 18,000 meets the criterion, which may be precise enough in most applications.

3.2 A validation of the Beta approximation for the posterior density

The posterior density of the parameter of interest is approximated by a Beta density, no matter which parameter is used. This approximation has shown to be very good in each problem we have studied, but its appropriateness has not been shown to be universal by any theoretical demonstration. Therefore, it is always a good idea, once an optimal sample size for a particular problem was determined through **PropMisclassSampleSize**, to validate the appropriateness of that ad hoc approximation.

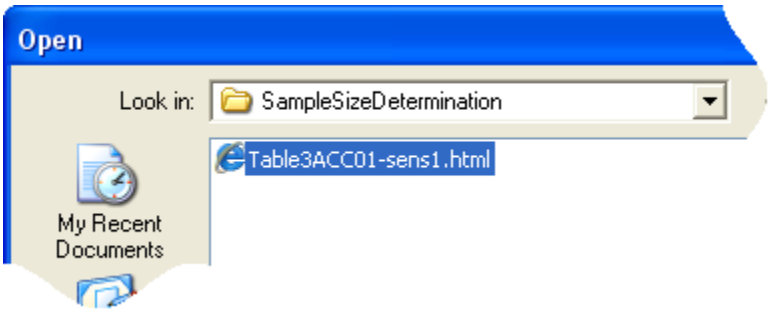
In this section, we will run **PropMisclassSampleSize** to validate the Beta approximation for the optimal sample size obtained for the problem registered in section above.

From **PropMisclassSampleSize** initial form, select the *Run/Validation of beta approximation to posterior distribution in problem analyzed in past output file…* top-left menu item.



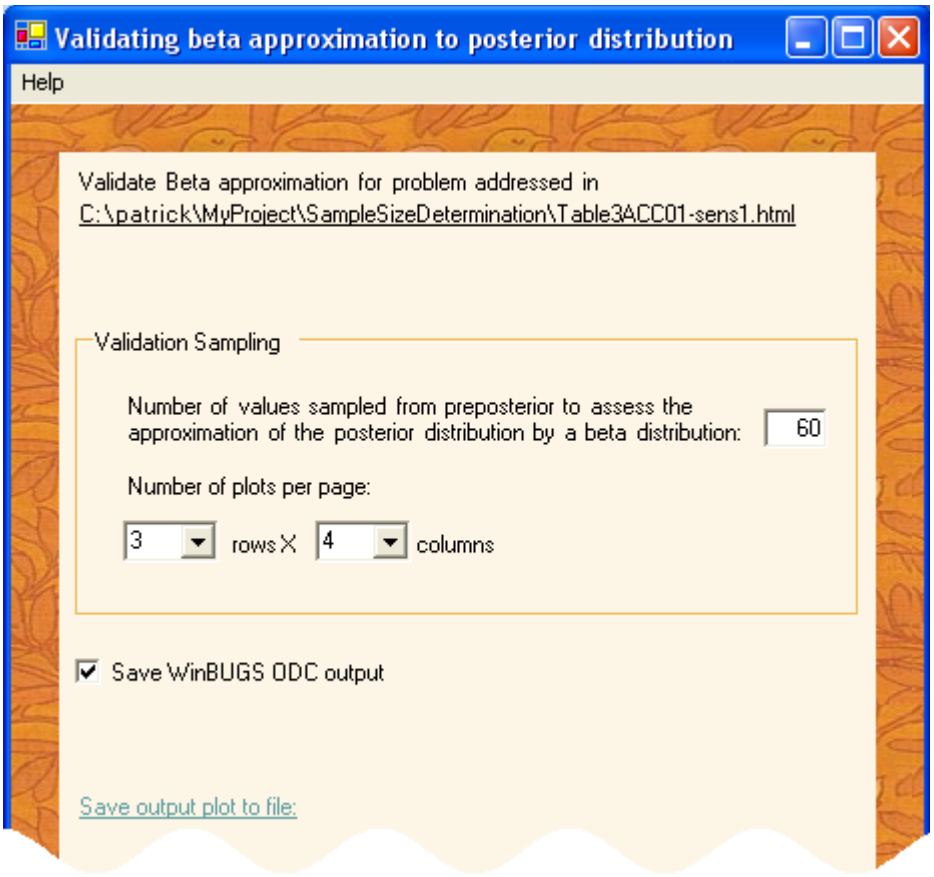
Browse through previous output file for which the Beta approximation is to be validated.

Click *Open*.



The next form allows the user to enter the number of values sampled from preposterior to assess the Beta approximation for the posterior density of the parameter under study.

Even though a large preposterior sample size (of the magnitude of thousands of samples) was used in the original problem, this time the sample does not need to be very large to form an opinion on the appropriateness of the Beta approximation. We suggest to run it for 60 samples by default, but this can of course be changed. Remember, however, that a histogram of the values obtained in the MCMC WinBUGS program run will be drawn for each sample, with the distribution function of the best-fitting Beta density superimposed. This means we have to monitor and save the values of the parameter of interest for each WinBUGS iteration, which is demanding in both computer time and resources.



By default, each page of the pdf output file will display 3 rows and 4 columns of histograms.

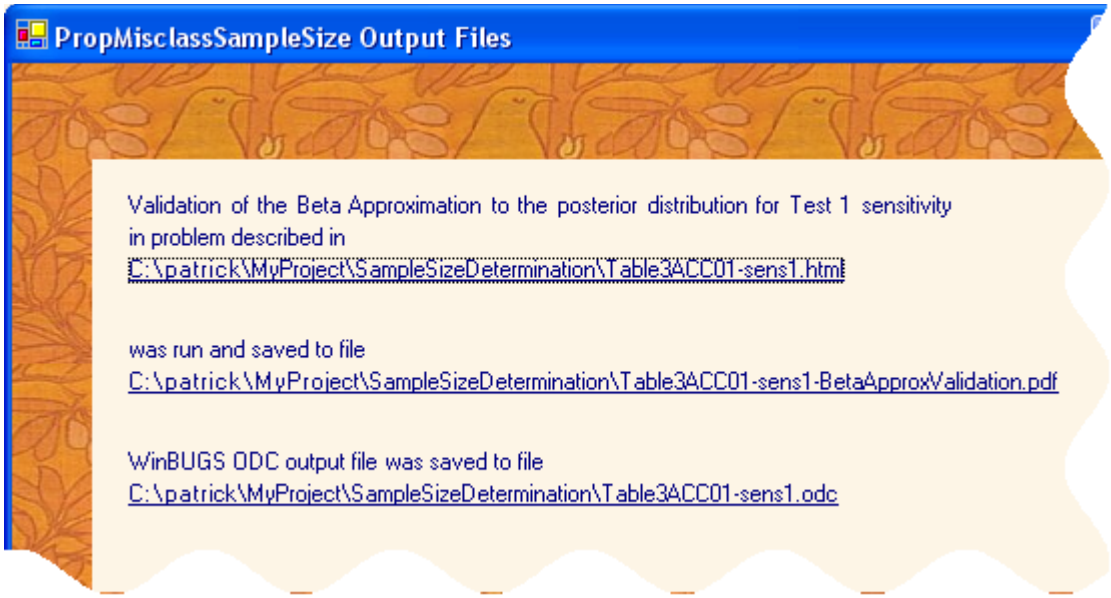
The WinBUGS ODC output file can also be saved, allowing the informed user to monitor the appropriateness of the chosen number of burn-in and monitored iterations, among other things. Note that the size of the latter odc output file can become humongous with larger preposterior sample sizes. Click the Save output plot to file item in the lower part of the form to select the pdf output file location: make sure not to overwrite an already existing pdf file. When done, click the *Ok>>* button.

The next form displays the optimal sample size obtained for this problem.

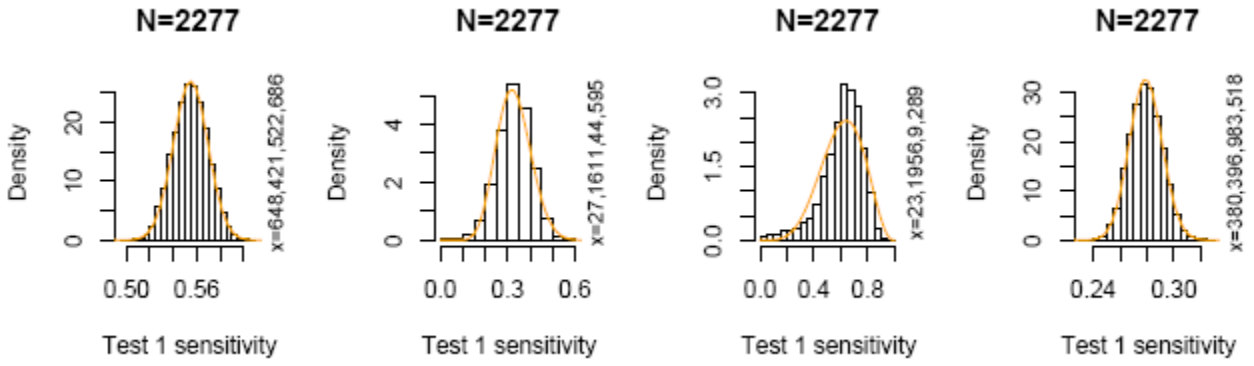
By default, the Beta approximation validation check will be run for that optimal sample size only, but you can also add additional sample sizes, should you consider sampling more or less subjects than indicated and wish to validate the Beta approximation for these alternative sample sizes as well. Click the *Next>>* button to proceed with optimal sample size only.

A Problem Reviewal form will be displayed, allowing you to modify the parameters entered. Some parameters, however – such as number of burn-in and monitored iterations – are not modifiable as it would not make sense to check the Beta approximation validation with technical parameters different from those used in original problem. Click *[Run now] Proceed>>* when ready.

Upon completion, a form will pop-up, offering you links to the (modified) main html output file, the .pdf file with the Beta Approximation overimposed on the histograms and, if requested, to the WinBUGS odc output file.



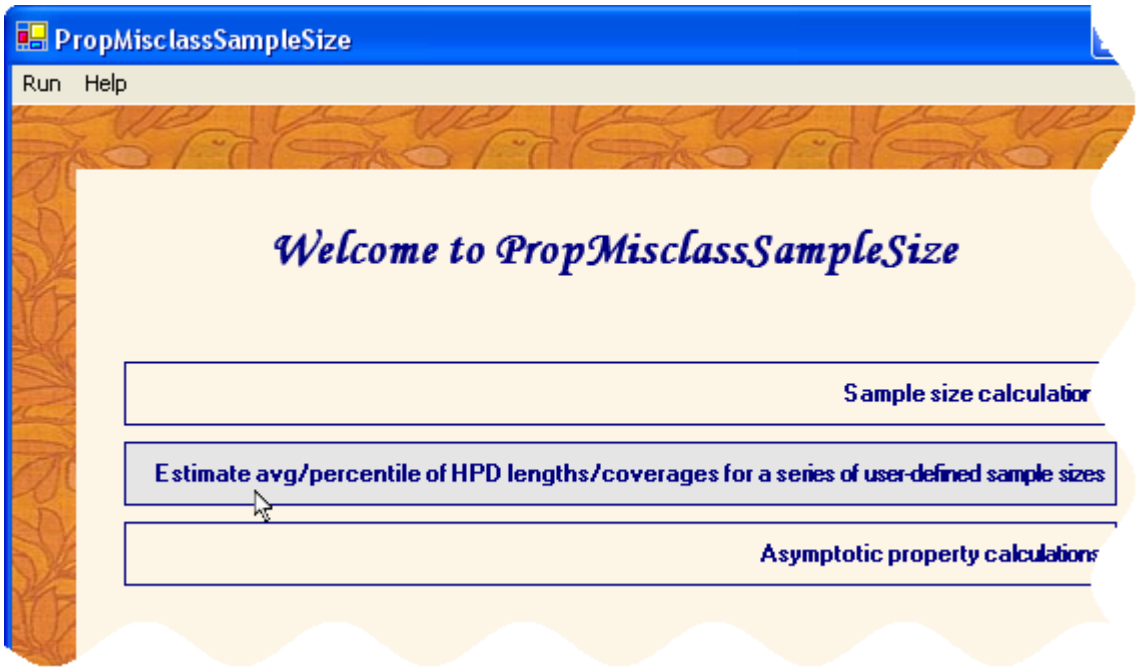
A link to the Beta approximation validation check pdf output file will be added to the main html output file.
Below is an excerpt of the Beta approximation validation check pdf output file: the four figures display, respectively, the histogram for Test 1 Sensitivity values sampled in the WinBUGS MCMC run for four samples with N=2277: the superimposed orange lines show the distribution function of the best-fitting Beta distributions and all show a more than decent fit, which should reassure the user of both the appropriateness of the Beta approximation for the posterior distribution for Test 1 Sensitivity and for the appropriateness of the optimal sample size returned, given the prior information at hand. Each histogram represents the results from one two by two table. The x values listed provide the values in the table, given in row order.



3.3 Running PropMisclassSampleSize with pre-determined sample sizes

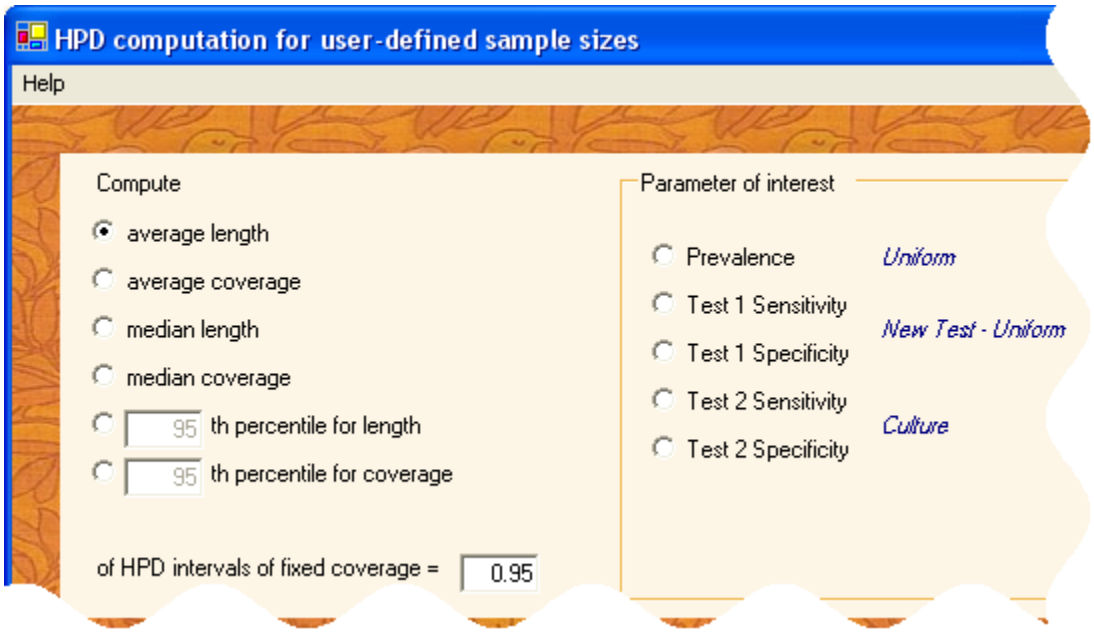
PropMisclassSampleSize can also be used to estimate a given outcome (e.g. the average coverage of HPD regions of fixed length) for fixed sample sizes. Indeed, suppose one knows it will not be possible to recruit more than 2000 subjects in a study, but would still like to obtain an idea, beforehand, of the coverage of HPD regions of fixed length 0.1 in the same context as that illustrated in section 3.1.

From **PropMisclassSample** initial form, click the second box.

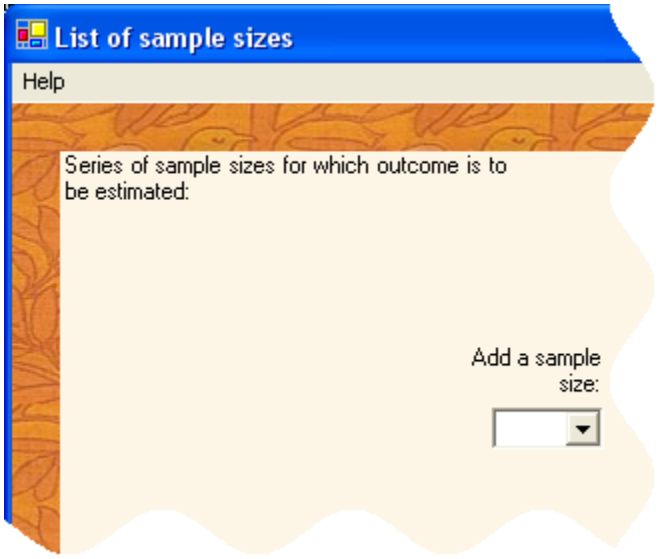


The next forms are identical to those presented in section 3.1 and are used to enter the prior distributions for disease prevalence and the different test(s) sensitivities and specificities. We therefore skip showing these steps here.

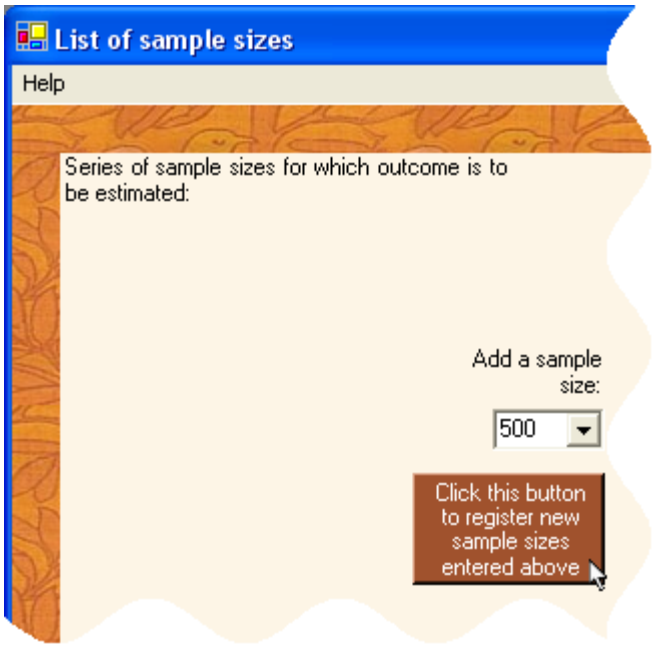
The next form allows the user to pick the outcome of interest and specify either of the fixed HPD length or coverage, as well as to pick the parameter of interest.



The next form is used to specify the sample sizes for which the above-specified outcome is to be estimated.

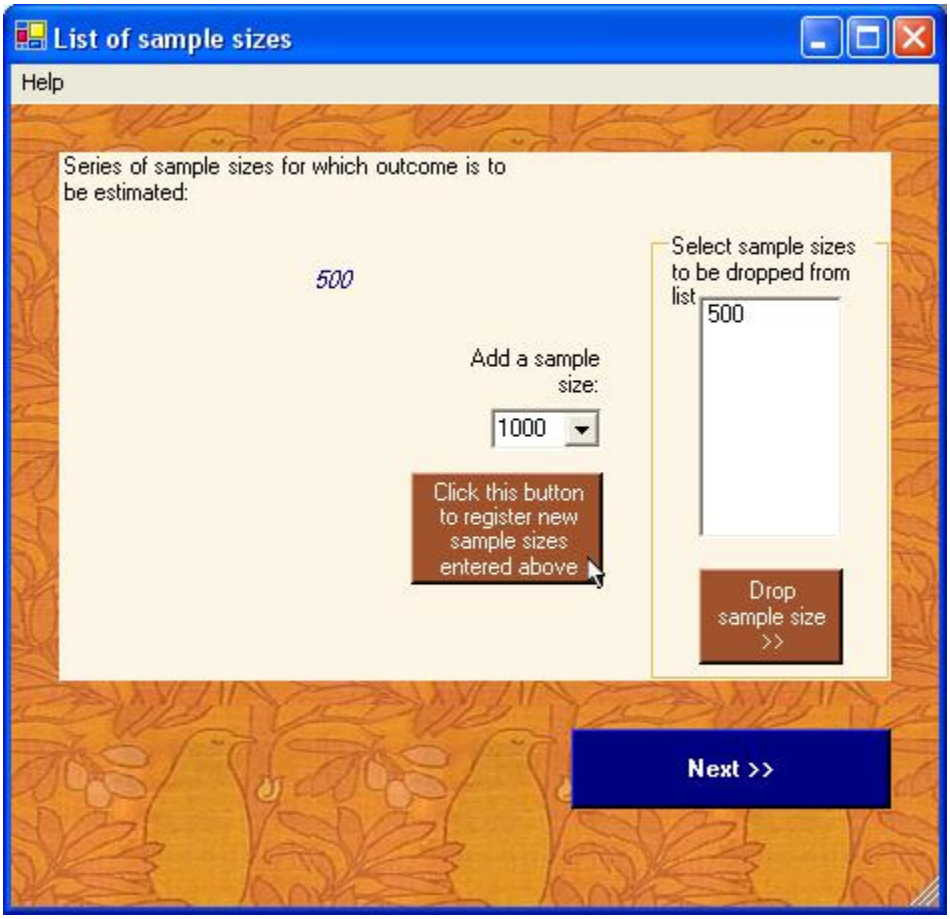


Enter 500 in the *Add a sample size* text box and click the button underneath to register this sample size.

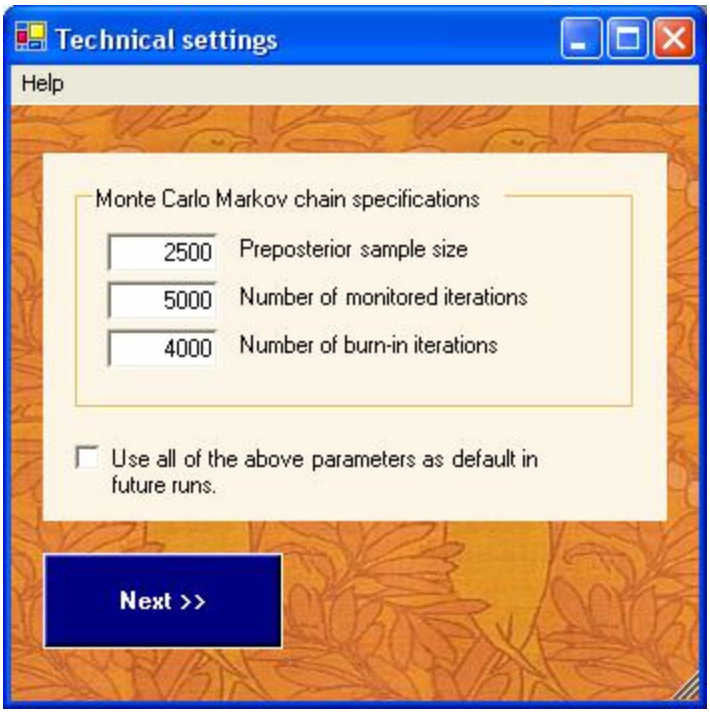


Proceed the same way for each sample size of interest.

Click *Next>>* when done.



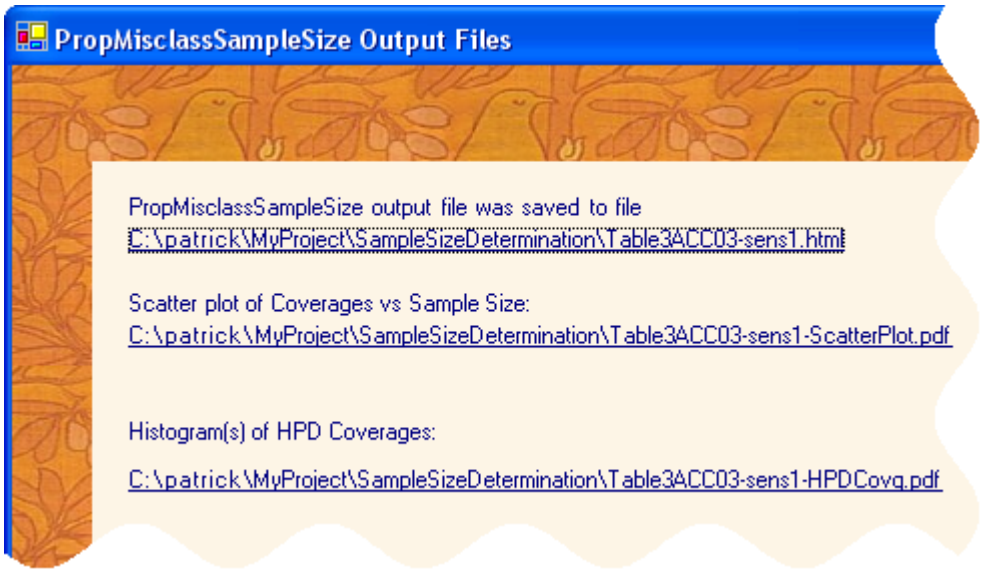
The next form – Technical settings – is simplified when compared to that presented in section 3.1, as the sample size search algorithm settings are irrelevant here.



The last form is the Problem Reviewal form, similar to that presented in section 3.1.

When done, **PropMisclassSampleSize** will prompt a form with links to the main html output file and to secondary graphical pdf output files.

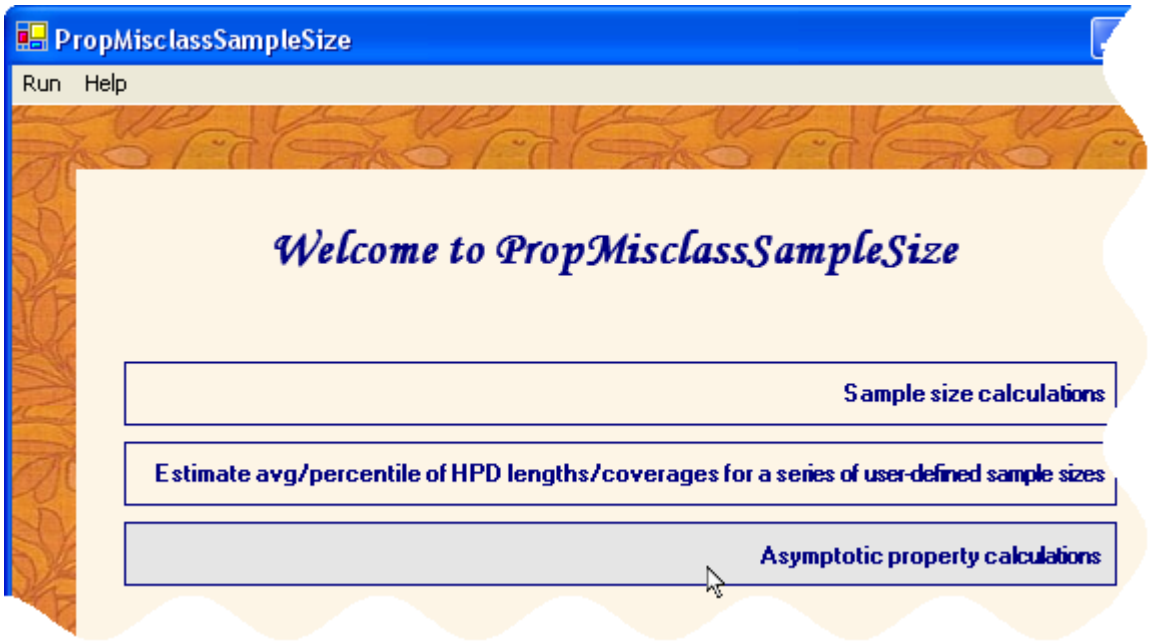
Whether this form will be prompted full size or minimized (then only noticeable in the taskbar) depends on the corresponding option in the Problem Reviewal form.



When running multiple scripts through the *Run from script…* top-left menu from the initial form, the above final form with links to output files will NOT appear.

3.4 Running asymptotic calculation

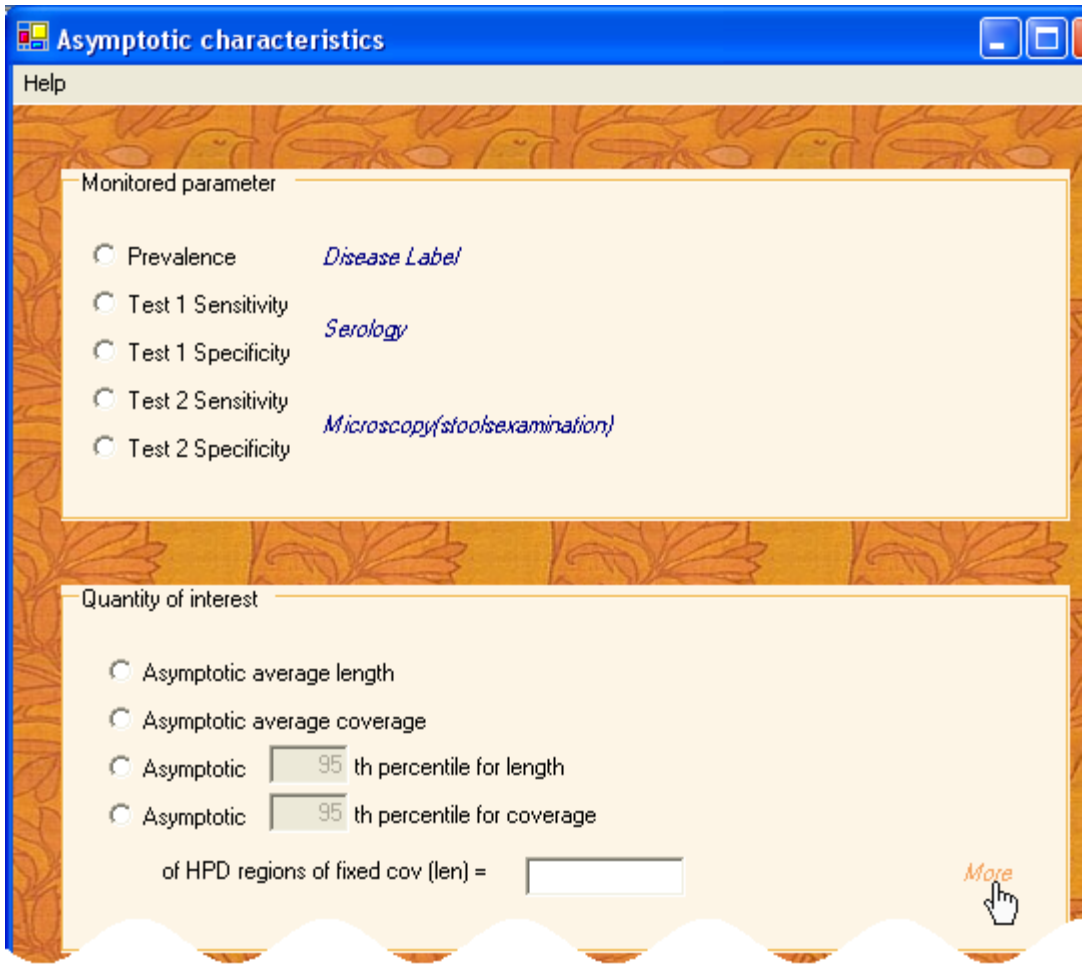
The third menu choice in the **PropMisclassSampleSize** initial form is to run asymptotic property calculations.



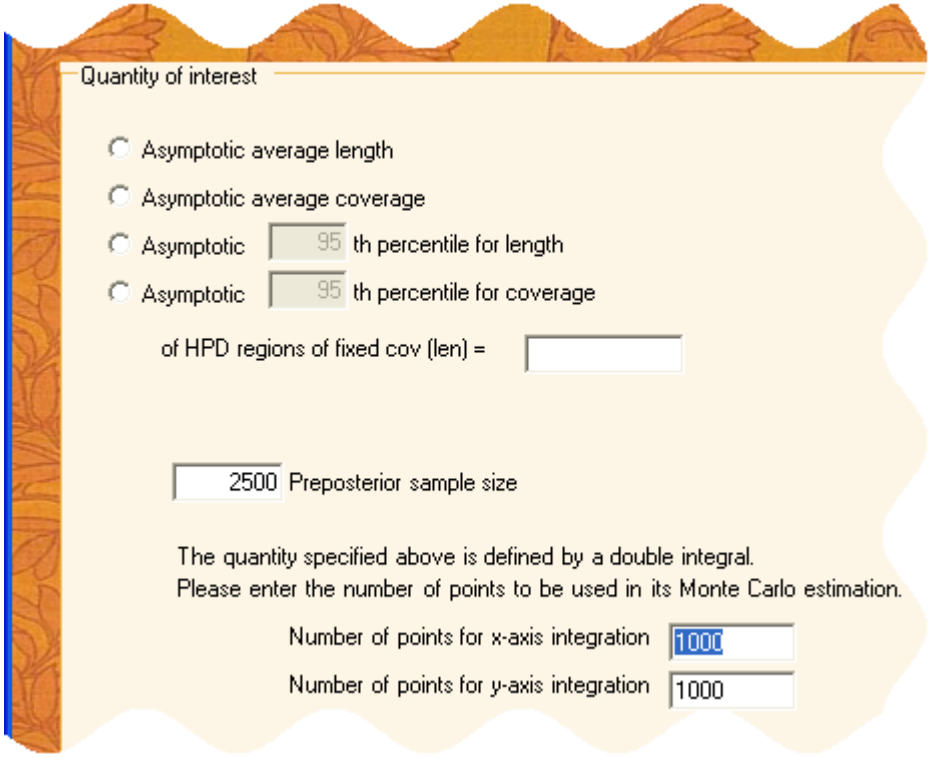
After selecting that option, the user is asked for prior parameters for prevalence and for each test used, through the same forms as presented in earlier sections. We therefore do not display these steps here, and move on to forms that are new.

The first form to differ from previous options is that where the asymptotic outcome is to be selected (right).

One extra option is modifiable if you click the *More* item in the bottom right portion of the form.



When computing asymptotic HPD lengths or coverages, a double integral must be computed, which is estimated by numerical integration; x-axis (representing the main parameter of interest) is divided in a number of equal length sections, the number of sections being given by the value of *Number of points for x-axis integration*. The y-axis, representing the only other free parameter (to be integrated out), is also divided in a number of equal length intervals, the number of intervals being fixed by the value defined by *Number of points for y-axis integration*.

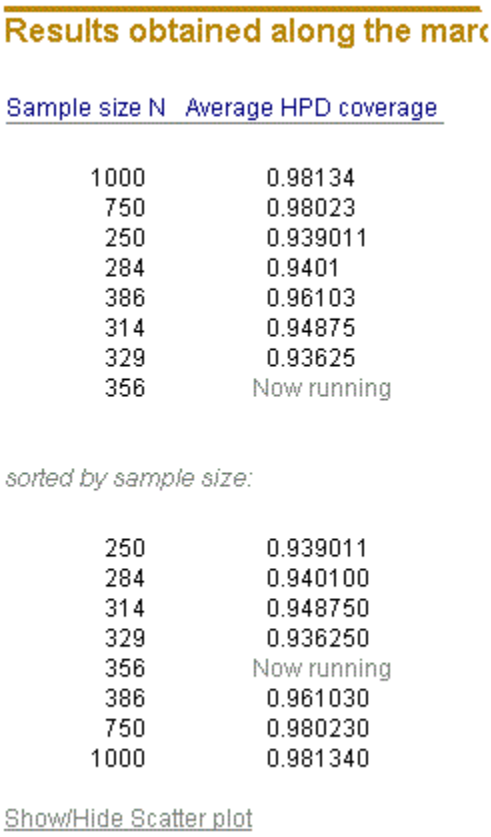


The larger the values for these parameters, the greater the precision in the HPD lengths or coverages estimations, but the longer the computations will take to run.

Finally, a Problem Reviewal form allows the user to revisit each parameter entered in the problem description and to launch the asymptotic calculations now or to register the problem description to a script for a future run.

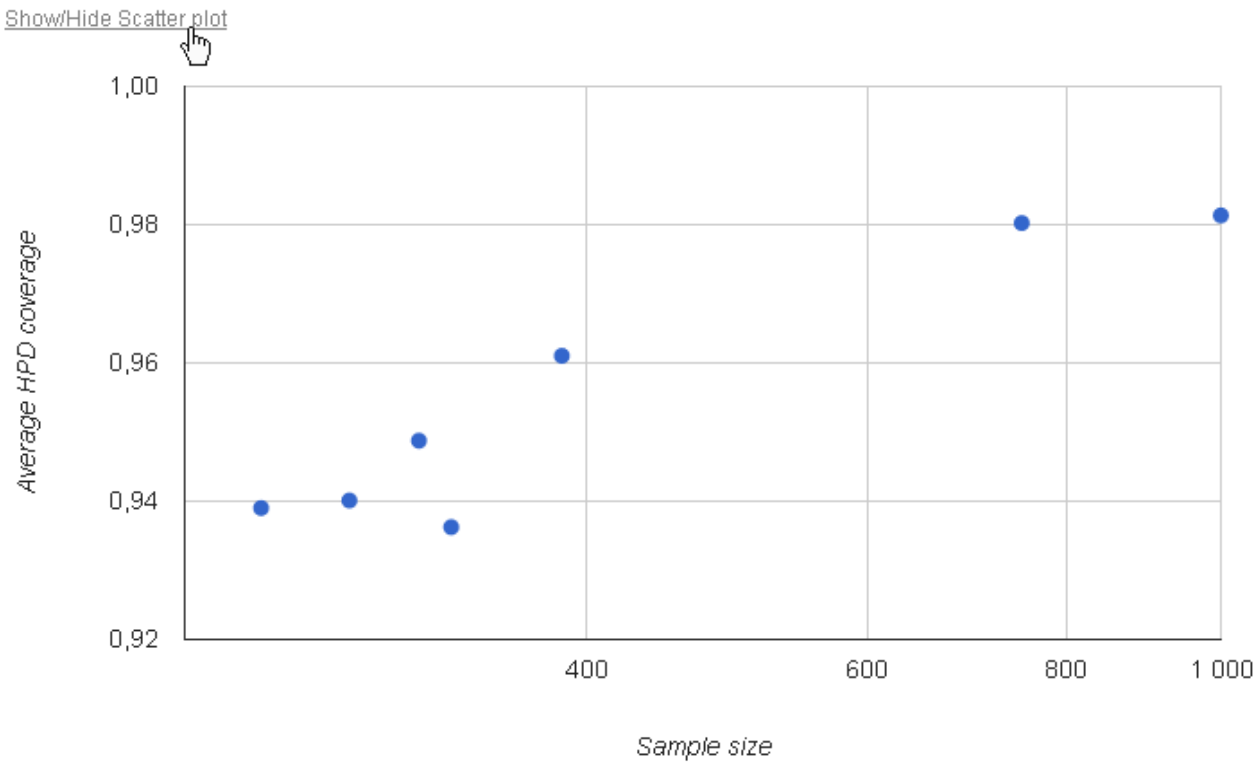
3.5 Iterative html output file updates

Whether you are running **PropMisclassSampleSize** for a sample size calculation or for outcome estimation for a series of predetermined sample sizes, you may be interested in having a look at intermediate results while **PropMisclassSampleSize** is running. The main html output file is updated after the outcome of interest is estimated for each sample size, and viewing it in your favorite browser is possible at any point in time.



The output section labeled **Results obtained along the march to optimal sample size** is divided into two subsections: the upper portion lists the series of sample sizes along with their corresponding estimated outcome, in the order in which they were run. This is followed by a subsection where the same results are listed in ascending order of sample size, which may be easier to follow. Note that if the outcome had to be re-estimated for a given sample size (along the search for optimal sample size), that sample size will appear two or more times in the upper section, once per re-estimation, while only the final estimate (which summarizes information from each intermediate outcome estimate) will appear in the lower (sorted) section.

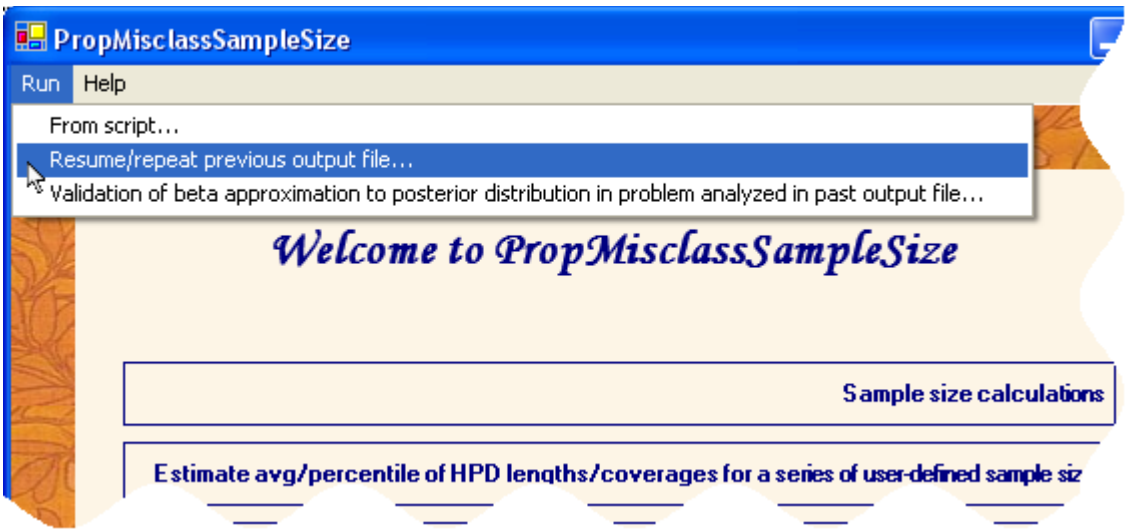
Finally, a link labeled *Show/Hide Scatter plot* opens a scatter plot (see example below) which helps visualize the relationship between outcome and sample size; it may even give enough information to the user with regards to the final sample size, even though the program has not formally reached convergence, and the user may decide to stop the program before it does converge. Note that when **PropMisclassSampleSize** finishes, a .pdf document presents the same scatter plot with a little bit more information.



3.6 Resuming/repeating a previous analysis

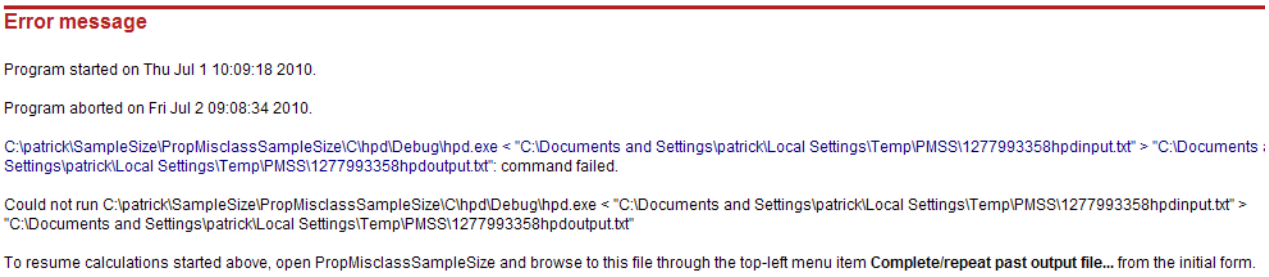
It is possible and easy to resume or repeat (with modification to the prior distribution of one ore more parameters, or with a new sample size criterion, for example) a problem that was already run with **PropMisclassSampleSize**.

Indeed, the top-left menu from the initial form offers the option of doing so.



3.6.1 Resuming a previous analysis

Some errors, such as due to a computer crash, to the need to reboot your system while **PropMisclassSampleSize** was still running, or if you inadvertently stopped a WinBUGS script that was launched by **PropMisclassSampleSize,** might lead you to want to continue running a previous run of the program. Regardless of the reason for an interruption of the program, an error message such as the one reproduced below will be printed at the bottom of the html output file.



Error message

Program started on Thu Jul 1 10:09:18 2010.

Program aborted on Fri Jul 2 09:08:34 2010.

C:\patrick\SampleSize\PropMisclassSampleSize\C\hpd\Debug\hpd.exe < "C:\Documents and Settings\patrick\Local Settings\Temp\PMSS\1277993358hpdinput.txt" > "C:\Documents a
Settings\patrick\Local Settings\Temp\PMSS\1277993358hpdoutput.txt": command failed.

Could not run C:\patrick\SampleSize\PropMisclassSampleSize\C\hpd\Debug\hpd.exe < "C:\Documents and Settings\patrick\Local Settings\Temp\PMSS\1277993358hpdinput.txt" >
"C:\Documents and Settings\patrick\Local Settings\Temp\PMSS\1277993358hpdoutput.txt"

To resume calculations started above, open PropMisclassSampleSize and browse to this file through the top-left menu item **Complete/repeat past output file...** from the initial form.

To resume the problem, select the *Resume/repeat previous output file…* top-left menu item from the initial **PropMisclassSampleSize** form and load the incomplete html output file: **PropMisclassSampleSize** will then resume from where it stopped (or was stopped!).

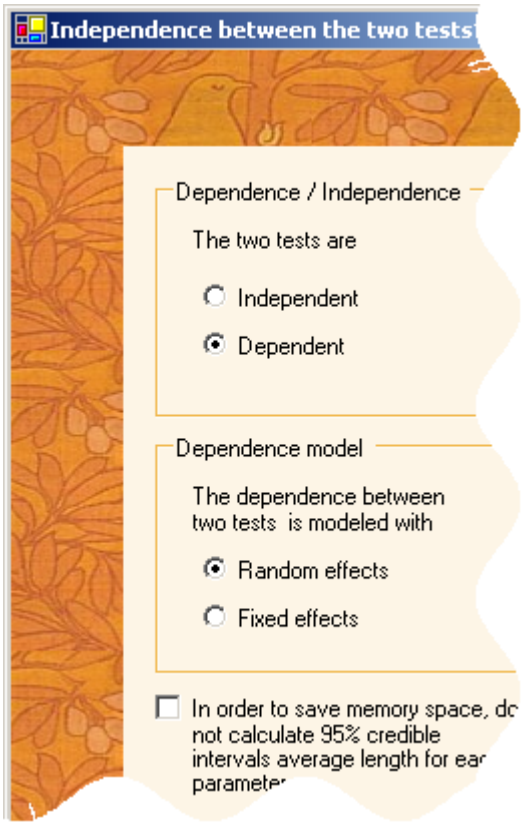
If the same error occurs again, it might be a sign of a problem inherent to **PropMisclassSampleSize**. Please do not hesitate to contact us if you cannot think of a solution to the problem or if the error message is unclear. See our contact email address at the end of this document.

3.6.2 Repeating a previous analysis

The *Resume/repeat previous output file…,* as its name indicates, can also be used to repeat an analysis previously done with **PropMisclassSampleSize**. If the original analysis was done with fair precision (i.e., with a large number of samples from preposterior and a decent number of monitored iterations), there is not much interest in actually repeating the same analysis (even though you would almost surely obtain at least slightly different results by doing so, as the whole process is subject to Monte Carlo error). This option, however, can advantageously be used to rerun an analysis with slight modifications, such as different hyperparameters for the prior distribution of one or more parameter, a different sample size criterion, or even for focusing the inference on a different parameter. When loading a **PropMisclassSampleSize** html output file that completed with success, this option will bring you directly to the final Problem Reviewal form which, as already seen earlier in this document, allows the user to modify almost every single problem description parameter.

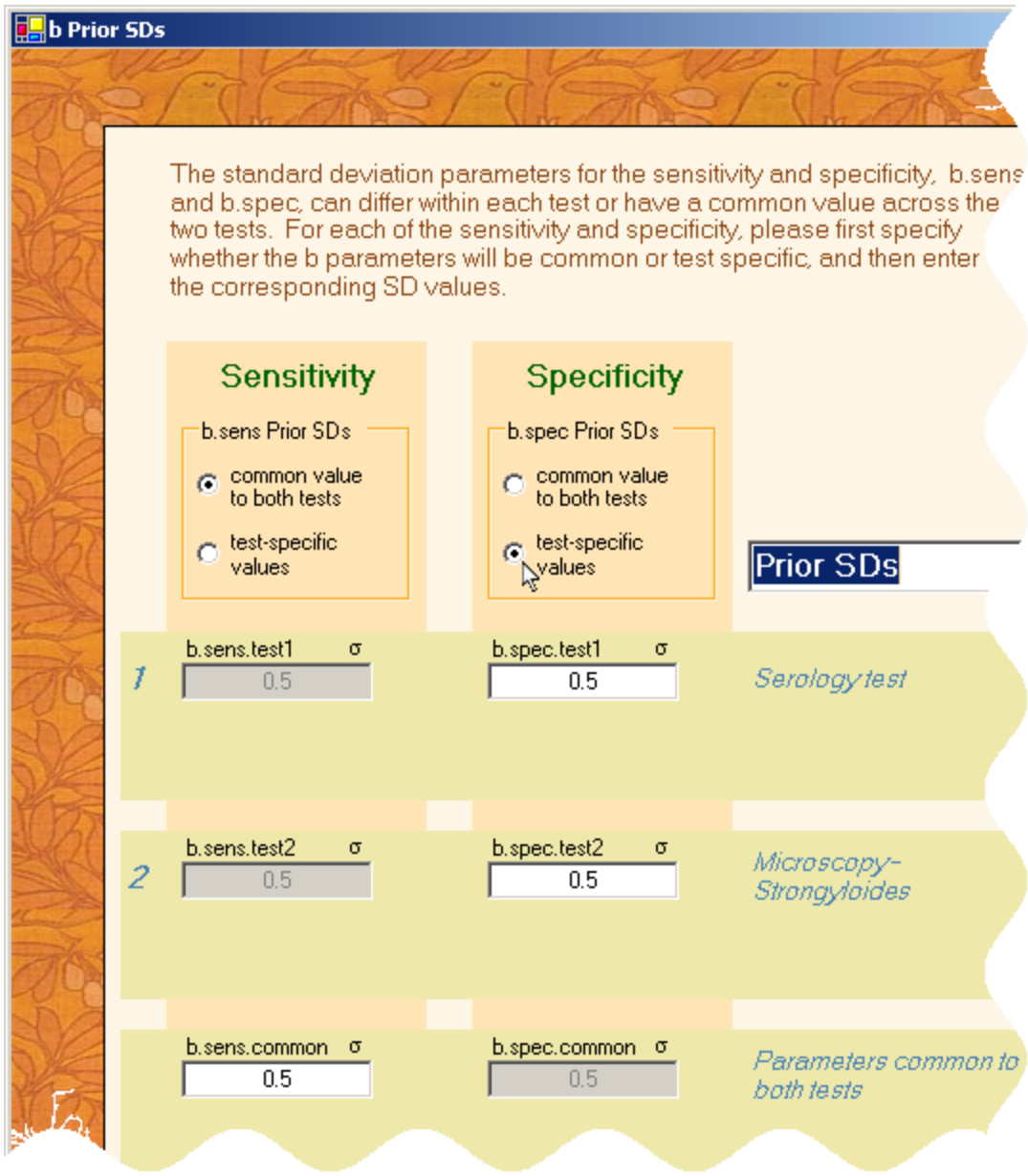
4. Modeling conditional dependence between two diagnostic tests

When two diagnostic tests are used, it is possible to model conditional dependence between them using either a Random or a Fixed effects model.



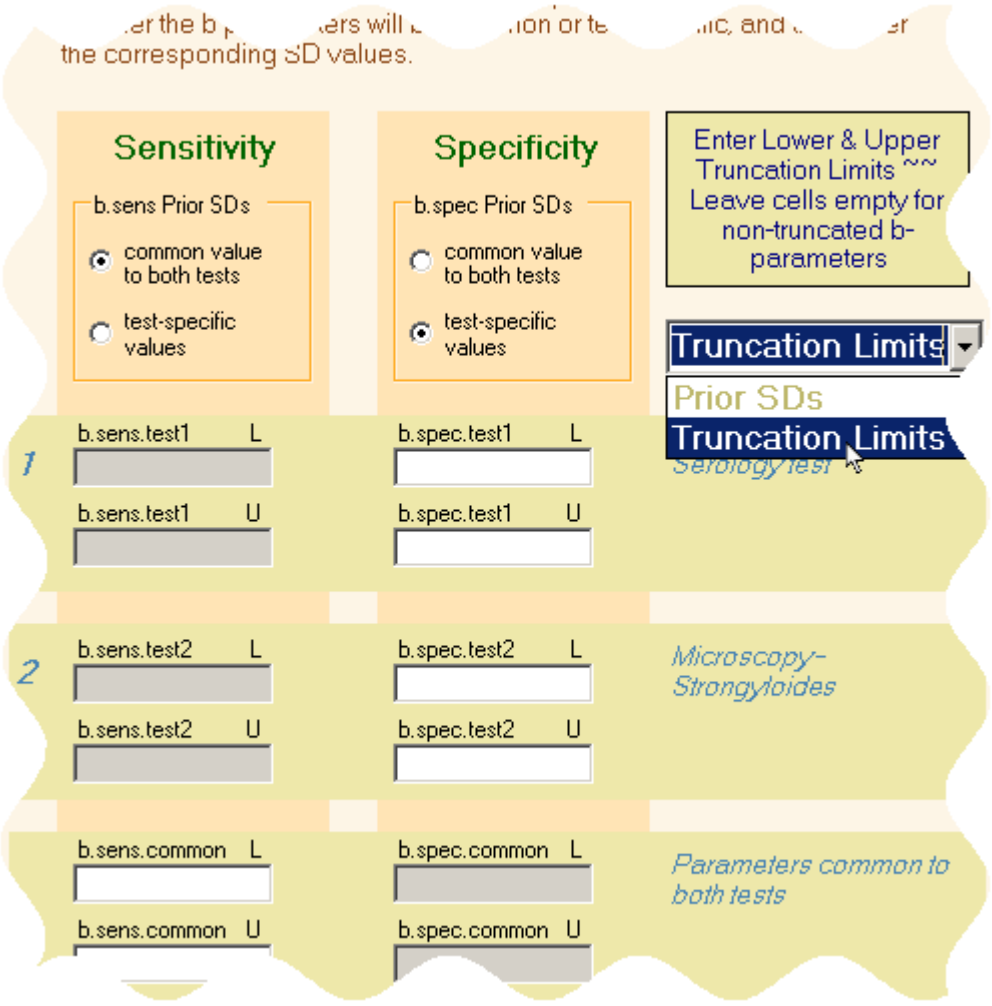
4.1 Random effects model

In a Random effects model, the *b* parameters for both sensitivity and specificity are given normal prior distributions with zero mean; their prior SDs, however, which largely influence the degree of dependence between tests, may depend on prior knowledge and are hence modifiable through next form, below. Also note that the *b* parameters for both sensitivity and specificity can be test-specific, but are assumed to the same for both tests if the "common values" radio button is selected. The form below shows that if the *b* parameter for specificity has test-specific values, selected by clicking the "test-specific" radio-button, at which point two values for *b* must be filled, one for each test.


b Prior SDs
The standard deviation parameters for the sensitivity and specificity, b.sens and b.spec, can differ within each test or have a common value across the two tests. For each of the sensitivity and specificity, please first specify whether the b parameters will be common or test specific, and then enter the corresponding SD values.
Sensitivity
b.sens Prior SDs
common value to both tests
test-specific values
Specificity
b.spec Prior SDs
common value to both tests
test-specific values
Prior SDs
b.sens.test1 σ
0.5
b.spec.test1 σ
0.5
1
Serology test
b.sens.test2 σ
0.5
b.spec.test2 σ
0.5
2
Microscopy-Strongyloides
b.sens.common σ
0.5
b.spec.common σ
0.5
Parameters common to both tests

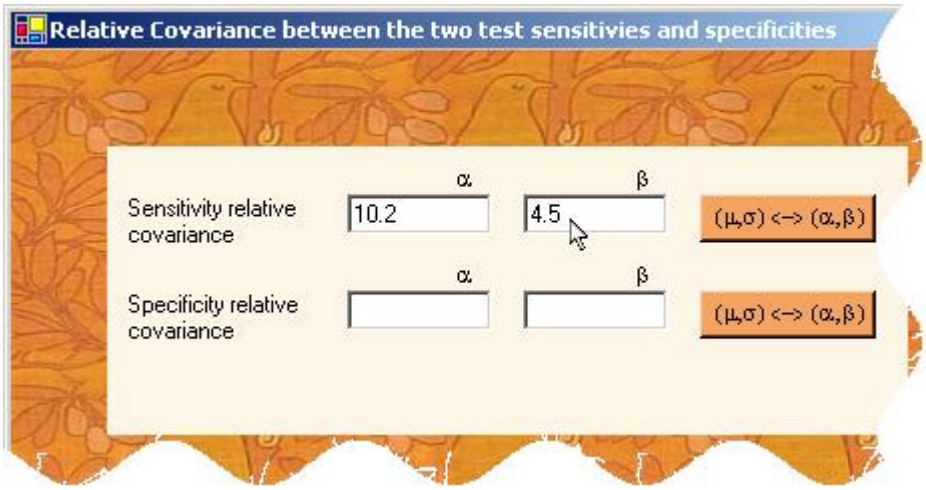
In the same form, it is possible to specify truncation limits relating to the normal prior distribution for each *b* parameter. Select Truncation Limits from the menu item and enter lower and upper limits as needed.

Leave cells empty for non-truncated parameters.



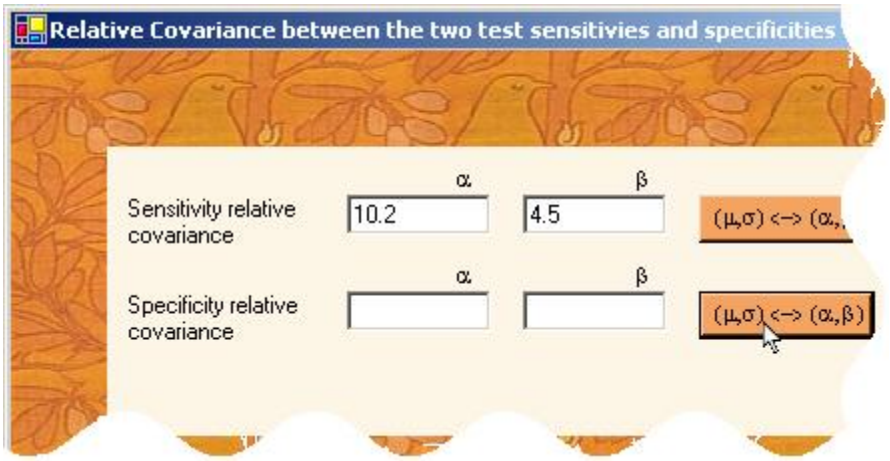
4.2 Fixed effects model

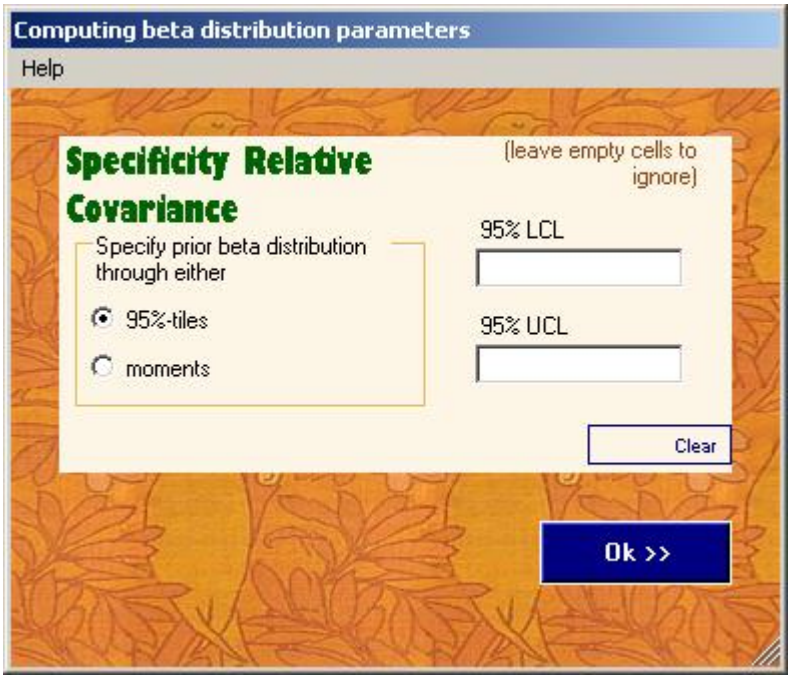
In the two test conditional dependence fixed effects model, the relative covariance between the two test sensitivities and specificities are given beta prior distributions, for which the standard (α, β) parameters can be entered directly.



Alternatively, the beta prior distribution parameters can be calculated to fit prior mean & sd or prior 95% credible interval limits by clicking the appropriate (μ, σ) <-> (α,β) button.

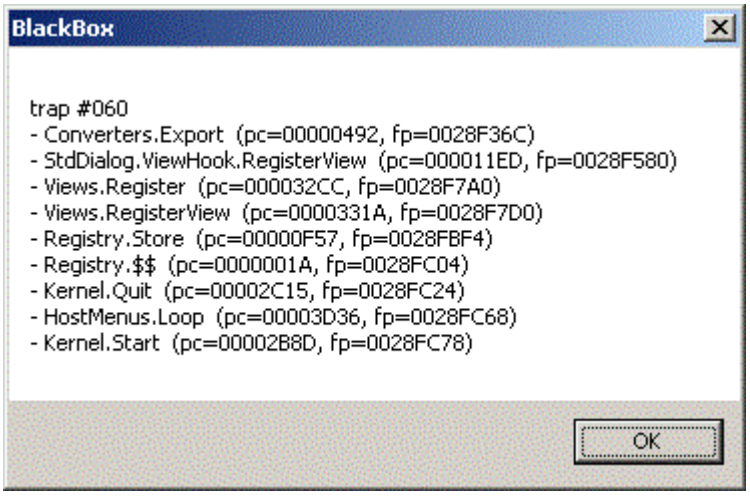
Note however that while **PropMisclassSampleSize** is compliant with both 64-bytes and 32-bytes R versions, this feature will work only if the 32-bytes version of R is installed.





5. Avoiding trap errors from permission settings on Windows 7 and Windows Vista platforms

If you are working on a Windows 7 or Windows Vista platform and have run WinBUGS before, you may have already run into the cryptic **Trap #060** error message illustrated to the right. This is due to restricted write permissions in c:\Program Files, where you may have installed WinBUGS.



WinBUGS **must** be installed in a directory where you have write permissions (e.g. C:\Users\user name \Documents) for **PropMisclassSampleSize** to run smoothly.

6. Stopping criterion

PropMisclassSampleSize iterates over N until

a) the desired parameter accuracy is met for sample size N but not for $N-1$ or

b) in a series of six consecutive sample sizes, the larger three satisfy the sample size criterion while the smaller three do not, and these six consecutive sample sizes do not span more than 2% of their midpoint value.

Stopping criterion (b) proves useful when the final sample size is large (e.g. more than a thousand).

7. Change log

Version 1.3 (September 2005). A minor fix update: previous versions would cause a WinBUGS trap message to appear if the path where files were saved or input from were too long (longer than 119 characters). Version 1.3 avoids this trap by detecting troublesome long path names, and saving some files in temporary locations at WinBUGS exit (for output files) or prior to running WinBUGS (for input files), and removing input file copies and moving output files to their proper locations after WinBUGS closes. This change is invisible and has no consequences for the user.

Version 1.3.1 (September 2005). A minor fix update: the path to an executable file used was incorrect in internal code.

Version 2.0 (March 2007). Improved model-based sample size search algorithm: a more intuitive stopping criterion is now used, resulting in convergence to optimal sample size in fewer steps. A bug was fixed for the MWOC.
Finally, when run interactively (that is, not when running from a DOS-prompt [except with –v and –plot options] or from batch files), a GUI opens at program completion to allow immediate consultation of all output files.

Version 2.1 (April 2008). In earlier versions, default paths to R and WinBUGS may not have been defined properly on Windows x64 platforms. This has now been corrected.

Version 2.2 (June 2008). Earlier versions accepted commas in numeric inputs, which, depending on their placement, could have led to unintended inputs being used. If a comma is found, a pop-up box now asks you to remove it, eliminating all ambiguity.
Version 2.3 (January 2009). The Graphical User Interface now accepts either commas or periods as decimal symbols (in numeric inputs), depending on the value set for **Decimal symbol** in the **Customized Regional Options** form of the **Regional and Language Options** in the **Control Panel**. Both commas and periods cannot be used at the same time, you must use the option chosen for your computer.

Version 3.0 (September 2010). The main output file is now an html file, providing an easier to read summary of the problem and output. Management of scripts (for preparing sample size calculations to be run later) is made much easier than in earlier versions, as well as resumption of previous sample size calculations.

Versions 3.1 and 3.1.1 (November 2010). It is now possible to enter your prior information in terms of (μ, σ) without first entering values for (α, β) in each entry form. Previously, you counter-intuitively had to enter values for (α, β) before being allowed to switch to the alternative (μ, σ) parameterization in the prevalence and diagnostic tests forms.

Version 3.2 (February 2011)
Some users were receiving the message *This application has failed to start because R.dll was not found* with previous version. That problem is now resolved.

Version 3.2.1 (May 2011)
A minor update that may help in correctly identifying the path to R in the initial run (especially for Windows 7 and Windows Vista users).

Versions 3.3 - 3.3.2 (December 2011)
The previous default application folder (c:\Program Files) caused write permission problems for some Windows 7 and Vista users. Default application folder now changed to C:\Users\user name\Documents.

Version 3.4 (January 2012)
Minor update. We have slightly improved the algorithm that searches for the next sample size.

Versions 3.5--3.5.2 (February 2012)
Minor technical problem solved from previous version.

Version 3.5.3 (March 2012)
Minor bug fix update.

Versions 3.6 and 3.6.1 (April 2012)
We suggest a solution to prevent Trap errors for Windows 7 and Windows Vista users.

Versions 4.0 -- 4.0.4 (July 2012)
A better-fitting model is used in the model-based sample size search algorithm, potentially leading to faster convergence.

Version 4.1 (July 2012)
Minor update: cmd.exe now closes automatically when program terminates.

Versions 4.2 and 4.2.1 (August 2012)
Minor esthetical modification was brought to html output file and presentation was changed in Problem Reviewal form.

Version 4.3 (August 2012)
The path to the sub-directory where temporary files are stored was added to the Help menu of the initial form. While you can usually ignore these files, they can sometimes be helpful in troubleshooting when there are problems.

Version 4.4 (November 2012)
PropMisclassSampleSize must limit the length of temporary paths, since it uses WinBUGS scripts, which limits file paths to a maximum of 119 characters. Longer names will cause WinBUGS to freeze. Therefore, if the default temp directory path is too long, **PropMisclassSampleSize** will ask the user to enter a path with a shorter name.

Version 4.5 (January 2013)
Minor bug fix (in best-fitting curve plot).

Version 4.5.1 (November 2013)
Minor update: outcome quantile would not be printed in the html output file when sample size criterion is MBL/MWOC.

Versions 5.0--5.0.2 (December 2013)
We have improved the stopping criterion to avoid very long runs of the program (see section 6). We have also made more efficient use of information from previous runs when repeating an outcome estimate for a given simple size.

Versions 5.1, 5.1.1 and 5.1.2 (February 2014)
We have corrected the html output file (of versions 5.0.x) when reporting results for user-defined sample sizes.

Version 5.2 (February 2014)
Minor bug fix update.

Versions 5.3 and 5.3.1 (February 2014)
Scatter plot of outcome vs sample size is now embedded in the main html output file.

Version 5.4 (April 2014)
Minor improvement to the model-based search algorithm.

Version 5.5 (April 2014)
Minor update.

Versions 5.6 -- 5.6.3 (May 2014)
Added automated sequences for user-defined sample sizes for which outcome is to be estimated.

Versions 5.7 and 5.7.1 (January 2015)
Minor update.

Versions 5.7.2 and 5.7.3 (April 2015)
Minor bug fix update: a potential installation problem was solved.

Versions 5.8 and 5.8.1 (December 2015)
Minor update.

Version 5.9 (January 2016)
Minor update.

Version 5.10 (June 2016)
Technical bug fixed from previous version.

Version 6.0 (February 2019)
It is now possible to run sample size calculations when two conditionally **dependent** diagnostic tests are used. This option requires that the free software package NIMBLE is installed as part of R.

Also solved a technical problem: depending on user's R settings, some temporary R output files were not saved along the expected lines, causing **PropMisclassSampleSize** to crash in previous version.

Questions? Comments? Please send email to: Lawrence.Joseph@McGill.ca

---

Other Bayesian software packages are available at
http://www.medicine.mcgill.ca/epidemiology/Joseph