Appendix

CrossMark 2.0.0 User Manual

The program *CrossMark* is designed to estimate transition probabilities using data from repeated cross sections. Given a dichotomous Y variable, *CrossMark* estimates the effects of predictor variables X on the entry and exit probabilities using a Markov model.

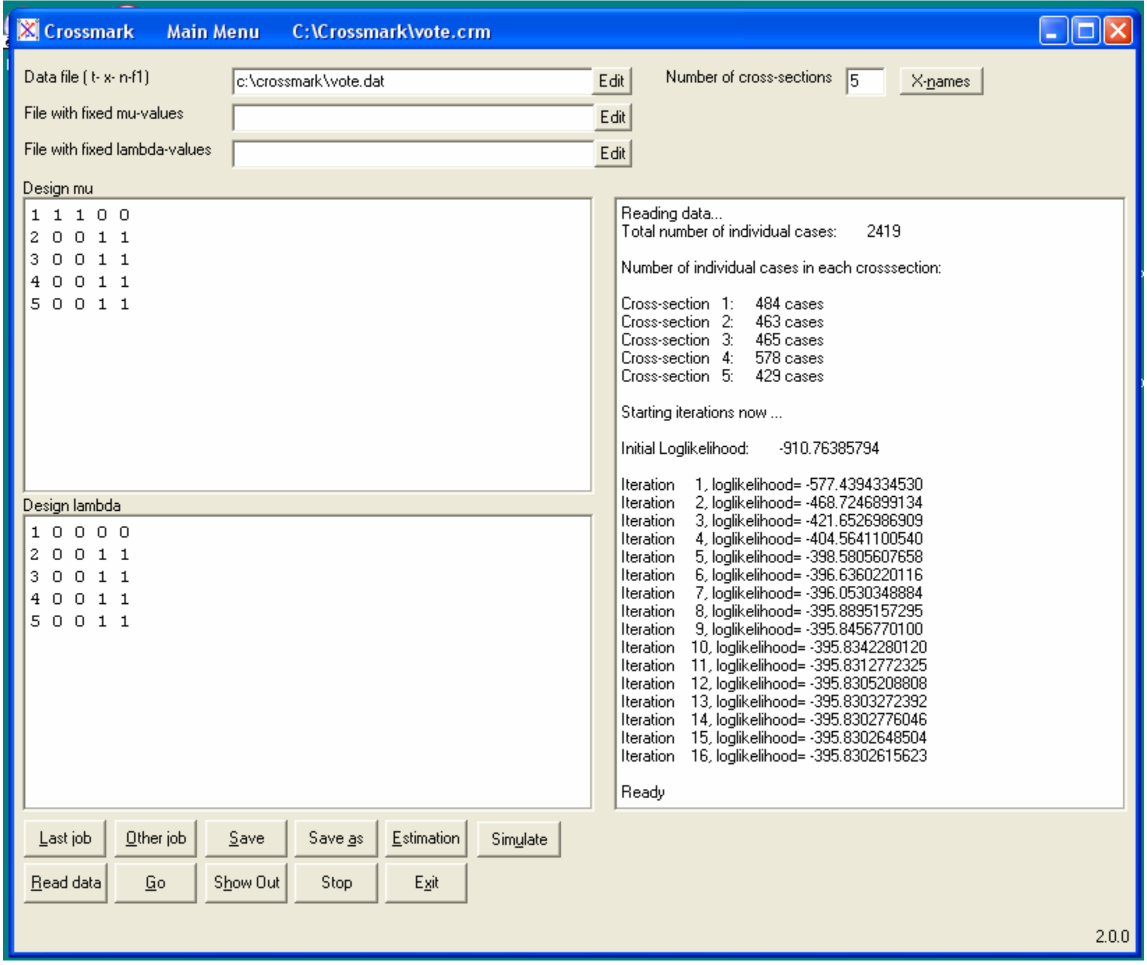
CrossMark is available for Windows 95, 98, 2000 an XP. The program needs not be installed: simply place file *CrossMark.exe* in a directory of your choice and double-click on this file (in Windows Explorer) to start CrossMark. The **Main Menu** then appears on the screen. This menu looks like the one in Figure 1, except that all fields are still empty.

1 Standard analysis

We shall describe how a standard analysis with *CrossMark* proceeds using a fictitious example on vote intention. To highlight all the options of the program, we use **bold face** characters for buttons that must be clicked and fields or menu's that have to be filled in.

Suppose the data to be analyzed are from 5 cross-sections, gathered in consecutive years, i.e., from 1996 to 2000. The dependent variable is the 'intention to vote for political party A' (code 1 = 'vote for', 0 = 'not vote for') and the independent variable is the respondent’s age (ranging from 18 to 70 years). The file containing the data is named 'c:\crossmark\vote.dat'. This filename has to be entered on the Main Menu in the field **Data file (t-x-n-f1)**. The data file can be inspected by clicking the **Edit** button, which opens the data file in WordPad. The total number of cross-sections (5) has to be

Figure 1   Main Menu



Data file (t- x- n-f1) c:\crossmark\vote.dat

File with fixed mu-values

File with fixed lambda-values

Number of cross-sections 5

Design mu

```
1 1 1 0 0
2 0 0 1 1
3 0 0 1 1
4 0 0 1 1
5 0 0 1 1
```

Design lambda

```
1 0 0 0 0
2 0 0 1 1
3 0 0 1 1
4 0 0 1 1
5 0 0 1 1
```

```
Reading data...
Total number of individual cases:      2419

Number of individual cases in each crosssection:

Cross-section  1:    484 cases
Cross-section  2:    463 cases
Cross-section  3:    465 cases
Cross-section  4:    578 cases
Cross-section  5:    429 cases

Starting iterations now ...

Initial Loglikelihood:      -910.76385794

Iteration    1, loglikelihood= -577.4394334530
Iteration    2, loglikelihood= -468.7246899134
Iteration    3, loglikelihood= -421.6526986909
Iteration    4, loglikelihood= -404.5641100540
Iteration    5, loglikelihood= -398.5805607658
Iteration    6, loglikelihood= -396.6360220116
Iteration    7, loglikelihood= -396.0530348884
Iteration    8, loglikelihood= -395.8895157295
Iteration    9, loglikelihood= -395.8456770100
Iteration   10, loglikelihood= -395.8342280120
Iteration   11, loglikelihood= -395.8312772325
Iteration   12, loglikelihood= -395.8305208808
Iteration   13, loglikelihood= -395.8303272392
Iteration   14, loglikelihood= -395.8302776046
Iteration   15, loglikelihood= -395.8302648504
Iteration   16, loglikelihood= -395.8302615623

Ready
```

entered in the field **Number of cross-sections**. The abbreviation 't-x-n-f1' behind 'Data file' stands for t=time index, x=X or predictor variables, n=number of cases and f1=number of cases in category $Y=1$, respectively, and reflects the order in which the data must appear in the data file. The first three lines of the example data of each cross-section are shown below:

```
1    1 18       1   18 19 20 21 22       9     2
1    1 19       1   19 20 21 22 23       5     0
1    1 20       1   20 21 22 23 24       3     0
2    1 18       1   18 19 20 21 22       4     1
2    1 19       1   19 20 21 22 23      13     5
2    1 20       1   20 21 22 23 24       8     0
3    1 18       1   18 19 20 21 22       4     2
3    1 19       1   19 20 21 22 23       5     1
3    1 20       1   20 21 22 23 24       8     3
4    1 18       1   18 19 20 21 22       4     2
```

```
4   1 19       1  19 20 21 22 23       12     11
4   1 20       1  20 21 22 23 24        8      5
5   1 18       1  18 19 20 21 22        7      5
5   1 19       1  19 20 21 22 23        4      1
5   1 20       1  20 21 22 23 24        2      1
```

The first data column is the time index *t.* As there are five cross-sections the time index has to have the values 1, 2, 3, 4, and 5 denoting the years 1996, 1997, 1998, 1999, 2000, respectively. *CrossMark* expects the data to be ordered in time, with the data of the first cross section located at the top of the file, those of the second cross-section following underneath and so on until the data of the last cross-section which must be located at the end of the file.

The next 8 data columns of the data file in this example, i..e., column 2 through 9, contain the values of the predictor variables X. There are 4 predictor variables here:

1. An intercept, having the value 1 for each case. It is located in column 2 of the data file. In the sequel we will refer to it as ‘intercept 1’.
2. The respondents age in 1996, located in column 3. For the respondents of the cross sections 1997 and following, the age in 1996 has been computed by ‘backcasting’ their age to the year 1996. We shall explain below why we use ‘age in 1996’ as a separate predictor, which we call ‘age 1996’.
3. A second intercept in column 4, which is called ‘intercept 2’.
4. The respondents age in each of the five years, located in columns 5 through 9. These five age values together constitute a single predictor variable, the values of which change over time. We call this predictor 'age'.

We will return to the characteristics of the 4 predictors and the way they affect the transition probabilities in more detail below. The last two columns, 10 and 11, of the data file concern the total number of cases and the number of cases in Y category 1, respectively. For example, the first record of the cross section at t=5, i.e., the record

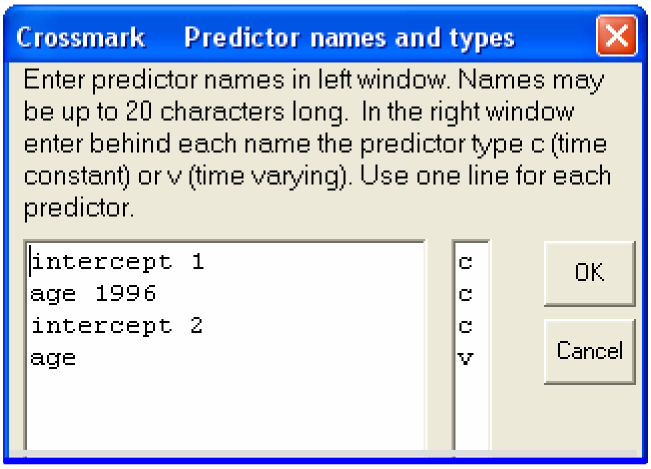
```
5   1 18       1  18 19 20 21 22        7      5
```

specifies that there are 7 cases in this cross section who were 18 years old in 1996, 19 years old in 1997 etc., and that 5 of them are in Y category 1 (at $t=5$) while the other 2 are in category 0. If each row in the datafile would contain data for just a single case, then the last but one column (here column 10) would be 1 for all cases while the last column would be either 1 or 0. There is no need to aggregate over t, X and Y. However, aggregating the data, as is done in this example, can speed up the estimation process considerably.

We now return to the predictor variables X. The predictors numbered 1, 2 and 3 above are constant over time, while predictor 4 takes a different value in each of the five years. Time constant predictors occupy a single column in the data file, while time varying predictors occupy as many columns as there are cross-sections, i.e. five in the example. The names and types (constant or varying) of the predictors have to be specified in the submenu **Predictor names and types,** which shows up after clicking the **X-names** button of the Main Menu and is shown in Figure 2. The left field of the submenu **Predictor names and types** contains the predictor's name and the right field the predictor's type. For a time constant predictor enter the character **c**, and for a time varying predictor enter **v**. Having done so, click OK to return to the Main Menu.

To understand why we use two intercepts and two age predictors (instead of just one intercept and one age predictor, which would be possible too)

Figure 2 Predictor names and types



Enter predictor names in left window. Names may be up to 20 characters long. In the right window enter behind each name the predictor type c (time constant) or v (time varying). Use one line for each predictor.

```
intercept 1     c
age 1996        c
intercept 2     c
age             v
```

we take a closer look at the model equations for p_1, p_2, p_3, p_4 and p_5 or, in words, the probabilities to vote for political party A in each of the five years. In general, the basic equations *CrossMark* uses are, with five cross-sections:

$$\begin{aligned}
p_1 &= \mu_1 \\
p_2 &= p_1\ (1-\lambda_2) + (1-p_1)\ \mu_2 \\
p_3 &= p_2\ (1-\lambda_3) + (1-p_2)\ \mu_3 \\
p_4 &= p_3\ (1-\lambda_4) + (1-p_3)\ \mu_4 \\
p_5 &= p_4\ (1-\lambda_5) + (1-p_4)\ \mu_5
\end{aligned}$$

In the example, the transition probabilities μ and λ depend on the respondents ages as follows:

$$\begin{aligned}
&\text{logit}(\mu_1) = \beta_1 + \beta_2\, Age_{1996} \\
&\text{logit}(\mu_2) = \beta_3 + \beta_4\, Age_{1997} \qquad && \text{logit}(1-\lambda_2) = \beta_1^* + \beta_2^*\, Age_{1997} \\
&\text{logit}(\mu_3) = \beta_3 + \beta_4\, Age_{1998} && \text{logit}(1-\lambda_3) = \beta_1^* + \beta_2^*\, Age_{1998} \\
&\text{logit}(\mu_4) = \beta_3 + \beta_4\, Age_{1999} && \text{logit}(1-\lambda_4) = \beta_1^* + \beta_2^*\, Age_{1999} \\
&\text{logit}(\mu_5) = \beta_3 + \beta_4\, Age_{2000} && \text{logit}(1-\lambda_5) = \beta_1^* + \beta_2^*\, Age_{2000}
\end{aligned}$$

Age$_{1996}$ refers to the respondent's age in 1996, Age$_{1997}$ to the age in 1997, etcetera. The symbol λ indicates the exit probability: λ_3 is the probability not to vote for party A in 1998 given a 'vote for A' in 1997. For the complement of λ, or the probability to stay in state $Y=1$, the term '1-exit' probability is used in the sequel. The symbol μ indicates the entry probability: μ_3 is the probability to vote for A in 1998 given a 'not vote for A' in 1997.

Speaking of $\mu_1 = p_1$ as an entry probability can be problematic. Generally spoken, p_1 is the probability to be in state $Y=1$ at $t=1$ and this need not to be the same as the probability to be in state $Y=1$ given that the previous state was $Y=0$. Only if one knows that each respondent's previous state was $Y=0$, one may truly consider p_1 an entry probability. This would e.g. be the case if political party A did not exist before 1996. In many applications, of course, the $Y=1$ state does exist prior to $t=1$ and respondents could have been in that state. In such situations, one may prefer to model p_1 as a state probability, rather than an entry probability. This is accomplished by estimating different sets of parameters for μ_1 and for μ_2 and following, as is done in the model above, where the parameters β_1 and β_2 only apply to μ_1.

In *CrossMark* the model equations can be specified in the **Design mu** and **Design lambda** fields of the Main Menu. In **Design mu** we indicate which predictor variable acts upon which entry probability μ. For the example this is done as follows:

```
1  1 1  0 0
2  0 0  1 1
3  0 0  1 1
4  0 0  1 1
5  0 0  1 1
```

The first column is the time index t and the other four columns correspond to the four predictor variables in the model. The second column corresponds to 'intercept 1', and the value 1 for $t = 1$ indicates that ‘intercept 1’ has an effect on μ_1; the 0 scores in the second column for $t = 2, 3, 4$ and 5 indicate that ‘intercept 1’ does not have an effect on μ_2, μ_3, μ_4 and μ_5. The rightmost column is related to the time varying predictor ‘age’; the 0 value for $t = 1$ indicates that 'age' does not occur in the equation for μ_1 while the 1 values for $t = 2, 3, 4$ and 5 indicate that 'age' does occur in the equations for μ_2, μ_3, μ_4 and μ_5.

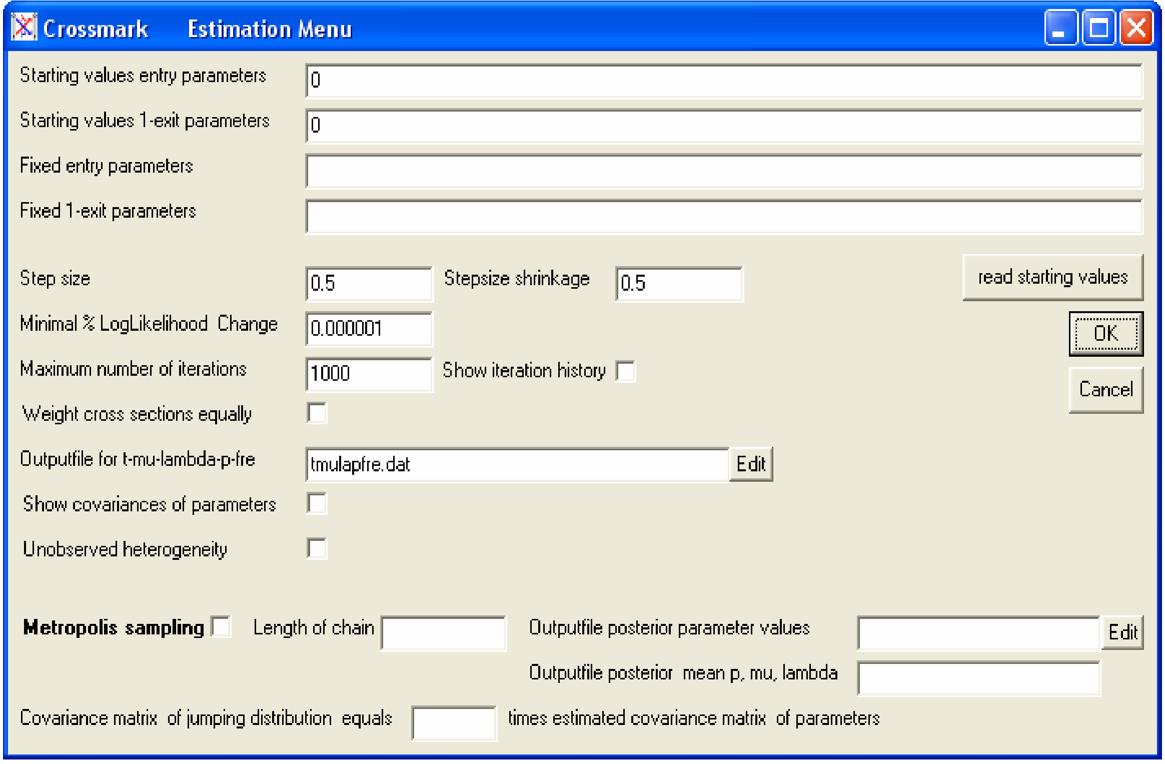
In general, the **Design mu** matrix must have as many rows as there are cross-sections. Each row starts with the time index t and is followed by a 1 or 0 value for each predictor variable indicating whether (1) or not (0) the predictor acts upon entry probability μ_t. In the same way a **Design lambda** matrix has to be specified indicating which predictor acts upon which exit probability λ. For the present example the lambda matrix is specified as:

```
1  0 0  0 0
2  0 0  1 1
3  0 0  1 1
4  0 0  1 1
5  0 0  1 1
```

Note that the first row of the **Design lambda** matrix contains the value 1 for the time index $t = 1$ and else only 0 values to indicate that none of the four predictor variables has an effect on λ_1. This is just to specify that λ_1 does not play a part in the model equations.

We proceed by clicking the **Estimation** button of the Main menu to invoke the Estimation Menu as shown in Figure 3. The upper two fields in this Estimation Menu specify the **starting values** for the iterative Fisher scoring scheme. The default values are 0 for all β and β^* parameters of the entry and 1-exit probabilities respectively. Good starting values, i.e., values close to the final ML estimates, speed up the estimation process. Starting values far removed from the final estimates slow down this process or may cause the estimates to be caught in a local maximum or not to reach convergence at all. When convergence has been reached, it is advisable to choose other starting values and let *CrossMark* run again to check whether the same parameter estimates are found. If this turns out to be the case, one can be more confident that the estimates are indeed the true global ML estimates instead of estimates associated with a local maximum.

Figure 3 Estimation Menu



Crossmark Estimation Menu

Starting values entry parameters: 0

Starting values 1-exit parameters: 0

Fixed entry parameters:

Fixed 1-exit parameters:

Step size: 0.5

Stepsize shrinkage: 0.5

Minimal % LogLikelihood Change: 0.000001

Maximum number of iterations: 1000

Show iteration history □

Weight cross sections equally □

Outputfile for t-mu-lambda-p-fre: tmulapfre.dat

Show covariances of parameters □

Unobserved heterogeneity □

Metropolis sampling □ Length of chain:

Outputfile posterior parameter values:

Outputfile posterior mean p, mu, lambda:

Covariance matrix of jumping distribution equals ____ times estimated covariance matrix of parameters

When analyzing complex models, in the sense of having many predictors, starting values become more of an issue. The final estimates of a previous, relatively simple model can be used as starting values for a new model having additional predictors. To this end the button **read starting values** can be helpful. After clicking, the final estimates of the previous model are filled in as starting values in both fields. The starting values for the additional predictors in the second model are defined to be zero and automatically added to the list. If a predictor that was present in the previous model does not appear in the second, the user has to remove the relevant starting values from both lines.

If, for some reason, one would like to fix the parameters of one or more predictors to certain predefined values instead of having them estimated by CrossMark, one can be proceed as follows. In the field named **Fixed entry parameters** enter a value 0 or 1 for each predictor parameter that has to be estimated (enter 0) or not (enter 1). Be sure to enter a value 0 or 1 for *all* predictors and to use the same *order* for the predictors as was used in the menu Predictor names and types. For predictors that have a value 0 specified, CrossMark will estimate a parameter starting from the starting value. For predictors that have a value 1 specified, CrossMark will not estimate a parameter but substitutes the given starting value as the parameter value to be used for this predictor's effect on the entry probability. In CrossMark's output, fixed parameters are denoted by the character 'f' and have a Wald Significance and Std. error of 1.0. In the same manner, one can fix parameters for the 1-exit probability.

The **Step size** field in the Estimation Window refers to the step size ε of the Fisher scoring algorithm employed for iteratively updating the parameter estimates. The algorithm is given by $\hat{\theta}_{k+1} = \hat{\theta}_k + \varepsilon \;\; \hat{I}_k^{-1} \;\; (\delta LL / \delta \theta)_k$, where $\hat{\theta}_k$ and $\hat{\theta}_{k+1}$ are the parameter estimates at the iterations k and $k+1$, $\hat{I}_k^{-1}$ is the inverse of the Fisher information matrix evaluated at $\theta = \hat{\theta}_k$, and $(\delta LL / \delta \theta)_k$ are the derivatives of the log likelihood with respect to the parameters, evaluated at $\theta = \hat{\theta}_k$. By default, the value of the step size ε is 0.5. If the log likelihood function has a single mode, the optimal value for the step size would be 1. It is not unusual, however, for the log likelihood function to have multiple modes in which case a step size of 1 could easily cause the algorithm to jump over the parameter region with the highest mode. For this reason, a default step size of 0.5 is chosen. A much smaller step size value may slow down the algorithm too much. There is no rule of thumb given here as to the choice of the most efficient step size value.

The **Step size shrinkage** (s) also deals with the problem of the step size being too large. If the log likelihood based on $\hat{\theta}_{k+1}$ is lower than the one based on $\hat{\theta}_k$, the current step size has apparently been too large. In that case *CrossMark* produces the message "Not converging, back to parameter estimates of previous iteration" and takes as the new step size the product $s \cdot \varepsilon$. If this smaller step size also leads to $\hat{\theta}_{k+1}$ estimates with a lower log likelihood than the one based on $\hat{\theta}_k$, the step size $s \cdot s \cdot \varepsilon$ is tried. In short, the step size is multiplied by s as many times as needed to produce an increase in log likelihood.

The iterative estimation process ends if either the percentage change in log likelihood is less than the **Minimal % LogLikelihood Change** specified, which by default is 0.000001%, or the **Maximum number of iterations** has been reached, which by default is 1000. Also by default, CrossMark only shows the parameter estimates of the final iteration and not those of previous iterations. To force CrossMark showing the estimates of each iteration, check the **Show iteration history** option.

By default CrossMark applies caseweights resulting in the same weighted number of cases for each cross section. The sum of all caseweights is equal to the total number of cases in all cross sections. To prevent this weighting procedure uncheck the option **Weight cross sections equally**.

CrossMark produces an output file, the name of which can be specified in the field **Outputfile for t-mu-lambda-p-fre**. By default it is labeled 'tmulapfre' and placed in the directory where the 'crossmark.exe' resides. The output file contains one line for each case in the data file. For case i, this line has the following information from left to right:

- the time index of the cross-section case i belongs to,
- the predicted values of μ_{i1} to μ_{iT},
- the predicted values of λ_{i1} to λ_{iT},

- the predicted values of p_{i1} to p_{iT},
- the frequency of case i, equal to the frequency specified in the rightmost column of the data file.

Predicted μ, λ and p values that do not apply to a particular case (e.g., μ_3 for a case of cross-section 2, or λ_1 for all cases) are assigned the 'missing value' 9.

By default, in CrossMark's output no (co)variances of parameter estimates are shown. They will be, if the option **Show covariances of parameters** is checked before running the model.

The options for **Unobserved heterogeneity** and **Metropolis sampling** will be discussed below in separate sections.

After clicking the **OK** button of the Estimation Menu the Main Menu reappears. To save all the specifications entered, click the **Save** button and specify a file name, e.g. 'vote.crm' which then appears in the top line of the Main Menu. Using the **Save as** button enables saving the job under a different name. The most recently saved job can be opened by clicking on the button **Last job** while older jobs may be opened with **Other job**.

To start the analysis the data have to be read first. This is done by clicking on **Read data**. When finished reading, *CrossMark* presents the total number of cases as well as the number of cases for each cross-section in the rightmost window of the Main Menu. After reading the data, the estimation can be carried out by clicking on **Go.** The initial log likelihood, based on the starting values of the parameters, appears on the screen after a few moments, as does the log likelihood of each subsequent iteration. When the last iteration is finished, a 'Ready' message is delivered. The estimation may take some time, especially when many cases and/or predictor variables are involved. In the mean time the user may want to look at intermediate results by clicking the **Show Out** button or pressing **Ctrl+Tab** on the keyboard. The **Output window** then appears, with the parameter estimates of each iteration scrolling over the screen, accompanied by the log likelihood and, possibly, messages concerning corrective actions undertaken by the estimation algorithm. Pressing **Ctrl+Tab** again (or clicking the cross X in the upper right corner of the screen) closes the Output window.

Back in the Main Menu the estimation process - if still running - can be stopped by using the **Stop** button. This may be useful if e.g. the log likelihood does not change substantially anymore. Another reason to stop the iterations is that the algorithm does not converge, which may happen if the model contains too many (i.e., not uniquely identified) parameters.

To leave *CrossMark* click **Exit** or the cross X in the upper right corner of the screen.

2 Nonbackcastable variables

It may be that the respondent's value on a predictor variable at time t is known, but the values at $t-1$, $t-2$ and so on are not. Take e.g. the variable ‘monthly income’. Given the income of a respondent of cross-section t, usually little, if anything, is know about his or her income at earlier points in time. To put it another way: the variable income cannot be 'backcasted'. Such a nonbackcastable variable can be used as a predictor for the entry and exit probability only at the time the respondent was observed but not at preceding points in time. We will show using a simple example how such variables can be handled in *CrossMark*.

Suppose that we have three cross-sections and the nonbackcastable predictor we would like to use is named Inc, representing the monthly personal income of a respondent at the time of observation. Also, we have the backcastable predictor age specified as $Age(t)$, where the t between brackets denotes that there are three age vectors, one for each of the three points in time. For simplicity, we omit the intercept in the equations for μ below. For any respondent of the second and subsequent cross-sections, the following two equations apply to $\text{logit}(\mu_t)$, depending on whether t relates to the time the respondent is actually observed or to a preceding point in time:

observed: logit(μ_t) = $\beta_1 \cdot Age(t)$ + $\beta_3 \cdot$ Inc (1)

preceding: logit(μ_t) = $\beta_2 \cdot Age(t)$ (2)

In equation (1) we can use Inc as a predictor, whereas in equation (2) this is not possible. Of course the Age effects β_1 and β_2 need not necessarily be the same. In order to estimate β_1, β_2 and β_3 with *CrossMark* a single equation for $\text{logit}(\mu_t)$ must be specified that applies to all points in time. To achieve this we construct three ancillary time varying predictors, which we shall call $Age\_obs(t)$, $Age\_pre(t)$ and $Inc\_obs(t)$ to be discussed below. The construction of these predictors must precede the analysis with *CrossMark* and the user must add the predictors to the data file and treat them like any normal predictor variable: their names and types (v) have to be entered (using the **X-names** button in the Main Menu) and also, three columns, one for each predictor, have to be added to the **Design mu** and **Design lambda** matrices.

The predictor $Age\_obs(t)$ has to be constructed such that $Age\_obs(t) = Age(t)$ for cases observed at time point t and $Age\_obs(t) = 0$ for all other cases. For predictor $Age\_pre(t)$ it must hold that $Age\_pre(t) = Age(t)$ for cases observed *after* time point t and $Age\_pre(t) = 0$ for all other cases.

For 6 randomly chosen cases, two of each cross-section, the values of $Age(t)$, $Age\_obs(t)$ and $Age\_pre(t)$ might be those shown in the upper part of Table 1. Note that, put next to one another, the three $Age\_\ obs(t)$ vectors form a block-diagonal matrix and the $Age\_pre(t)$ vectors a 'sub-block diagonal' one. For Inc and $Inc\_obs(t)$ the values of the 6 cases might be the ones in the lower part of Table 1, with now the $Inc\_obs(t)$ vectors forming a block-diagonal matrix. Instead of the two separate equations (1) and (2), we can write a single equation, holding for time observed as well as preceding points in time:

$$\text{logit}(\mu_t) = \beta_4 \cdot Age\_\ obs(t) + \beta_5 \cdot Age\_pre(t) + \beta_6 \cdot Inc\_obs(t) \qquad (3)$$

Why (1) and (2) are equivalent to (3) becomes clear when equation (3) is worked out for the observed and preceding time points separately:

Table 1 Ancillary predictors for Age

	$Age(t)$			$Age\_obs(t)$			$Age\_pre(t)$		
	(1)	(2)	(3)	(1)	(2)	(3)	(1)	(2)	(3)
$t=1$	19	0	0	19	0	0	0	0	0
	45	0	0	45	0	0	0	0	0
$t=2$	37	38	0	0	38	0	37	0	0
	21	22	0	0	22	0	21	0	0
$t=3$	42	43	77	0	0	44	42	43	0
	66	67	68	0	0	68	66	67	0

	Inc	$Inc\_obs(t)$		
		(1)	(2)	(3)
$t=1$	1500	1500	0	0
	7300	7300	0	0
$t=2$	3500	0	3500	0
	9400	0	9400	0
$t=3$	1200	0	0	1200
	2200	0	0	2200

$$\begin{aligned}\text{observed:}\quad \text{logit}(\mu_t) &= \beta_4 \cdot Age\_obs(t) + \beta_5 \cdot Age\_pre(t) + \beta_6 \cdot Inc\_obs(t)\\ &= \beta_4 \cdot Age(t) \qquad + \beta_5 \cdot 0 \qquad\quad + \beta_6 \cdot Inc\\ &= \beta_4 \cdot Age(t) \qquad\qquad\qquad\qquad + \beta_6 \cdot Inc \end{aligned} \tag{3a}$$

$$\begin{aligned}\text{preceding:}\quad \text{logit}(\mu_t) &= \beta_4 \cdot Age\_obs(t) + \beta_5 \cdot Age\_pre(t) + \beta_6 \cdot Inc\_obs(t)\\ &= \beta_4 \cdot 0 \qquad\qquad + \beta_5 \cdot Age(t) \qquad + \beta_6 \cdot 0\\ &= \qquad\qquad\qquad \beta_5 \cdot Age(t) \end{aligned} \tag{3b}$$

Thus, equations (3a) and (3b) appear to be equivalent to (1) and (2), respectively. Since *CrossMark* uses a single equation for μ we employ the generic equation (3). Parameter β_4 can be interpreted as β_1, i.e., the effect of age controlled for income, at observation time; β_5 is interpreted like β_2 as the effect of age at preceding points in time not controlled for income; β_6 has the same interpretation as β_3, i.e., the effect of income controlled for age at the time of observation.

Instead of (3) way may also use another generic equation in *CrossMark*:

$$\mathrm{logit}(\mu_t) = \beta_7 \cdot Age(t) \quad + \beta_8 \cdot Age\_obs(t) \; + \beta_9 \cdot Inc\_obs(t) \tag{4}$$

Working out (4) for observation time and preceding timepoints results in:

$$\begin{aligned} \text{observed:}\quad \mathrm{logit}(\mu_t) &= \beta_7 \cdot Age(t) \quad + \beta_8 \cdot Age\_obs(t) \; + \beta_9 \cdot Inc\_obs(t) \\ &= \beta_7 \cdot Age(t) \quad + \beta_8 \cdot Age(t) \quad\;\; + \beta_9 \cdot Inc \\ &= (\beta_7 + \beta_8) \cdot Age(t) \qquad\qquad + \beta_9 \cdot Inc \end{aligned} \tag{4a}$$

$$\begin{aligned} \text{preceding:}\quad \mathrm{logit}(\mu_t) &= \beta_7 \cdot Age(t) \quad + \beta_8 \cdot Age\_obs(t) \; + \beta_9 \cdot Inc\_obs(t) \\ \mathrm{logit}(\mu_t) &= \beta_7 \cdot Age(t) \quad + \beta_8 \cdot 0 \qquad\qquad + \beta_9 \cdot 0 \\ \mathrm{logit}(\mu_t) &= \beta_7 \cdot Age(t) \end{aligned} \tag{4b}$$

As can be seen (4a) is equivalent to (3a) and (1), while (4b) is equivalent to (3b) and (2). Therefore, both equation (3) and (4) can be used to model $\mathrm{logit}(\mu_t)$. They differ only in parameterization. The sum $\beta_7 + \beta_8$ has the same interpretation as β_4 (or β_1); β_7 is interpreted in the same way as β_5 (or β_2). Finally, the interpretation of β_9 is similar to the one of β_5 (or β_3). A minor advantage of using (4) instead of (3), is that (4) needs on construction of the $Age\_pre(t)$ vectors.

2.1 Testing the null-hypothesis $\mathrm{H}_0 : \beta_1 = \beta_2$

Looking at the equations (1) and (2) the question arises as to the equality of the two Age effects β_1 and β_2. When applying equation (4) the above null hypothesis translates into $\mathrm{H}_0 : \beta_7 + \beta_8 = \beta_7$ or, more simply, to $\mathrm{H}_0 : \beta_8 = 0$. This test is automatically performed by *CrossMark* and the significance level of the related Wald statistic is reported in the Output window. When, on the other hand, equation (3) is applied, the above hypothesis translates into $\mathrm{H}_0 : \beta_4 - \beta_5 = 0$. Given the hypothesis is true, the sample outcome of the statistic $(\hat{\beta}_4 - \hat{\beta}_5)^2 / \mathrm{var}(\hat{\beta}_4 - \hat{\beta}_5)$, with $\mathrm{var}(\hat{\beta}_4 - \hat{\beta}_5)$ being the estimated sample variance of $\hat{\beta}_4 - \hat{\beta}_5$, follows a χ^2 distribution with 1 degree of freedom. The value of $\hat{\beta}_4 - \hat{\beta}_5$ can of course be derived from the ML estimates produced by *CrossMark* in the final iteration. To derive $\mathrm{var}(\hat{\beta}_4 - \hat{\beta}_5)$ the formula $\mathrm{var}(\hat{\beta}_4 - \hat{\beta}_5) = \mathrm{var}(\hat{\beta}_4) + \mathrm{var}(\hat{\beta}_5) - 2\,\mathrm{cov}(\hat{\beta}_4, \hat{\beta}_5)$ can be applied with $\mathrm{var}(\hat{\beta}_4)$, $\mathrm{var}(\hat{\beta}_5)$

Table 2 Ancillary intercept predictors

$Intercept$		$Intercept\_obs(t)$			$Intercept\_pre(t)$		
		(1)	(2)	(3)	(1)	(2)	(3)
$t=1$	1	1	0	0	0	0	0
	1	1	0	0	0	0	0
$t=2$	1	0	1	0	1	0	0
	1	0	1	0	1	0	0
$t=3$	1	0	0	1	1	1	0
	1	0	0	1	1	1	0

and $\operatorname{cov}(\hat{\beta}_4, \hat{\beta}_5)$ representing the estimated variances of $\hat{\beta}_4$ and $\hat{\beta}_5$ and their estimated covariance respectively. These variances and covariance are given by *CrossMark* by checking the option **Show covariances of parameters** in the **Estimation Menu**.

If the test outcome leads to not rejecting the null hypothesis, the ancillary variables for the predictor in question are no longer needed and the original predictor, $Age(t)$ in the example, can be used, possibly along with ancillary variables of other predictors for which the hypothesis does not hold.

The equations above did not include an intercept, for simplicity. Of course, in most applications an intercept will be present and we will have to decide which type of intercept vector(s) to employ. If we have no nonbackcastable predictors, the intercept is simply a single vector containing the value 1 for all cases of all cross-sections. If, however, nonbackcastable predictors are utilized, we may want to estimate one intercept for time observed and another one for preceding time, just as was done for $Age(t)$ in equations (1) and (2). In that case we would have to construct two ancillary (time varying) intercept predictors, according to the scheme in Table 2.

3 Fixed μ and λ values

CrossMark has the option of entering fixed μ and/or fixed λ values for some (or all) cases on some (or all) points in time. We start with discussing three situations in which this option can be utilized to adjust the basic equations for the state probabilities p. We also explain how the option has to be specified in *CrossMark*.

In some applications, the values for μ and/or λ may be considered fixed and hence need not be estimated. This would e.g. be the case when the (backcasted) age of a respondent is 17 or younger in a study on voting behavior, given that the voting age is 18. Suppose, in the example given earlier, a respondent is 18 years old at the time that the third cross-section was observed (i.e., on $t = 3$). For this respondent we would like p_1 and p_2 to be zero; also, since p_3 is an entry probability (the respondent could not have voted for party A at $t = 2$) we would like p_3 to equal the entry probability μ_3. To implement these restrictions in the model equations, we fix $\mu_1 = \mu_2 = 0$ for this respondent, which implies the following adjusted equations for p_1 to p_5:

$$
\begin{aligned}
p_1 &= \mu_1 = 0\\
p_2 &= p_1(1-\lambda_2) + (1-p_1)\mu_2 = 0(1-\lambda_2) + 1\cdot 0 \quad = 0\\
p_3 &= p_2(1-\lambda_3) + (1-p_2)\mu_3 = 0(1-\lambda_3) + 1\cdot \mu_3 = \mu_3\\
p_4 &= p_3(1-\lambda_4) + (1-p_3)\mu_4\\
p_5 &= p_4(1-\lambda_5) + (1-p_4)\mu_5
\end{aligned}
$$

The equations for p_4 and p_5 have the usual Markov form, while those for p_1, p_2 and p_3 are adjusted in the sense specified above. We shall explain below how the fixed 0values for the μ probabilities in question for respondents younger than 18 have to be entered in *CrossMark*.

A second example of adjusting the basic equations for p is the following. Suppose all predictor variables we would like to use are constant over time, but only for a short time period. To be more specific, we assume that the predictor values for a case observed at time t also apply to $t-1$ and $t-2$, but not further back in time. Therefore, we let the Markov chain for each case start two time points preceding to the one the case was observed, instead of starting at time point $t = 1$ as we would have done, had the predictors been perfectly stable. This implies that the first state probability estimated for the cases of the cross-section at $t = 5$ will be p_3. For the cases of the cross-section at $t = 4$, p_2 will be the first estimated state probability, and for those of the cross-section at $t = 3$, $t = 2$ and $t = 1$, p_1 will be the first estimated state probability. This is different from the more general situation where, for all cases of all cross-sections, p_1 is the first estimated state probability. Remember that for p_1 we used a logistic equation, $p_1 = \mu_1$, with specific β parameters, different from the ones of μ_2 through μ_5. Here, we would like the same to hold for p_2 and p_3, as far as the cases of the cross-sections at $t = 4$ and $t = 5$ respectively are involved. To achieve this, we shall again use the equation $p_1 = \mu_1$ to estimate p_1 as the first estimated state probability for all cases of all cross-sections and then (i) let p_2 have the same value as p_1 for the cases of the cross-section at $t = 4$ and (ii) let p_3 have the same value as p_1 for the cases of the cross-section at $t = 5$. By doing so, we estimate three first state probabilities, p_1, p_2 and p_3, using the logistic equations $p_1 = \mu_1$, $p_2 = \mu_1$ and $p_3 = \mu_1$. At the same time p_2 and p_3 are also estimated by a Markov equation for the cases of the cross-sections at $t = 3$ and $t = 4$ respectively.

To specify the model we exploit fixed μ and λ values. Let us take a look at a case of the cross-section at $t = 5$ for which we want to estimate p_3 using the equation $p_3 = \mu_1$. We let $\lambda_2 = \lambda_3 = 0$ and $\mu_2 = \mu_3 = 0$, which results in:

$$
\begin{aligned}
p_1 &= \mu_1 \\
p_2 &= p_1(1-\lambda_2) + (1-p_1)\mu_2 && = \mu_1(1-0) + (1-\mu_1)\cdot 0 = \mu_1 \\
p_3 &= p_2(1-\lambda_3) + (1-p_2)\mu_3 && = \mu_1(1-0) + (1-\mu_1)\cdot 0 = \mu_1 \\
p_4 &= p_3(1-\lambda_4) + (1-p_3)\mu_4 \\
p_5 &= p_4(1-\lambda_5) + (1-p_4)\mu_5
\end{aligned}
$$

As can be seen, the equations for p_5 and p_4 are the usual Markov equations, while for p_3 we have $p_3 = \mu_1$. For cases of cross-section at $t = 4$ we proceed in a similar way by fixing $\lambda_2 = 0$ and $\mu_2 = 0$ which leads to $p_2 = \mu_1$. For the cases of the cross-sections at $t = 3, t = 2$ and

$t = 1$, we automatically have $p_1 = \mu_1$, so for these cases we do not need to fix any μ or λ.

The last example of using fixed μ and λ values concerns the analysis of discrete panel data. Consider a situation in which we have at our disposal a five wave panel data set without any inflow or outflow. The Markov model for discrete panel data reads as

$$p_t = y_{t-1}(1-\lambda_t) \ + \ (1-y_{t-1})\mu_t, \qquad t = 2,\ldots,5,$$

while for cross-sections, it reads as

$$p_t = p_{t-1}(1-\lambda_t) \ + \ (1-p_{t-1})\mu_t, \qquad t = 2,\ldots,5,$$

the difference being the use of y_{t-1} in the case of panel data and p_{t-1} when using cross-sectional data. As stated earlier, *CrossMark* uses the second equation since it was designed for the analysis of cross-sectional data. However, the program can simply be tricked to analyze panel data as well and thus to apply the first equation.

To do so, we first have to construct the data file in the way *CrossMark* expects it to be, i.e., according to the t-y-x-fre format. Each 'cross-section' in this data file corresponds to a particular wave of the panel data. The data for the first wave have to be placed at the top of the data file, followed by the data for the second wave, the third wave and so on. The order in which the respondents appear within the data for each wave is irrelevant and need not be the same for each wave.

Second, we need to define $p_{t-1} = y_{t-1}$ for $t = 2,\ldots,5$ or, to put it simply, $p_t = y_t$ for $t = 1,\ldots,4$. To do so we use fixed μ and fixed λ values. To make sure that $p_1 = y_1$, we simply let $\mu_1 = y_1$, resulting in $p_1 = \mu_1 = y_1$. For p_2 through p_4 we proceed as follows. If for a certain case $y_t = 0$ $(t = 2,\ldots,4)$, we let $\lambda_t = 1$ and $\mu_t = 0$, which results in $p_t = p_{t-1}(1-\lambda_t) + (1-p_{t-1})\mu_t = p_{t-1}(1-1) + (1-p_{t-1})\ 0 = 0$; thus $p_t = y_t = 0$, as was meant to be the case. If, on the other hand, $y_t = 1$, we let $\lambda_t = 0$ and $\mu_t = 1$, so that $p_t = p_{t-1}(1-0) + (1-p_{t-1}) = 1$; thus $p_t = y_t = 1$.

The third and final point concerns the fact that in models for panel data the likelihood is commonly computed for the data of $t \geq 2$, while in

CrossMark, the likelihood for $t=1$ is used as well. To delete the likelihood contribution of the cases for $t=1$ in *CrossMark*, we assign a very small frequency to the cases of the first wave (i.e., 0.0000000001) in the (t-y-x-fre) data file. We can also delete all cases of the first wave from the data file except one case, and assign the small frequency value to this single case. This single remaining case for $t=1$ may have any values on the Y and X variables since it only acts as a dummy case, having (virtually) no influence on the parameter estimates.

3.1 Specifying fixed μ and λ values in CrossMark

The fields **File with fixed mu-values** and **File with fixed lambda-values** in the Main Menu can be used to enter the names of the data files containing fixed μ and λ values for some or all cases of some or all cross-sections. The ‘file with fixed mu-values’ must contain one line for each case to which fixed μ values are assigned. Each line starts with the sequence number the case has in the (t-y-x-fre) data file and is followed by as many values 0, 1 or 9 as there are cross-sections. In the first example given above, where the age of a respondent (say the 316th respondent in the data file) was 18 years at the time point of the third cross-section, the line to enter in the ‘file with fixed-mu values’ for this respondent is the first of the two following lines:

```
316  0  0  9 9 9
925  0  0  0 0 9
```

Value 316 in the first line refers to the sequence number of the respondent; the two 0 values that follow are assigned to μ_1 and μ_2 and the three 9 values indicate that μ_3, μ_4 and μ_5 are not fixed, but have to be estimated. The second line refers to another respondent with sequence number 925 in the data file, who was 18 years old at $t=5$. In this example a ‘file with fixed lambda values' need not be specified, since only values of μ are fixed.

The ‘file with fixed lambda-values’ must contain one line for each case to which fixed λ values are assigned. Each line starts with the sequence number of the case in the data file and is followed by as many values 0, 1 or 9 as there are cross-sections minus 1, since these values relate to λ_2 through λ_T, T being the total number of cross-sections. The third example given above concerned the analysis of five-wave panel data without inflow and outflow. If we assume there are 500 respondents then the data file consists of 2500 lines, 500 lines for each wave. Suppose a particular respondent has the Y pattern 01100 for $t = 1,\ldots,5$. If the sequence number of the respondent in the first wave is 29, then the other four sequence numbers are 529, 1029, 1529 and 2029. In the 'file with fixed mu-values' and the 'File with fixed lambda-values' we have to enter the lines given in the box below.

File with fixed mu-values						File with fixed lambda-values					Wave
seqnr	μ_1	μ_2	μ_3	μ_4	μ_5	seqnr	λ_2	λ_3	λ_4	λ_5	
529	0	9	9	9	9						2
1029	9	1	9	9	9	1029	0	9	9	9	3
1529	9	9	1	9	9	1529	9	0	9	9	4
2029	9	9	9	0	9	2029	9	9	1	9	5

As can be seen, for the data of wave t we specify a fixed μ_{t-1} value in the 'file with fixed mu-values' equal to value of Y_{t-1}; e.g. for wave 3 we specify $\mu_2 = y_2 = 1$. The fixed λ_{t-1} value that has to be specified in the 'File with fixed lambda-values' for the data of wave t is equal to the complement of Y_{t-1}.

4 Unobserved heterogeneity

CrossMark offers the possibility to account for the influence of unobserved variables on the entry and exit probabilities. In doing so the assumption is made that the overall contribution of these variables to the logits of the transition probabilities is constant for the time period considered. The logit equations for μ and $1-\lambda$ including the contributions of unobserved variabels can be written as follows:

$$\text{logit}\,(\mu_t) = x\beta + \delta_1$$
$$\text{logit}\,(1-\lambda_t) = x\beta^* + \delta_2\,,$$

where x is a row vector with the values of the observed (potentially backcasted) predictors, β and β^* are the column vectors with the parameters associated with x, and finally δ_1 and δ_2 represent the total contribution of the unobserved variables. The values of δ_1 and δ_2 for all respondents (or cases) are considered to be drawn from a normal distribution with zero mean and variances γ_1^2 en γ_2^2. The above equations therefore can also be written as:

$$\text{logit}\,(\mu_t) = x\beta + \gamma_1 z$$
$$\text{logit}\,(1-\lambda_t) = x\beta^* + \gamma_2 z\,,$$

with $z \sim N(0,1)$ being the standardized contribution of the unobserved variables and γ_1 and γ_2 the parameters associated with the ‘predictor’ z. Since the z values for all cases are unknown the parameters β, β^*, $\hat{\gamma}_1$ en $\hat{\gamma}_2$ cannot be estimated. However, given a set of parameter values and the value of z, it is of course easy to determine the log likelihood contribution $\ell\ell$ of that case. Also, for a given set of parameter values, the expected (or marginal) log likelihood contribution $E(\ell\ell)$ of a case can be determined, where the expectation is taken over all possible values of z taken from $N(0,1)$. For a case of e.g. the cross-section at $t = 2$ it holds that:

$$E(\ell\ell) = \int_{-\infty}^{\infty} [\quad p_1(1-\lambda_2) + (1-p_1)\,\mu_2 \quad]\; f(z)\; dz \;\text{ if } y_2 = 1, \text{ and}$$
$$E(\ell\ell) = \int_{-\infty}^{\infty} [\quad (1-p_1)(1-\mu_2) + p_1\,\lambda_2 \quad]\, f(z)\; dz \;\text{ if } y_2 = 0$$

Here, μ_2 and λ_2 are defined as above (i.e., including z), p_1 is defined as usual (i.e., $p_1 = \mu_1$) without z (in *CrossMark*, controlling for unobserved variables is only possibly for the transitions probabilities at $t \geq 2$.), and $f(z)$ is the height of the standard normal pdf at z. The integrals cannot be derived analytically, but are approximated by CrossMark using Gaussian quadrature with 20 mass points. Utilizing the $E(\ell\ell)$ values of all cases of all cross-sections it is possible to estimate those values $\hat{\beta}$, $\hat{\beta}*$, $\hat{\gamma}_1$ en $\hat{\gamma}_2$ that, averaged over all values that z can take, have the highest expected (or marginal) log likelihood. The criterion to maximize in this estimation is the sum of the $E(\ell\ell)$ values of all cases of all cross-sections. The resulting estimates $\hat{\beta}$ en $\hat{\beta}*$ can be interpreted as the effects of the predictors x, corrected for the average influence of the unobserved variables. Using the above equations and estimation procedure has consequences for the standard errors of $\hat{\beta}$ and $\hat{\beta}*$, which can be quite different from the ones estimated without taking into account unobserved heterogeneity. The values of $\hat{\gamma}_1$ and $\hat{\gamma}_2$ are the estimates of the standard errors of δ_1 and δ_2 respectively, i.e., of the contributions of the unobserved variables to the logits of the entry and exit transition probabilities.

4.1 Testing the hypothesis $H_0 : \gamma_1 = \gamma_2 = 0$

To test this hypothesis we may use a test-procedure described by Snijders and Bosker (1999). We first calculate the value of $A = -2 \cdot \text{loglikelihood}$ for the model including $\gamma_1 z$ and $\gamma_2 z$. Then we compute $B = -2 \cdot \text{loglikelihood}$ for the model without $\gamma_1 z$ and $\gamma_2 z$ and obtain the difference $D = B - A$. Finally we test the difference D to be significant using a χ^2 distribution with 2 degrees of freedom, but halve the right tail probability associated with the value of D.

The standard estimation procedure in *CrossMark* does not take into account the possible influence of unobserved heterogeneity. If we wish to perform an analysis as described above, including the $\gamma_1 z$ and $\gamma_2 z$ terms in the equations for the transition probabilities, we have to go the Estimation Menu and click on the option called **Extra Bernoulli variance**. After running the model we will find the estimates $\hat{\gamma}_1$ en $\hat{\gamma}_2$ in the Output window.

5 Metropolis sampling

In the **Estimation window** an MCMC procedure can be performed that uses pure Meropolis sampling. To do so, check the option **Metropolis sampling** and specify a filename after **Outputfile posterior parameter values** for the file that the sampled parameter-points are written to. We only implemented this option in a very basic sense. There is e.g. no prior distribution that can be specified for the parameters: the implicit prior used for all parameters is the uniform distribution. After **Length of chain** specify the number of samples that has to be drawn from the posterior distributions of all parameters. Note that no burn-in period can be provided and, hence, the length of the chain must be large enough to also contain the desired burn-in period.

After pushing button **OK** *CrossMark* first performs the usual maximum likelihood (ML) estimation process. Once this is finished, the metropolis sampler is started. Consequently, metropolis sampling begins by default at the ML point. To start metropolis sampling from any other parameter-point, specify the parameter values for this point as the starting values to be used and also set the maximum number of iterations to 0.

It is possible to let *CrossMark*, for each sampled parameter-point, calculate the mean values of p_{it}, μ_{it} and λ_{it} over all cases i for each timepoint t. To this end, a filename must be entered after **Outputfile posterior mean p, mu, lambda**.

The value to be entered on the Estimation window in the sentence

Covariance matrix of the jumping distribution equals ...
times estimated covariance matrix of parameters

refers to what is discussed by Gelman, Stern and Rubin in 'Bayesian data analysis', 1995, on page 334 at the bottom where *c* = 2.4/sqrt(d). Value 2.4 for *c* is the default *CrossMark* uses if you don't specify another value in the above sentence. After the metropolis sampler is finished, inspection of the chain of sampled parameter-points (in the file specified after Output posterior parameter values) is always recommended, to make sure that the chain has changed fast enough. If the same parameter-points are resampled many times, a smaller value for *c* is probably more appropriate.

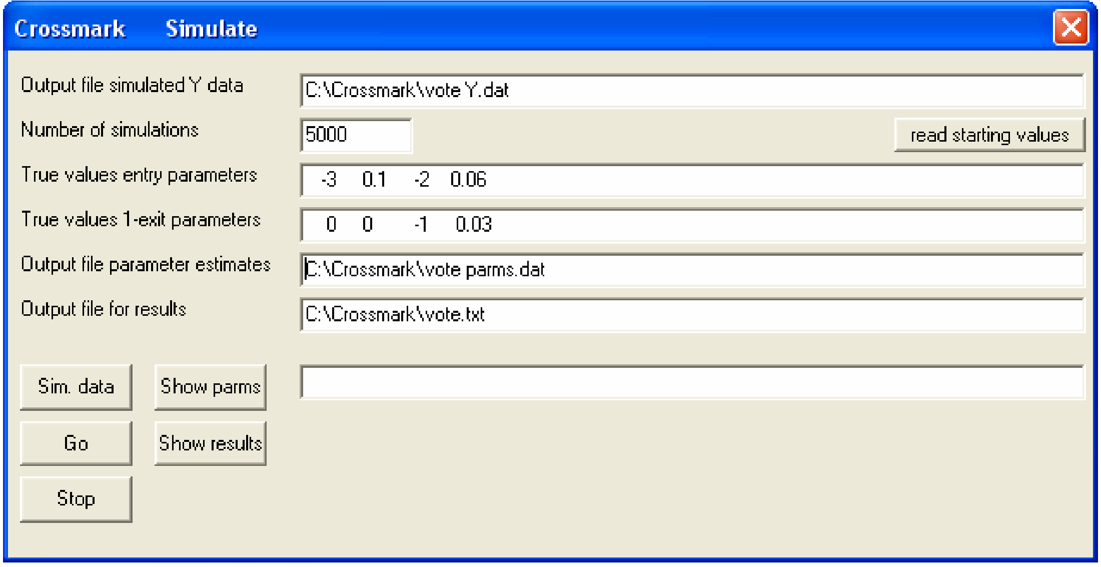
The parameter values that are sampled by the metropolis algorithm, are written out (to the file specified after **Outputfile posterior parameter values**) in the following format: sequence number (1,2,3,4,..., 100000, or more, depending on the length of the chain that was entered) followed by the values of the parameters of all predictors on the entry probabilities, followed by those on the 1-exit probabilities, followed finally by the loglikelihood value associated with these parameter values.

To evaluate the output files with posterior parameter values and/or means of p_{it}, μ_{it} and λ_{it}, other statistical software must be used. *CrossMark* itself does not perform any chain-evaluation, produces no histogram's of posterior parameter estimates and/or means, nor calculates means or standard deviations of the samples that were taken from the posterior distributions.

6 Parametric bootstrap

The **Simulate** button on the **Main Menu** opens the **Simulate** window where a parametric bootstrapping procedure can be performed. This window is shown in Figure 4. In the first step of the parametric bootstrap procedure a number of Y datasets are simulated, based on the observed X values in the data file and a set of true parameter values that must be specified after **True values entry parameters** and **True values 1-exit parameters.** The number Y datasets that have to be simulated is specified after **Number of simulations**. A name is generated automatically (but can be modified) for the output file that will contain the simulated Y data. After clicking the button **Sim. data** the simulation process starts, during which the Y data are generated and written to the file specified. Once the simulation has been finished, the next step can be started, during which the parameters will be estimated for samples that were simulated in the

Figure 4 Simulate window



Output file simulated Y data: C:\Crossmark\vote Y.dat
Number of simulations: 5000
True values entry parameters: -3   0.1   -2   0.06
True values 1-exit parameters: 0   0   -1   0.03
Output file parameter estimates: C:\Crossmark\vote parms.dat
Output file for results: C:\Crossmark\vote.txt

previous step. Estimation is started by pushing button **Go**. The estimated parameter values for all simulated Y datasets are written to the file specified after **Output file parameter estimates** in the following format: the sample number, the parameter values of all predictors for the entry probability, the parameter values of all predictors for the 1-exit probability, and, finally, the value of the loglikelihood. The file specified after **Output file for results** contains the final results, for all simulated Y datasets, similar to the ones that are generally shown in the Output window. As with the metropolis sampler, here again one will have to evaluate the estimated parameters with other statistical software. The two buttons **Show parms** and **Show results** show the corresponding files in Wordpad.