Event-History Analysis: Some Exercises

Ulrich Pötter, Götz Rohwer

The following exercises are intended to supplement an introductory course on event-history analysis. We assume that participants have access to the computer program TDA (Transition Data Analysis). This program is in the public domain and can be obtained from our home page, `www.stat.ruhr-uni-bochum.de`.

1. Assuming that you have access to a computer, begin with creating an environment for work on the exercises.

a) Create your private working directory. (Always work only in this private directory!)

b) Try to execute TDA. Simply type `tda`. The program should show up with a short message.

c) Invoke TDA in 'interactive mode'. Simply type

```
tda i
```

The program should show up with a command line, beginning with a colon, that allows you to enter commands. Try simple commands like `time` or `mpr(3+4)`. *Don't forget that each command must be finished by a semicolon.*

d) Try the '`help;`' command.

e) Leave the program with '`quit;`' or '`exit;`'.

2. Most often we shall use TDA in 'batch mode'. This means that one first creates a *command file* containing the commands to be executed by the program and then call the program to execute the commands in the command file.

a) Become familiar with one of the editors that you can find on your computer.

Box 1 Data file `eha1.dat`

```
ID   DUR   CEN
-------------
 1    17     1
 2     5     0
 3    22     1
 4    13     1
 5     2     0
 6     9     1
 7    12     0
 8    15     1
```

b) Create a command file, say `my.cf`, containing some commands to be executed by `TDA`.

c) Invoke TDA to execute the commands in the following way:

```
tda cf=my.cf
```

The program should then show the results on the screen.

d) You can save the results into an output file by invoking the program in the following way:

```
tda cf=my.cf > out
```

Try this and investigate the contents of `out`.

3. Create a data file that contains the data shown in Box 1. Then create a TDA command file that performs the following tasks.

a) Create an internal data matrix, using the `nvar` command.

b) Create a frequency distribution of the censoring variable, `CEN`, using the `freq` command. (Remember the convention: `CEN`=0 if the observation is censored, `CEN`=1 if the observation is not censored.)

c) Calculate the mean value of uncensored durations, using first the `tsel` command to select uncensored cases and then the `dstat` command for descriptive statistics.

Solution: `eha1.cf`

4. Create a data file that contains the data shown in Box 2. Then create a TDA command file that performs the following tasks.

Box 2 Data file `eha2.dat`

```
ID     X
---------
 1    17
 2    -5
 3    22
 4    13
 5    -2
 6     9
 7   -12
 8    15
```

a) Create an internal data matrix, using the `nvar` command.

b) Create new variables, `DUR` and `CEN`, where `DUR` is the absolute value of `X` and `CEN`=1 if `X` is positive and `CEN=0` if `X` is negative.[1]

c) Create a new data file that contains TDA's internal data matrix, using the `pdata` command.

d) Create another new data file that contains only the variables `ID`, `DUR`, and `CEN`, using the `pdata` command and, in addition, the `keep` parameter. This output file should be identical with `eha1.dat` as shown in Box 1.

Solution: `eha2.cf`

5. Use TDA's `edef` command (see `help edef`) to create an episode data structure based on data file `eha1.dat` (Box 1). Try two different ways to do this.

a) Origin state is 0, destination state is 1.

a) Origin state is 3, destination state is 9.

Solution: `eha3.cf`

6. Having defined an episode data structure with the `edef` command, one can use the `epdat` command (see `help epdat`) to write the episode data into an output file. In addition, one can request a TDA command file that describes the data in the output file and can be used to create a new internal data matrix.

[1] This can be done by defining new variables inside the first `nvar` command, or by using a new `nvar` command.

a) Try the `epdat` command with the episode data structure created in the previous exercise.

b) Use the command file created by the `epdat` command to read the output file into a new internal data matrix.

Solution: `eha27.cf`

7. Use the data shown in Box 1 and calculate the Kaplan-Meier survivor function for the variable `DUR`.

a) Do this with paper and pencil.

b) Do this with TDA's `ple` command (see `help ple`).

Solution: `eha4.cf`

Check whether you get the same result.

8. Consider the output file that you got from the `ple` command in the previous exercise.

a) Calculate an estimate of the median of the distribution by using linear interpolation of the survivor function values. You should get the same result as written at the end of the file (14.2, in this example).

b) Will it always be possible to estimate the median of the distribution?

c) Use the `qo` and `qt` parameters that are offered by the `ple` command to create a table with quantiles.[2]

Solution: `eha5.cf`

9. Use the output file from the `ple` command in Example 7 to create a plot of the survivor function. The steps are:

a) Use the `nvar` command to create an internal data matrix that contains variables for t and $\hat{G}(t)$, as found in the output file from the `ple` command.

b) Use the `xplot` command (see `help xplot`) to create a PostScript file. For example, if the variables are called `T` and `G`, use

```
xplot = T,G;
```

to create a scatter plot, or

```
xplot(opt=2) = T,G;
```

to create a line plot.

c) Use the `xshow` command (see `help xshow`) to see the plot on the screen.

See `eha6.cf` for an example. Also try to use this in interactive mode.

10. The `xplot` command is mainly intended for interactive use. In order to use all commands that `TDA` offers to create PostScript plots one should work in batch mode. While we do not intend here to discuss the creation of PostScript plots systematically, you may find an example in the command file `eha7.cf`.

11. Create a macro (see `help macro`) that can be used to plot a survivor function that has been estimated with the `ple` command. See the file `macro1.cf` for an example. Assume that you have used the `ple` command to create an output file, say `ple1.out` (see Exercise 7). You may then use the macro in interactive mode, or simply by calling `small TDA` in the following way:

```
tda cf=macro1.cf Plotple=ple1.out
```

Notice that a macro must first be loaded before it can be used.

12. The Kaplan-Meier procedure does not directly provide estimates of the rate. An estimate of the rate can be recovered, however, by differentiating a smoothed version of the estimated survivor function. Use paper and pencil to become familiar with this idea.

a) Use the results from Exercise 7 and plot a smoothed version of the survivor function, say $\hat{G}_s(t)$.

[2] Both, `qo` and `qt`, are optional parameters for the `ple` command, but only one of these parameters can be used in each `ple` command. `qt` must be given with a sequence of time points,

`qt` $= t_1, t_2, t_3, \ldots$

and then provides the corresponding values of the estimated survivor function. `qo` must be given with a *descending* sequence of values between 1 and 0 and then provides the corresponding quantiles.

Box 3 Data file `eha3.dat`

```
ID   DUR   CEN
-------------
 1    17     1
 2     5     0
 3    22     2
 4    13     1
 5     2     0
 6     9     2
 7    12     0
 8    15     1
 9    13     2
10     8     2
11    11     1
12     8     1
```

b) Graphically differentiate $-\hat{G}_s(t)$ to get an estimate of the density function, say $\hat{f}_s(t)$.

c) Plot $\hat{f}_s(t)/\hat{G}_s(t)$ to get an idea about the rate function.

13. Calculate lower and upper bounds for the Kaplan-Meier estimate of the survivor function (Exercise 7).

a) Calculate a lower bound by assuming that censored observations end with the observed censored duration.

b) Calculate an upper bound by assuming that all censored observations end at the longest observed duration.

c) Create a plot that shows the Kaplan-Meier estimate of the survivor function and its bounds.

Solution: `eha10.cf`

14. Box 3 shows a data file where episodes may end in one of two different destination states, 1 and 2.

a) Create a data file, `eha3.dat`, that contains these data.

b) Create a command file to set up a corresponding episode data structure. Use the `nvar` command to create an internal data matrix, then use the `edef` command to create an episode data structure with two destination states.

c) Create another episode data structure that recognizes only a single destination state (1 or 2).

Solution: `eha8.cf`

15. Use the episode data structures created in the previous exercise.

a) Based on the first episode data structure that distinguishes two different destination states, use the `ple` command to estimate corresponding sub-survivor functions.

b) Based on the second episode data structure that combines both destination states into a single one, use the `ple` command to estimate a standard survivor functions.

c) Check that the relationship is not additive, but multiplicative:

$$\hat{G}(t) \approx \hat{G}_1(t)\,\hat{G}_2(t)$$

Solution: `eha9.cf`

16. Consider a discrete rate defined by

$$r(t) = \Pr(T = t \,|\, T \geq t)$$

and the corresponding survivor function

$$G(t) = \Pr(T > t)$$

Derive the equation

$$G(t) \;=\; \prod_{\tau=1}^{t} (1 - r(\tau))$$

17. Consider the data in Box 1. Assume that you can only observe events if they occur at time point 10 or later, resulting in so-called *left truncated* data.

a) Set up a command file that uses only those cases from `eha1.dat` where `DUR` is at least 10.

b) Set up an episode data structure for left truncated data by explicitly providing a positive value (10, in this example) for the starting time.

c) Use the `ple` command to get a Kaplan-Meier estimate of the survivor function for the left truncated data.

d) Compare the result with the survivor function that was estimated from the complete data set. Verify that you have estimated

$$\Pr(T > t \,|\, T \geq 10) \;=\; \Pr(T > t) / \Pr(T \geq 10)$$

Solution: `eha11.cf`

18. Use the equation

$$G(t) = \exp\left\{ -\int_0^t r(\tau)\, d\tau \right\}$$

to derive the survivor and density functions of an exponential distribution, i.e., $r(\tau) = \theta$ (constant).

19. Create a table that contains three columns:

a) Values of a time variable, $t = 0\,(0.1)\,5$.

b) Corresponding values of the survivor function of an exponential distribution with $\theta = 2$.

c) Corresponding values of the density function.

Solution: `eha12.cf`

20. Create a plot for the survivor function of an exponential distribution with $\theta = 2$, in the range $0 \leq t \leq 4$.

Solution: `eha13.cf`. To see the plot, use TDA in interactive mode. First, use

```
xopen = plot4.ps;
```

to make `plot4.ps` (or whatever the name of your PostScript file) the currently active plot file. Then use `xshow` to see the plot. Alternatively, you can call TDA as

```
tda xopen=plot4.ps xshow
```

21. Consider fitting an exponential distribution to the data shown in Box 1. The maximum likelihood estimate of the parameter, say θ, is given by

$$\hat{\theta}_{\text{ML}} = \frac{N_u}{T_w} \tag{1}$$

where N_u is the number of uncensored observations and T_w is the summed duration of all observations. Calculate $\hat{\theta}_{\text{ML}}$ for the data in Box 1.

22. Use TDA's `rate` command (see `help rate`) to fit an exponential distribution to the data in Box 1. The command is `rate=2` to estimate a model without covariates. The model is then parameterized as

$$\theta = \exp(\alpha)$$

where θ is the parameter of an exponential distribution. The command provides an ML estimate for the model parameter, α. Check whether you get the same result as you have found in the previous exercise.

Solution: `eha14.cf`

23. Try to prove formula (1). Proceed as follows.

a) Set up the likelihood function for fitting an exponential distribution, being a function of the parameter θ.

b) Derive the log likelihood function, say $\ell(\theta)$.

c) Calculate a solution for

$$\frac{\partial \ell(\theta)}{\partial \theta} = 0$$

As a by-product, derive a formula that allows to directly calculate the maximum of the log-likelihood function. Calculate this maximum for the data in Box 1 and compare with the output from the `rate` command.

24. Derive the likelihood for fitting an exponential distribution to censored data. The parameterization should be

$$\theta = \exp(\alpha)$$

where θ refers to the parameter of the exponential distribution and α is the parameter to be estimated. If `DUR` denotes the duration and `CEN` the censoring indicator, the contribution of the ith observation to the log-likelihood should then be derivable as

$$\texttt{CEN}(i)\,\alpha - \mathrm{DUR}(i)\,\exp(\alpha)$$

Use TDA's `fml` command (see `help fml`) to maximize the log-likelihood function (i.e., the sum over all individual contributions to the log-likelihood) and find an estimate of α. Compare the result with the estimate found in Exercise 22.

Solution: `eha36.cf`

25. Refer to the exponential distribution fitted in Exercise 21.

a) Derive the formula for the mean value of an exponential distribution,

$$E(T) = \int_0^\infty t\,f(t)\,dt = \int_0^\infty \theta t \exp(-\theta t)\,dt = \frac{1}{\theta}$$

and calculate the estimated mean value.[3]

b) Derive a formula for the median of an exponential distribution. and use the formula to estimate the median from the fitted distribution.

c) Compare with the estimate of the median based on the Kaplan-Meier procedure for the survivor function (see Exercise 7).

26. When fitting transition rate models to single episode data, TDA's `rate` command uses the likelihood

$$\mathcal{L}(\theta) = \prod_{i \in \mathcal{E}} f(t_i;\theta) \prod_{i \in \mathcal{Z}} G(t_i \,|\, s_i;\theta) \tag{2}$$

where $\mathcal{E}$ and $\mathcal{Z}$ denote, respectively, the index sets for the uncensored and censored observations; t_i is the ending time and s_i is the starting time in the ith observation. $G(t\,|s;\theta)$ denotes the conditional survivor function, defined by

$$G(t \,|\, s;\theta) = \frac{G(t;\theta)}{G(s;\theta)}$$

[3] Try to prove the formula by using partial integration. The rule for partial integration is

$$\int F(t)g(t)\,dt = F(t)G(t) - \int f(t)G(t)\,dt$$

where $f(t) = dF(t)/dt$ and where $g(t) = dG(t)/dt$. Use $F(t) = \theta t$, $g(t) = \exp(-\theta t)$.

Box 4 Data file `eha8.dat`

```
ID   TS   TF    CEN
-------------------
 1    0   10     0
 1   10   17     1
 2    0    3     0
 2    3    5     0
 3    0   11     0
 3   11   22     1
 4    0   12     0
 4   12   13     1
 5    0    1     0
 5    1    2     0
 6    0    6     0
 6    6    9     1
 7    0    8     0
 7    8   12     0
 8    0   10     0
 8   10   15     1
```

a) Consider the corresponding likelihood function for the exponential model and derive that parameter estimates will not change when one adds a constant value to all starting and ending times.

b) Check whether TDA does this correctly by adding a constant value, say 10, to the starting and ending times of the durations in Box 1. This can be done by modifying the command file `eha14.cf` discussed in Exercise 22.

Solution: `eha30.cf`

27. The fact that TDA uses the likelihood function (2) allows to apply the so-called method of *episode splitting*. Assume that an observation has starting time s_i and ending time t_i. Its contribution to the likelihood should therefore be $G(t_i \,|\, s_i;\theta)$.[4] Now, the same contribution can also be given by

$$G(t_i \,|\, s_i;\theta) = G(t_i \,|\, \tau_i;\theta)\,G(\tau_i \,|\, s_i;\theta)$$

where τ_i is some time point that splits the period from s_i to t_i into two parts ($s_i < \tau_i < t_i$). For example, consider the data in Box 4. These data have been derived from the data in Box 1 by arbitrarily splitting

[4] And, if the observation is not censored, also $f(t_i;\theta)$.

Box 5 Data file `eha9.dat`

```
ID   DUR   CEN    S
-------------------
 1    17     1   10
 2     5     0    3
 3    22     1   11
 4    13     1   12
 5     2     0    1
 6     9     1    6
 7    12     0    8
 8    15     1   10
```

each duration into two parts. Of course, the first part does not end in an event and should always be treated as a censored (sub-) episode.

a) Set up an episode data structure for the data shown in Box 4 and estimate an exponential model.

b) Check whether estimation results are identical with those from Exercise 22.

Solution: `eha31.cf`

28. Episode splitting can be performed with the `edef` command (see `help edef`). One only needs to supply variables containing the time points for splitting. To illustrate this option, consider the data in Box 5. The data are identical to those in Box 1, we only added a further column (`S`) containing time points for splitting the episodes.

a) Set up a command file that reads data file `edat9.dat` (Box 5).

b) Use the `edef` command to create an episode data structure and the `split=S` parameter to request episode splitting at the time points given by variable `S`.

c) Use the `epdat` command to create a new output file containing the splitted episodes. Check that the resulting output file contains the same information as the data in Box 4.

Solution: `eha32.cf`

29. Also TDA's Kaplan-Meier procedure uses conditional survivor functions. (See the description of the `ple` command in the manual.) One can therefore apply the `ple` command to episode data that have been splitted and should get the same result as if the episodes were not split. Check this with the data file created in the previous exercise. Assume that the command file `eha32.cf` contains the command

```
epdat(dtda=t) = eha9a.dat
```

You can then make the file `t` to become the starting point for a new command file, say `eha33.cf`, that reads the data file `eha9a.dat`, creates an episode data structure with the `edef` command, and then requests a Kaplan-Meier estimate of the survivor function with the `ple` command. The resulting survivor function should be identical with the estimate produced in Exercise 7.

Solution: `eha33.cf`

30. The technique of episode splitting is mainly used to provide a simple way of incorporating time-varying covariates. It therefore suffices to split episodes at time points where a covariate changes its value. Since episode splitting does not change the information contained in a set of episode data it is possible, however, to split episodes at each possible time point. This is sometimes done when the data are defined on a discrete time axis. It would then be possible to apply, for example, standard procedures for estimating logit and probit models.

a) Set up a command file that splits the episodes in Box 1 at all integral time points and write the data into a new output file, say `eha1a.dat`.

   Solution: `eha34.cf`

b) Set up a command file that uses `eha1a.dat` to estimate a simple logit model for the event that occurs when an episode ends. (The command is `qreg`, see `help qreg`.) If the state space is $\{0, 1\}$, where $Y = 1$ denotes the destination state, the model would be

$$\Pr(Y = 1) = \frac{\exp(\alpha)}{1 + \exp(\alpha)}$$

   Solution: `eha35.cf`

c) Calculate the estimated probability for the occurrence of an event and derive a comparable estimate from fitting an exponential transition rate model. Compare both estimates.

31. Derive the log-likelihood function for the simple logit model without covariates that was used in the previous exercise. Then use the `fml` command to estimate the parameter, α. Compare with the parameter estimate that you got in the previous exercise.

Solution: `eha37.cf`

32. Let $F(t)$ denote a distribution function. Then, if r is a random variable that is equally distributed in $[\,0,1\,]$, $F^{-1}(r)$ is a random variable with a distribution described by F.

a) Use this idea to derive a formula that can be used to create exponentially distributed random numbers.

b) Use TDA's operator for equally distributed random numbers (`rd`) and the formula derived under (a) to create 100 exponentially distributed random numbers ($\theta = 2$).[5]

c) Fit an exponential distribution and check the estimated value of θ.

Solution: `eha15.cf`

33. Continue with the previous exercise and introduce some censored observations. One possibility is as follows: Let t_i denote the original uncensored duration for case i. Then, for each case i, draw another random number, say r_i, equally distributed in $[\,0,1\,]$, and assume that case i is censored at duration 0.5 if $r_i \leq 0.5$ and $t_i \geq 0.5$.

Solution: `eha16.cf`

34. Create data for $n = 100$ cases. Define a dummy variable, say `GRP`, that takes the value 1 for the first 50 cases and value 0 for the remaining 50 cases. For each case create an exponentially distributed duration, $\theta = 2$ if `GRP` $= 1$ and $\theta = 3$ if `GRP` $= 0$.

a) Estimate an exponential model that contains `GRP` as a covariate. Check whether you can recover estimates of the parameters that have been used for data generation from the results of the model estimation.

Solution: `eha17.cf`

[5] Use the `nvar` command. The number of cases can then be fixed with the parameter `noc=100`.

Box 6 Data file `eha4.dat`

```
ID  T1  T2  T3  CEN
-------------------
1   50  66  73   1
2   56  71  81   0
3   45  63  88   1
4   70  87  97   1
5   72  90  -1   0
6   58  75  80   1
7   60  77  82   1
8   65  82  -1   0
```

b) Estimate an exponential model for each group separately and compare the parameter estimates with the estimates you got in (a).

Solution: `eha18.cf`

35. Having fitted an exponential distribution to a set of durations, one can use a simple graphical method to check goodness-of-fit. The method uses the survivor function of the exponential distribution,

$$G(t) = \exp(-\theta t)$$

The graphical check uses the transformation[6]

$$-\log(G(t)) = \theta t$$

One first estimates the survivor function non-parametrically, e.g., with the Kaplan-Meier procedure, resulting in an estimate $\hat{G}(t)$, and then plots $-\log(\hat{G}(t)$ against t. If the exponential distribution fits the data one should get, approximately, a straight line through the origin.

a) Apply this check to the data created in Exercise 32.

Solution: `eha28.cf`

b) Apply this check to the data in Box 1.

Solution: `eha29.cf`

In both examples, add a straight line representing the fitted exponential distribution. For the first example, use $\theta = 2$; for the second example use $\theta = 0.0526$ as resulting from Exercise 22.

[6] In this text, log() always means the natural logarithm.

Box 7 Data file and required output file

```
Data file: eha5.dat

 I   T   X
-----------
 1   3   1
 2   4   2
 3   2   7

Required output file: eha6.dat

 I   T   X  Cnt
---------------
 1   3   1   1
 1   3   1   2
 1   3   1   3
 2   4   2   1
 2   4   2   2
 2   4   2   3
 2   4   2   4
 3   2   7   1
 3   2   7   2
```

36. Consider the data shown in Box 6. Each case is described by two, or three, dates, given in calendar time. (You may assume that `T1` records birth date, `T2` records end of schooling, and `T3` records first marriage.) `T2` is censored if `T3` is missing, `T3` is censored if `CEN` $= 0$. `T1` is always observed.

a) Create a data file that contains, for each case, information about its first episode, recorded in process time.[7]

b) Create a data file that contains, for each case that has a second episode, information about its second episode, recorded in process time.

Solution: `eha19.cf`

37. Modify the command file that you have created in the previous exercise in order to set up an episode data structure, both for first and second episodes.

Solution: `eha20.cf`

[7] The term 'process time' is used to refer to a time axis where the first episode always begins at time zero.

Box 8 Command file `eha21.cf`

```
nvar(
    dfile = eha5.dat,
    I = c1,
    T = c2,
    X = c3,
);
mfmt = 2.0;
repeat(n = noc,Case);
    repeat(n = T(Case,1),TCnt);
        mcath(I(Case,1),T(Case,1),X(Case,1),TCnt,Tmp);
        mpra(Tmp) = eha6.dat;
    endrepeat;
endrepeat;
```

38. The next step is to create multi-episode data. This can be done with TDA's matrix and loop commands. To learn some of these options, consider the data file, `eha5.dat`, shown in Box 7.[8] There is an ID variable (`I`), a variable that counts time periods (`T`), and some further covariate (`X`). The file contains a single record for each ID number. Now assume that you want a new data file that contains, for each ID number i, `T`(i) records, as shown in the lower part of Box 7. This can be done with the command file `eha21.cf` shown in Box 8.

a) The `nvar` commands reads the input data file, `eha5.dat`, and creates the three variables, `I`, `T`, and `X`.

b) The `mfmt` command specifies a print format for the `mpra` command which is used later in the command file.

c) Then follows a `repeat` command that repeats the following commands, until the matching `endrepeat`, a number of times as defined by the `n` parameter. In this case, `n = noc`, that is, the number of cases in the data matrix. In addition, the command creates a $(1, 1)$ matrix `Case` that gets the value `Case` $= 1, \ldots, n$ while being in the repeat loop.

d) Then follows a second `repeat` command where the repeat variable,

[8] This example is taken from the paper *Using TDA Matrix Commands and Loops for Data Generation and Selection*. The paper is available in the `contrib` directory of the TDA homepage. We recommend that you also study the other examples discussed in that paper.

Box 9 Data file `eha7.dat`

```
ID NS SN  TS TF CEN
-------------------
 1  2  1  50 66  1
 1  2  2  66 73  1
 2  2  1  56 71  1
 2  2  2  71 81  0
 3  2  1  45 63  1
 3  2  2  63 88  1
 4  2  1  70 87  1
 4  2  2  87 97  1
 5  1  1  72 90  0
 6  2  1  58 75  1
 6  2  2  75 80  1
 7  2  1  60 77  1
 7  2  2  77 82  1
 8  1  1  65 82  0
```

`TCnt`, now runs in the range $1, \ldots,$ `T(Case,1)`. The latter expression refers to the value of variable `T` in the current data matrix row as given by `Case`.

e) The inner repeat loop contains two commands. The first one, `mcath` (= horizontal concatenation), creates a row vector, `Tmp`, that consists of the current values of the three variables and, in addition, the current value of `TCnt`.

f) The second command in the inner loop, `mpra`, appends the row vector `Tmp` to the output file `eha6.dat`.

The final result is the output file `eha6.dat` as shown in the lower part of Box 7. Note that when running the command file, the matrix and loop commands will not, by default, give any echo in the standard output. Such an echo might be helpful when debugging a command file and can be requested with the `silent=-1` command.

39. Now try to transform the data file `eha4.dat` (Box 6) into a multi-episode data file that should look similar to the file `eha7.dat` shown in Box 9.

Solution: `eha22.cf`

40. Use the data file `eha7.dat`, created in the previous exercise, and the `edef` command, to set up a multi-episode data structure. This should be done on a process time axis where the first episode for each individual begins at time 0.

Solution: `eha23.cf`

41. Continue with the previous exercise and consider, for each time point on the process time axis, the cross-sectional distribution of cases in the state space, $\{0, 1, 2\}$, in this example. This will be called a *state distribution*. Use the `epsdat` command (see `help epsdat`) to calculate a state distribution for the time points $t = 0, 1, 2, \ldots, 50$.

Solution: `eha24.cf`

42. Continue with the multi-episode data created in Exercise 40.

a) Estimate an exponential model without covariates simultaneously for first and second episodes.

b) Estimate separate exponential models for first and second episodes.

c) Derive from the likelihoods of the models that one should get identical parameter estimates.

Solution: `eha25.cf`

43. Transform the multi-episode data created in Exercise 40 into sequence data, on a process time axis that runs from 0 to 50. Use the `seqpe` command (see `help seqpe`).

Solution: `eha26.cf`

44. Consider the Weibull distribution.

a) Show that the exponential distribution is a special case of the Weibull distribution.

b) Create a command file that plots the survivor function of the Weibull distribution,

$$G_{a,b}(t) = \exp(-(at)^b) \tag{3}$$

for parameter values $a = 1$ and $b = 2$, in the range $0 \leq t \leq 2$. Use the `plotf` command (see `help plotf`).

Solution: `eha38.cf`

45. Continue with the Weibull distribution.

a) Derive a formula for the inverse survivor function,

$$t = \exp\left\{\frac{\log\left(-\log(G(t))/a^b\right)}{b}\right\} \tag{4}$$

b) Use this formula to create 100 random durations which are distributed according to a Weibull distribution with $a = 1$, $b = 2$.

Solution: `eha39.cf`

46. Use the random data created in the previous exercise.

a) Use the `ple` command to find a Kaplan-Meier estimate of the survivor function.

Solution: `eha40.cf`

b) Use the macro created in Exercise 11 to see a plot of the estimated survivor function. Assuming that you have written the estimated survivor function into an output file, `wei.ple`, you may use

```
tda cf=macro1.cf Plotple=wei.ple
```

c) Create a plot that shows, simultaneously, the theoretical and the estimated survivor function.

Solution: `eha41.cf`

47. Use TDA's `rate` command to estimate a Weibull model for the data created in Exercise 45. The model number is `rate=7` (see `help rate model number`). Notice that TDA's Weibull model uses the parameterization

$$a = \exp(\alpha) \quad b = \exp(\beta)$$

Calculate the estimated values for a and b and compare with the values that were used for data generation.

Solution: `eha42.cf`

48. Consider the Weibull model parameterized with $a = \exp(\alpha)$ and $b = \exp(\beta)$.

a) Derive the log-likelihood for ML estimation of α and β.

b) Use the `fml` command to estimate α and β with the data created in Exercise 45.

Solution: `eha43.cf`

49. Remember the graphical method to check goodness-of-fit of an exponential distribution that was discussed in Exercise 35. Think of a similar method for the Weibull distribution.

a) Derive the formula

$$\log(-\log(G(t))) = b\log(a) + b\log(t)$$

from the survivor function of the Weibull distribution.

b) Use the Kaplan-Meier estimate of the survivor function that was created in Exercise 46 to plot

$$\log(-\log(G(t))) \text{ vs. } \log(t)$$

If the Weibull model fits the data (what should be the case in this example), the plot should exhibit a straight line.

Solution: `eha44.cf`

c) Use the plot to graphically determine estimates of a and b and compare with the values that were used to create the data.

50. Continue with the Weibull distribution.

a) Derive a general formula for the median of the Weibull distribution in terms of the parameters, a and b.

b) Calculate the median of a Weibull distribution with $a = 1$ and $b = 2$.

c) Compare with the Kaplan-Meier estimate of the median that was calculated in Exercise 46.

51. We now discuss some difficulties that occur when one tries to fit a Weibull model to the data in Box 1.

a) Try to estimate a Weibull model with TDA's `rate` command. (Use a suitably modified version of command file `eha14.cf` that was used in Exercise 22.) You will find that TDA is not able to estimate a Weibull model with these data when beginning with default starting values.

b) Fix the value for the b parameter to estimate an exponential model as a special case of the Weibull model (see Exercise 44). Since in the TDA parameterization we have $b = \exp(\beta)$, use the constraint

```
con = b2 = 0,
```

You should then get the same estimate for α as was found in Exercise 22.

Solution: `eha45.cf`

c) Now try to fix β at some other value, say $\beta = 1.5$, and check whether you get a better fit. Use the value of the log-likelihood as a criterion. Also write the estimated parameter values into an output file, say `sv`, using the `ppar` parameter for the `rate` command.

Solution: `eha46.cf`

d) Now use these parameter values as starting values to fit an unrestricted Weibull model.

Solution: `eha47.cf`

e) Use a graphical method to check whether the finally estimated Weibull model fits the data in Box 1.

52. We now discuss some options provided by TDA's `rate` command.

a) Add the parameter

$$\text{prate (tab=0(1)20)} = \text{rate.dat},$$

to the command file `eha47.cf` that was used in the previous exercise. You will get an output file, `rate.dat`, containing the estimated rate, survivor and density functions for the time points $t = 0, 1, \ldots, 20$.

b) Add also the parameter

$$\text{pres} = \text{res.dat},$$

You will get an output file, `res.dat`, containing the so-called *generalized residuals*. For information about the contents of this file, see Section 6.17.1.6 of the TDA manual.

Solution: `eha48.cf`

53. Generalized residuals can be used to check whether a transition rate model fits the data. The basic idea is quite simple. If the model provides a good fit one can expect that generalized residuals behave like random numbers drawn from a standard ($\theta = 1$) exponential distribution. Therefore, to perform the check, one uses the Kaplan-Meier procedure to create an estimate of the survivor function for the residuals and then graphically checks whether one gets a straight line.

54. We have now finished with a selection of most basic exercises. It remains to apply what we have learnt to more complex data sets that also provide an opportunity to include covariates. For this task we continue with an example data set, `rrdat.1`, that provides observations of job histories for 201 individuals.[9] The variables contained in this data set are shown in Box 10.

a) Begin with investigating the first records of the data file shown in Box 11.

b) Set up a command file that reads the data into an internal data matrix.

c) Use the `edef` command to create different versions of single and multi-episode data structures.

d) Find survivor functions with the `ple` command.

e) Estimate transition rate models with the `rate` command.

[9] This data set has been used by Blossfeld and Rohwer, Techniques of Event History Modeling (Lawrence Erlbaum 1995), and is also used in the TDA manual, Section 3.3.3.

Box 10 Variables in data file `rrdat.1`

```
Variable  Column  Description
-----------------------------------------------------
  ID        C1    ID of individual
  NOJ       C2    Serial number of the job
  TS        C3    Starting time of the job
  TF        C4    Ending time of the job
  SEX       C5    Sex (1 men, 2 women)
  TI        C6    Date of interview
  TB        C7    Date of birth
  TE        C8    Date of entry into the labor market
  TM        C9    Date of marriage (0 if no marriage)
  PRES      C10   Prestige score of job i
  PRES1     C11   Prestige score of job i + 1
  EDU       C12   Highest educational attainment
```

Box 11 First records of data file `rrdat.1`

```
ID NOJ   TS   TF  SEX   TI   TB   TE   TM PRES PRES1 EDU
--------------------------------------------------------
 1   1  555  982    1  982  351  555  679   34    -1  17
 2   1  593  638    2  982  357  593  762   22    46  10
 2   2  639  672    2  982  357  593  762   46    46  10
 2   3  673  892    2  982  357  593  762   46    -1  10
 3   1  688  699    2  982  473  688  870   41    41  11
 3   2  700  729    2  982  473  688  870   41    44  11
 3   3  730  741    2  982  473  688  870   44    44  11
 3   4  742  816    2  982  473  688  870   44    44  11
 3   5  817  828    2  982  473  688  870   44    -1  11
```