Management of Event History Data and Episode Splitting Using the AGIPEB-Dataset

Daniel Baron

Christina Krüger

1.   Introduction[1]

During the AGIPEB-Project several event history analyses were implemented to find answers to the question to what extent fixed-term employment as well as other sociostructural or labour market variables influence transition-rates into partnership-stabilizing events (Baron 2015, Baron, Rapp 2017b, Baron et al. 2015). These events under consideration were: cohabitation, marriage, parenthood and homeownership. While we relied on cox regressions and exponential models in earlier studies, we went on to use piecewise-constant models in later phases of the project.

Other than cox regression models or exponential models, piecewise-constant models allow for calculating flexible transition rates for each time-span under consideration (Blossfeld et al. 2007: 116-127). In our models, process-time and, thus, all the event history episodes were modeled using each participant's reported occupational and partnership biography.

Furthermore, piecewise-constant models give the opportunity to implement time-invariant as well time-dependent covariates. Examples for the former variable-type would be gender or levels of education, examples for the latter would be age or type of employment spell, such as fixed-term employment, permanent employment or being in education.

Event history analysis, especially piecewise-constant models, often require detailed data transformation and sophisticated data management techniques. First, each interviewee's sub-dataset has to be reshaped from wide- into long-format. This means that there is not only one row per interview in the dataset containing each participant's cross sectional data, but each participant is recorded with each of his reported event history episode. If there are, for example, five employment spells – one educational spell, two spells of fixed-term and two of permanent employment – then this particular participant's sub-dataset will contain five rows of reported covariates.

Second, to allow for calculating – flexible as well as constant – transition-rates the change of every time-dependent covariate has to be taken into consideration for each participant's sub-dataset. This that means that whenever a time-dependent covariate changes its value – for example: if a participant changes from fixed-term to permanent employment – this change of state has to be resembled in the sub-dataset. This aim is being achieved by introducing a split in the sub-dataset at the time-point whenever the state of a time-dependent variable changes (Blossfeld et al. 2007: 137f.). These time-splits, then, contain information on the origin state of the interesting episode, the values of each covariate at the start of the episode, information on starting and ending points of the episode and information whether the episode under consideration is (right-)censored or reaches its end after the event occurred (Blossfeld et al. 2007: 138).

Third, piecewise-constant exponential models in particular require a somewhat more detailed approach to episode splitting. Here, each participant's sub-dataset contains not only all episodes – for example: 10 months of being in education, five months of fixed-term em-

---

ployment and 35 months of being in permanent employment – but all the reported time-spells. If, for example, a participant reported a total amount of 50 months that he or she spent in his current partnership, his or her sub-dataset will contain 50 rows within the main-dataset. Each row contains information on the participant's time-invariant as well time-dependent covariates.

In this paper we document and describe the data transformation and episode splitting-procedure in the course of our later piecewise constant-exponential models (Blossfeld et al. 2007). First, a short overview over the used variables as well as the sample is given. Then, we document the transformation of our data. All data transformation-procedures as well as statistical analyses were conducted using Stata (Version 14, IC). For details on the empirical background as well theoretical assumptions of our analyses see the overview in Baron (Baron 2015). A general overview over the sample and the study-design is provided by Eickemeier et al. (2016). Due to restrictions in space and time, we could not provide a deeper methodical insight into the basics of event history-analyses. For introductory purposes, we recommend the textbooks by Blossfeld et al. (2007) and Cleves et al. . (2010)

2.   Sample description

The empirical analyses within the study rely on a random sample consisting of 1,083 adults. Primarily 13,483 people passed the screening-questionnaire. All remaining participants were determined by complying with certain criteria. We had to make sure that all participants were not only in an age-rage of twenty to thirty-five years, but also being in a relationship and working either under a fixed-term contract or under a permanent contract throughout the survey. The interviews for our data collection were conducted in a standardized way from November 2012 until March 2013.

Off the 1,083 adults, 543 indicated that they had worked at least once under a fixed-term contract during their occupational biography, whereas 530 participants reported that they had worked under a permanent contract most of their working life so far, but never under a fixed-term contract. These data could be used to separate the sample into a treatment group and a control group. The treatment group consisted those actors who experienced working under a fixed-term contract at least once during their occupational biography, the control group covered the participants who had worked under a permanent contract most of the time. Overall, 617 participants reported they were female, which makes a total of 57% of all interviewees. Furthermore, selection bias throughout the sampling process caused overrepresentation of highly educated persons: 63.2% got A-levels respectively university of applied sciences entrance qualification.

3.   Used variables

Within the study we mainly focused on four partnership events: cohabitation, marriage, transition to parenthood and real estate purchase. The spells measuring time until each of those partnership events occurred during each participant's relationship were set as dependent variables. During the interview each participant was asked, if the aforementioned events

already occurred during their partnership[2] . The answers were dichotomized whereat the codes were used as follows: 1 means 'event did occur', 0 means 'event did not occur'. In addition to that we collected precisely the number of months it took until the event did occur[3] . Those allegations were not listed in a separate variable, but were calculated in the course of data preparation (see section 4). Regarding transition to parenthood, we only inquired the years when the children were born, and stated all birthdates on July of the respective years, in order to ensure the participants' privacy.

The employment status was set as independent variable. All participants were asked about their employment status during their occupational biography. The employment status is a time-dependent variable. The interviewees were queried about four employment status: episode of working under a permanent contract (Eb_unbefr), episode of working under a fixed-term contract (Eb_befr), episode of unemployment (Eb_al) and being in educational system (Eb_bild). For all four status, the answer was dichotomized with answers coded as1, when the status occurred during each participant's occupational biography and 0 when it did not. The length of each spell was calculated on a monthly basis by cumulating the spells for each status.

The length of the each participants' relationship was also measured in months (pb_dauer). Herefrom, spells and covariates were calculated for the piecewise constant models during the AGIPEB-project (pb1_2jahr, pb2_2_kind_jahr, etc.). For more detailed explanation see section 4.

As already mentioned during the sample description, gender (v8), age (v7_jahre) and educational status (v192) are important for our analyses. Concerning the age, we composed an additional variable by squaring the age (v7_jahre_quadriert). Furthermore, we calculated a time-variant age-variable based on the length of each participant's reported occupational biography. Additionally, the variable educational status was dichotomized into high (A-levels and university of applied sciences entrance qualification) and low (middle school and lower secondary school) educational level. The variable subjective precariousness (skala_spw) was based on a likert scale (Baron 2015). It was mean centered for the later analyses during the AGIPEB-project (for technical information see section 4).

4.   Data transformation

In this section we present the syntax used for episode-splitting. All data transformation was conducted using Stata IC, Version 14. The do-file and data set can be obtained after containing the corresponding author of this paper. The resulting event history data was used for piecewise constant models conducted in a series of studies based on the AGIPEB-project (Baron 2015, Baron, Rapp 2017a; b).

---

[2]   pb4_haushalt_event, pb1_hochzeit_event, pb2_kind_event, pb8_immobilie_event
[3]   pb4_haushalt_risktime, pb1_hochzeit_risktime, pb2_kind_risktime, pb8_immobilie_risktime

4.1 Preparing data for episode splitting

For all the data transformation and analyses in our recent papers (Baron 2015, Baron, Rapp 2017a; b, Baron et al. 2015), we used version 26 of the AGIPEB-dataset. After loading the data into Stata's memory processor we deactivated manual page turning of the results by using the set more off-command.

```
clear
use \\DATENKRAKE\projekte$\Prekaer\MANUSKRIPTE\Artikel-
Ingmar_Daniel\Aufsatz_International\STATA-
Datensätze\Daten_AGIPEB_26-STATA_new.dta
set more off
```

Then, we shortened the names of all the variables that provide information on wether a partnership-event has taken place or not.

```
rename pb1_hochzeit_event pb1_event
rename pb2_kind_event pb2_event
rename pb3_urlaub_event pb3_event
rename pb4_haushalt_event pb4_event
rename pb5_kasse_event pb5_event
rename pb6_spareinlage_event pb6_event
rename pb7_auto_event pb7_event
rename pb8_immobilie_event pb8_event
```

In order to facilitate the later reshaping of the data from wide- to long-format we rearranged the names of all the indicators of occupational spells.

```
rename eb1beginn eb_beginn1
rename eb2beginn eb_beginn2
rename eb3beginn eb_beginn3
rename eb4beginn eb_beginn4
rename eb5beginn eb_beginn5
rename eb6beginn eb_beginn6
rename eb7beginn eb_beginn7
rename eb8beginn eb_beginn8
rename eb9beginn eb_beginn9
rename eb10beginn eb_beginn10
rename eb11beginn eb_beginn11
rename eb12beginn eb_beginn12

rename eb1ende eb_ende1
rename eb2ende eb_ende2
rename eb3ende eb_ende3
rename eb4ende eb_ende4
rename eb5ende eb_ende5
rename eb6ende eb_ende6
rename eb7ende eb_ende7
rename eb8ende eb_ende8
rename eb9ende eb_ende9
```

```
rename eb10ende eb_ende10
rename eb11ende eb_ende11
rename eb12ende eb_ende12

rename eb1dauer eb_dauer1
rename eb2dauer eb_dauer2
rename eb3dauer eb_dauer3
rename eb4dauer eb_dauer4
rename eb5dauer eb_dauer5
rename eb6dauer eb_dauer6
rename eb7dauer eb_dauer7
rename eb8dauer eb_dauer8
rename eb9dauer eb_dauer9
rename eb10dauer eb_dauer10
rename eb11dauer eb_dauer11
rename eb12dauer eb_dauer12

rename eb1_typ eb_typ1
rename eb2_typ eb_typ2
rename eb3_typ eb_typ3
rename eb4_typ eb_typ4
rename eb5_typ eb_typ5
rename eb6_typ eb_typ6
rename eb7_typ eb_typ7
rename eb8_typ eb_typ8
rename eb9_typ eb_typ9
rename eb10_typ eb_typ10
rename eb11_typ eb_typ11
rename eb12_typ eb_typ12
```

In preparation of the final episode splitting (see section 4.6) we conducted a provisional stset-command in order to normalize all the different process times of our participants. Therefore, we generated a new variable measuring the general time ("gt") within our data set. Each participant's process time was normalized by subtracting the starting point of his or her occupational biography from the greatest possible process time that of the person with the longest occupational biography within the AGIPEB-dataset (296 months). Afterwards we moved the newly generated gt-variable to an appropriate position within our variable list and, then, checked the normalization procedure by using the tab-command. Later, we used this provisional variable to check for erroneous biographical data among each participant's partnership biography (see below).

```
gen gt = .
replace gt = 296-eb_beginn1
move gt eb_beginn1
tab gt if gt==.
```

Afterwards, we generated a provisional variable measuring wether a partnership event had occured. Later in the data transformation procedure, this variable was used to determine,

whether a partnership event, based on our participants' information, had occured within a certain occupational spell or not (see section 4.2).

```
gen ee = 0
move ee eb_beginn1
tab ee
```

Using this provisional event variable we conducted a provisional reshape of our dataset. Each row of the reshaped dataset now contains information on a single month of each participant's occupational biography ("ever(1)").

```
stset gt, failure(ee=1) id(id)
stsplit T0, every(1)
```

This provisional reshaping of our dataset from wide- to long-format allows for calculating the correct partnership starting points for each participant. First, we renamed the original variable containing time-constant measures of partnership duration (from "pb_dauer" to "pb_dauer_konstant"). Then, we copied the general process time, which is in fact the amount of months for each participants' duration of his or her occupational biography, into the newly created, now time-variant variable. We subtracted 12 months from the general time because we had originally asked our participants to start their reports of occupational events one year before their current relationship had begun ("gt-12"). Furthermore, we set values fot pb_dauer to missing if, due to erroneous reports of starting points of occupational biographies, the data transformation yielded negative values for partnership duration or values equal zero.

```
rename pb_dauer pb_dauer_konstant
gen pb_dauer=.
move pb_dauer eb_beginn1
replace pb_dauer=gt-12
tab pb_dauer
replace pb_dauer=. if pb_dauer <= 0
tab pb_dauer
```

To prepare for implementing time-varying data on each participant's process time we filled each row for each participant with his or her personal process time. This calculation was based on the total amount of each indivudal's process time and his or her length of occupation biography.

```
replace gt = gt+eb_beginn1-1
tab gt
```

Then, we prepared for the time-varying measures of type of employment to be implemented. Based on our theoretical considerations (Baron 2015) differentiated between five types of employment: fixed term employment ("eb_befr"), permanent employment ("eb_unbefr"), unemployed ("eb_al"), in education ("eb_bild") or other types of employment ("eb_sonst"). All the variables were coded with 0 (type of employment="no") on default.

```
gen eb_befr=0
gen eb_unbefr=0
gen eb_al=0
gen eb_bild=0
gen eb_sonst=0

move eb_befr eb_beginn1
move eb_unbefr eb_beginn1
move eb_al eb_beginn1
move eb_bild eb_beginn1
move eb_sonst eb_beginn1
```

After rearranging our newly generated variables, we changed the value for each of the five variables from 0 to 1 if a change in employment type took place in a given episode. Therefore, we used the forval-command with i indicating each spell for every employment indicator (during the interviews, a maxcimum of 12 spells per employment indicator was gathered).

```
forval i = 1/12 {            // forval = forvalues /

replace eb_befr=1 if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==1
replace eb_befr=. if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==9
replace eb_unbefr=1 if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==2
replace eb_unbefr=. if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==9
replace eb_al=1 if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==3
replace eb_al=. if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==9
replace eb_bild=1 if gt >= eb_beginn`i' & gt <= eb_ende`i' &
(eb_typ`i'==4 | eb_typ`i'==5 | eb_typ`i'==6 | eb_typ`i'==7)
replace eb_bild=. if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==9
replace eb_sonst=1 if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==8
replace eb_sonst=. if gt >= eb_beginn`i' & gt <= eb_ende`i' &
eb_typ`i'==9

}

tab eb_befr
tab eb_unbefr
tab eb_al
tab eb_bild
tab eb_sonst
```

After checking for erroneous data transformation using the tab-command the preparatory transformation of the data for episode splitting was finished.

4.2 Time-variant measure for employment status

Due to theoretical considerations further transformation of the data was conducted to find out about each participant's employment status at the beginning of the partnership. Thus, we built four new variables indication wether a participant was a fixed-term employee ("eb_befr_pbeginn"), permanent employee ("eb_unbefr_pbeginn") or in education ("eb_bild") at the start of his or her partnership. Due to low amounts of cases within the subgroups of unemployed participants as well as among other types of employment both these groups were not considered for this data transformation.

```
sort id gt

by id: gen eb_befr_pbeginn = eb_befr[13]
move eb_befr_pbeginn eb_befr

by id: gen eb_unbefr_pbeginn = eb_unbefr[13]
move eb_unbefr_pbeginn eb_unbefr

by id: gen eb_bild_pbeginn = eb_bild[13]
move eb_bild_pbeginn eb_bild
```

From these newly conducted variables we constructed an index measuring each participant's type of employment at the start of his or her partnerschip using the folloqing codes: 1 = in education, 2 = fixed-term employment and 3 = permanent employment. This newly constructed variable was named "eb_bstat_pbegin".

```
gen eb_bstat_pbeginn=.
move eb_bstat_pbeginn eb_befr_pbeginn

replace eb_bstat_pbeginn=1 if eb_bild_pbeginn==1
replace eb_bstat_pbeginn=2 if eb_befr_pbeginn==1
replace eb_bstat_pbeginn=3 if eb_unbefr_pbeginn==1

tab eb_befr_pbeginn
tab eb_unbefr_pbeginn
tab eb_bild_pbeginn
tab eb_bstat_pbeginn
```

4.3 Data transformation for piecewise-constant exponential model

With piecewise-constant exponential models it is possible to calculate flexible transition rates for each modeled time-span (Blossfeld et al. 2007: 116). In our model, partnership-time is the process time for all our participants. We built twelve variables with each of them encompassing twelve months. Only the last of our time-span variables ("pb_jahr10") encompasses more than twelve months due to this being the variable comprising right-censored cases.

Time-span variable 1: Partnership duration 1 to 12 months:

```
gen pb_jahr1=.
move pb_jahr1 eb_beginn1
replace pb_jahr1=1 if pb_dauer >= 1 & pb_dauer <= 12
replace pb_jahr1=0 if pb_dauer > 12 & pb_dauer < 1000
```

Time-span variable 2: Partnership duration 13 to 24 months:

```
gen pb_jahr2=.
move pb_jahr2 eb_beginn1
replace pb_jahr2=1 if pb_dauer >= 13 & pb_dauer <= 24
replace pb_jahr2=0 if pb_dauer < 13 | (pb_dauer > 24 & pb_dauer <
1000)
```

Time-span variable 3: Partnership duration 25 to 36 months:

```
gen pb_jahr3=.
move pb_jahr3 eb_beginn1
replace pb_jahr3=1 if pb_dauer >= 25 & pb_dauer <= 36
replace pb_jahr3=0 if pb_dauer < 25 | (pb_dauer > 36 & pb_dauer <
1000)
```

Time-span variable 4: Partnership duration 37 to 48 months:

```
gen pb_jahr4=.
move pb_jahr4 eb_beginn1
replace pb_jahr4=1 if pb_dauer >= 37 & pb_dauer <= 48
replace pb_jahr4=0 if pb_dauer < 37 | (pb_dauer > 48 & pb_dauer <
1000)
```

Time-span variable 5: Partnership duration 49 to 60 months:

```
gen pb_jahr5=.
move pb_jahr5 eb_beginn1
replace pb_jahr5=1 if pb_dauer >= 49 & pb_dauer <= 60
replace pb_jahr5=0 if pb_dauer < 49 | (pb_dauer > 60 & pb_dauer <
1000)
```

Time-span variable 6: Partnership duration 61 to 72 months:

```
gen pb_jahr6=.
move pb_jahr6 eb_beginn1
replace pb_jahr6=1 if pb_dauer >= 61 & pb_dauer <= 72
replace pb_jahr6=0 if pb_dauer < 61 | (pb_dauer > 72 & pb_dauer <
1000)
```

Time-span variable 7: Partnership duration 73 to 84 months:

```
gen pb_jahr7=.
move pb_jahr7 eb_beginn1
replace pb_jahr7=1 if pb_dauer >= 73 & pb_dauer <= 84
```

```
replace pb_jahr7=0 if pb_dauer < 73 | (pb_dauer > 84 & pb_dauer <
1000)
```

Time-span variable 8: Partnership duration 85 to 96 months:

```
gen pb_jahr8=.
move pb_jahr8 eb_beginn1
replace pb_jahr8=1 if pb_dauer >= 85 & pb_dauer <= 96
replace pb_jahr8=0 if pb_dauer < 85 | (pb_dauer > 96 & pb_dauer <
1000)
```

Time-span variable 9: Partnership duration 97 to 108 months:

```
gen pb_jahr9=.
move pb_jahr9 eb_beginn1
replace pb_jahr9=1 if pb_dauer >= 97 & pb_dauer <= 108
replace pb_jahr9=0 if pb_dauer < 97 | (pb_dauer > 108 & pb_dauer <
1000)
```

Time-span variable 10: Partnership duration 109 monaths or more:

```
gen pb_jahr10=.
move pb_jahr10 eb_beginn1
replace pb_jahr10=1 if pb_dauer >= 109 & pb_dauer <= 296
replace pb_jahr10=0 if pb_dauer < 109
```

Regarding our models, we decided to use two year-intervals for our time-span variables. So we put together each of two consecutive time-span variables to one new time span-variable ("pb2_jahr1", "pb2_jahr2" etc.). By using the move-command we placed each newly constructed variable at an appropriate position in the dataset.

Composite time-span variable 1

```
gen pb_2jahr1=0
move pb_2jahr1 pb_jahr1
replace pb_2jahr1=1 if pb_jahr1==1 | pb_jahr2==1
tab pb_jahr1 pb_jahr2
```

Composite time-span variable 2

```
gen pb_2jahr2=0
move pb_2jahr2 pb_jahr1
replace pb_2jahr2=1 if pb_jahr3==1 | pb_jahr4==1
tab pb_jahr3 pb_jahr4
```

Composite time-span variable 3

```
gen pb_2jahr3=0
move pb_2jahr3 pb_jahr1
replace pb_2jahr3=1 if pb_jahr5==1 | pb_jahr6==1
tab pb_jahr5 pb_jahr6
```

Composite time-span variable 4

```
gen pb_2jahr4=0
move pb_2jahr4 pb_jahr1
replace pb_2jahr4=1 if pb_jahr7==1 | pb_jahr8==1
tab pb_jahr7 pb_jahr8
```

Composite time-span variable 5

```
gen pb_2jahr5=.
move pb_2jahr5 pb_jahr1
replace pb_2jahr5=1 if pb_dauer >= 97 & pb_dauer <= 120
replace pb_2jahr5=0 if pb_dauer < 97 | (pb_dauer > 120 & pb_dauer <
1000)
```

Composite time-span variable 6

```
gen pb_2jahr6=.
move pb_2jahr6 pb_jahr1
replace pb_2jahr6=1 if pb_dauer >= 121 & pb_dauer <= 296
replace pb_2jahr6=0 if pb_dauer < 121
```

We constructed a control variable "pb_2jahr_pruef" to check whether all time-span variables had been constructed correctly.

```
gen
pb_2jahr_pruef=pb_2jahr1+pb_2jahr2+pb_2jahr3+pb_2jahr4+pb_2jahr5+pb_
2jahr6
tab pb_2jahr_pruef
```

In the next step we changed the variable measuring age in years to a variable measuring age in months. This allowed for implementing a time-variant measure of age into our later piecewise-constant model. After sorting our dataset according to the general process-time ("gt") we replaced the yearly by a monthly measure

```
sort id gt
gen v7_monate=.
replace v7_monate=v7*12
move v7_monate v8
```

Then, we constructed an auxiliary variable "pb_dauer_max".

```
gen pb_dauer_max=.
move pb_dauer_max v8
by id, sort: replace pb_dauer_max = pb_dauer[_N]
```

After resorting our dataset by using the by-command we calculated the tima-variant measures for age in months.

```
replace v7_monate=v7_monate-pb_dauer_max+pb_dauer
```

Then, we calculated further measures that allow for different modellings of age in our later analyses. First we constructed a time-variant measure for age in years.

```
gen v7_jahre=v7_monate/12
move v7_jahre v7_monate
```

Furthermore, we constructed a time-variant variable measuring squared age in years.

```
gen v7_jahre_q=v7_jahre*v7_jahre
move v7_jahre_q v7_monate
```

Finally, we calculated age in years at the beginning of each participant's partnership.

```
gen v7_pb_beginn=.
move v7_pb_beginn v8
replace v7_pb_beginn=(v7*12)-pb_dauer_max+1
```

For our later analyses, a dummy variable for participants' level of education was needed. The resulting variable was coded as follows: participants with A-levels were assigned the value 1 and all others the value 0.

```
gen v192_abifh=0
move v192_abifh v192
replace v192_abifh=1 if v192==4 | v192==5
replace v192_abifh=. if v192==6 | v192==9
tab v192_abifh
```

Then, we filled in Stata-readable missing values into the newly constructed dummy-variable as well as the the original variables.

```
replace v192=. if v192==8 | v192==9
replace v192_abifh=. if v192==8 | v192==9
replace v192_dich=. if v192==8 | v192==9
```

Furthermore, we recoded the scale measuring subjective precariousness from the original metrics (1 = subjective precariousness very low to 5 = subjective precariousness very high) into a more common (0 to 4)

```
gen skala_spw_alt=skala_spw
replace skala_spw_alt=. if skala_spw_alt==8 | skala_spw_alt==9
move skala_spw_alt skala_spw
replace skala_spw=skala_spw_alt-1
tab skala_spw_alt skala_spw
```

4.4 Excursion: Time-Variant Shares of Different Types of Employment

For all types of employment status we calculated shares of months spent in each type among each participant's total amount of time-spells. Using these calculated shares, we built a new time-varying variable for each type of employment status. We did not use this variable in our later model, however it could be useful for future analyses to fraw on these variables

when, for example, testing for effects of different types of labor market integration on partnership-stabilizing events (Baron 2015).

First, we built our time-variant variable for measuring cumulated shares of fixed-term employment among each participant's occupational life-course ("eb_befr_rel") with values of 0 for each spell on default.

```
gen eb_befr_rel=0
move eb_befr_rel eb_unbefr
```

We needed an auxiliary variable ("eb_befr_cum") to sort each participant's cumulated shares of fixed-term employment by using the by-command-

```
gen eb_befr_cum=0
move eb_befr_cum eb_unbefr
by id, sort: replace eb_befr_cum=sum(eb_befr)
```

Then, we constructed a second auxiliary variable that contained each participant's total amount of reported months spend in occupation ("beob_dauer").

```
gen beob_dauer=0
move beob_dauer eb_befr
by id, sort: replace beob_dauer=_n
```

Dividing each participant's cumulated shares of fixed-term employment by each participant's total amount of reported months spend in occupation yielded our time-variant shares of fixed-term employment ///
among each participant's occupational life-course. Note that we used the variable built in the first step of this excursion ("eb_befr_rel") as the target-variable for the calculation process.

```
replace eb_befr_rel=eb_befr_cum/beob_dauer
```

Finally, we constructed a variable that tells us whether a participant has ever been in fixed-term employment (eb_befr_je=1) or not (eb_befr_je=0).

```
gen eb_befr_je=0
move eb_befr_je eb_befr_rel
replace eb_befr_je = 1 if eb_befr_rel > 0
tab eb_befr_je
```

We used the same commands with differing variable names to transform the remaining variables measuring types of employment status during each participant's life-course. Thus, we continued with calculating each participant's time-variant shares of permanent employment ("eb_unbefr_rel").

```
gen eb_unbefr_rel=0
move eb_unbefr_rel eb_al
```

Auxiliary variable ("eb_unbefr_cum):

```
gen eb_unbefr_cum=0
move eb_unbefr_cum eb_al
by id, sort: replace eb_unbefr_cum=sum(eb_unbefr)
```

Final transformation:

```
replace eb_unbefr_rel=eb_unbefr_cum/beob_dauer
```

Time-variant shares of permanent unemployment ("eb_al_rel"):

```
gen eb_al_rel=0
move eb_al_rel eb_bild
```

Auxiliary variable ("eb_al_cum"):

```
gen eb_al_cum=0
move eb_al_cum eb_bild
by id, sort: replace eb_al_cum=sum(eb_al)
```

Final transformation:

```
replace eb_al_rel=eb_al_cum/beob_dauer
```

Additional variable measuring whether participants have ever been unemployed:

```
gen eb_al_je=0
move eb_al_je eb_al_rel
replace eb_al_je = 1 if eb_al_rel > 0
tab eb_al_je
```

Time-variant shares of educational spells ("eb_bild_rel"):

```
gen eb_bild_rel=0
move eb_bild_rel eb_sonst
```

Auxiliary variable ("eb_bild_cum"):

```
gen eb_bild_cum=0
move eb_bild_cum eb_sonst
by id, sort: replace eb_bild_cum=sum(eb_bild)
```

Final transformation:

```
replace eb_bild_rel=eb_bild_cum/beob_dauer
```

Time-variant shares of other types of employment ("eb_sonst_rel"):

```
gen eb_sonst_rel=0
move eb_sonst_rel pb_jahr1
```

Auxiliary variable ("eb_sonst_cum"):

```
gen eb_sonst_cum=0
move eb_sonst_cum pb_jahr1
```

```
by id, sort: replace eb_sonst_cum=sum(eb_sonst)
```

Final transformation:

```
replace eb_sonst_rel=eb_sonst_cum/beob_dauer
```

4.5 Final Episode Splitting

To facilitate the final episode-splitting, we had to construct a variable measuring the occurence (or non-occurence) of the theoretically interesting partnership-stabilizing event for each spell. As an example, we focus on the variable "pb2_fail" that measures whether participants have trinsitioned into parenthood (pb2_fail=1) or not (pb2_fail=0). Building on the replace-command, we use each participant's risktime and the event-status to allow for constructing time-variant measures of the event-status. Afterwards we deleted ("drop") all those spells where each participant's general process-time exceeded the time at risk.

```
gen pb2_fail = 0
move pb2_fail eb_beginn1
replace pb2_fail = 1 if pb_dauer==pb2_risktime & pb2_event==1
drop if pb2_event==1 & pb_dauer>pb2_risktime
tab pb2_fail if pb2_fail==1
```

Then we constructed time-variant starting- and ending-variables for each spell for each participant. The maximum amount of spells and, thus, the endpoint of each participant's occupational biography is marked by the time-variant variable pb_dauer (shich means: duration of partnership, for transformation-instructions see section 4.1). Hence, we generated a variable measuring the final date ("pb_end") that used the data of the original variable "pb_dauer".

```
gen pb_end=pb_dauer
move pb_end pb_dauer
```

We calculated the time-variant starting points for each spell by subtracting one month from each spell's ending-point.

```
gen pb_start=pb_end-1
move pb_start pb_end
```

To avoid redundance, we deleted all the remaining system-variables that were constructed during the provisional stset-procedure (see section 4.1).

```
drop _st
drop _d
drop _t
drop _t0
drop T0
```

Also, the auxiliary variables from the first reshaping-procedure (see section 4.1) were deleted.

```
drop gt
drop ee
```

A new variable "pb_zeit" was then constructed to measure general process time for each participant.

```
gen pb_zeit=pb_end-pb_start
move pb_zeit pb_dauer
tab pb_zeit
```

Then, we dropped empty spells that, as a consequence of data transformation, occurred before each participant's biography-starting date.

```
drop if pb_dauer==.
```

Finally, our episode-splitting command relied on the aforementioned event- and time-variables. For general instructions on reshaping and episode-splitting with Stata see (Cleves et al. 2010: 193-197). We sorted our data along each participant's ID and duration of life-courses.

```
stset pb_end, id(id) time0(pb_start) origin(time pb_start)
failure(pb2_fail==1)
sort id pb_dauer
```

4.6 Additional data transformation

We used a shortened scale for subjective precariousness that contained only three instead of four of the original items. The shortened scale was constructed as an additive index. For details on the scale-building as well as reliability and validity analyses see (Baron 2015). Furthermore, we used codes ranging from 0 (very low subjective precariousness) to 4 (very strong subjective precariousness), thereby replacing the original codings.

```
gen v31_alt=v31
replace v31_alt=. if v31_alt==8 | v31_alt==9
move v31_alt v31
replace v31=v31_alt-1
tab v31 v31_alt

gen v32_alt=v32
replace v32_alt=. if v32_alt==8 | v32_alt==9
move v32_alt v32
replace v32=v32_alt-1
tab v32 v32_alt

gen v33_alt=v33
replace v33_alt=. if v32_alt==8 | v32_alt==9
move v33_alt v32
replace v33=v33_alt-1
tab v33 v33_alt

gen skala_spw3=.
```

```
replace skala_spw3=(v31+v32+v33)/3
tab skala_spw3
```

For our piecewise-constant model we needed mean-centred predictors in order to provide intuitive calculations of interaction effects (Aiken, West 1991). As an example we report data transformation for the construct subjective precariousness. This variable measures feelings of uncertainty and economic restrictions associated with fixed-term employment (Baron 2015). First, we calculated the scale-mean for subjective precariousness, excluding all missing values for each predictor used in the later model.

```
mean skala_spw3 if eb_befr!=. & eb_bild!=. & eb_al!=. & eb_sonst!=.
& pb_2jahr2!=. & pb_2jahr3!=. & pb_2jahr4!=. & pb_2jahr5!=. &
pb_2jahr6!=. & v7_jahre!=. & v7_jahre_q!=. & v8!=. & v192_abifh!=. &
skala_spw3!=. & v49_code_dich!=.

tab skala_spw3 if eb_befr!=. & eb_bild!=. & eb_al!=. & eb_sonst!=. &
pb_2jahr2!=. & pb_2jahr3!=. & pb_2jahr4!=. & pb_2jahr5!=. &
pb_2jahr6!=. & v7_jahre!=. & v7_jahre_q!=. & v8!=. & v192_abifh!=. &
skala_spw3!=. & v49_code_dich!=.
```

Then, we mean-centred each participant's value for subjective precariousness and later moved the new mean-centred scale to an appropriate position in the dataset.

```
generate skala_spw3_zentr=skala_spw3-2.757336
move skala_spw3_zentr skala_spw
```

Last, we cleared the sample from spells for unemployment and other types of employment for each of the three types of employment status under consideration. We decided to so because of very low incidences of unemployment ///
spells and other types of employment spells in our sample.

```
replace eb_befr=. if eb_al==1
replace eb_unbefr=. if eb_al==1
replace eb_bild=. if eb_al==1
replace eb_befr=. if eb_sonst==1
replace eb_unbefr=. if eb_sonst==1
replace eb_bild=. if eb_sonst==1
```

4.7 Syntax for piecewise-constant exponential model

Before calculating our piecewise-constant model, we ordered Stata to provide us with sample statistics.

```
sum eb_befr eb_unbefr eb_bild pb_2jahr1 pb_2jahr2 pb_2jahr3
pb_2jahr4 pb_2jahr5 pb_2jahr6 v7 v7_jahre v8 v192_abifh if v8!=. &
v192_abifh!=.
```

We used Kaplan-Meier method to estimated survival curves for each transition. Additionally, we plotted the function by using the sts graph-command.

```
sts list if v8!=. & v192_abifh!=. & eb_befr!=. & eb_unbefr!=. &
eb_bild!=.
```

```
sts graph if v8!=. & v192_abifh!=. & eb_befr!=. & eb_unbefr!=. &
eb_bild!=.
```

We calculated our model by using the streg-command and the option "dist(exp)", expecting a (piecewise) constant hazard rate (Blossfeld et al. 2007: 116ff.). Therefore, we implemented the time-span variables ("pb_2jahr2", "pb_2jahr3" etc.) as predictors.

```
streg eb_befr eb_bild pb_2jahr2 pb_2jahr3 pb_2jahr4 pb_2jahr5
pb_2jahr6 v7_jahre v7_jahre_q  v8 v192_abifh, dist(exp)
```

As a final step we calculated separated models for each gender. First, we calculated models for women only (v8==1), then for men only (v8==0).

```
streg eb_befr eb_bild pb_2jahr2 pb_2jahr3 pb_2jahr4 pb_2jahr5
pb_2jahr6 v7_jahre v7_jahre_q v192_abifh if v8==1, dist(exp)
```

```
streg eb_befr eb_bild pb_2jahr2 pb_2jahr3 pb_2jahr4 pb_2jahr5
pb_2jahr6 v7_jahre v7_jahre_q v192_abifh if v8==0, dist(exp)
```

5.   Conclusion

In this paper we presented data transformation procedures in the course of conducting several piecewise constant-models using event history data based on the AGIPEB-project. Therefore, we conducted several reshaping- as well as recoding-procedures to facilitate episode splitting of our event history data. For our data transformation as well as later empirical analyses we used Stata IC, Version 14. Empirical results using the aforementioned event history data can be obtained from several publications, one of them still being under review at the time of finishing this working paper (Baron 2015, Baron, Rapp 2017a; b).

References

Aiken, L. S. & West, S. G. 1991: Multiple Regression. Testing and Interpreting Interactions. Newbury Park/London/New Delhi: Sage.

Baron, D. 2015: Objective vs. Subjective Precariousness and the Problem of Family Institutionalization. An Empirical Analysis of Dörres Theory of Capitalist Landgrab. AGIPEB Working Paper IV. Aachen: Institute of Sociology.

Baron, D. & Rapp, I. 2017a: Effects of Fixed-term Employment on Partnership-Stabilizing Events in Germany. Comparing Transitions into Cohabitation, Marriage, Parenthood, and Real Estate Property (under review). In Jg.,

Baron, D. & Rapp, I. 2017b: Wie wirkt sich befristete Beschäftigung auf die Institutionalisierung von Partnerschaften aus? In Hill, Paul B./Baron, Daniel (Hg.): Atypische Beschäftigung. Folgen für Partnerschaft und Familie. Wiesbaden: Springer VS (im Erscheinen),

Baron, D., Schmidt, A. & Hill, P. B. 2015: Analyse geplanter Institutionalisierungsprozesse in Paarbeziehungen unter dem Einfluss prekärer Beschäftigungsverhältnisse. Zentrale

Ergebnisse der ersten Projektphase. AGIPEB-Working Paper III, 3/2015. Aachen: Institut für Soziologie.

Blossfeld, H.-P., Golsch, K. & Rohwer, G. 2007: Event History Analysis with Stata. New York: Taylor & Friends.

Cleves, M., Gould, W. & Gutierrez, R. G. 2010: An Introduction to Survival Analysis Using Stata. College Station: Stata Press.

Eickemeier, M.-L., Hermsdorf, A., Hartmann, J., Sleik, K., Baron, D. & Hill, P. B. 2016: AGIPEB. Design und Deskription. AGIPEB Working Paper V, 12/2016. RWTH Aachen: Institut für Soziologie.