mkduration: An Easy Way to Create Duration Variables in Binary Cross-Sectional Time Series Data[*]

Andrew Q. Philips[†]

June 2, 2020

Abstract: In cross-sectional time series data with a dichotomous dependent variable (B-CSTS), failing to account for duration dependence when it exists can lead to faulty inferences. A common solution is to include duration dummies, polynomials or splines to proxy for duration dependence. Since creating these is not easy for the common practitioner, I introduce `mkduration`, a straightforward way to generate a duration variable for B-CSTS data in Stata. This command can handle various forms of missing data, and allows the duration variable to easily be turned into common parametric and non-parametric approximations.

Word Count: 92 (abstract); 4180 (manuscript)
Keywords: binary cross-sectional time series; event history; duration

[*] I would like to thank the editor and an anonymous reviewer for their thoughtful comments and suggestions. Inspiration to write this program came from students in panel data courses at CU Boulder and the IPSA-USP Summer School held in São Paulo, Brazil. Despite this, all errors and omissions are my own.

[†] andrew.philips@colorado.edu. Assistant Professor, Department of Political Science, University of Colorado Boulder, UCB 333, Boulder, CO 80309-0333.

Introduction

It is well-known that when modeling a dichotomous dependent variable in cross-sectional time series data (or B-CSTS), failing to account for duration dependence—the phenomenon by which the occurrence of an event at time t in unit i may make the reoccurrence of an event at future time point more or less likely—can have severe consequences for estimation (Beck, Katz and Tucker 1998). At best, failing to model such dependence may induce serial autocorrelation, leading to standard errors that are anticonservative. At worst, it can produce omitted variable bias even if the included regressors are unrelated to the omitted duration dependence.

A common approach recommended by Beck, Katz and Tucker (1998) when dealing with B-CSTS data—when the occurrence of events are relatively rare—is to estimate a logistic regression with duration dummies in order to proxy for any duration dependence.[1] While alternative approaches exist (c.f., Zorn 2000; Box-Steffensmeier and Jones 2004, or fitting random-effects parametric survival models using the `xtstreg` command in Stata), in general the Beck, Katz and Tucker (1998) approach is appealing due to its simplicity; as of April 2020 this article had nearly 2900 citations.[2] Despite this popularity, creating the duration variable from a dichotomous dependent variable is not straightforward. For one, such time-since-last-event variables are not as simple as techniques such as including the lag of a series. Moreover, missing data can lead to additional complications since it is unknown whether an event has occurred during this period. And, creating nonparametric approximations of duration dependence through tools such as splines is far from straightforward (c.f., Carter and Signorino 2010).

In this paper I introduce `mkduration`, an easy way to generate duration variables for

[1] In lieu of duration dummies, other recommendations include turning durations into splines or polynomial terms (Beck, Katz and Tucker 1998; Carter and Signorino 2010).

[2] The most similar existing function in Stata is `xtstreg`, although it differs substantially from the program discussed here in several ways. Both allow for grouped durations by unit (i.e., shared frailties), although the former cannot handle missing data or delayed entries into the sample, while the latter can. `mkduration` does not require that the data are `stset`, unlike `xtstreg`. While `xtstreg` uses common parametric survival distributions (e.g., exponential, Weibull), models incorporating the duration produced by `mkduration` are closest to the Cox proportional hazards model (Beck, Katz and Tucker 1998); moreover, they are typically estimated using the logit link, making them far easier to interpret.

B-CSTS data in Stata using a single command. It can also handle missing data—in effect interpolating or extrapolating—depending on what the user specifies. Moreover, it can produce several different functional forms of duration commonly used in the literature. In the sections that follow, I first discuss duration dependence in the context of B-CSTS data, then introduce the `mkduration` command. I illustrate the utility of this command through an example using data from Philips (2020).

Duration dependence with B-CSTS

Consider a simple B-CSTS dataset in long form, like the one shown in Table 1. y_{it} is a dichotomous dependent variable for unit i observed at time t that does not occur relatively often.[3] This is commonly modeled using a generalized linear model with a logistic link in order to account for the dichotomous nature of the dependent variable (Beck, Katz and Tucker 1998). The problem that arises is with duration dependence, which exists if the $\Pr(y_{it}) = 1$ changes based on how long it has been since the last event (or entry into the sample). This is shown by the duration variable in Table 1, which records the time since the last event in the data.[4]

Failing to model duration dependence implies a constant hazard rate, meaning that the probability of event re-occurrence does not change over time. In other words, events are independent from one another. In real-world data however, such an assumption is probably almost always violated. For instance, duration dependence has been argued to exist in topics as varied as conflict onsets (Clare 2010; Bapat and Zeigler 2016), pursuit of nuclear weapons (Way and Weeks 2014), and firm-level bankruptcies (Hillegeist et al. 2004). Failing to model duration dependence when it exists can lead to a number of problems. At best, the estimator will be inefficient and the standard errors will be incorrect; at worst, biased and inconsistent estimates may result, since failing to include duration

[3] One suggestion is that the event occurs with less than a 25 percent probability in a given unit-year (Beck, Katz and Tucker 1998).

[4] Table 1 is an example of left-censored data, which is a reasonable strategy if data are balanced (e.g., all units i enter at time $t = 1$; alternative strategies may be advisable if some units enter into the sample at different times (Beck, Katz and Tucker 1998), and are discussed more in the example below.

Table 1: Durations in B-CSTS data

Unit	Time	Event (y_{it})	Duration
1	1	0	1
1	2	0	2
1	3	1	3
1	4	0	1
1	5	1	2
2	1	0	1
2	2	1	2
2	3	0	1
2	4	0	2
2	5	0	3
3	1	0	1
3	2	1	2
3	3	1	1
3	4	0	1
3	5	0	2
⋮	⋮	⋮	⋮

when it exists is a form of omitted variable bias (Beck, Katz and Tucker 1998).

Beck, Katz and Tucker (1998) note that a straightforward way to continue to model B-CSTS data in the logistic framework—but also account for duration dependence—is to simply create a time-since-last-event variable (i.e., the duration variable shown in Table 1), which is then turned into a vector of dummy variables. These are then included in the logit-GLM:[5]

$$\Pr(y_{it} = 1 | \mathbf{x}_{it}, \boldsymbol{\kappa}_{it}) = \frac{1}{1 + \exp\left[-(\mathbf{x}_{it}\boldsymbol{\beta} + \boldsymbol{\kappa}_{it}\boldsymbol{\gamma})\right]} \tag{1}$$

Now, in addition to the standard covariates ($\mathbf{x}_{it}$ is a matrix of k regressors with dimensions $(N \cdot T) \times k$ with k coefficients $\boldsymbol{\beta}$), $\boldsymbol{\kappa}_{it}$ is now included, and is a matrix of duration dummies with coefficients $\boldsymbol{\gamma}$.[6] An illustration of these dummy variables is shown in Table 2. For instance, $\kappa_1 = 1$ if the duration variable is equal to one, $\kappa_2 = 1$ if the duration variable is equal to two, and so on.

[5] As long as the number of events is relatively small, using the logit link is analogous to using the complementary log-log link, the latter of which is the Cox proportional hazards model for grouped duration data (Beck, Katz and Tucker 1998).

[6] Only d dummy variables are needed, where $d = \text{Max(Duration)}$. Note that either the constant or one of the dummy variables must be dropped in order to estimate Equation 1.

Table 2: Duration dummy variables

Unit	Time	Event (y_{it})	Duration	κ_1	κ_2	κ_3
1	1	0	1	1	0	0
1	2	0	2	0	1	0
1	3	1	3	0	0	1
1	4	0	1	1	0	0
1	5	1	2	0	1	0
2	1	0	1	1	0	0
2	2	1	2	0	1	0
2	3	0	1	1	0	0
2	4	0	2	0	1	0
2	5	0	3	0	0	1
3	1	0	1	1	0	0
3	2	1	2	0	1	0
3	3	1	1	1	0	0
3	4	0	1	1	0	0
3	5	0	2	0	1	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Since it is likely that some $\boldsymbol{\kappa}_i$ may be perfectly collinear with y_{it}, separation is likely to lead to estimation issues when using maximum likelihood; this will force Stata to drop any collinear dummy variables. To alleviate this, Carter and Signorino (2010) advocate for a simple approach of incorporating duration, duration squared, and duration cubed in the model instead of either splines (another approach that Beck, Katz and Tucker (1998) recommend) or dummy variables. While some consider $\boldsymbol{\kappa}_{it}$ to be nuisance parameters (c.f., Beck 2010), others contend that it is important to discuss and interpret the estimated dependence function as a feature of theoretical interest (Carter and Signorino 2010; Williams 2016).[7] Regardless, both lines of reasoning agree that it is necessary to include some functional form of duration in the model in order to account for duration dependence.

One difficulty with implementing the advice above is that incorporating some functional form of duration dependence requires the creation of a duration variable, which is far less straightforward than taking lags or including time dummmies in standard CSTS data with a continuous dependent variable. This difficulty is compounded if some data are

[7] Moreover, the estimated duration function can also be used to check if the assumption of proportional hazards has been violated (Carter and Signorino 2010).

missing, or if units enter or leave the sample at different times. Below, I show a straightforward way to create a duration variable, even in the presence of missing data, using the command `mkduration`. The resulting variable can easily be included in the model, either through the use of dummy variables, basis functions—most commonly polynomials—or non-parametric approximations such as splines.

Accounting for dependence with `mkduration`

Syntax

```
mkduration eventvar, [dname() spline() nknots() strict force lfill rfill]
```

This command requires the specification of a single variable, `eventvar`, which is a dichotomous dependent variable where "1" indicates the presence of some event occurring at time *t* for unit *i*, and "0" indicates the absence of this event; note that `eventvar` will become the dependent variable in the logit model. The data must also first be set using `xtset`.

Options

`dname()` is an option to name the duration variable generated by mkduration. By default, the duration variable is called simply "___duration".

`spline()` is an option to create a spline to model duration dependence. The resulting spline variables to include in the model will include a "_spl1", "_spl2" (and so on) suffix. Users can choose from the following:

- `spline(linear)` creates a linear spline. In effect, a piecewise linear model is run across the duration series, which can then be included in the logistic regression model. By default, five knots are used, meaning that duration will be split into six equal segments based on percentiles of the data; the first knot will be placed at about the 16.66 percentile, the next at the 33.33rd percentile, and so on. The assumption with a linear spline is that duration has a linear effect on the probability of an event occurring between the knots, although this effect may differ across knots. Increasing the number of knots allows for a more flexible approximation of the relationship, but has the potential to overfit the data.

- `spline(cubic)` creates a restricted cubic spline that creates a linear function before the first knot, a cubic polynomial function from the second and subsequent knots, while data past the final knot is assumed linear. By default, five knots are used, and are placed along percentiles of duration, as recommended by Harrell Jr (2015), which is the default in Stata.

`nknots()` is an option—to use if `spline()` is specified—to define the number of knots to include. By default, five knots are included for both the linear and cubic splines, although this can be changed anywhere from between three to seven knots.[8] Less knots are often more efficient, but offer less flexibility in modeling duration dependence. Greater numbers of knots increase flexibility, at the cost of (potentially) decreasing efficiency and overfitting duration.

There are four additional options to account for various types of missing data. By default, the duration variable is created for all non-missing values of the event variable; any gaps in the middle of the series are handled by replacing the duration variable with missings until the next event occurs.

- `strict` takes a more stringent approach than the default at the beginning of the series (they both account for gaps in the middle of the series in the same way). Duration data may be left-censored, in that events may have occurred before the start of the sample. As such, the true underlying duration at the start of the sample is unknown, although it is quite common to ignore this and instead start the duration at $t = 1$ (i.e., the default setting). Adding `strict` will leave duration missing until the first observed event occurs, since only then is the underlying duration truly known.

[8] Users wanting more knots, or more precise placements of the knots, should instead generate the duration variable and create splines using Stata's `mkspline` command (which is the same command as that used for `spline()`, but offers greater flexibility).

- `force` is an is an option to force the creation of duration data when gaps in time are present for one or more units. Specifying force will fill in any missing gaps in duration that are preceded and succeeded by non-missing values.[9] To do this, it must assume that no event occurred during the gap. This is described in greater detail in the example below. There are two additional options that may be used with `force`:[10]
  - By default, `force` only fills in gaps in the middle of a series. By including the option `lfill` in addition to `force`, the duration variable will start when the first time variable is observed, regardless of whether the event variable is missing. As with `force`, it is assumed that no events have occurred during this period.
  - `rfill` is similar to `lfill`, but will fill in duration in all available time points after the event variable is observed. For instance, if an event variable is not observed after $t = 10$, but the dataset includes time up to $t = 15$, including both `rfill` and `force` will tell `mkduration` to fill in the duration variable all the way to $t = 15$. As with `force` and `lfill`, it is assumed that no events have occurred during this period.

Example

For an applied example I use data from Philips (2020), who examines whether state governments in India time land reforms to occur just before state elections in order to appeal to voters. Passage of legislative land reforms is a relatively rare event, occurring in just 48 of the 515 state-years under observation, meaning that these B-CSTS data may exhibit some form of duration dependence; one intuitive expectation is that passage of reform in one year makes additional land reform passage quite unlikely in the near-term.[11]

[9] In other words, `force` will not extrapolate the beginning and ends of a series; `lfill` and `rfill` are needed for this.

[10] `lfill` and/or `rfill` can only be specified in addition to `force`.

[11] Although Philips (2020) does not include duration variables in his analysis, his key findings remain unchanged from their inclusion. For brevity, I drop the state and year fixed effects that Philips includes

To start, we will create the duration variable using the dependent variable, `landref`, and then summarize it using a histogram, being sure to first `xtset` the data.

```
xtset state year
   panel variable:  state (unbalanced)
    time variable:  year, 1957 to 1991
            delta:  1 unit


mkduration landref
histogram __duration, discrete frequency
```

The histogram is shown in Figure 1. Duration is a monotonically decreasing function, with a maximum duration of 32 years, meaning that no land reform occurred during 32 years "at risk" for one of the states.

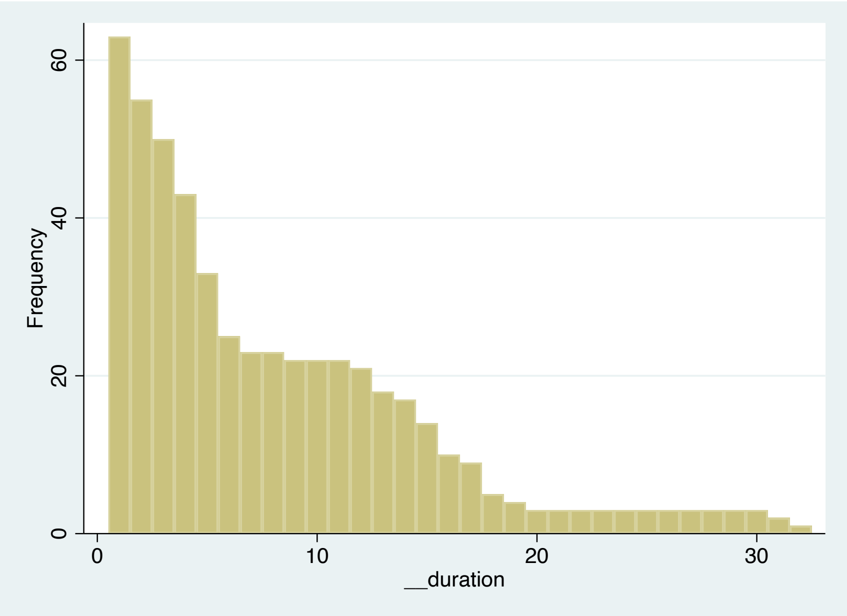

Figure 1: Histogram of `__duration`

By default the generated duration variable is called "___duration", although this can be changed using `dname()`. The duration variable can quickly be turned into dummy variables κ_{it} using Stata's categorical variable capabilities when specifying the model. In addition to the duration dummies (`i.__duration`), predictors in the random-effects

in the example below.

logit model include the following dichotomous variables: the year before an election (f1elecdum), the election year (elecdum), whether the state's government is either single-party dominant (onep_dom), a multi-party system of left, center, and right government (multp_leftcenright), a two-party system of left and center parties (twop_leftcenter) or of center and right parties (twop_centerright). There is also a continuous variable of the percentage of citizens in a state owning no land (noland).

```
xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland i.__duration
```

The results are shown in Table 3, Model 1. As is clear from the table, due to perfect collinearity (no land reform ever occurs for many of the duration-years), a large number of the duration dummies fall out of the model, reducing the number of observations. For the other covariates, it appears that land reform is more likely in the year before a state legislative election. Land reform is also more likely in multiparty competitive political systems than it is for two-party or single-party competition.

Instead of including duration dummies, we can use the recommendation of Carter and Signorino (2010) and create a cubic polynomial term of duration using Stata's interaction capabilities:

```
xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland c.__duration##c.__duration##c.__duration
```

The results using a cubic polynomial are shown in Model 2 in Table 3. None of the duration coefficients are statistically significant, which suggests they may not be needed. The results remain similar to those in Model 1, although multiparty government is no longer statistically significant, while two-party governments (specifically, one party of the left and a centrist party) are associated with an increased likelihood of land reform, although this effect is statistically significant only at the 10 percent level.

As an additional functional form choice, users can choose to model duration using splines:

Table 3: Different approaches to account for duration

	(1)	(2)	(3)	(4)	(5)
	Duration	Cubic	Cubic	Linear	No
	Dummies	Polynomial	Spline	Spline	Duration
Year before election	0.98** (0.39)	0.89** (0.36)	1.04*** (0.38)	0.98*** (0.38)	0.86** (0.36)
Election year	0.15 (0.45)	-0.01 (0.42)	0.35 (0.43)	0.20 (0.44)	0.03 (0.41)
Single-party dominant	0.31 (0.42)	0.25 (0.40)	0.34 (0.41)	0.33 (0.42)	0.43 (0.39)
Multiparty: Left-Center-Right	2.19** (0.97)	1.29 (0.87)	1.55* (0.91)	1.89** (0.93)	1.45* (0.86)
Two-party: Left-Center	0.74 (0.45)	0.75* (0.43)	0.75* (0.44)	0.73 (0.45)	0.96** (0.42)
Two-party: Center-Right	-0.08 (0.69)	-0.21 (0.67)	-0.19 (0.67)	-0.20 (0.68)	-0.43 (0.66)
Percentage owning no land	0.03 (0.02)	0.03 (0.02)	0.03 (0.02)	0.03 (0.02)	0.04* (0.02)
κ_2	-0.08 (0.65)				
κ_3	0.29 (0.62)				
κ_4	1.11* (0.58)				
κ_5	0.86 (0.64)				
κ_6	-0.34 (0.88)				
κ_8	-0.81 (1.13)				
κ_{11}	-0.79 (1.13)				
κ_{12}	0.13 (0.91)				
κ_{13}	-0.16 (1.14)				
κ_{14}	0.35 (0.92)				
κ_{16}	0.24 (1.19)				
κ_{17}	1.54 (0.95)				
Duration		-0.08 (0.21)			
Duration^2		0.00 (0.02)			
Duration^3		-0.00 (0.00)			
Spline 1			0.90*** (0.34)	-0.20 (0.62)	
Spline 2			-30.07*** (10.05)	0.72** (0.29)	
Spline 3			58.93*** (19.88)	-0.83*** (0.23)	
Spline 4			-35.20*** (12.42)	0.49** (0.20)	
Spline 5				-0.18 (0.14)	
Constant	-3.31*** (0.69)	-2.81*** (0.69)	-4.66*** (0.94)	-3.08*** (1.14)	-3.36*** (0.43)
N	389	515	515	515	515
States	15	15	15	15	15
LR-Test (vs. Model 5)		3.95	13.28**	20.91***	
χ^2	25.21***	21.09***	28.88***	33.00***	20.35***

Note: Dependent variable is equal to one if state i enacted land reform in year t, zero otherwise. LR-test results not available for Model 1 due to sample size difference. Random effects logistic regression with standard errors in parentheses. Two-tailed tests. * $p < 0.10$, ** $p < 0.05$, *** $p < 0.01$

```
* cubic spline
mkduration landref, spline(cubic) dname(dcubic)
xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland dcubic_spl*

* linear spline
mkduration landref, spline(linear) dname(dlinear) nknots(4)
xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland dlinear_spl*
```

Each command shows (respectively), a restricted cubic spline, using the default of five knots, and a piecewise linear spline with four knots (meaning that duration will be partitioned into quintiles). Note too that by specifying `dname()`, we can change the name of the resulting duration spline variables that are created. All generated spline variables have a "_spl" suffix followed by the spline number, e.g., `dlinear_spl1`, `dlinear_spl1`, and so on. The results for the cubic and linear splines are shown in Models 3 and 4 in Table 3. Most of the splines are statistically significant in both models. Last, in Model 5, a model without any form of duration is shown. Likelihood ratio tests at the bottom of Table 3 indicate that both the cubic and linear splines are preferred to the model with no duration dependence. Compared to incorporating a linear spline of duration (Model 4), the model omitting a duration function (Model 5) finds evidence that the percentage owning no land and two-party left-center governments make land reform more likely.

Given that interpreting the various approaches to duration in Table 3 is not straightforward, we can instead plot the dummy variables, splines and cubic polynomials to better understand the underlying nature of duration dependence in the data (Carter and Signorino 2010; Williams 2016). Here, we estimate each model and use `margins` to generate the predicted probability of conflict across duration, setting all other covariates to their modes or means:

```
xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland i.__duration
```

```
margins, at(__duration = (1(1)32) f1elecdum = (0) elecdum = (0) onep_dom = (1) ///
        multp_leftcenright = (0) twop_leftcenter = (0) twop_centerright = (0) ///
        noland = (13.48))
marginsplot, yline(0) title("Dummies")

xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland c.__duration##c.__duration##c.__duration
margins, at(__duration = (1(1)32) f1elecdum = (0) elecdum = (0) onep_dom = (1) ///
        multp_leftcenright = (0) twop_leftcenter = (0) twop_centerright = (0) ///
        noland = (13.48))
marginsplot, yline(0) title("Cubic Polynomial")

xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland dcubic_spl*
margins, at(f1elecdum = (0) elecdum = (0) onep_dom = (1) multp_leftcenright = (0) ///
        twop_leftcenter = (0) twop_centerright = (0) noland = (13.48)) over(__duration)
marginsplot, yline(0) title("Cubic Spline")

xtlogit landref f1elecdum elecdum onep_dom multp_leftcenright twop_leftcenter ///
        twop_centerright noland dlinear_spl*
margins, at(f1elecdum = (0) elecdum = (0) onep_dom = (1) multp_leftcenright = (0) ///
        twop_leftcenter = (0) twop_centerright = (0) noland = (13.48)) over(__duration)
marginsplot, yline(0) title("Linear Spline")
```

The resulting plot of these durations is shown in Figure 2. The estimated duration for land reform appears to be non-monotonic for all specifications except the cubic polynomial; the predicted probability of land reform increases through the first four or five years after a previous land reform, then tends to decline. For the dummy and spline durations, there appears to be another period about a dozen years after a previous land reform in which reform once again becomes more likely. After about 20 years after land reform passage, there is only a very small probability of an additional land reform. Figure 2 also shows how the inclusion of the duration dummies—especially in the context of separation—can result in "bumpy" durations; moreover, in this example, we are unable to obtain predicted probabilities beyond κ_{17} due to separation issues.

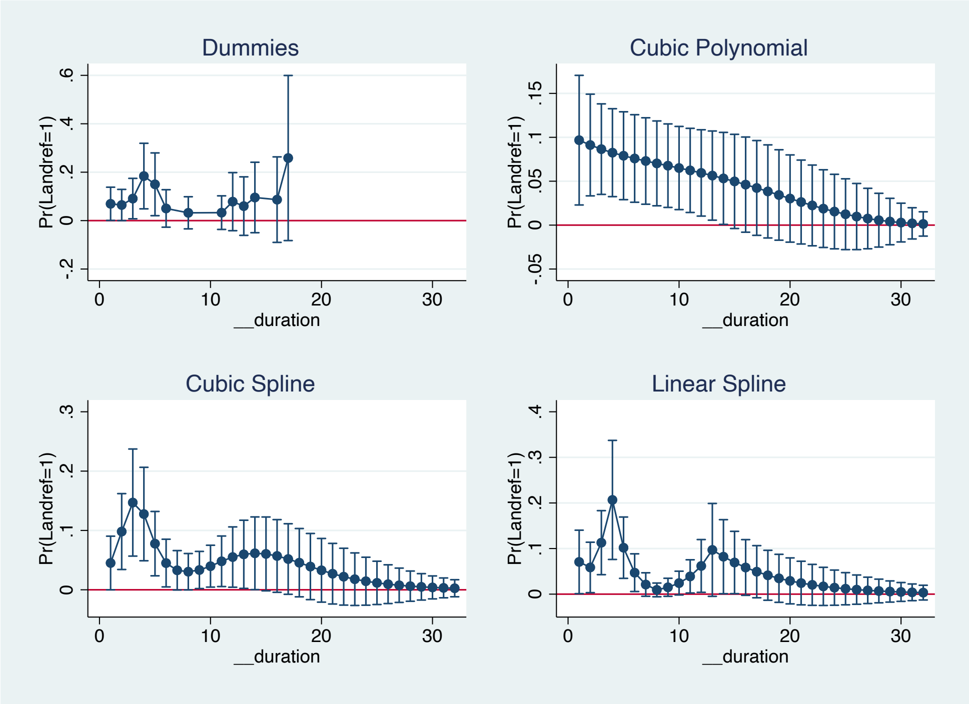

Figure 2: Different durations generated using `mkduration`

Note: Figure shows the predicted probability of land reform across duration, holding other covariates at their mean or modal value, using Models 1-4 from Table 3. 95 percent confidence intervals shown.

Missing data

One issue with duration dependence has to do with missing data. I discuss three types specific to B-CSTS data, using a stylized example shown in Table 4. First, the event variable may be missing at the beginning of the series; for instance, in Table 4, the event series is not observed for $t = 1, 2$. Second, data may be missing at the end of the series. In Table 4, data are not observed for time points $t = 17$ to $t = 20$. Third, data could be missing during the interval in which the series is observed; the event in Table 4 is not observed for the interval $t = 7, 8$, although prior and future values *are* observed. `mkduration` has several different options for handling 'left' (missing at the beginning of the series), 'interval' (missing in the middle of the series), and 'right' (missing at the end of the series) forms of missing data.

Table 4: `mkduration` and approaches to missing data

Unit	Time	Event	Default	`strict`	`force`	`force`	`force`	`force`, `lfill`
						& `lfill`	& `rfill`	& `rfill`
1	1	.	.	.	.	1	.	1
1	2	.	.	.	.	2	.	2
1	3	0	1	.	1	3	1	3
1	4	0	2	.	2	4	2	4
1	5	1	3	.	3	5	3	5
1	6	0	1	1	1	1	1	1
1	7	.	2	2	2	2	2	2
1	8	.	.	.	3	3	3	3
1	9	1	.	.	4	4	4	4
1	10	0	1	1	1	1	1	1
1	11	0	2	2	2	2	2	2
1	12	0	3	3	3	3	3	3
1	13	1	4	4	4	4	4	4
1	14	0	1	1	1	1	1	1
1	15	1	2	2	2	2	2	2
1	16	0	1	1	1	1	1	1
1	17	.	2	2	2	2	2	2
1	18	.	.	.	.	.	3	3
1	19	.	.	.	.	.	4	4
1	20	.	.	.	.	.	5	5

By default, `mkduration` will start the duration at the first non-missing event variable, and create a missing duration value for any instances between the missing values until after the next observed event. This is shown by the "Default" column. Note too that by default, the duration variable will revert to missing one period after the last observed value.[12]

A stricter interpretation might lead to us to replace duration with missings until the first event is actually observed, since events may have occurred before the start of the sample (i.e., left-censoring). Using the `strict` option will not start the duration variable until after the first event has been observed, For instance, since it is unknown how long it has been since the last event for non-missing values at $t = 3$ through $t = 5$ in Table 4, these are coded as missing in the `strict` column.

[12] In Table 4, duration is equal to two at $t = 17$ since is has been two time points after the last event, but is coded as missing at $t = 18$ since it is not known whether the event variable was zero or one at $t = 17$.

If the user is comfortable assuming that no events have occurred during the unobserved middle time period $t = 7, 8$, they can use the `force` option to fill in observed periods that contain missing values. As shown in Table 4, including this option will fill in the duration variable for time points $t = 8, 9$ (i.e., up until the next event occurs).

In addition to using the `force` option, two other options may be used. Adding `lfill` will start the duration at the first time period, not the first observed value of the event variable. As shown in Table 4, this will start the duration at $t = 1$, even though the event variable is not observed until $t = 3$. As with `force`, it is assumed that no event has occurred during this time. Similar to `lfill`, the option `rfill` can be used to continue the duration series after the last observed event variable. In Table 4, this means that values $t = 18$ to $t = 20$ are filled in, even though the las observed event variable is at $t = 16$. Like `force` and `lfill`, it is assumed that no events are occurring during this time. Last, users can use the `force`, `lfill` and `rfill` options together to fill in left, interval, and right forms of missingness.

Conclusion

In this paper, I have introduced `mkduration`, a simple, less error-prone way to create a duration variable in B-CSTS data when duration dependence is suspected. Replicating an example application that uses B-CSTS data, I have shown that this command allows users to easily account for duration dependence in whichever manner they choose, such as dummies, splines, or polynomials. Moreover, depending on the additional assumptions users are willing to make, `mkduration` can easily account for dependence in the context of missing data in a number of different ways.

References

Bapat, Navin A and Sean Zeigler. 2016. "Terrorism, dynamic commitment problems, and military conflict." *American Journal of Political Science* 60(2):337–351.

Beck, Nathaniel. 2010. "Time is not a theoretical variable." *Political Analysis* 18(3):293–294.

Beck, Nathaniel, Jonathan N Katz and Richard Tucker. 1998. "Taking time seriously: Time-series-cross-section analysis with a binary dependent variable." *American Journal of Political Science* pp. 1260–1288.

Box-Steffensmeier, Janet M and Bradford S Jones. 2004. *Event history modeling: A guide for social scientists.* Cambridge University Press.

Carter, David B and Curtis S Signorino. 2010. "Back to the future: Modeling time dependence in binary data." *Political Analysis* 18(3):271–292.

Clare, Joe. 2010. "Ideological fractionalization and the international conflict behavior of parliamentary democracies." *International Studies Quarterly* 54(4):965–987.

Harrell Jr, Frank E. 2015. *Regression modeling strategies: With applications to linear models, logistic and ordinal regression, and survival analysis.* Springer.

Hillegeist, Stephen A, Elizabeth K Keating, Donald P Cram and Kyle G Lundstedt. 2004. "Assessing the probability of bankruptcy." *Review of Accounting Studies* 9(1):5–34.

Philips, Andrew Q. 2020. "Just in time: Political policy cycles of land reform." *Politics* 40(2):207–226.

Way, Christopher and Jessica LP Weeks. 2014. "Making it personal: Regime type and nuclear proliferation." *American Journal of Political Science* 58(3):705–719.

Williams, Laron K. 2016. "Long-term effects in models with temporal dependence." *Political Analysis* 24(2):243–262.

Zorn, Christopher JW. 2000. "Modeling duration dependence." *Political Analysis* 8(4):367–380.