Modeling Relational Event Dynamics with statnet

Carter T. Butts

Contents

Last updated 2022-06-24

This tutorial is a joint product of the Statnet Development Team:

Pavel N. Krivitsky (University of New South Wales)
Martina Morris (University of Washington)
Mark S. Handcock (University of California, Los Angeles)
Carter T. Butts (University of California, Irvine)
David R. Hunter (Penn State University)
Steven M. Goodreau (University of Washington)
Chad Klumb (University of Washington)
Skye Bender de-Moll (Oakland, CA)

The network modeling software demonstrated in this tutorial is authored by Carter Butts (`relevent`, `sna`).

The `statnet` Project

All `statnet` packages are open-source, written for the **R** computing environment, and published on CRAN. The source repositories are hosted on GitHub. Our website is statnet.org

- Need help? For general questions and comments, please email the statnet users group at statnet_help@uw.edu. You'll need to join the listserv if you're not already a member. You can do that here: statnet_help listserve.
- Found a bug in our software? Please let us know by filing an issue in the appropriate package GitHub repository, with a reproducible example.
- Want to request new functionality? We welcome suggestions – you can make a request by filing an issue on the appropriate package GitHub repository. The chances that this functionality will be developed are substantially improved if the requests are accompanied by some proposed code (we are happy to review pull requests).
- For all other issues, please email us at contact@statnet.org.

Section 0. Introduction to the Tutorial

This workshop and tutorial provide an introduction to statistical modeling of relational event data using `statnet` software. This online tutorial is also designed for self-study, with example code and self-contained data. The `statnet` package we will be demonstrating is:

- `relevent` – modeling and simulation for relational event models

Additional background on the tools, modeling framework, and data used in this tutorial may be found in the references at the bottom of this document.

0.0 Prerequisites

This workshop assumes basic familiarity with **R**, experience with network concepts, terminology and data, and familiarity with the general framework for statistical modeling and inference. While previous experience with relational event models (REMs) is not required, some of the topics covered here may be difficult to understand without a strong background in linear and generalized linear models in statistics.

0.1 Software installation

Minimally, you will need to install the latest version of **R** (available here) and the `statnet` packages `relevent`, `sna` and `network` to run the code presented here (`sna` will automatically install `network` when it is installed).

The full set of installation instructions with details can be found on the `statnet` workshop wiki.

If you have not already downloaded the `statnet` packages for this workshop, the quickest way to install these (and the other most commonly used packages from the `statnet` suite), is to open an R session and type:

```
install.packages(c("relevent","sna"))
```

```
library(relevent)
```

```
Loading required package: trust

Loading required package: sna

Loading required package: statnet.common

Attaching package: 'statnet.common'

The following objects are masked from 'package:base':

    attr, order

Loading required package: network

'network' 1.17.2 (2022-05-19), part of the Statnet Project
* 'news(package="network")' for changes since last version
* 'citation("network")' for citation information
* 'https://statnet.org' for help, support, and other information

sna: Tools for Social Network Analysis
Version 2.6 created on 2020-10-5.
copyright (c) 2005, Carter T. Butts, University of California-Irvine
 For citation information, type citation("sna").
 Type help(package="sna") to get started.

Loading required package: coda

relevent: Relational Event Models
Version 1.1 created on 2021-09-20.
copyright (c) 2007, Carter T. Butts, University of California-
```

```
Irvine
 For citation information, type citation("relevent").
 Type help(package="relevent") to get started.
```

```r
library(sna)
```

You can check the version number with:

```r
packageVersion("relevent")
```

```
[1] '1.1'
```

Throughout, we will set a random seed via `set.seed()` for commands in tutorial that require simulating random values—this is not necessary, but it ensures that you will get the same results as the online tutorial.

Section 1. Dyadic Relational Event Models with `rem.dyad`: Ordinal Timing

Dyadic relational event models are intended to capture the behavior of systems in which individual social units (persons, organizations, animals, etc.) direct discrete actions towards other individuals in their environment. Within the `relevent` package, the `rem.dyad` function is the primary workhorse for modeling dyadic data. Although less flexible than `rem` (another `relevent` tool, not covered in this tutorial), `rem.dyad` contains many features that make it easier to work with in the dyadic case.

Data for use with `rem.dyad` consists of dynamic edge lists, each edge being characterized by a sender, a recipient, and an event time. (Currently, self-edges and undirected edges are not supported – this will change in future versions!) Ideally, event times are known exactly; however, under the piecewise constant hazard assumption (per Butts, 2008) the relational event family can still be identified up to a pacing constant so long as the order of events is known. Since the case of ordinal timing is somewhat simpler than that of exact timing, we consider it first.

1.0 Loading the `relevent` package, and the workshop data

```r
library(relevent)  #Load the relevent library
load("relevent_workshop.Rdata")  #Load the workshop data - may need to change directory!
```

1.1 Getting a look at the WTC Police radio data

The data we will use here comes from the World Trade Center radio communication data set coded by Butts et al. (2007). It consists of radio calls among 37 named communicants belonging to a police unit at the World Trade Center complex on the morning of 9/11/2001. The edgelist is contained in an object called `WTCPoliceCalls`; printing it should yield output like the following:

```r
WTCPoliceCalls
```

```
    number source recipient
1        1     16        32
2        2     32        16
3        3     16        32
4        4     16        32
5        5     11        32
6        6     11        32
7        7     11        32
8        8     36        32
9        9      8        32
10      10      8        32
11      11     32         8
12      12     16        32
```

13	13	8	32
14	14	26	32
15	15	32	26
16	16	26	32
17	17	32	26
18	18	26	32
19	19	32	26
20	20	16	32
21	21	16	32
22	22	27	32
23	23	20	32
24	24	32	20
25	25	20	32
26	26	32	20
27	27	32	16
28	28	16	32
29	29	32	16
30	30	32	16
31	31	16	32
32	32	32	22
33	33	3	32
34	34	32	3
35	35	3	32
36	36	32	3
37	37	32	16
38	38	16	32
39	39	32	16
40	40	3	32
41	41	3	32
42	42	32	3
43	43	3	32
44	44	16	3
45	45	16	11
46	46	11	16
47	47	16	11
48	48	11	16
49	49	16	11
50	50	11	16
51	51	24	36
52	52	24	36
53	53	15	32
54	54	32	15
55	55	15	32
56	56	32	15
57	57	15	32
58	58	32	15
59	59	22	32
60	60	32	22
61	61	15	32
62	62	32	15
63	63	15	32
64	64	32	15
65	65	18	32
66	66	32	18

67	67	18	32
68	68	19	32
69	69	32	19
70	70	19	32
71	71	32	19
72	72	19	32
73	73	16	32
74	74	32	16
75	75	16	32
76	76	32	16
77	77	36	16
78	78	16	36
79	79	36	16
80	80	16	36
81	81	36	16
82	82	16	36
83	83	27	32
84	84	32	16
85	85	16	32
86	86	32	16
87	87	16	32
88	88	32	16
89	89	22	15
90	90	15	22
91	91	22	15
92	92	15	22
93	93	22	15
94	94	16	22
95	95	22	16
96	96	16	22
97	97	22	11
98	98	11	22
99	99	36	32
100	100	32	36
101	101	36	32
102	102	32	36
103	103	36	32
104	104	32	36
105	105	27	32
106	106	37	32
107	107	32	37
108	108	37	32
109	109	32	37
110	110	5	32
111	111	32	5
112	112	5	32
113	113	32	5
114	114	31	36
115	115	36	31
116	116	31	36
117	117	36	31
118	118	37	32
119	119	16	32
120	120	32	16

121	121	16	32
122	122	32	16
123	123	29	32
124	124	32	29
125	125	37	14
126	126	29	32
127	127	31	32
128	128	32	37
129	129	16	32
130	130	32	16
131	131	16	32
132	132	32	16
133	133	16	32
134	134	36	16
135	135	16	36
136	136	36	16
137	137	16	36
138	138	29	32
139	139	8	35
140	140	32	16
141	141	8	35
142	142	32	16
143	143	16	32
144	144	32	16
145	145	16	32
146	146	22	32
147	147	32	22
148	148	22	32
149	149	32	22
150	150	27	32
151	151	32	27
152	152	27	32
153	153	32	26
154	154	22	32
155	155	32	22
156	156	22	32
157	157	32	22
158	158	22	32
159	159	32	22
160	160	22	32
161	161	32	22
162	162	16	32
163	163	32	16
164	164	16	32
165	165	32	16
166	166	16	32
167	167	16	11
168	168	27	32
169	169	32	16
170	170	16	32
171	171	32	16
172	172	36	32
173	173	32	36
174	174	36	32

175	175	32	36
176	176	16	32
177	177	32	16
178	178	16	32
179	179	32	16
180	180	16	32
181	181	32	16
182	182	16	32
183	183	10	2
184	184	2	10
185	185	10	26
186	186	16	32
187	187	32	16
188	188	16	32
189	189	16	32
190	190	32	16
191	191	32	16
192	192	16	32
193	193	32	16
194	194	16	32
195	195	32	16
196	196	16	32
197	197	32	16
198	198	16	32
199	199	32	16
200	200	16	32
201	201	32	16
202	202	22	32
203	203	32	22
204	204	24	32
205	205	32	24
206	206	24	32
207	207	32	24
208	208	16	32
209	209	32	16
210	210	16	32
211	211	32	24
212	212	24	32
213	213	16	32
214	214	30	16
215	215	16	30
216	216	30	16
217	217	16	30
218	218	30	16
219	219	16	30
220	220	32	15
221	221	15	32
222	222	32	15
223	223	15	32
224	224	32	15
225	225	32	15
226	226	15	32
227	227	32	15
228	228	15	32

229	229	32	23
230	230	23	32
231	231	32	23
232	232	23	32
233	233	32	23
234	234	23	32
235	235	32	23
236	236	23	32
237	237	32	23
238	238	23	32
239	239	32	19
240	240	19	32
241	241	32	19
242	242	19	32
243	243	32	18
244	244	15	16
245	245	32	18
246	246	16	32
247	247	32	16
248	248	16	32
249	249	32	16
250	250	15	16
251	251	16	15
252	252	15	16
253	253	16	15
254	254	15	16
255	255	16	15
256	256	25	32
257	257	32	25
258	258	25	32
259	259	32	25
260	260	1	4
261	261	4	1
262	262	1	4
263	263	4	1
264	264	1	4
265	265	4	1
266	266	1	4
267	267	4	1
268	268	1	4
269	269	16	32
270	270	32	16
271	271	16	32
272	272	32	16
273	273	16	32
274	274	32	16
275	275	16	32
276	276	18	32
277	277	32	18
278	278	18	32
279	279	32	18
280	280	18	32
281	281	32	18
282	282	18	32

283	283	32	18
284	284	18	32
285	285	32	18
286	286	18	32
287	287	32	18
288	288	18	32
289	289	25	32
290	290	32	16
291	291	16	32
292	292	32	16
293	293	16	32
294	294	32	16
295	295	16	32
296	296	32	16
297	297	16	32
298	298	32	16
299	299	16	32
300	300	32	16
301	301	16	32
302	302	32	16
303	303	22	32
304	304	32	22
305	305	22	32
306	306	25	32
307	307	32	25
308	308	25	32
309	309	22	32
310	310	32	22
311	311	22	32
312	312	32	16
313	313	25	32
314	314	32	25
315	315	25	32
316	316	32	25
317	317	21	32
318	318	32	21
319	319	21	32
320	320	32	21
321	321	21	32
322	322	32	21
323	323	21	32
324	324	25	32
325	325	32	25
326	326	16	36
327	327	36	16
328	328	36	16
329	329	16	36
330	330	36	16
331	331	16	36
332	332	32	16
333	333	16	32
334	334	31	32
335	335	32	31
336	336	31	32

337	337	32	31
338	338	31	32
339	339	32	31
340	340	32	16
341	341	16	32
342	342	32	16
343	343	16	32
344	344	30	32
345	345	32	30
346	346	30	32
347	347	9	32
348	348	6	32
349	349	22	32
350	350	32	22
351	351	22	32
352	352	32	22
353	353	34	32
354	354	32	34
355	355	34	32
356	356	32	34
357	357	32	22
358	358	22	32
359	359	21	36
360	360	16	21
361	361	16	32
362	362	32	16
363	363	16	32
364	364	32	16
365	365	16	32
366	366	32	22
367	367	22	32
368	368	32	22
369	369	22	32
370	370	33	32
371	371	33	32
372	372	32	16
373	373	32	33
374	374	16	32
375	375	32	16
376	376	16	32
377	377	32	33
378	378	33	32
379	379	16	15
380	380	15	16
381	381	16	15
382	382	15	16
383	383	32	16
384	384	16	32
385	385	17	32
386	386	32	17
387	387	16	17
388	388	21	36
389	389	36	21
390	390	21	36

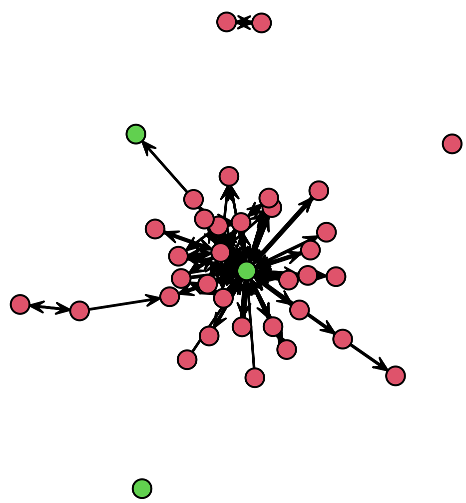
391	391	36	21
392	392	21	36
393	393	36	21
394	394	21	36
395	395	32	16
396	396	16	32
397	397	32	16
398	398	16	32
399	399	16	32
400	400	32	16
401	401	32	16
402	402	16	32
403	403	32	16
404	404	16	32
405	405	32	16
406	406	24	16
407	407	16	24
408	408	24	16
409	409	16	24
410	410	25	32
411	411	32	16
412	412	16	32
413	413	32	16
414	414	16	32
415	415	32	16
416	416	21	32
417	417	32	21
418	418	21	32
419	419	21	30
420	420	32	16
421	421	16	32
422	422	32	16
423	423	16	32
424	424	32	21
425	425	21	32
426	426	32	21
427	427	21	36
428	428	36	21
429	429	21	36
430	430	36	21
431	431	21	36
432	432	36	21
433	433	21	36
434	434	30	32
435	435	32	30
436	436	30	32
437	437	32	30
438	438	30	32
439	439	16	32
440	440	32	16
441	441	16	32
442	442	32	16
443	443	24	16
444	444	16	24

```
445    445    24     16
446    446    16     24
447    447    24     16
448    448    16     24
449    449    34     32
450    450    32     34
451    451    34     32
452    452    12     34
453    453    16     15
454    454    16     32
455    455    12     32
456    456    32     12
457    457    12     32
458    458    32     12
459    459    32     34
460    460    34     32
461    461    29     32
462    462    32     29
463    463    29     32
464    464    32     29
465    465    29     32
466    466    32     29
467    467    32     16
468    468    16     32
469    469    32     16
470    470    16     32
471    471    32     16
472    472    16     32
473    473    28     16
474    474    16     28
475    475    28     16
476    476    28     16
477    477    16     28
478    478    28     16
479    479    15     16
480    480    32     16
481    481    16     32
```

Note the form of the data: a matrix with the timing information, source (numbered from 1 to 37), and recipient (again numbered from 1 to 37) for each event (i.e., radio call). It is important to note that the WTC radio data was coded from transcripts that lacked detailed timing information; we do not therefore know precisely when these calls were made. We do, however, know the order in which calls were made, and can use this to fit relational event models with `rem.dyad`.

Before analyzing the data, it is helpful to consider what it looks like in time aggregated form. The workshop-supplied helper function `as.sociomatrix.eventlist` is useful for this purpose: it converts an event list into a valued sociomatrix, of the form used by other `statnet` routines. Let's convert the data to sociomatrix form, and visualize it using the `gplot` function of the `sna` package:

```r
WTCPoliceNet <- as.sociomatrix.eventlist(WTCPoliceCalls, 37)
gplot(WTCPoliceNet, edge.lwd = WTCPoliceNet^0.75, vertex.col = 2 +
   WTCPoliceIsICR, vertex.cex = 1.25)
```



In this visualization, we have scaled edge widths by communication volume – clearly, some pairs interact much more than others. Note also that we have colored vertices based on whether or not they occupy an institutionalized coordinative role (ICR), as indicated by the vector `WTCPoliceIsICR`. Those for whom this vector is TRUE (green) occupy roles within the police organization that would be expected to participate in coordinative activities; other actors were not identified as occupying such roles, based on the transcript data. In the analyses below, we will employ this covariate (as well as various endogenous mechanisms) to model the dynamics of radio communication within the WTC police network.

1.2 A first model: exploring ICR effects

Let's begin by fitting a very simple covariate model, in which the propensity of individuals to send and receive calls depends on whether they occupy institutionalized coordinative roles:

```
# First ICR effect - total interaction
wtcfit1 <- rem.dyad(WTCPoliceCalls, n = 37, effects = c("CovInt"),
   covar = list(CovInt = WTCPoliceIsICR), hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(wtcfit1)
```

```
Relational Event Model (Ordinal Likelihood)

         Estimate  Std.Err Z value  Pr(>|z|)
```

```
CovInt.1 2.104464 0.069817  30.142 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 6921.048 on 481 degrees of freedom
Residual deviance: 6193.998 on 480 degrees of freedom
    Chi-square: 727.0499 on 1 degrees of freedom, asymptotic p-value 0
AIC: 6195.998 AICC: 6196.007 BIC: 6200.174
```

The output gives us the covariate effect, as well as some uncertainty and goodness-of-fit information. The format is much like the output for a regression model, but coefficients should be interpreted per the relational event framework. In particular, the ICR role coefficient is the logged multiplier for the hazard of an event involving an ICR, versus a non-ICR event. (The effect is cumulative: an event in which one actor in an ICR calls another actor in an ICR gets twice the log increment.) We can see this impact in real terms as follows:

```
exp(wtcfit1$coef)  #Relative hazard for a non-ICR/ICR vs. a non-ICR/non-ICR event
```

```
CovInt.1
8.202706
```

```
exp(2 * wtcfit1$coef)  #Relative hazard for an ICR/ICR vs. a non-ICR/non-ICR event
```

```
CovInt.1
67.28438
```

We have here considered a homogeneous effect of ICR status on sending and receiving; is it worth treating these effects separately? To do so, we enter the ICR covariate as a sender and receiver covariate (respectively):

```
wtcfit2 <- rem.dyad(WTCPoliceCalls, n = 37, effects = c("CovSnd",
   "CovRec"), covar = list(CovSnd = WTCPoliceIsICR, CovRec = WTCPoliceIsICR),
   hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(wtcfit2)
```

```
Relational Event Model (Ordinal Likelihood)

         Estimate  Std.Err Z value  Pr(>|z|)
CovSnd.1 1.979172 0.095745  20.671 < 2.2e-16 ***
CovRec.1 2.225716 0.092862  23.968 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 6921.048 on 481 degrees of freedom
Residual deviance: 6190.175 on 479 degrees of freedom
    Chi-square: 730.8731 on 2 degrees of freedom, asymptotic p-value 0
AIC: 6194.175 AICC: 6194.2 BIC: 6202.527
```

Does the effect seem to differ? Let's see if fit improves (using the BIC):

```
wtcfit1$BIC - wtcfit2$BIC  #Model 1 a bit lower - we prefer it
```

```
[1] -2.352663
```

Model selection criteria are the preferred way to compare models, but one can also use a test of equality on the coefficients:

```r
wtcfit2$coef  #Extract the coefficients
```

```
CovSnd.1 CovRec.1
1.979172 2.225716
```

```r
wtcfit2$cov  #Likewise, the posterior covariance matrix
```

```
            [,1]        [,2]
[1,] 0.009167101 0.000900543
[2,] 0.000900543 0.008623350
```

```r
# Heuristic Wald test of equality (not Bayesian, but
# whatever)
z <- diff(wtcfit2$coef)/sqrt(sum(diag(wtcfit2$cov)) - 2 * wtcfit2$cov[1,
   2])
z
```

```
CovRec.1
1.949747
```

```r
2 * (1 - pnorm(abs(z)))  #Not conventionally significant - not strongly detectable
```

```
  CovRec.1
0.05120631
```

There might be some difference between the ICR sender and receiver effects, but it doesn't seem large enough to worry about. For now, we'll just stick with the simpler model (with a uniform effect on total interaction).

1.3 Bringing in endogenous social dynamics

One of the attractions of the relational event framework is its ability to capture endogenous social dynamics. In the following examples, we will examine several kinds of mechanisms that could conceivably impact communication among participants in the WTC police network. In each case, we first fit a candidate model, then compare that model to our best fitting model thus far identified. Where effects result in an improvement (as judged by the BIC), we include them in subsequent models.

To begin, we note that this is radio communication data. Radio communication is governed by strong conversational norms (in particular, radio SOP), which among other things mandate systematic turn-taking reciprocity. We can test for this via the use of participation shifts, particularly the AB-BA shift (a tendency for B to call A, given that A has just called B).

```r
wtcfit3 <- rem.dyad(WTCPoliceCalls, n = 37, effects = c("CovInt",
   "PSAB-BA"), covar = list(CovInt = WTCPoliceIsICR), hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```r
summary(wtcfit3)  #Looks like a strong effect...
```

```
Relational Event Model (Ordinal Likelihood)

         Estimate Std.Err Z value  Pr(>|z|)
CovInt.1  1.60405 0.11500  13.949 < 2.2e-16 ***
PSAB-BA   7.32695 0.10552  69.436 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Null deviance: 6921.048 on 481 degrees of freedom
Residual deviance: 2619.115 on 479 degrees of freedom
    Chi-square: 4301.933 on 2 degrees of freedom, asymptotic p-value 0
AIC: 2623.115 AICC: 2623.14 BIC: 2631.467
```

```
wtcfit1$BIC - wtcfit3$BIC  #We prefer model 3 to model 1 - reciprocity is in!
```

```
[1] 3568.707
```

```
exp(wtcfit3$coef["PSAB-BA"])  #Reciprocating events are >1500 times as likely
```

```
PSAB-BA
1520.73
```

What about other conversational norms? In general, we may expect that the current participants in an interaction may be likely to initiate the next call, a tendency that can also be captured with P-shift effects.

```
wtcfit4 <- rem.dyad(WTCPoliceCalls, n = 37, effects = c("CovInt",
   "PSAB-BA", "PSAB-BY", "PSAB-AY"), covar = list(CovInt = WTCPoliceIsICR),
   hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(wtcfit4)  #Seems like the effects are present, but let's test GOF...
```

```
Relational Event Model (Ordinal Likelihood)

         Estimate Std.Err Z value  Pr(>|z|)
CovInt.1  1.54283 0.11818 13.0549 < 2.2e-16 ***
PSAB-BA   7.49955 0.11418 65.6831 < 2.2e-16 ***
PSAB-BY   1.25941 0.25131  5.0115 5.402e-07 ***
PSAB-AY   0.87215 0.30612  2.8491  0.004384 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 6921.048 on 481 degrees of freedom
Residual deviance: 2595.135 on 477 degrees of freedom
    Chi-square: 4325.913 on 4 degrees of freedom, asymptotic p-value 0
AIC: 2603.135 AICC: 2603.219 BIC: 2619.839
```

```
wtcfit3$BIC - wtcfit4$BIC  #Yes, definite improvement
```

```
[1] 11.62806
```

P-shift effects are "local," in that they depend only on the prior event. What about effects of recency (from the point of view of ego) on the tendency to send calls to others?

```
wtcfit5 <- rem.dyad(WTCPoliceCalls, n = 37, effects = c("CovInt",
   "PSAB-BA", "PSAB-BY", "PSAB-AY", "RRecSnd", "RSndSnd"), covar = list(CovInt = WTCPoliceIsICR),
   hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(wtcfit5)  #Looks good; note that AB-BA is much smaller than before
```

```
Relational Event Model (Ordinal Likelihood)

         Estimate Std.Err Z value  Pr(>|z|)
RRecSnd   2.38496 0.27447  8.6892 < 2.2e-16 ***
RSndSnd   1.34623 0.22307  6.0350 1.590e-09 ***
CovInt.1  1.07058 0.14244  7.5160 5.640e-14 ***
PSAB-BA   4.88714 0.15293 31.9569 < 2.2e-16 ***
PSAB-BY   1.67938 0.26116  6.4304 1.273e-10 ***
PSAB-AY   1.39017 0.31057  4.4762 7.598e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 6921.048 on 481 degrees of freedom
Residual deviance: 2307.413 on 475 degrees of freedom
    Chi-square: 4613.635 on 6 degrees of freedom, asymptotic p-value 0
AIC: 2319.413 AICC: 2319.591 BIC: 2344.469
```

```
wtcfit4$BIC - wtcfit5$BIC  #Substantial improvement
```

```
[1] 275.3701
```

Finally, recall what our relational event data looked like when viewed in time-aggregated form. We observed a strongly hub-dominated network, with a few actors doing most of the communication. Could this be explained in part via a preferential attachment mechanism (per de Sola Price and others), in which those having the most air time become the most attractive targets for others to call? We can investigate this by including normalized total degree as a predictor of tendency to receive calls:

```
wtcfit6 <- rem.dyad(WTCPoliceCalls, n = 37, effects = c("CovInt",
   "PSAB-BA", "PSAB-BY", "PSAB-AY", "RRecSnd", "RSndSnd", "NTDegRec"),
   covar = list(CovInt = WTCPoliceIsICR), hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(wtcfit6)  #PA is drawing from recency, ICR effect, but not P-shifts
```

```
Relational Event Model (Ordinal Likelihood)

         Estimate Std.Err Z value  Pr(>|z|)
NTDegRec  3.13454 0.56678  5.5305 3.194e-08 ***
RRecSnd   2.02903 0.28500  7.1194 1.084e-12 ***
RSndSnd   0.87115 0.23846  3.6533 0.0002589 ***
CovInt.1  0.70734 0.16400  4.3129 1.611e-05 ***
PSAB-BA   5.32576 0.18236 29.2042 < 2.2e-16 ***
PSAB-BY   1.86024 0.26321  7.0674 1.579e-12 ***
PSAB-AY   1.64806 0.31092  5.3005 1.155e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 6921.048 on 481 degrees of freedom
Residual deviance: 2276.263 on 474 degrees of freedom
    Chi-square: 4644.785 on 7 degrees of freedom, asymptotic p-value 0
AIC: 2290.263 AICC: 2290.5 BIC: 2319.494
```

```
wtcfit5$BIC - wtcfit6$BIC  #Model is preferred
```

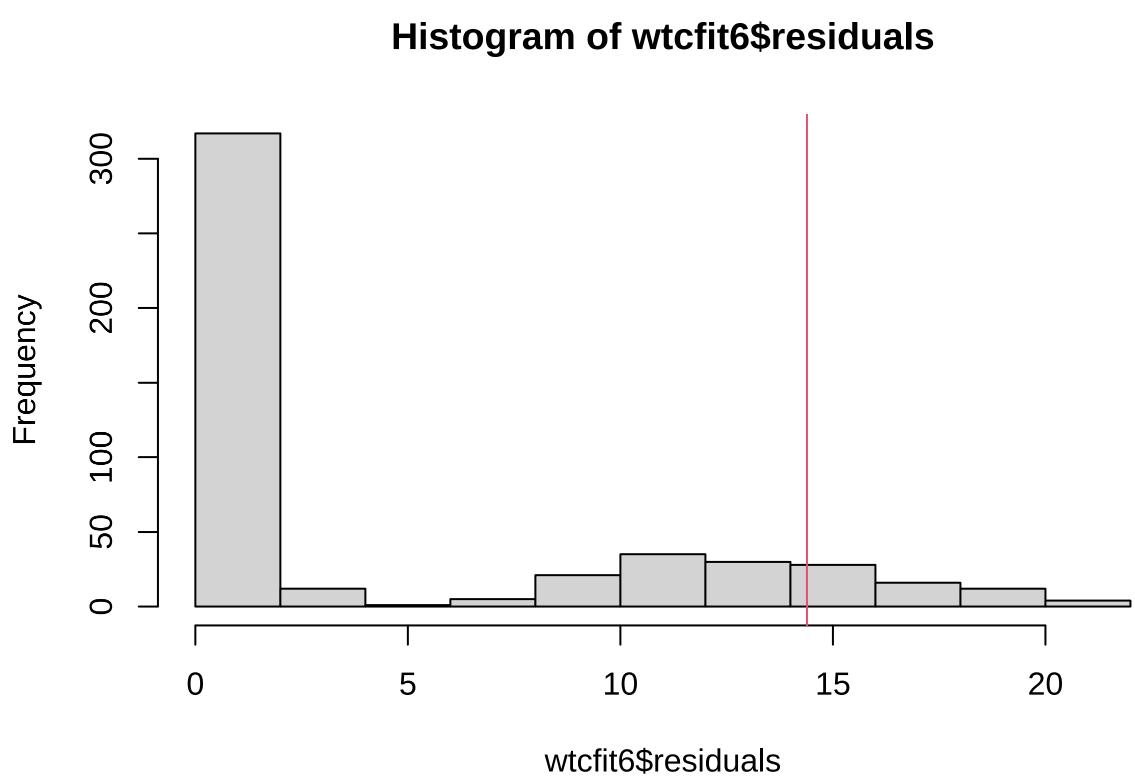
```
[1] 24.97434
```

At this point, we've got a decent quorum of effects, and the deviance reduction is substantial. Of course, we could continue to investigate other mechanisms; see `?rem.dyad` for the full range of options.

1.4 Assessing model adequacy

Model adequacy is an important consideration: even given that our model is the best of those we've seen, is it good enough for our purposes? There are many ways to assess model adequacy; here, we focus on the ability of the relational event model to predict the next event in the sequence, given those that have come before. A natural question to ask when assessing the model is to ask when it is "surprised:" when does it encounter observations that are relatively poorly predicted? To investigate this, we can examine the deviance residuals:

```
nullresid <- 2 * log(37 * 36)  #What would be the deviance residual for the null?
hist(wtcfit6$residuals)  #Deviance residuals - most well-predicted, some around chance levels
abline(v = nullresid, col = 2)
```




```
mean(wtcfit6$residuals < nullresid)  #Beating chance on almost all...
```

```
[1] 0.8898129
```

```
mean(wtcfit6$residuals < 3)  #Upper limit of lower cluster is about 3
```

```
[1] 0.6839917
```

We seem to be doing pretty well here. As another way of evaluating the deviance residuals for the ordinal model, it is useful to note that the quantity $\exp(DR/2)$ (where DR is the deviance residual) is a "random guessing equivalent," or an effective number of events such that a random guess among such events as to which is coming next would be right as often as the model expects to be. We can easily compute this as follows:

```
quantile(exp(wtcfit6$residuals/2))  #'Random guessing equivalent' (ref is 1332)
```

```
          0%          25%          50%          75%         100%
    1.073634     1.268661     1.739723   204.538319 31633.040883
```

Note that there are 1332 possible events, so we are doing much, much better than an uninformative baseline. Likewise, we've come a long way from our initial model:

```
quantile(exp(wtcfit1$residuals/2))  #By comparison, first model much worse!
```

```
       0%       25%       50%       75%      100%
 390.0004  390.0004  390.0004  390.0004 3199.0582
```

In addition to overall examination of residuals, it can be useful to ask which particular events seem to be sources of surprise:

```
cbind(WTCPoliceCalls, wtcfit6$residuals > nullresid)  #Which are the more surprising cases?
```

```
    number source recipient wtcfit6$residuals > nullresid
1        1     16        32                         FALSE
2        2     32        16                         FALSE
3        3     16        32                         FALSE
4        4     16        32                         FALSE
5        5     11        32                          TRUE
6        6     11        32                         FALSE
7        7     11        32                         FALSE
8        8     36        32                         FALSE
9        9      8        32                         FALSE
10      10      8        32                         FALSE
11      11     32         8                         FALSE
12      12     16        32                         FALSE
13      13      8        32                         FALSE
14      14     26        32                          TRUE
15      15     32        26                         FALSE
16      16     26        32                         FALSE
17      17     32        26                         FALSE
18      18     26        32                         FALSE
19      19     32        26                         FALSE
20      20     16        32                         FALSE
21      21     16        32                         FALSE
22      22     27        32                         FALSE
23      23     20        32                         FALSE
24      24     32        20                         FALSE
25      25     20        32                         FALSE
26      26     32        20                         FALSE
27      27     32        16                         FALSE
28      28     16        32                         FALSE
29      29     32        16                         FALSE
30      30     32        16                         FALSE
31      31     16        32                         FALSE
32      32     32        22                         FALSE
33      33      3        32                          TRUE
34      34     32         3                         FALSE
```

35	35	3	32	FALSE
36	36	32	3	FALSE
37	37	32	16	FALSE
38	38	16	32	FALSE
39	39	32	16	FALSE
40	40	3	32	FALSE
41	41	3	32	FALSE
42	42	32	3	FALSE
43	43	3	32	FALSE
44	44	16	3	TRUE
45	45	16	11	FALSE
46	46	11	16	FALSE
47	47	16	11	FALSE
48	48	11	16	FALSE
49	49	16	11	FALSE
50	50	11	16	FALSE
51	51	24	36	TRUE
52	52	24	36	TRUE
53	53	15	32	FALSE
54	54	32	15	FALSE
55	55	15	32	FALSE
56	56	32	15	FALSE
57	57	15	32	FALSE
58	58	32	15	FALSE
59	59	22	32	FALSE
60	60	32	22	FALSE
61	61	15	32	FALSE
62	62	32	15	FALSE
63	63	15	32	FALSE
64	64	32	15	FALSE
65	65	18	32	TRUE
66	66	32	18	FALSE
67	67	18	32	FALSE
68	68	19	32	TRUE
69	69	32	19	FALSE
70	70	19	32	FALSE
71	71	32	19	FALSE
72	72	19	32	FALSE
73	73	16	32	FALSE
74	74	32	16	FALSE
75	75	16	32	FALSE
76	76	32	16	FALSE
77	77	36	16	TRUE
78	78	16	36	FALSE
79	79	36	16	FALSE
80	80	16	36	FALSE
81	81	36	16	FALSE
82	82	16	36	FALSE
83	83	27	32	FALSE
84	84	32	16	FALSE
85	85	16	32	FALSE
86	86	32	16	FALSE
87	87	16	32	FALSE
88	88	32	16	FALSE

89	89	22	15	TRUE
90	90	15	22	FALSE
91	91	22	15	FALSE
92	92	15	22	FALSE
93	93	22	15	FALSE
94	94	16	22	TRUE
95	95	22	16	FALSE
96	96	16	22	FALSE
97	97	22	11	TRUE
98	98	11	22	FALSE
99	99	36	32	FALSE
100	100	32	36	FALSE
101	101	36	32	FALSE
102	102	32	36	FALSE
103	103	36	32	FALSE
104	104	32	36	FALSE
105	105	27	32	TRUE
106	106	37	32	FALSE
107	107	32	37	FALSE
108	108	37	32	FALSE
109	109	32	37	FALSE
110	110	5	32	TRUE
111	111	32	5	FALSE
112	112	5	32	FALSE
113	113	32	5	FALSE
114	114	31	36	TRUE
115	115	36	31	FALSE
116	116	31	36	FALSE
117	117	36	31	FALSE
118	118	37	32	FALSE
119	119	16	32	FALSE
120	120	32	16	FALSE
121	121	16	32	FALSE
122	122	32	16	FALSE
123	123	29	32	TRUE
124	124	32	29	FALSE
125	125	37	14	TRUE
126	126	29	32	FALSE
127	127	31	32	TRUE
128	128	32	37	FALSE
129	129	16	32	FALSE
130	130	32	16	FALSE
131	131	16	32	FALSE
132	132	32	16	FALSE
133	133	16	32	FALSE
134	134	36	16	TRUE
135	135	16	36	FALSE
136	136	36	16	FALSE
137	137	16	36	FALSE
138	138	29	32	FALSE
139	139	8	35	TRUE
140	140	32	16	FALSE
141	141	8	35	TRUE
142	142	32	16	FALSE

143	143	16	32	FALSE
144	144	32	16	FALSE
145	145	16	32	FALSE
146	146	22	32	FALSE
147	147	32	22	FALSE
148	148	22	32	FALSE
149	149	32	22	FALSE
150	150	27	32	TRUE
151	151	32	27	FALSE
152	152	27	32	FALSE
153	153	32	26	FALSE
154	154	22	32	FALSE
155	155	32	22	FALSE
156	156	22	32	FALSE
157	157	32	22	FALSE
158	158	22	32	FALSE
159	159	32	22	FALSE
160	160	22	32	FALSE
161	161	32	22	FALSE
162	162	16	32	FALSE
163	163	32	16	FALSE
164	164	16	32	FALSE
165	165	32	16	FALSE
166	166	16	32	FALSE
167	167	16	11	TRUE
168	168	27	32	FALSE
169	169	32	16	FALSE
170	170	16	32	FALSE
171	171	32	16	FALSE
172	172	36	32	TRUE
173	173	32	36	FALSE
174	174	36	32	FALSE
175	175	32	36	FALSE
176	176	16	32	FALSE
177	177	32	16	FALSE
178	178	16	32	FALSE
179	179	32	16	FALSE
180	180	16	32	FALSE
181	181	32	16	FALSE
182	182	16	32	FALSE
183	183	10	2	TRUE
184	184	2	10	FALSE
185	185	10	26	FALSE
186	186	16	32	FALSE
187	187	32	16	FALSE
188	188	16	32	FALSE
189	189	16	32	FALSE
190	190	32	16	FALSE
191	191	32	16	FALSE
192	192	16	32	FALSE
193	193	32	16	FALSE
194	194	16	32	FALSE
195	195	32	16	FALSE
196	196	16	32	FALSE

197	197	32	16	FALSE
198	198	16	32	FALSE
199	199	32	16	FALSE
200	200	16	32	FALSE
201	201	32	16	FALSE
202	202	22	32	FALSE
203	203	32	22	FALSE
204	204	24	32	TRUE
205	205	32	24	FALSE
206	206	24	32	FALSE
207	207	32	24	FALSE
208	208	16	32	FALSE
209	209	32	16	FALSE
210	210	16	32	FALSE
211	211	32	24	FALSE
212	212	24	32	FALSE
213	213	16	32	FALSE
214	214	30	16	TRUE
215	215	16	30	FALSE
216	216	30	16	FALSE
217	217	16	30	FALSE
218	218	30	16	FALSE
219	219	16	30	FALSE
220	220	32	15	TRUE
221	221	15	32	FALSE
222	222	32	15	FALSE
223	223	15	32	FALSE
224	224	32	15	FALSE
225	225	32	15	FALSE
226	226	15	32	FALSE
227	227	32	15	FALSE
228	228	15	32	FALSE
229	229	32	23	FALSE
230	230	23	32	FALSE
231	231	32	23	FALSE
232	232	23	32	FALSE
233	233	32	23	FALSE
234	234	23	32	FALSE
235	235	32	23	FALSE
236	236	23	32	FALSE
237	237	32	23	FALSE
238	238	23	32	FALSE
239	239	32	19	FALSE
240	240	19	32	FALSE
241	241	32	19	FALSE
242	242	19	32	FALSE
243	243	32	18	FALSE
244	244	15	16	TRUE
245	245	32	18	FALSE
246	246	16	32	TRUE
247	247	32	16	FALSE
248	248	16	32	FALSE
249	249	32	16	FALSE
250	250	15	16	TRUE

251	251	16	15	FALSE
252	252	15	16	FALSE
253	253	16	15	FALSE
254	254	15	16	FALSE
255	255	16	15	FALSE
256	256	25	32	TRUE
257	257	32	25	FALSE
258	258	25	32	FALSE
259	259	32	25	FALSE
260	260	1	4	TRUE
261	261	4	1	FALSE
262	262	1	4	FALSE
263	263	4	1	FALSE
264	264	1	4	FALSE
265	265	4	1	FALSE
266	266	1	4	FALSE
267	267	4	1	FALSE
268	268	1	4	FALSE
269	269	16	32	FALSE
270	270	32	16	FALSE
271	271	16	32	FALSE
272	272	32	16	FALSE
273	273	16	32	FALSE
274	274	32	16	FALSE
275	275	16	32	FALSE
276	276	18	32	FALSE
277	277	32	18	FALSE
278	278	18	32	FALSE
279	279	32	18	FALSE
280	280	18	32	FALSE
281	281	32	18	FALSE
282	282	18	32	FALSE
283	283	32	18	FALSE
284	284	18	32	FALSE
285	285	32	18	FALSE
286	286	18	32	FALSE
287	287	32	18	FALSE
288	288	18	32	FALSE
289	289	25	32	FALSE
290	290	32	16	FALSE
291	291	16	32	FALSE
292	292	32	16	FALSE
293	293	16	32	FALSE
294	294	32	16	FALSE
295	295	16	32	FALSE
296	296	32	16	FALSE
297	297	16	32	FALSE
298	298	32	16	FALSE
299	299	16	32	FALSE
300	300	32	16	FALSE
301	301	16	32	FALSE
302	302	32	16	FALSE
303	303	22	32	FALSE
304	304	32	22	FALSE

305	305	22	32	FALSE
306	306	25	32	FALSE
307	307	32	25	FALSE
308	308	25	32	FALSE
309	309	22	32	FALSE
310	310	32	22	FALSE
311	311	22	32	FALSE
312	312	32	16	FALSE
313	313	25	32	FALSE
314	314	32	25	FALSE
315	315	25	32	FALSE
316	316	32	25	FALSE
317	317	21	32	TRUE
318	318	32	21	FALSE
319	319	21	32	FALSE
320	320	32	21	FALSE
321	321	21	32	FALSE
322	322	32	21	FALSE
323	323	21	32	FALSE
324	324	25	32	FALSE
325	325	32	25	FALSE
326	326	16	36	TRUE
327	327	36	16	FALSE
328	328	36	16	FALSE
329	329	16	36	FALSE
330	330	36	16	FALSE
331	331	16	36	FALSE
332	332	32	16	TRUE
333	333	16	32	FALSE
334	334	31	32	FALSE
335	335	32	31	FALSE
336	336	31	32	FALSE
337	337	32	31	FALSE
338	338	31	32	FALSE
339	339	32	31	FALSE
340	340	32	16	FALSE
341	341	16	32	FALSE
342	342	32	16	FALSE
343	343	16	32	FALSE
344	344	30	32	TRUE
345	345	32	30	FALSE
346	346	30	32	FALSE
347	347	9	32	TRUE
348	348	6	32	FALSE
349	349	22	32	FALSE
350	350	32	22	FALSE
351	351	22	32	FALSE
352	352	32	22	FALSE
353	353	34	32	TRUE
354	354	32	34	FALSE
355	355	34	32	FALSE
356	356	32	34	FALSE
357	357	32	22	FALSE
358	358	22	32	FALSE

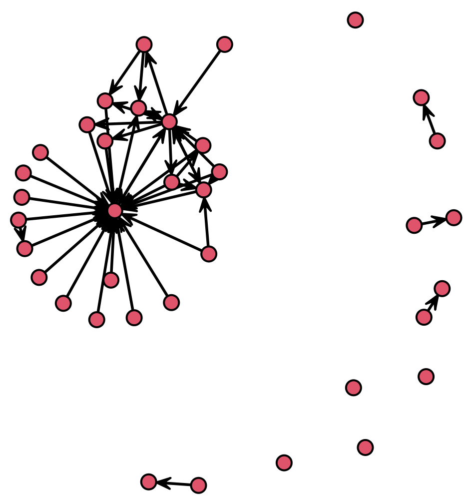
359	359	21	36	TRUE
360	360	16	21	TRUE
361	361	16	32	FALSE
362	362	32	16	FALSE
363	363	16	32	FALSE
364	364	32	16	FALSE
365	365	16	32	FALSE
366	366	32	22	FALSE
367	367	22	32	FALSE
368	368	32	22	FALSE
369	369	22	32	FALSE
370	370	33	32	TRUE
371	371	33	32	FALSE
372	372	32	16	FALSE
373	373	32	33	FALSE
374	374	16	32	FALSE
375	375	32	16	FALSE
376	376	16	32	FALSE
377	377	32	33	FALSE
378	378	33	32	FALSE
379	379	16	15	TRUE
380	380	15	16	FALSE
381	381	16	15	FALSE
382	382	15	16	FALSE
383	383	32	16	FALSE
384	384	16	32	FALSE
385	385	17	32	TRUE
386	386	32	17	FALSE
387	387	16	17	TRUE
388	388	21	36	TRUE
389	389	36	21	FALSE
390	390	21	36	FALSE
391	391	36	21	FALSE
392	392	21	36	FALSE
393	393	36	21	FALSE
394	394	21	36	FALSE
395	395	32	16	FALSE
396	396	16	32	FALSE
397	397	32	16	FALSE
398	398	16	32	FALSE
399	399	16	32	FALSE
400	400	32	16	FALSE
401	401	32	16	FALSE
402	402	16	32	FALSE
403	403	32	16	FALSE
404	404	16	32	FALSE
405	405	32	16	FALSE
406	406	24	16	TRUE
407	407	16	24	FALSE
408	408	24	16	FALSE
409	409	16	24	FALSE
410	410	25	32	FALSE
411	411	32	16	FALSE
412	412	16	32	FALSE

413	413	32	16	FALSE
414	414	16	32	FALSE
415	415	32	16	FALSE
416	416	21	32	TRUE
417	417	32	21	FALSE
418	418	21	32	FALSE
419	419	21	30	TRUE
420	420	32	16	FALSE
421	421	16	32	FALSE
422	422	32	16	FALSE
423	423	16	32	FALSE
424	424	32	21	FALSE
425	425	21	32	FALSE
426	426	32	21	FALSE
427	427	21	36	FALSE
428	428	36	21	FALSE
429	429	21	36	FALSE
430	430	36	21	FALSE
431	431	21	36	FALSE
432	432	36	21	FALSE
433	433	21	36	FALSE
434	434	30	32	FALSE
435	435	32	30	FALSE
436	436	30	32	FALSE
437	437	32	30	FALSE
438	438	30	32	FALSE
439	439	16	32	FALSE
440	440	32	16	FALSE
441	441	16	32	FALSE
442	442	32	16	FALSE
443	443	24	16	FALSE
444	444	16	24	FALSE
445	445	24	16	FALSE
446	446	16	24	FALSE
447	447	24	16	FALSE
448	448	16	24	FALSE
449	449	34	32	FALSE
450	450	32	34	FALSE
451	451	34	32	FALSE
452	452	12	34	TRUE
453	453	16	15	TRUE
454	454	16	32	FALSE
455	455	12	32	TRUE
456	456	32	12	FALSE
457	457	12	32	FALSE
458	458	32	12	FALSE
459	459	32	34	FALSE
460	460	34	32	FALSE
461	461	29	32	FALSE
462	462	32	29	FALSE
463	463	29	32	FALSE
464	464	32	29	FALSE
465	465	29	32	FALSE
466	466	32	29	FALSE

```
467    467     32          16                              FALSE
468    468     16          32                              FALSE
469    469     32          16                              FALSE
470    470     16          32                              FALSE
471    471     32          16                              FALSE
472    472     16          32                              FALSE
473    473     28          16                               TRUE
474    474     16          28                              FALSE
475    475     28          16                              FALSE
476    476     28          16                              FALSE
477    477     16          28                              FALSE
478    478     28          16                              FALSE
479    479     15          16                              FALSE
480    480     32          16                              FALSE
481    481     16          32                              FALSE
```

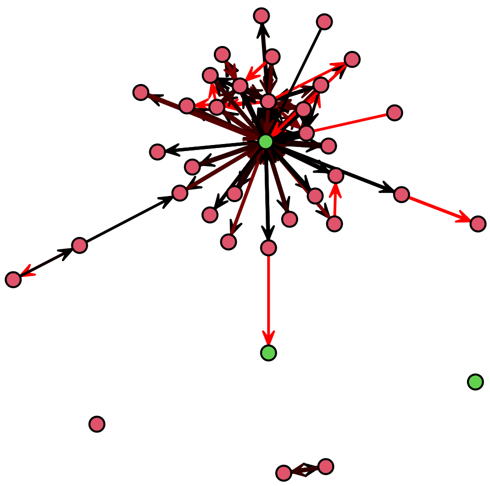
Using `as.sociomatrix.eventlist`, we can even pull out these events and view them in time-aggregated form. This can give us a better sense of the structural context in which they occur:

```
surprising <- as.sociomatrix.eventlist(WTCPoliceCalls[wtcfit6$residuals >
   nullresid, ], 37)
gplot(surprising)  #Plot in network form
```



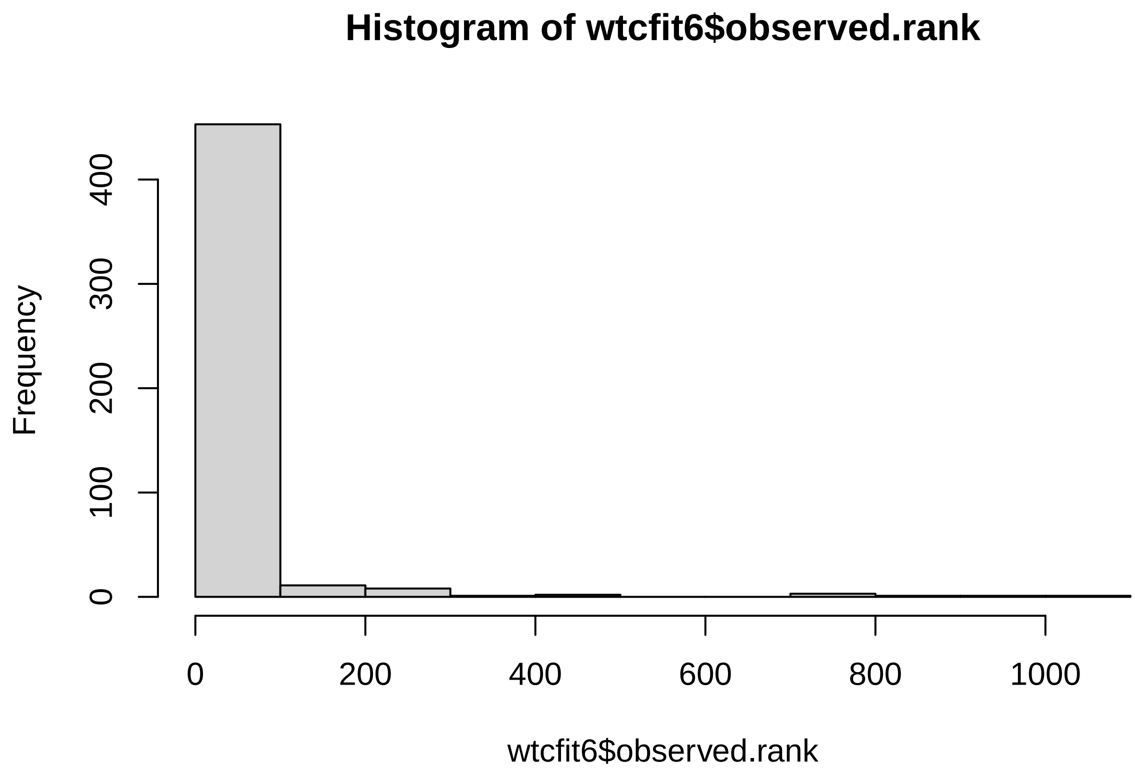
```
# Can also superimpose on the original network (coloring
# edges by fraction surprising)
edgecol <- matrix(rgb(surprising/(WTCPoliceNet + 0.01), 0, 0),
   37, 37)  #Color me surprised
```

```
gplot(WTCPoliceNet, edge.col = edgecol, edge.lwd = WTCPoliceNet^0.75,
   vertex.col = 2 + WTCPoliceIsICR)
```



Yet another approach to adequacy assessment is to consider the rank of the observed events in the predicted rate structure: that is, we ask to what extent the events viewed most likely to occur are in fact those that are observed.

```
hist(wtcfit6$observed.rank)
```




```
cbind(WTCPoliceCalls, wtcfit6$observed.rank)  #Histogram of ranks
```

```
   number source recipient wtcfit6$observed.rank
1       1     16        32                     7
2       2     32        16                     1
3       3     16        32                     1
4       4     16        32                     2
5       5     11        32                    42
6       6     11        32                     6
7       7     11        32                     6
8       8     36        32                    44
9       9      8        32                    45
10     10      8        32                     8
11     11     32         8                     1
12     12     16        32                     2
13     13      8        32                     2
14     14     26        32                    45
15     15     32        26                     1
16     16     26        32                     1
17     17     32        26                     1
18     18     26        32                     1
19     19     32        26                     1
20     20     16        32                     2
21     21     16        32                     2
22     22     27        32                    46
23     23     20        32                    47
```

24	24	32	20	1
25	25	20	32	1
26	26	32	20	1
27	27	32	16	6
28	28	16	32	1
29	29	32	16	1
30	30	32	16	5
31	31	16	32	1
32	32	32	22	19
33	33	3	32	49
34	34	32	3	1
35	35	3	32	1
36	36	32	3	1
37	37	32	16	7
38	38	16	32	1
39	39	32	16	1
40	40	3	32	2
41	41	3	32	2
42	42	32	3	1
43	43	3	32	1
44	44	16	3	276
45	45	16	11	79
46	46	11	16	1
47	47	16	11	1
48	48	11	16	1
49	49	16	11	1
50	50	11	16	1
51	51	24	36	465
52	52	24	36	128
53	53	15	32	28
54	54	32	15	1
55	55	15	32	1
56	56	32	15	1
57	57	15	32	1
58	58	32	15	1
59	59	22	32	5
60	60	32	22	1
61	61	15	32	2
62	62	32	15	1
63	63	15	32	1
64	64	32	15	1
65	65	18	32	58
66	66	32	18	1
67	67	18	32	1
68	68	19	32	57
69	69	32	19	1
70	70	19	32	1
71	71	32	19	1
72	72	19	32	1
73	73	16	32	13
74	74	32	16	1
75	75	16	32	1
76	76	32	16	1
77	77	36	16	248

78	78	16	36	1
79	79	36	16	1
80	80	16	36	1
81	81	36	16	1
82	82	16	36	1
83	83	27	32	16
84	84	32	16	2
85	85	16	32	1
86	86	32	16	1
87	87	16	32	1
88	88	32	16	1
89	89	22	15	279
90	90	15	22	1
91	91	22	15	1
92	92	15	22	1
93	93	22	15	1
94	94	16	22	434
95	95	22	16	1
96	96	16	22	1
97	97	22	11	29
98	98	11	22	1
99	99	36	32	28
100	100	32	36	1
101	101	36	32	1
102	102	32	36	1
103	103	36	32	1
104	104	32	36	1
105	105	27	32	25
106	106	37	32	62
107	107	32	37	1
108	108	37	32	1
109	109	32	37	1
110	110	5	32	72
111	111	32	5	1
112	112	5	32	1
113	113	32	5	1
114	114	31	36	286
115	115	36	31	1
116	116	31	36	1
117	117	36	31	1
118	118	37	32	3
119	119	16	32	13
120	120	32	16	1
121	121	16	32	1
122	122	32	16	1
123	123	29	32	75
124	124	32	29	1
125	125	37	14	158
126	126	29	32	3
127	127	31	32	66
128	128	32	37	15
129	129	16	32	2
130	130	32	16	1
131	131	16	32	1

132	132	32	16	1
133	133	16	32	1
134	134	36	16	118
135	135	16	36	1
136	136	36	16	1
137	137	16	36	1
138	138	29	32	3
139	139	8	35	789
140	140	32	16	15
141	141	8	35	128
142	142	32	16	15
143	143	16	32	1
144	144	32	16	1
145	145	16	32	1
146	146	22	32	42
147	147	32	22	1
148	148	22	32	1
149	149	32	22	1
150	150	27	32	30
151	151	32	27	1
152	152	27	32	1
153	153	32	26	32
154	154	22	32	3
155	155	32	22	1
156	156	22	32	1
157	157	32	22	1
158	158	22	32	1
159	159	32	22	1
160	160	22	32	1
161	161	32	22	1
162	162	16	32	2
163	163	32	16	1
164	164	16	32	1
165	165	32	16	1
166	166	16	32	1
167	167	16	11	62
168	168	27	32	3
169	169	32	16	2
170	170	16	32	1
171	171	32	16	1
172	172	36	32	28
173	173	32	36	1
174	174	36	32	1
175	175	32	36	1
176	176	16	32	2
177	177	32	16	1
178	178	16	32	1
179	179	32	16	1
180	180	16	32	1
181	181	32	16	1
182	182	16	32	1
183	183	10	2	821
184	184	2	10	1
185	185	10	26	60

186	186	16	32	3
187	187	32	16	1
188	188	16	32	1
189	189	16	32	2
190	190	32	16	1
191	191	32	16	12
192	192	16	32	1
193	193	32	16	1
194	194	16	32	1
195	195	32	16	1
196	196	16	32	1
197	197	32	16	1
198	198	16	32	1
199	199	32	16	1
200	200	16	32	1
201	201	32	16	1
202	202	22	32	2
203	203	32	22	1
204	204	24	32	72
205	205	32	24	1
206	206	24	32	1
207	207	32	24	1
208	208	16	32	2
209	209	32	16	1
210	210	16	32	1
211	211	32	24	14
212	212	24	32	1
213	213	16	32	2
214	214	30	16	260
215	215	16	30	1
216	216	30	16	1
217	217	16	30	1
218	218	30	16	1
219	219	16	30	1
220	220	32	15	136
221	221	15	32	1
222	222	32	15	1
223	223	15	32	1
224	224	32	15	1
225	225	32	15	16
226	226	15	32	1
227	227	32	15	1
228	228	15	32	1
229	229	32	23	50
230	230	23	32	1
231	231	32	23	1
232	232	23	32	1
233	233	32	23	1
234	234	23	32	1
235	235	32	23	1
236	236	23	32	1
237	237	32	23	1
238	238	23	32	1
239	239	32	19	35

240	240	19	32	1
241	241	32	19	1
242	242	19	32	1
243	243	32	18	37
244	244	15	16	265
245	245	32	18	91
246	246	16	32	22
247	247	32	16	1
248	248	16	32	1
249	249	32	16	1
250	250	15	16	121
251	251	16	15	1
252	252	15	16	1
253	253	16	15	1
254	254	15	16	1
255	255	16	15	1
256	256	25	32	46
257	257	32	25	1
258	258	25	32	1
259	259	32	25	1
260	260	1	4	981
261	261	4	1	1
262	262	1	4	1
263	263	4	1	1
264	264	1	4	1
265	265	4	1	1
266	266	1	4	1
267	267	4	1	1
268	268	1	4	1
269	269	16	32	22
270	270	32	16	1
271	271	16	32	1
272	272	32	16	1
273	273	16	32	1
274	274	32	16	1
275	275	16	32	1
276	276	18	32	2
277	277	32	18	1
278	278	18	32	1
279	279	32	18	1
280	280	18	32	1
281	281	32	18	1
282	282	18	32	1
283	283	32	18	1
284	284	18	32	1
285	285	32	18	1
286	286	18	32	1
287	287	32	18	1
288	288	18	32	1
289	289	25	32	2
290	290	32	16	17
291	291	16	32	1
292	292	32	16	1
293	293	16	32	1

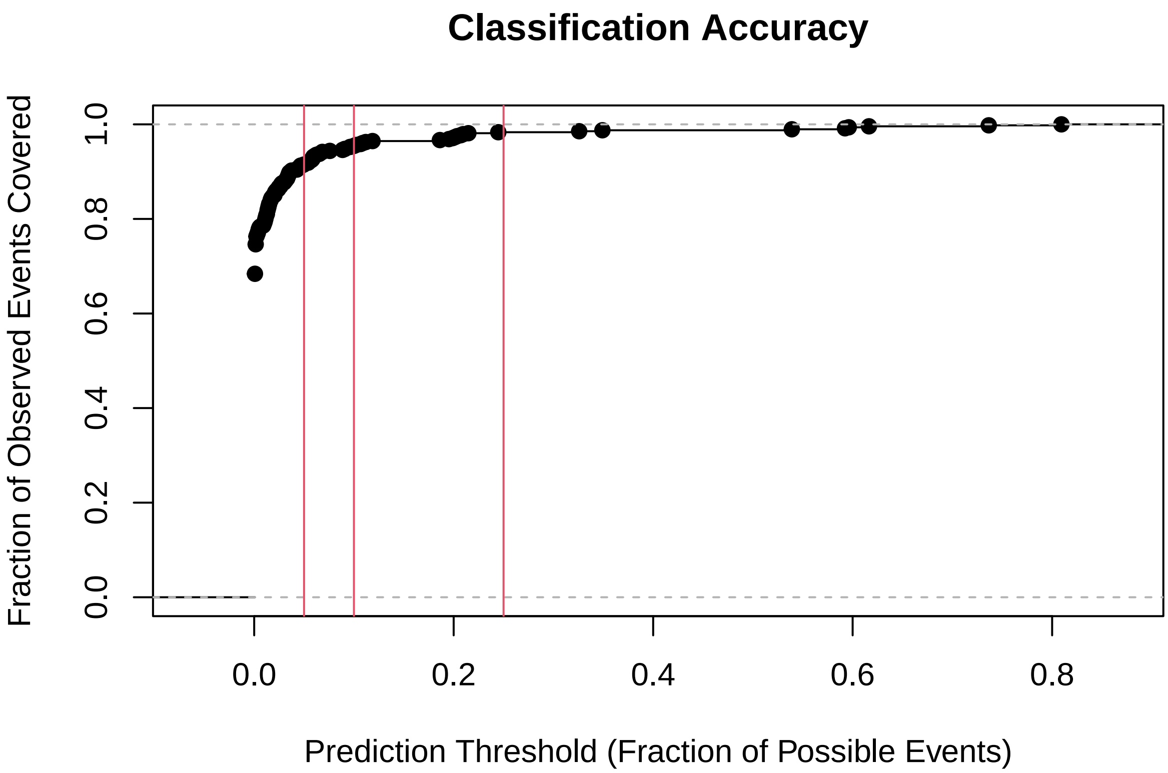
294	294	32	16	1
295	295	16	32	1
296	296	32	16	1
297	297	16	32	1
298	298	32	16	1
299	299	16	32	1
300	300	32	16	1
301	301	16	32	1
302	302	32	16	1
303	303	22	32	2
304	304	32	22	1
305	305	22	32	1
306	306	25	32	2
307	307	32	25	1
308	308	25	32	1
309	309	22	32	2
310	310	32	22	1
311	311	22	32	1
312	312	32	16	16
313	313	25	32	3
314	314	32	25	1
315	315	25	32	1
316	316	32	25	1
317	317	21	32	78
318	318	32	21	1
319	319	21	32	1
320	320	32	21	1
321	321	21	32	1
322	322	32	21	1
323	323	21	32	1
324	324	25	32	2
325	325	32	25	1
326	326	16	36	142
327	327	36	16	1
328	328	36	16	20
329	329	16	36	1
330	330	36	16	1
331	331	16	36	1
332	332	32	16	47
333	333	16	32	1
334	334	31	32	40
335	335	32	31	1
336	336	31	32	1
337	337	32	31	1
338	338	31	32	1
339	339	32	31	1
340	340	32	16	14
341	341	16	32	1
342	342	32	16	1
343	343	16	32	1
344	344	30	32	79
345	345	32	30	1
346	346	30	32	1
347	347	9	32	75

348	348	6	32	77
349	349	22	32	2
350	350	32	22	1
351	351	22	32	1
352	352	32	22	1
353	353	34	32	81
354	354	32	34	1
355	355	34	32	1
356	356	32	34	1
357	357	32	22	18
358	358	22	32	1
359	359	21	36	326
360	360	16	21	718
361	361	16	32	2
362	362	32	16	1
363	363	16	32	1
364	364	32	16	1
365	365	16	32	1
366	366	32	22	18
367	367	22	32	1
368	368	32	22	1
369	369	22	32	1
370	370	33	32	83
371	371	33	32	44
372	372	32	16	19
373	373	32	33	17
374	374	16	32	2
375	375	32	16	1
376	376	16	32	1
377	377	32	33	19
378	378	33	32	1
379	379	16	15	133
380	380	15	16	1
381	381	16	15	1
382	382	15	16	1
383	383	32	16	33
384	384	16	32	1
385	385	17	32	87
386	386	32	17	1
387	387	16	17	1078
388	388	21	36	145
389	389	36	21	1
390	390	21	36	1
391	391	36	21	1
392	392	21	36	1
393	393	36	21	1
394	394	21	36	1
395	395	32	16	36
396	396	16	32	1
397	397	32	16	1
398	398	16	32	1
399	399	16	32	2
400	400	32	16	1
401	401	32	16	18

402	402	16	32	1
403	403	32	16	1
404	404	16	32	1
405	405	32	16	1
406	406	24	16	268
407	407	16	24	1
408	408	24	16	1
409	409	16	24	1
410	410	25	32	4
411	411	32	16	2
412	412	16	32	1
413	413	32	16	1
414	414	16	32	1
415	415	32	16	1
416	416	21	32	33
417	417	32	21	1
418	418	21	32	1
419	419	21	30	101
420	420	32	16	36
421	421	16	32	1
422	422	32	16	1
423	423	16	32	1
424	424	32	21	18
425	425	21	32	1
426	426	32	21	1
427	427	21	36	27
428	428	36	21	1
429	429	21	36	1
430	430	36	21	1
431	431	21	36	1
432	432	36	21	1
433	433	21	36	1
434	434	30	32	20
435	435	32	30	1
436	436	30	32	1
437	437	32	30	1
438	438	30	32	1
439	439	16	32	2
440	440	32	16	1
441	441	16	32	1
442	442	32	16	1
443	443	24	16	22
444	444	16	24	1
445	445	24	16	1
446	446	16	24	1
447	447	24	16	1
448	448	16	24	1
449	449	34	32	4
450	450	32	34	1
451	451	34	32	1
452	452	12	34	794
453	453	16	15	149
454	454	16	32	3
455	455	12	32	91

456	456	32	12	1
457	457	12	32	1
458	458	32	12	1
459	459	32	34	25
460	460	34	32	1
461	461	29	32	2
462	462	32	29	1
463	463	29	32	1
464	464	32	29	1
465	465	29	32	1
466	466	32	29	1
467	467	32	16	22
468	468	16	32	1
469	469	32	16	1
470	470	16	32	1
471	471	32	16	1
472	472	16	32	1
473	473	28	16	271
474	474	16	28	1
475	475	28	16	1
476	476	28	16	23
477	477	16	28	1
478	478	28	16	1
479	479	15	16	23
480	480	32	16	20
481	481	16	32	1

```
# Rank on a per-event basis (low is good) Sometimes useful
# to plot the ECDF of the observed ranks....
plot(ecdf(wtcfit6$observed.rank/(37 * 36)), xlab = "Prediction Threshold (Fraction of Possible Events)"
   ylab = "Fraction of Observed Events Covered", main = "Classification Accuracy")
abline(v = c(0.05, 0.1, 0.25), col = 2)
```



As the above indicates, we sometimes (in fact often) manage to get things exactly right: that is, the event predicted most likely to be the next in the sequence is in fact the one that is observed. Examining the match rate is a very strict notion of adequacy, but can be useful for assessing models that are strongly predictive.

```
wtcfit6$predicted.match   #Exactly correct src/target
```

```
       source recipient
 [1,]  FALSE     FALSE
 [2,]   TRUE      TRUE
 [3,]   TRUE      TRUE
 [4,]  FALSE     FALSE
 [5,]  FALSE     FALSE
 [6,]  FALSE     FALSE
 [7,]  FALSE     FALSE
 [8,]  FALSE     FALSE
 [9,]  FALSE     FALSE
[10,]  FALSE     FALSE
[11,]   TRUE      TRUE
[12,]  FALSE      TRUE
[13,]  FALSE     FALSE
[14,]  FALSE     FALSE
[15,]   TRUE      TRUE
[16,]   TRUE      TRUE
[17,]   TRUE      TRUE
[18,]   TRUE      TRUE
[19,]   TRUE      TRUE
[20,]  FALSE      TRUE
```

```
[21,]  FALSE   FALSE
[22,]  FALSE   FALSE
[23,]  FALSE   FALSE
[24,]   TRUE    TRUE
[25,]   TRUE    TRUE
[26,]   TRUE    TRUE
[27,]  FALSE   FALSE
[28,]   TRUE    TRUE
[29,]   TRUE    TRUE
[30,]  FALSE   FALSE
[31,]   TRUE    TRUE
[32,]   TRUE   FALSE
[33,]  FALSE    TRUE
[34,]   TRUE    TRUE
[35,]   TRUE    TRUE
[36,]   TRUE    TRUE
[37,]  FALSE   FALSE
[38,]   TRUE    TRUE
[39,]   TRUE    TRUE
[40,]  FALSE    TRUE
[41,]  FALSE   FALSE
[42,]   TRUE    TRUE
[43,]   TRUE    TRUE
[44,]  FALSE    TRUE
[45,]  FALSE   FALSE
[46,]   TRUE    TRUE
[47,]   TRUE    TRUE
[48,]   TRUE    TRUE
[49,]   TRUE    TRUE
[50,]   TRUE    TRUE
[51,]  FALSE   FALSE
[52,]  FALSE   FALSE
[53,]  FALSE   FALSE
[54,]   TRUE    TRUE
[55,]   TRUE    TRUE
[56,]   TRUE    TRUE
[57,]   TRUE    TRUE
[58,]   TRUE    TRUE
[59,]  FALSE    TRUE
[60,]   TRUE    TRUE
[61,]  FALSE    TRUE
[62,]   TRUE    TRUE
[63,]   TRUE    TRUE
[64,]   TRUE    TRUE
[65,]  FALSE    TRUE
[66,]   TRUE    TRUE
[67,]   TRUE    TRUE
[68,]  FALSE   FALSE
[69,]   TRUE    TRUE
[70,]   TRUE    TRUE
[71,]   TRUE    TRUE
[72,]   TRUE    TRUE
[73,]  FALSE   FALSE
[74,]   TRUE    TRUE
```

```
 [75,]   TRUE       TRUE
 [76,]   TRUE       TRUE
 [77,]  FALSE      FALSE
 [78,]   TRUE       TRUE
 [79,]   TRUE       TRUE
 [80,]   TRUE       TRUE
 [81,]   TRUE       TRUE
 [82,]   TRUE       TRUE
 [83,]  FALSE      FALSE
 [84,]   TRUE      FALSE
 [85,]   TRUE       TRUE
 [86,]   TRUE       TRUE
 [87,]   TRUE       TRUE
 [88,]   TRUE       TRUE
 [89,]  FALSE      FALSE
 [90,]   TRUE       TRUE
 [91,]   TRUE       TRUE
 [92,]   TRUE       TRUE
 [93,]   TRUE       TRUE
 [94,]  FALSE       TRUE
 [95,]   TRUE       TRUE
 [96,]   TRUE       TRUE
 [97,]   TRUE      FALSE
 [98,]   TRUE       TRUE
 [99,]  FALSE      FALSE
[100,]   TRUE       TRUE
[101,]   TRUE       TRUE
[102,]   TRUE       TRUE
[103,]   TRUE       TRUE
[104,]   TRUE       TRUE
[105,]  FALSE       TRUE
[106,]  FALSE      FALSE
[107,]   TRUE       TRUE
[108,]   TRUE       TRUE
[109,]   TRUE       TRUE
[110,]  FALSE       TRUE
[111,]   TRUE       TRUE
[112,]   TRUE       TRUE
[113,]   TRUE       TRUE
[114,]  FALSE      FALSE
[115,]   TRUE       TRUE
[116,]   TRUE       TRUE
[117,]   TRUE       TRUE
[118,]  FALSE      FALSE
[119,]  FALSE      FALSE
[120,]   TRUE       TRUE
[121,]   TRUE       TRUE
[122,]   TRUE       TRUE
[123,]  FALSE       TRUE
[124,]   TRUE       TRUE
[125,]  FALSE      FALSE
[126,]  FALSE      FALSE
[127,]  FALSE      FALSE
[128,]   TRUE      FALSE
```

```
[129,] FALSE  TRUE
[130,]  TRUE  TRUE
[131,]  TRUE  TRUE
[132,]  TRUE  TRUE
[133,]  TRUE  TRUE
[134,] FALSE  TRUE
[135,]  TRUE  TRUE
[136,]  TRUE  TRUE
[137,]  TRUE  TRUE
[138,] FALSE FALSE
[139,] FALSE FALSE
[140,] FALSE FALSE
[141,] FALSE FALSE
[142,] FALSE FALSE
[143,]  TRUE  TRUE
[144,]  TRUE  TRUE
[145,]  TRUE  TRUE
[146,] FALSE FALSE
[147,]  TRUE  TRUE
[148,]  TRUE  TRUE
[149,]  TRUE  TRUE
[150,] FALSE  TRUE
[151,]  TRUE  TRUE
[152,]  TRUE  TRUE
[153,]  TRUE FALSE
[154,] FALSE  TRUE
[155,]  TRUE  TRUE
[156,]  TRUE  TRUE
[157,]  TRUE  TRUE
[158,]  TRUE  TRUE
[159,]  TRUE  TRUE
[160,]  TRUE  TRUE
[161,]  TRUE  TRUE
[162,] FALSE  TRUE
[163,]  TRUE  TRUE
[164,]  TRUE  TRUE
[165,]  TRUE  TRUE
[166,]  TRUE  TRUE
[167,] FALSE FALSE
[168,] FALSE FALSE
[169,]  TRUE FALSE
[170,]  TRUE  TRUE
[171,]  TRUE  TRUE
[172,] FALSE  TRUE
[173,]  TRUE  TRUE
[174,]  TRUE  TRUE
[175,]  TRUE  TRUE
[176,] FALSE  TRUE
[177,]  TRUE  TRUE
[178,]  TRUE  TRUE
[179,]  TRUE  TRUE
[180,]  TRUE  TRUE
[181,]  TRUE  TRUE
[182,]  TRUE  TRUE
```

```
[183,] FALSE  FALSE
[184,]  TRUE   TRUE
[185,]  TRUE  FALSE
[186,] FALSE  FALSE
[187,]  TRUE   TRUE
[188,]  TRUE   TRUE
[189,] FALSE  FALSE
[190,]  TRUE   TRUE
[191,] FALSE  FALSE
[192,]  TRUE   TRUE
[193,]  TRUE   TRUE
[194,]  TRUE   TRUE
[195,]  TRUE   TRUE
[196,]  TRUE   TRUE
[197,]  TRUE   TRUE
[198,]  TRUE   TRUE
[199,]  TRUE   TRUE
[200,]  TRUE   TRUE
[201,]  TRUE   TRUE
[202,] FALSE   TRUE
[203,]  TRUE   TRUE
[204,] FALSE   TRUE
[205,]  TRUE   TRUE
[206,]  TRUE   TRUE
[207,]  TRUE   TRUE
[208,] FALSE   TRUE
[209,]  TRUE   TRUE
[210,]  TRUE   TRUE
[211,]  TRUE  FALSE
[212,]  TRUE   TRUE
[213,] FALSE  FALSE
[214,] FALSE   TRUE
[215,]  TRUE   TRUE
[216,]  TRUE   TRUE
[217,]  TRUE   TRUE
[218,]  TRUE   TRUE
[219,]  TRUE   TRUE
[220,] FALSE  FALSE
[221,]  TRUE   TRUE
[222,]  TRUE   TRUE
[223,]  TRUE   TRUE
[224,]  TRUE   TRUE
[225,] FALSE  FALSE
[226,]  TRUE   TRUE
[227,]  TRUE   TRUE
[228,]  TRUE   TRUE
[229,]  TRUE  FALSE
[230,]  TRUE   TRUE
[231,]  TRUE   TRUE
[232,]  TRUE   TRUE
[233,]  TRUE   TRUE
[234,]  TRUE   TRUE
[235,]  TRUE   TRUE
[236,]  TRUE   TRUE
```

```
[237,]  TRUE   TRUE
[238,]  TRUE   TRUE
[239,]  TRUE  FALSE
[240,]  TRUE   TRUE
[241,]  TRUE   TRUE
[242,]  TRUE   TRUE
[243,]  TRUE  FALSE
[244,] FALSE  FALSE
[245,] FALSE  FALSE
[246,] FALSE   TRUE
[247,]  TRUE   TRUE
[248,]  TRUE   TRUE
[249,]  TRUE   TRUE
[250,] FALSE  FALSE
[251,]  TRUE   TRUE
[252,]  TRUE   TRUE
[253,]  TRUE   TRUE
[254,]  TRUE   TRUE
[255,]  TRUE   TRUE
[256,] FALSE  FALSE
[257,]  TRUE   TRUE
[258,]  TRUE   TRUE
[259,]  TRUE   TRUE
[260,] FALSE  FALSE
[261,]  TRUE   TRUE
[262,]  TRUE   TRUE
[263,]  TRUE   TRUE
[264,]  TRUE   TRUE
[265,]  TRUE   TRUE
[266,]  TRUE   TRUE
[267,]  TRUE   TRUE
[268,]  TRUE   TRUE
[269,] FALSE  FALSE
[270,]  TRUE   TRUE
[271,]  TRUE   TRUE
[272,]  TRUE   TRUE
[273,]  TRUE   TRUE
[274,]  TRUE   TRUE
[275,]  TRUE   TRUE
[276,] FALSE  FALSE
[277,]  TRUE   TRUE
[278,]  TRUE   TRUE
[279,]  TRUE   TRUE
[280,]  TRUE   TRUE
[281,]  TRUE   TRUE
[282,]  TRUE   TRUE
[283,]  TRUE   TRUE
[284,]  TRUE   TRUE
[285,]  TRUE   TRUE
[286,]  TRUE   TRUE
[287,]  TRUE   TRUE
[288,]  TRUE   TRUE
[289,] FALSE  FALSE
[290,]  TRUE  FALSE
```

```
[291,]   TRUE       TRUE
[292,]   TRUE       TRUE
[293,]   TRUE       TRUE
[294,]   TRUE       TRUE
[295,]   TRUE       TRUE
[296,]   TRUE       TRUE
[297,]   TRUE       TRUE
[298,]   TRUE       TRUE
[299,]   TRUE       TRUE
[300,]   TRUE       TRUE
[301,]   TRUE       TRUE
[302,]   TRUE       TRUE
[303,]  FALSE       TRUE
[304,]   TRUE       TRUE
[305,]   TRUE       TRUE
[306,]  FALSE      FALSE
[307,]   TRUE       TRUE
[308,]   TRUE       TRUE
[309,]  FALSE      FALSE
[310,]   TRUE       TRUE
[311,]   TRUE       TRUE
[312,]   TRUE      FALSE
[313,]  FALSE       TRUE
[314,]   TRUE       TRUE
[315,]   TRUE       TRUE
[316,]   TRUE       TRUE
[317,]  FALSE       TRUE
[318,]   TRUE       TRUE
[319,]   TRUE       TRUE
[320,]   TRUE       TRUE
[321,]   TRUE       TRUE
[322,]   TRUE       TRUE
[323,]   TRUE       TRUE
[324,]  FALSE      FALSE
[325,]   TRUE       TRUE
[326,]  FALSE      FALSE
[327,]   TRUE       TRUE
[328,]  FALSE      FALSE
[329,]   TRUE       TRUE
[330,]   TRUE       TRUE
[331,]   TRUE       TRUE
[332,]  FALSE       TRUE
[333,]   TRUE       TRUE
[334,]  FALSE      FALSE
[335,]   TRUE       TRUE
[336,]   TRUE       TRUE
[337,]   TRUE       TRUE
[338,]   TRUE       TRUE
[339,]   TRUE       TRUE
[340,]  FALSE      FALSE
[341,]   TRUE       TRUE
[342,]   TRUE       TRUE
[343,]   TRUE       TRUE
[344,]  FALSE      FALSE
```

```
[345,]  TRUE   TRUE
[346,]  TRUE   TRUE
[347,] FALSE  FALSE
[348,] FALSE  FALSE
[349,] FALSE  FALSE
[350,]  TRUE   TRUE
[351,]  TRUE   TRUE
[352,]  TRUE   TRUE
[353,] FALSE   TRUE
[354,]  TRUE   TRUE
[355,]  TRUE   TRUE
[356,]  TRUE   TRUE
[357,] FALSE  FALSE
[358,]  TRUE   TRUE
[359,] FALSE  FALSE
[360,] FALSE   TRUE
[361,] FALSE  FALSE
[362,]  TRUE   TRUE
[363,]  TRUE   TRUE
[364,]  TRUE   TRUE
[365,]  TRUE   TRUE
[366,]  TRUE  FALSE
[367,]  TRUE   TRUE
[368,]  TRUE   TRUE
[369,]  TRUE   TRUE
[370,] FALSE  FALSE
[371,] FALSE  FALSE
[372,]  TRUE  FALSE
[373,] FALSE  FALSE
[374,] FALSE   TRUE
[375,]  TRUE   TRUE
[376,]  TRUE   TRUE
[377,]  TRUE  FALSE
[378,]  TRUE   TRUE
[379,] FALSE  FALSE
[380,]  TRUE   TRUE
[381,]  TRUE   TRUE
[382,]  TRUE   TRUE
[383,] FALSE  FALSE
[384,]  TRUE   TRUE
[385,] FALSE  FALSE
[386,]  TRUE   TRUE
[387,] FALSE  FALSE
[388,] FALSE  FALSE
[389,]  TRUE   TRUE
[390,]  TRUE   TRUE
[391,]  TRUE   TRUE
[392,]  TRUE   TRUE
[393,]  TRUE   TRUE
[394,]  TRUE   TRUE
[395,] FALSE  FALSE
[396,]  TRUE   TRUE
[397,]  TRUE   TRUE
[398,]  TRUE   TRUE
```

```
[399,]  FALSE     FALSE
[400,]   TRUE      TRUE
[401,]  FALSE     FALSE
[402,]   TRUE      TRUE
[403,]   TRUE      TRUE
[404,]   TRUE      TRUE
[405,]   TRUE      TRUE
[406,]  FALSE     FALSE
[407,]   TRUE      TRUE
[408,]   TRUE      TRUE
[409,]   TRUE      TRUE
[410,]  FALSE     FALSE
[411,]   TRUE     FALSE
[412,]   TRUE      TRUE
[413,]   TRUE      TRUE
[414,]   TRUE      TRUE
[415,]   TRUE      TRUE
[416,]  FALSE      TRUE
[417,]   TRUE      TRUE
[418,]   TRUE      TRUE
[419,]  FALSE     FALSE
[420,]  FALSE     FALSE
[421,]   TRUE      TRUE
[422,]   TRUE      TRUE
[423,]   TRUE      TRUE
[424,]   TRUE     FALSE
[425,]   TRUE      TRUE
[426,]   TRUE      TRUE
[427,]   TRUE     FALSE
[428,]   TRUE      TRUE
[429,]   TRUE      TRUE
[430,]   TRUE      TRUE
[431,]   TRUE      TRUE
[432,]   TRUE      TRUE
[433,]   TRUE      TRUE
[434,]  FALSE     FALSE
[435,]   TRUE      TRUE
[436,]   TRUE      TRUE
[437,]   TRUE      TRUE
[438,]   TRUE      TRUE
[439,]  FALSE     FALSE
[440,]   TRUE      TRUE
[441,]   TRUE      TRUE
[442,]   TRUE      TRUE
[443,]  FALSE     FALSE
[444,]   TRUE      TRUE
[445,]   TRUE      TRUE
[446,]   TRUE      TRUE
[447,]   TRUE      TRUE
[448,]   TRUE      TRUE
[449,]  FALSE     FALSE
[450,]   TRUE      TRUE
[451,]   TRUE      TRUE
[452,]  FALSE      TRUE
```

```
[453,]  FALSE     FALSE
[454,]  FALSE     FALSE
[455,]  FALSE     FALSE
[456,]   TRUE      TRUE
[457,]   TRUE      TRUE
[458,]   TRUE      TRUE
[459,]  FALSE     FALSE
[460,]   TRUE      TRUE
[461,]  FALSE     FALSE
[462,]   TRUE      TRUE
[463,]   TRUE      TRUE
[464,]   TRUE      TRUE
[465,]   TRUE      TRUE
[466,]   TRUE      TRUE
[467,]  FALSE     FALSE
[468,]   TRUE      TRUE
[469,]   TRUE      TRUE
[470,]   TRUE      TRUE
[471,]   TRUE      TRUE
[472,]   TRUE      TRUE
[473,]  FALSE      TRUE
[474,]   TRUE      TRUE
[475,]   TRUE      TRUE
[476,]  FALSE     FALSE
[477,]   TRUE      TRUE
[478,]   TRUE      TRUE
[479,]  FALSE     FALSE
[480,]  FALSE     FALSE
[481,]   TRUE      TRUE
```

```r
mean(apply(wtcfit6$predicted.match, 1, any))  #Fraction for which something is right
```

```
[1] 0.7941788
```

```r
mean(apply(wtcfit6$predicted.match, 1, all))  #Fraction entirely right
```

```
[1] 0.6839917
```

```r
colMeans(wtcfit6$predicted.match)  #Fraction src/target, respectively
```

```
   source recipient
0.7234927 0.7546778
```

Despite its simplicity, this model seems to fit extremely well. Further improvement is possible, but for many purposes we might view it as an adequate representation of the event dynamics in this WTC police network.

1.5 Simulating from the fitted model

In addition to fitting REMs, `relevent` has tools for simulating from them. These work a bit like the `simulate` commands in the `ergm` library, in that they can be used in two modes: we can simulate draws from a fitted `rem.dyad` model; or we can simulate draws from an *a priori* specified model. For now, let's consider this first case.

The syntax for the `rem.dyad` `simulate` method is as follows:

```r
simulate(object, nsim = object$m, seed = NULL, coef = NULL, covar = NULL,
   verbose = FALSE, ...)
```

`object` here is our fitted model object, `nsim` is the number of events to draw from the model (the length of the event series to simulate), `seed` is an optional random number seed to specify, `coef` is a (here optional) coefficient vector, `covar` is our usual covariate list, and `verbose` says whether we want to print tracking information. By default, the coefficients used are taken from the fitted model, but specifying `coef` will allow them to be overridden (a useful tool for performing scenario analyses, as illustrated below). Likewise, the function will by default simulate as many events as were in the original data, but this can be altered by changing `nsim`. Note that we *do* have to specify any covariates being used when simulating, both because `rem.dyad` does not save the input covariates, and because (even if it did) the size of the covariate set in some cases depends on the number of events to be produced.

Let's begin with the most basic use case: simulating a synthetic replicate of our original data, using our final model. For this, we only need pass our model, and the covariates used:

```
set.seed(1331)
simwtc <- simulate(wtcfit6, covar = list(CovInt = WTCPoliceIsICR),
   verbose = TRUE)
```

```
Working on event 25 of 481
Working on event 50 of 481
Working on event 75 of 481
Working on event 100 of 481
Working on event 125 of 481
Working on event 150 of 481
Working on event 175 of 481
Working on event 200 of 481
Working on event 225 of 481
Working on event 250 of 481
Working on event 275 of 481
Working on event 300 of 481
Working on event 325 of 481
Working on event 350 of 481
Working on event 375 of 481
Working on event 400 of 481
Working on event 425 of 481
Working on event 450 of 481
Working on event 475 of 481
```

We now have a simulated event sequence from the `wtcfit6` model! Let's see what it looks like:

```
simwtc
```

```
               [,1] [,2] [,3]
  [1,] 0.0004468037   27   13
  [2,] 0.0004626365   13   27
  [3,] 0.0004704297   27   13
  [4,] 0.0004819716   13   27
  [5,] 0.0004923628   27   13
  [6,] 0.0005274594   13   27
  [7,] 0.0005329693   27   13
  [8,] 0.0005379387   13   27
  [9,] 0.0005687593   27   13
 [10,] 0.0006085595   13   27
 [11,] 0.0006466945   27   13
 [12,] 0.0006974053   13   27
 [13,] 0.0007084273   27   13
 [14,] 0.0007339463   13   27
 [15,] 0.0007466829   27   13
```

```
[16,] 0.0007492683   13   27
[17,] 0.0007884909   27   13
[18,] 0.0008027625   13   27
[19,] 0.0008181865   27   13
[20,] 0.0008316952   13   27
[21,] 0.0008386959   27   13
[22,] 0.0008486556   13   27
[23,] 0.0008972990   27   13
[24,] 0.0009089347   13   27
[25,] 0.0009255862   27   13
[26,] 0.0009356594   13   27
[27,] 0.0009481628   27   13
[28,] 0.0009701587   13   27
[29,] 0.0010908130   27   13
[30,] 0.0011515702   13   27
[31,] 0.0011593253   27   13
[32,] 0.0012039061   13   27
[33,] 0.0012429317   27   13
[34,] 0.0012671525   13   27
[35,] 0.0012826752   27   13
[36,] 0.0012830963   13   27
[37,] 0.0012937784   27   13
[38,] 0.0013168737   13   27
[39,] 0.0013233223   27   13
[40,] 0.0013545025   13   27
[41,] 0.0013680706   27   13
[42,] 0.0014074779   13   27
[43,] 0.0014247731   27   13
[44,] 0.0014539612   13   27
[45,] 0.0015243125   27   13
[46,] 0.0015514799   13   27
[47,] 0.0015694444   27   13
[48,] 0.0015723405   14    7
[49,] 0.0016150093    7   14
[50,] 0.0019005495   14    7
[51,] 0.0019436751   14    1
[52,] 0.0026720071   10   15
[53,] 0.0027797163   15   27
[54,] 0.0028284331   27   13
[55,] 0.0028393703   13   27
[56,] 0.0028778669   27   13
[57,] 0.0028872745   13   27
[58,] 0.0029061574   27   13
[59,] 0.0029642788   13   27
[60,] 0.0029984275   27   13
[61,] 0.0030100931   13   27
[62,] 0.0030511732   13   10
[63,] 0.0031090736   10   13
[64,] 0.0032220003   13   10
[65,] 0.0032321823   10   13
[66,] 0.0032814063   13   10
[67,] 0.0032917865   10   13
[68,] 0.0033707913   19   13
[69,] 0.0034818291   13   19
```

```
 [70,] 0.0034824391   19   13
 [71,] 0.0034995456   13   19
 [72,] 0.0035038241   19   13
 [73,] 0.0036111921   13   19
 [74,] 0.0036167909   19   13
 [75,] 0.0036274942   13   19
 [76,] 0.0036565874   19   13
 [77,] 0.0037396404   13   19
 [78,] 0.0037581203   19   13
 [79,] 0.0037626679   13   19
 [80,] 0.0038189882   19   13
 [81,] 0.0039261284   13   19
 [82,] 0.0039629404   19   13
 [83,] 0.0040927924   13   19
 [84,] 0.0040931646   19   13
 [85,] 0.0041057193   13   19
 [86,] 0.0041215224   19   13
 [87,] 0.0041289047   13   19
 [88,] 0.0041427527   19   13
 [89,] 0.0042120045   13   32
 [90,] 0.0042217457   32   13
 [91,] 0.0042779780   13   32
 [92,] 0.0042783679   32   13
 [93,] 0.0042967949   13   32
 [94,] 0.0043393786   32   13
 [95,] 0.0044448128    1   13
 [96,] 0.0047968558   27   13
 [97,] 0.0048509162   13   27
 [98,] 0.0048628311   27   13
 [99,] 0.0049557717   13   27
[100,] 0.0049983464   27   13
[101,] 0.0050064196   13   27
[102,] 0.0050738140   27   13
[103,] 0.0050807734   13   27
[104,] 0.0051075827   27   13
[105,] 0.0051297565   13   27
[106,] 0.0051722794   27   13
[107,] 0.0052835176   13   27
[108,] 0.0052870769   27   13
[109,] 0.0053668566   13   27
[110,] 0.0053920768   27   13
[111,] 0.0053967348   13   27
[112,] 0.0053997387   27   13
[113,] 0.0054173571   13   27
[114,] 0.0054407495   27   13
[115,] 0.0054851996   13   27
[116,] 0.0054943310   27   13
[117,] 0.0055459587   13   27
[118,] 0.0055745344   27   13
[119,] 0.0055822387   13   27
[120,] 0.0055973198   22   19
[121,] 0.0056548113    7   16
[122,] 0.0056565772    7   32
[123,] 0.0057062067   32    7
```

[124,]	0.0058585200	7	32
[125,]	0.0058586059	19	27
[126,]	0.0060430914	19	8
[127,]	0.0062320149	34	9
[128,]	0.0064461376	9	34
[129,]	0.0064966523	34	9
[130,]	0.0065614688	20	13
[131,]	0.0067258537	13	20
[132,]	0.0067578971	20	13
[133,]	0.0068017018	13	20
[134,]	0.0068169609	20	13
[135,]	0.0068175603	13	20
[136,]	0.0068193375	20	13
[137,]	0.0072490023	13	36
[138,]	0.0073217099	36	13
[139,]	0.0073334576	8	14
[140,]	0.0075354927	37	19
[141,]	0.0080144305	19	37
[142,]	0.0081275539	37	19
[143,]	0.0082557445	19	34
[144,]	0.0084110183	23	13
[145,]	0.0084193262	13	3
[146,]	0.0084699743	3	13
[147,]	0.0084824236	13	3
[148,]	0.0084888033	3	13
[149,]	0.0085827016	13	3
[150,]	0.0086069518	3	13
[151,]	0.0087828726	35	34
[152,]	0.0089764370	7	32
[153,]	0.0091843670	36	13
[154,]	0.0091999470	34	13
[155,]	0.0095757076	13	34
[156,]	0.0095844298	34	13
[157,]	0.0096696495	13	34
[158,]	0.0097051639	34	13
[159,]	0.0097855855	12	25
[160,]	0.0098027304	25	12
[161,]	0.0098266242	36	13
[162,]	0.0099684114	13	36
[163,]	0.0100833282	36	13
[164,]	0.0101338236	13	36
[165,]	0.0101348208	36	13
[166,]	0.0101359436	13	36
[167,]	0.0101362160	36	13
[168,]	0.0102493013	34	13
[169,]	0.0103154482	13	34
[170,]	0.0103373270	34	13
[171,]	0.0103487979	13	34
[172,]	0.0103762911	34	14
[173,]	0.0103797891	14	34
[174,]	0.0104068093	34	14
[175,]	0.0104986921	22	9
[176,]	0.0105404012	9	5
[177,]	0.0105561170	32	6

```
[178,] 0.0111850679    6   32
[179,] 0.0112744069   32    6
[180,] 0.0114502292    6   32
[181,] 0.0115323191   17   32
[182,] 0.0115771734   32   17
[183,] 0.0116842323   17   32
[184,] 0.0117025022    8   21
[185,] 0.0118103739   21    8
[186,] 0.0118195300   36   27
[187,] 0.0118529142   36   13
[188,] 0.0119964117   13   34
[189,] 0.0119970428   34   13
[190,] 0.0121343062   25   18
[191,] 0.0121393506   18   12
[192,] 0.0121894876   11   12
[193,] 0.0123057281    7   32
[194,] 0.0125862617   32   14
[195,] 0.0125951288   14   32
[196,] 0.0126633957    6   20
[197,] 0.0128275415   25   18
[198,] 0.0132555835    9    6
[199,] 0.0132829885    3   13
[200,] 0.0134291758   13    3
[201,] 0.0134839210   36   13
[202,] 0.0135437201   17   32
[203,] 0.0136436819   14   13
[204,] 0.0138417769   13   14
[205,] 0.0138438397   14   13
[206,] 0.0139496944   13   14
[207,] 0.0139804066   14   13
[208,] 0.0139867062   13    3
[209,] 0.0140538515    3   13
[210,] 0.0140623483   14   13
[211,] 0.0144145036   13   14
[212,] 0.0144300544   14   13
[213,] 0.0144495355   13   14
[214,] 0.0144640063   14   13
[215,] 0.0145017638   13   14
[216,] 0.0145029027   14   13
[217,] 0.0145118854   13   14
[218,] 0.0145119762   14   13
[219,] 0.0145319720   13   14
[220,] 0.0145356619   14   13
[221,] 0.0145938272   13   14
[222,] 0.0145953873   14   13
[223,] 0.0146065457   13   14
[224,] 0.0146465378   14   13
[225,] 0.0146650777   13   14
[226,] 0.0147036105   14   13
[227,] 0.0147131933   13   14
[228,] 0.0147587748   14    7
[229,] 0.0147680605    7   14
[230,] 0.0150733687   14    7
[231,] 0.0151593320    7   14
```

[232,]	0.0151611674	14	7
[233,]	0.0153084712	7	14
[234,]	0.0154132771	14	7
[235,]	0.0154630741	7	14
[236,]	0.0154683185	14	7
[237,]	0.0158290609	7	21
[238,]	0.0159298269	3	13
[239,]	0.0162762023	13	3
[240,]	0.0163345305	36	27
[241,]	0.0164640819	27	17
[242,]	0.0165578216	17	27
[243,]	0.0166365518	34	32
[244,]	0.0167307895	8	9
[245,]	0.0169132338	12	14
[246,]	0.0169183259	14	12
[247,]	0.0170049871	14	32
[248,]	0.0170257977	32	14
[249,]	0.0171313075	14	32
[250,]	0.0172037723	32	14
[251,]	0.0172433546	14	32
[252,]	0.0172553841	32	14
[253,]	0.0172588572	14	32
[254,]	0.0173394824	32	14
[255,]	0.0174859588	14	32
[256,]	0.0176055546	32	14
[257,]	0.0176171234	14	32
[258,]	0.0176355326	32	14
[259,]	0.0176363899	14	5
[260,]	0.0178551296	30	25
[261,]	0.0179038912	24	3
[262,]	0.0179614156	3	13
[263,]	0.0179934458	13	3
[264,]	0.0180305524	3	13
[265,]	0.0180429788	13	3
[266,]	0.0181813951	3	13
[267,]	0.0184782074	13	3
[268,]	0.0184845219	3	13
[269,]	0.0184972893	13	3
[270,]	0.0184988343	3	13
[271,]	0.0185055797	5	3
[272,]	0.0187918664	3	5
[273,]	0.0189023340	5	3
[274,]	0.0192145298	10	13
[275,]	0.0192505078	13	10
[276,]	0.0192688123	10	13
[277,]	0.0194784659	13	10
[278,]	0.0194887014	10	13
[279,]	0.0195002548	13	10
[280,]	0.0195378845	24	19
[281,]	0.0195719229	19	24
[282,]	0.0197782835	24	19
[283,]	0.0199081357	19	24
[284,]	0.0199500457	24	19
[285,]	0.0200982578	19	24

[286,]	0.0201604806	24	19
[287,]	0.0201623866	29	16
[288,]	0.0207869219	24	30
[289,]	0.0210273775	17	32
[290,]	0.0210622046	32	17
[291,]	0.0210828130	17	32
[292,]	0.0210952752	32	17
[293,]	0.0211329695	17	32
[294,]	0.0212798052	33	13
[295,]	0.0213271700	23	8
[296,]	0.0217965511	8	23
[297,]	0.0218333401	35	13
[298,]	0.0219790111	13	35
[299,]	0.0220608649	35	13
[300,]	0.0220865274	13	35
[301,]	0.0221579945	35	13
[302,]	0.0221665006	13	35
[303,]	0.0221694719	35	13
[304,]	0.0222068245	13	35
[305,]	0.0222089425	35	13
[306,]	0.0222403088	2	29
[307,]	0.0234768598	3	26
[308,]	0.0235433074	26	3
[309,]	0.0235718927	30	25
[310,]	0.0235745023	25	30
[311,]	0.0236604695	36	33
[312,]	0.0236673593	12	32
[313,]	0.0237699169	20	13
[314,]	0.0238429667	13	6
[315,]	0.0238765856	6	13
[316,]	0.0239138126	13	6
[317,]	0.0239983877	6	13
[318,]	0.0240305884	13	6
[319,]	0.0240332773	6	13
[320,]	0.0240570425	13	6
[321,]	0.0240782290	6	13
[322,]	0.0243157765	13	6
[323,]	0.0243718037	6	13
[324,]	0.0245154928	13	6
[325,]	0.0245407958	6	13
[326,]	0.0245484017	36	13
[327,]	0.0246745495	15	11
[328,]	0.0248602998	11	15
[329,]	0.0248914984	15	11
[330,]	0.0249525295	36	13
[331,]	0.0249679162	13	36
[332,]	0.0249824877	36	13
[333,]	0.0250038325	13	36
[334,]	0.0250310501	21	28
[335,]	0.0252419834	8	20
[336,]	0.0256013720	23	6
[337,]	0.0256986523	23	37
[338,]	0.0258426251	17	27
[339,]	0.0259476944	27	17

[340,]	0.0259488282	17	27
[341,]	0.0264904744	12	13
[342,]	0.0265228257	13	12
[343,]	0.0265818520	12	13
[344,]	0.0266614725	3	29
[345,]	0.0267612812	37	14
[346,]	0.0268845215	14	37
[347,]	0.0268879343	37	14
[348,]	0.0269286884	14	37
[349,]	0.0269360921	17	8
[350,]	0.0278964242	8	17
[351,]	0.0280965550	17	8
[352,]	0.0281559822	8	17
[353,]	0.0285466952	7	30
[354,]	0.0285679607	35	13
[355,]	0.0285992632	13	35
[356,]	0.0286171657	12	13
[357,]	0.0290322387	13	12
[358,]	0.0291357700	12	13
[359,]	0.0292442398	13	12
[360,]	0.0293683434	12	13
[361,]	0.0294945773	31	14
[362,]	0.0297095344	14	31
[363,]	0.0298376213	10	13
[364,]	0.0300455572	13	10
[365,]	0.0301098837	10	13
[366,]	0.0301205439	13	17
[367,]	0.0302045183	6	8
[368,]	0.0303019986	8	6
[369,]	0.0303115659	6	8
[370,]	0.0303131470	12	13
[371,]	0.0304611049	29	3
[372,]	0.0305443415	3	29
[373,]	0.0306824337	18	19
[374,]	0.0307545726	7	32
[375,]	0.0308680302	34	12
[376,]	0.0309930084	12	34
[377,]	0.0310660590	34	12
[378,]	0.0311366402	11	15
[379,]	0.0312986374	15	11
[380,]	0.0313265824	11	15
[381,]	0.0315919583	18	33
[382,]	0.0316031844	25	35
[383,]	0.0317072850	12	14
[384,]	0.0317276851	14	12
[385,]	0.0317654391	37	14
[386,]	0.0317947398	14	37
[387,]	0.0318287335	37	14
[388,]	0.0318921579	14	37
[389,]	0.0318968843	37	14
[390,]	0.0319175497	14	37
[391,]	0.0319322077	37	14
[392,]	0.0320606525	14	37
[393,]	0.0320616003	37	14

```
[394,] 0.0321529897   14   37
[395,] 0.0321957564   37   14
[396,] 0.0321971256   14   37
[397,] 0.0322595744   37   14
[398,] 0.0323020836   14   37
[399,] 0.0323578709   14   37
[400,] 0.0324794103   37   14
[401,] 0.0326005557   14   37
[402,] 0.0326094890   37   14
[403,] 0.0326346505   10   13
[404,] 0.0326597593   13   10
[405,] 0.0329174360   10   13
[406,] 0.0329720093   13   10
[407,] 0.0329972551   10   13
[408,] 0.0330169603   13   10
[409,] 0.0330238512   10   13
[410,] 0.0330583756   13   10
[411,] 0.0330733820   10   13
[412,] 0.0331370859    8   32
[413,] 0.0331711939   18   19
[414,] 0.0332798044   25    5
[415,] 0.0333958345    5   25
[416,] 0.0334240531   20    3
[417,] 0.0337391377   20   31
[418,] 0.0338543561   20   33
[419,] 0.0338626343   37   14
[420,] 0.0338645847   14   37
[421,] 0.0339029649   35   18
[422,] 0.0339428986   25    6
[423,] 0.0341417510   25    5
[424,] 0.0341505481   10   13
[425,] 0.0343134147   13   10
[426,] 0.0343159159   10   13
[427,] 0.0343269222   13   10
[428,] 0.0343773847   10   11
[429,] 0.0344291065   11   10
[430,] 0.0345942633    9   23
[431,] 0.0346363482   27   17
[432,] 0.0348390638   17   13
[433,] 0.0350104540   13   17
[434,] 0.0350182481   17   13
[435,] 0.0350464416    3   16
[436,] 0.0351329717   36   13
[437,] 0.0353440306   13   36
[438,] 0.0353784161   36   13
[439,] 0.0354021733   13   36
[440,] 0.0354351543   36   13
[441,] 0.0354445556   13   36
[442,] 0.0355089797   36   13
[443,] 0.0356242282   13   36
[444,] 0.0357127140   36   13
[445,] 0.0357545637   13   36
[446,] 0.0357551721   36   13
[447,] 0.0358335781   13   36
```

```
[448,] 0.0358677486   36   13
[449,] 0.0358935560    7   32
[450,] 0.0359377202   32    7
[451,] 0.0361369654    7   14
[452,] 0.0363378537   36   13
[453,] 0.0365084886   13   36
[454,] 0.0365349292    9   28
[455,] 0.0370824937   12   14
[456,] 0.0373067522   14   12
[457,] 0.0373914736   12   14
[458,] 0.0374009077   14   12
[459,] 0.0374550551   12   14
[460,] 0.0375279317   14   12
[461,] 0.0376051262   12   14
[462,] 0.0376379543   14   12
[463,] 0.0376565397   12   13
[464,] 0.0377124981   33   13
[465,] 0.0377720269   13   33
[466,] 0.0378468068   33   13
[467,] 0.0383414810   13   33
[468,] 0.0384142692   33   13
[469,] 0.0385017871   10   33
[470,] 0.0387486634   13    7
[471,] 0.0390431414   32   20
[472,] 0.0391026534   20   32
[473,] 0.0392090664   32   20
[474,] 0.0392492569   20   32
[475,] 0.0393944110   32   20
[476,] 0.0394789064   20   32
[477,] 0.0395283224   32   20
[478,] 0.0396423750   32   23
[479,] 0.0399125590   23   32
[480,] 0.0399974886   20   32
[481,] 0.0400171775    1   13
attr(,"n")
[1] 37
```

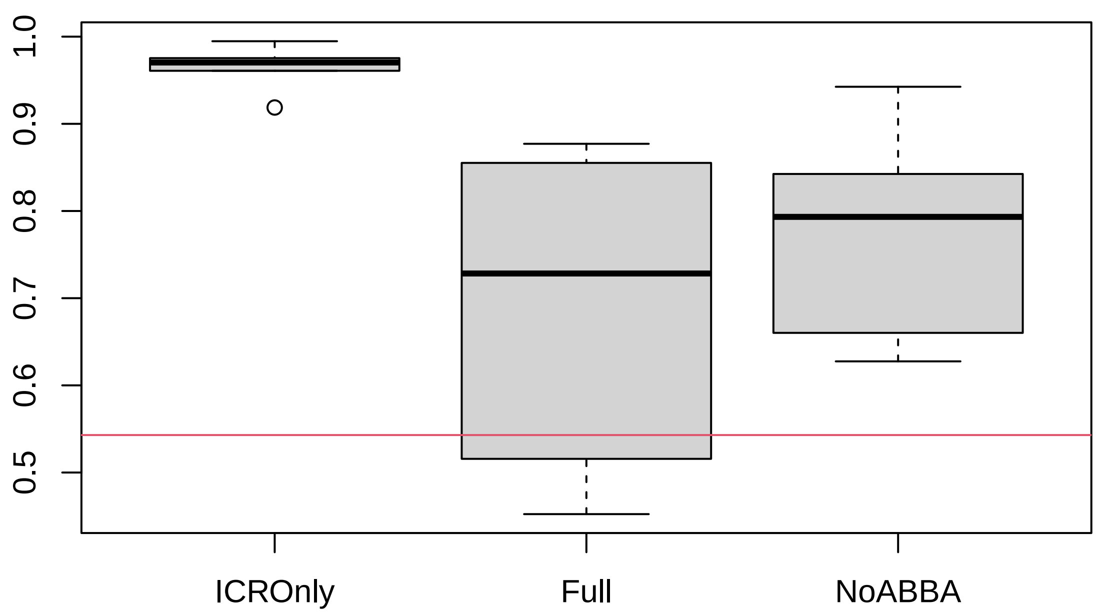
As we can see, we now have an event list that looks just like our original data (but that is synthetic). Such synthetic replicates can be used for many purposes, including exploratory simulation, model adequacy checking, and aiding in model interpretation. For instance, let's perform a very small simulation study to look at the relationship between occupying an ICR and betweenness, and probe the role of the AB-BA P-shift term in impacting that relationship. We'll do this by simulating data first from our ICR-only model, then our final model, and lastly a version of the final model with the P-shift term zeroed out. This is called an *in silico* "knock-out" experiment, and can be useful for understanding the role that specific effects play in generating aggregate outcomes.

```
set.seed(1331)
reps <- 6  #Number of replicate series to take
kocoef <- wtcfit6$coef  #Knock-out coefs
kocoef["PSAB-BA"] <- 0
ICRBetCor <- matrix(nrow = reps, ncol = 3)
for (i in 1:reps) {
   print(i)
   simwtc <- simulate(wtcfit1, covar = list(CovInt = WTCPoliceIsICR))  #ICR only
   ICRBetCor[i, 1] <- cor(betweenness(as.sociomatrix.eventlist(simwtc,
```

```
      37)), WTCPoliceIsICR)
   simwtc <- simulate(wtcfit6, covar = list(CovInt = WTCPoliceIsICR))  #Final
   ICRBetCor[i, 2] <- cor(betweenness(as.sociomatrix.eventlist(simwtc,
      37)), WTCPoliceIsICR)
   simwtc <- simulate(wtcfit6, covar = list(CovInt = WTCPoliceIsICR),
      coef = kocoef)  #Knockout
   ICRBetCor[i, 3] <- cor(betweenness(as.sociomatrix.eventlist(simwtc,
      37)), WTCPoliceIsICR)
}
```

```
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
[1] 6
```

```
boxplot(ICRBetCor, names = c("ICROnly", "Full", "NoABBA"))
abline(h = cor(betweenness(as.sociomatrix.eventlist(WTCPoliceCalls,
   37)), WTCPoliceIsICR), col = 2)
```



We can see here that (perhaps unsurprisingly) the ICR-only model overstates the relationship between occupying an ICR and having high betweenness; our full model does much better, generally producing realizations that cover the observed data (though, with only a few replicates, you may find that it sometimes doesn't!). What happens when we "turn off" the AB-BA shift? It turns out that this greatly increases the relative betweenness of ICRs, telling us that the AB-BA shifts are helping to play a role in keeping ICRs from inappropriately dominating the network. Why should turn taking matter here? The short answer is that turn-taking effects create opportunities for non-ICR responders to gain airtime, and end up as emergent coordinators. Taking out the AB-BA effect reduces emergent coordination, which in turn increases the relative centrality of the few individuals in institutionalized coordinative roles.

Section 2. Dyadic Relational Event Models with `rem.dyad`: Exact Timing

In the previous section, we considered dyadic relational event models in the case for which only ordinal timing information is available. We now proceed to the case of exact timing, in which we know the time at which each event occurs (relative to the onset of observation, which is treated as time 0).

2.0 The McFarland classroom data

For this section, we will make use of data collected by Dan McFarland (and published in Bender-deMoll and McFarland, 2006) on interaction among students and instructors within a high school classroom. (Note that the data employed here has been slightly modified from the original for illustrative purposes, in that small timing adjustments have been made to separate closely spaced events; those interested in using it for purposes other than practice are directed to the above paper in the Journal of Social Structure.) To see the event data itself, we may print it as follows:

```
head(Class)
```

```
  StartTime FromId ToId
1     0.135     14   12
2     0.270     12   14
3     0.405     18   12
4     0.540     12   18
5     0.675      1   12
6     0.810     12    1
```

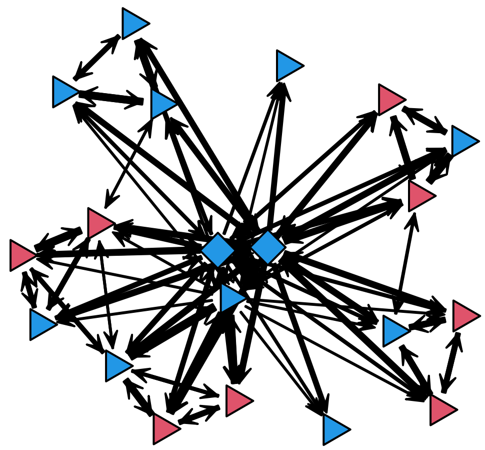
```
tail(Class)
```

```
    StartTime FromId ToId
687    50.426      1    3
688    50.547      3    1
689    50.668      6   17
690    50.789      6   17
691    50.910     17    6
692    50.920     NA   NA
```

As before, we have three columns: the event time, the event source (numbered from 1 to 20), and the event target (again, numbered 1 to 20). In this case, event time is given in increments of minutes from onset of observation. Note that the last row of the event list contains the time at which observation was terminated; it (and only it!) is allowed to contain `NA`s, since it has no meaning except to set the period during which events could have occurred. Where exact timing is used, the final entry in the edgelist is always interpreted in this way, and any source/target information on this row is ignored.

In addition to the `Class` edgelist, we also observe the covariates `ClassIsTeacher` (an indicator for instructor role) and `ClassIsFemale` (an indicator for gender). Visualizing the data in time-aggregate form gives us the following:

```
ClassNet <- as.sociomatrix.eventlist(Class, 20)
gplot(ClassNet, vertex.col = 4 - 2 * ClassIsFemale, vertex.sides = 3 +
   ClassIsTeacher, vertex.cex = 2, edge.lwd = ClassNet^0.75)
```



A dynamic visualization for this data is also available in the above-cited paper, and is well worth examining! (The `ndtv` package in `statnet` can be used to produce visualizations of this kind.)

2.1 Modeling with covariates

We begin our investigation of classroom dynamics with a trivial intercept model, containing only a vector of 1s (`ClassIntercept`) as a sending effect:

```
classfit1 <- rem.dyad(Class, n = 20, effects = c("CovSnd"), covar = list(CovSnd = ClassIntercept),
   ordinal = FALSE, hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(classfit1)
```

```
Relational Event Model (Temporal Likelihood)

          Estimate   Std.Err Z value  Pr(>|z|)
CovSnd.1 -3.332287  0.038042 -87.596 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 5987.221 on 691 degrees of freedom
Residual deviance: 5987.221 on 691 degrees of freedom
    Chi-square: -7.003109e-11 on 0 degrees of freedom, asymptotic p-value 1
```

```
AIC: 5989.221 AICC: 5989.227 BIC: 5993.759
```

Note that we must tell `rem.dyad` that we do not want to discard timing information (`ordinal=FALSE`). The model does not fit any better than the null because it is equivalent to the null model (but you must supply your own intercept, regardless!). As one would expect from first principles, this is really just an exponential waiting time model, calibrated to the observed communication rate:

```
(classfit1$m - 1)/max(Class[, 1])  #Events per minute (on average)
```

```
[1] 13.57031
```

```
20 * 19 * exp(classfit1$coef)  #Predicted events per minute (matches well!)
```

```
CovSnd.1
13.57031
```

To make things more interesting, let's add effects for role and gender:

```
classfit2 <- rem.dyad(Class, n = 20, effects = c("CovSnd", "CovRec"),
   covar = list(CovSnd = cbind(ClassIntercept, ClassIsTeacher,
      ClassIsFemale), CovRec = cbind(ClassIsTeacher, ClassIsFemale)),
   ordinal = FALSE, hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(classfit2)
```

```
Relational Event Model (Temporal Likelihood)

          Estimate   Std.Err  Z value Pr(>|z|)
CovSnd.1 -3.834231  0.078842 -48.6319  < 2e-16 ***
CovSnd.2  1.672565  0.091679  18.2437  < 2e-16 ***
CovSnd.3  0.123898  0.094931   1.3051  0.19185
CovRec.1  0.373733  0.127028   2.9421  0.00326 **
CovRec.2  0.165735  0.080896   2.0487  0.04049 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 5987.221 on 691 degrees of freedom
Residual deviance: 5652.318 on 687 degrees of freedom
    Chi-square: 334.9034 on 4 degrees of freedom, asymptotic p-value 0
AIC: 5662.318 AICC: 5662.405 BIC: 5685.008
```

```
classfit1$BIC - classfit2$BIC  #Model is preferred
```

```
[1] 308.7508
```

Note that covariate effects correspond to the order in which they were specified within the covar argument. It doesn't look here like gender affects propensity to send; given this, we might wonder whether dropping it gives us a better model.

```
classfit3 <- rem.dyad(Class, n = 20, effects = c("CovSnd", "CovRec"),
   covar = list(CovSnd = cbind(ClassIntercept, ClassIsTeacher),
      CovRec = cbind(ClassIsTeacher, ClassIsFemale)), ordinal = FALSE,
   hessian = TRUE)
```

```
Prepping edgelist.
```

```
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```r
summary(classfit3)
```

```
Relational Event Model (Temporal Likelihood)

          Estimate   Std.Err  Z value  Pr(>|z|)
CovSnd.1 -3.775225  0.063622 -59.3379 < 2.2e-16 ***
CovSnd.2  1.615761  0.079933  20.2139 < 2.2e-16 ***
CovRec.1  0.371753  0.127019   2.9267  0.003425 **
CovRec.2  0.161156  0.080815   1.9941  0.046138 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 5987.221 on 691 degrees of freedom
Residual deviance: 5654.016 on 688 degrees of freedom
    Chi-square: 333.2049 on 3 degrees of freedom, asymptotic p-value 0
AIC: 5662.016 AICC: 5662.074 BIC: 5680.169
```

```r
classfit2$BIC - classfit3$BIC  #Reduced model is indeed preferred
```

```
[1] 4.839661
```

2.2 Endogenous social dynamics

The above model is still relatively poor, in the sense that the reduction in deviance is unimpressive. What else might explain classroom communication? Recency effects would seem to be a reasonable bet:

```r
classfit4 <- rem.dyad(Class, n = 20, effects = c("CovSnd", "CovRec",
   "RRecSnd", "RSndSnd"), covar = list(CovSnd = cbind(ClassIntercept,
   ClassIsTeacher), CovRec = cbind(ClassIsTeacher, ClassIsFemale)),
   ordinal = FALSE, hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```r
summary(classfit4)
```

```
Relational Event Model (Temporal Likelihood)

          Estimate   Std.Err  Z value  Pr(>|z|)
RRecSnd   4.153291  0.119900  34.6397 < 2.2e-16 ***
RSndSnd  -1.399517  0.133149 -10.5109 < 2.2e-16 ***
CovSnd.1 -4.467620  0.075244 -59.3749 < 2.2e-16 ***
CovSnd.2  1.448491  0.080958  17.8919 < 2.2e-16 ***
CovRec.1 -1.364383  0.139346  -9.7913 < 2.2e-16 ***
CovRec.2  0.270105  0.083327   3.2415  0.001189 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 5987.221 on 691 degrees of freedom
Residual deviance: 4522.646 on 686 degrees of freedom
    Chi-square: 1464.575 on 5 degrees of freedom, asymptotic p-value 0
```

```
AIC: 4534.646 AICC: 4534.769 BIC: 4561.875
```

```
classfit3$BIC - classfit4$BIC  #Enhanced model is preferred
```

```
[1] 1118.294
```

This certainly helps, but we may suspect that more structure is present. Although a classroom is not as structured as a radio channel, we might reasonably expect to see at least modest adherence to conversational norms such as turn-taking. Moreover, sequential address and "hand-offs" might also be expected to occur more frequently here than would be expected by chance. To examine these possibilities, we incorporate the appropriate P-shift effects into our cumulative model:

```
classfit5 <- rem.dyad(Class, n = 20, effects = c("CovSnd", "CovRec",
   "RRecSnd", "RSndSnd", "PSAB-BA", "PSAB-AY", "PSAB-BY"), covar = list(CovSnd = cbind(ClassIntercept,
   ClassIsTeacher), CovRec = cbind(ClassIsTeacher, ClassIsFemale)),
   ordinal = FALSE, hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```
summary(classfit5)
```

```
Relational Event Model (Temporal Likelihood)

          Estimate   Std.Err  Z value  Pr(>|z|)
RRecSnd   2.429211  0.155365  15.6355 < 2.2e-16 ***
RSndSnd  -0.986714  0.144667  -6.8206 9.068e-12 ***
CovSnd.1 -5.003427  0.090609 -55.2203 < 2.2e-16 ***
CovSnd.2  1.253884  0.085160  14.7239 < 2.2e-16 ***
CovRec.1 -0.722692  0.141951  -5.0912 3.559e-07 ***
CovRec.2  0.047961  0.081325   0.5898    0.5554
PSAB-BA   4.622108  0.137600  33.5909 < 2.2e-16 ***
PSAB-BY   1.677543  0.164932  10.1711 < 2.2e-16 ***
PSAB-AY   2.869969  0.103113  27.8334 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 5987.221 on 691 degrees of freedom
Residual deviance: 2803.315 on 683 degrees of freedom
    Chi-square: 3183.906 on 8 degrees of freedom, asymptotic p-value 0
AIC: 2821.315 AICC: 2821.58 BIC: 2862.158
```

```
classfit4$BIC - classfit5$BIC  #Enhanced model is again preferred
```

```
[1] 1699.716
```

Note that, while P-shift effects are certainly present, including them has led the remaining gender effect to fall out. This suggests the possibility that what seemed at first to be a difference in communication receipt tendency by gender was in fact a result of social dynamics (perhaps stemming from the fact that the instructors are male, with their inherent tendency to communicate more often amplified by local conversational norms). Does dropping gender now result in improved model fit? Let's check.

```
classfit6 <- rem.dyad(Class, n = 20, effects = c("CovSnd", "CovRec",
   "RRecSnd", "RSndSnd", "PSAB-BA", "PSAB-AY", "PSAB-BY"), covar = list(CovSnd = cbind(ClassIntercept,
   ClassIsTeacher), CovRec = ClassIsTeacher), ordinal = FALSE,
   hessian = TRUE)
```

```
Prepping edgelist.
Checking/prepping covariates.
Computing preliminary statistics
Fitting model
Obtaining goodness-of-fit statistics
```

```r
summary(classfit6)
```

```
Relational Event Model (Temporal Likelihood)

          Estimate   Std.Err  Z value  Pr(>|z|)
RRecSnd   2.430694  0.155292  15.6524 < 2.2e-16 ***
RSndSnd  -0.984647  0.144654  -6.8069 9.972e-12 ***
CovSnd.1 -4.983905  0.084196 -59.1944 < 2.2e-16 ***
CovSnd.2  1.257296  0.084966  14.7976 < 2.2e-16 ***
CovRec.1 -0.745128  0.136612  -5.4543 4.915e-08 ***
PSAB-BA   4.623687  0.137503  33.6261 < 2.2e-16 ***
PSAB-BY   1.677815  0.164940  10.1723 < 2.2e-16 ***
PSAB-AY   2.870503  0.103103  27.8411 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Null deviance: 5987.221 on 691 degrees of freedom
Residual deviance: 2803.662 on 684 degrees of freedom
    Chi-square: 3183.558 on 7 degrees of freedom, asymptotic p-value 0
AIC: 2819.662 AICC: 2819.874 BIC: 2855.968
```

```r
classfit5$AICC - classfit6$AICC  #Reduced model is indeed preferred
```

```
[1] 1.705912
```

At this point, we have a relatively simple model that incorporates some plausible social mechanisms. We could continue to elaborate it, but for instructional purposes we stop our search here.

2.3 Using a fitted model to investigate event timing

One use of a fitted relational event model is to consider the inter-event times predicted to be observed under various scenarios. For this purpose, it is useful to remember that, under the piecewise constant hazard assumption, event waiting times are conditionally exponentially distributed. This allows us to easily work out the consequences of various model effects for social dynamics, at least within the context of a particular scenario.

In interpreting coefficient effects, recall that they act as logged hazard multipliers. For instance:

```r
exp(classfit6$coef["PSAB-BA"])  #Response events have apx 100 times the hazard of other events
```

```
 PSAB-BA
101.8689
```

Remember, however, that the fact that an event has an unusually high hazard does not mean that it will necessarily occur. For instance, while a response of B to a communication from A has a hazard that is (*ceteris paribus*) about 100 times as great as the hazard of a non B→A event, there are many more events of the latter type. Here, indeed, there are 379 other events "competing" with the B→A response, and thus the chance that the latter will occur next is smaller than it may appear. Both relative rates and combinatorics (i.e., the number of possible ways that an event type may occur) govern the result.

One basic use of the model coefficients is to examine the expected inter-event times under specific scenarios. E.g.:

```r
# Mean inter-event time if nothing else going on....
1/(20 * 19 * exp(classfit6$coef["CovSnd.1"]))
```

```
CovSnd.1
0.384325
```

```r
# Mean teacher-student time (again, if nothing else
# happened)
1/(2 * 18 * exp(sum(classfit6$coef[c("CovSnd.1", "CovSnd.2")])))
```

```
[1] 1.153833
```

```r
# Sequential address by teacher w/out prior interaction,
# given a prior teacher-student interaction, and assuming
# nothing else happened
1/(17 * exp(sum(classfit6$coef[c("CovSnd.1", "CovSnd.2", "PSAB-AY")])))
```

```
[1] 0.138469
```

```r
# Teacher responding to a specific student, given an
# immediate event
1/(exp(sum(classfit6$coef[c("CovSnd.1", "CovSnd.2", "PSAB-BA",
   "RRecSnd")])))
```

```
[1] 0.03587293
```

```r
# Student responding to a specific teacher, given an
# immediate event
1/(exp(sum(classfit6$coef[c("CovSnd.1", "CovRec.1", "PSAB-BA",
   "RRecSnd")])))
```

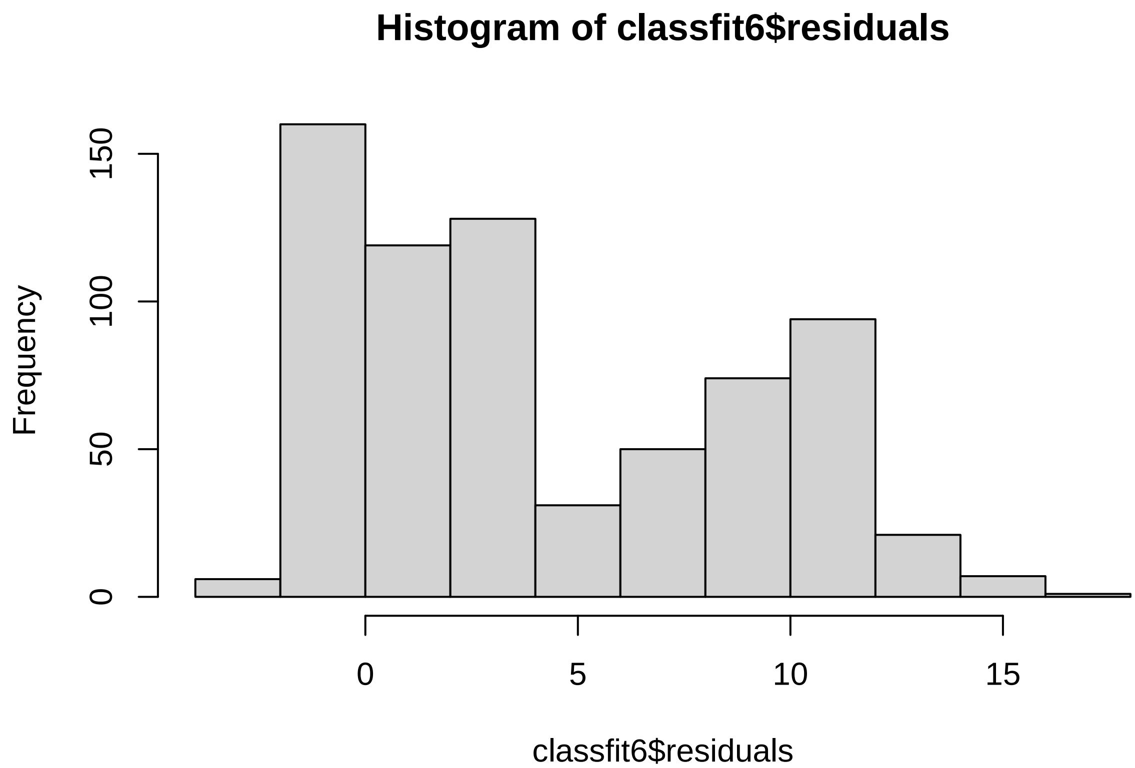
```
[1] 0.2657105
```

Again, the number of ways that an event type can occur and the propensity of such events to occur both matter!

2.4 Assessing model adequacy

Model adequacy assessment in the exact timing case is much like that of the ordinal case. We cannot here use a fixed null residual or guessing equivalent, but can still look at "surprise" based on deviance residuals:

```r
# Where is the model 'surprised'? Can't use null residual
# trick, but can see what the distribution looks like
hist(classfit6$residuals)  #Deviance residuals - lumpier by far, most smallish
```



The fit here doesn't seem to be as good as it was for the WTC police data. Let's look at classification:

```
mean(apply(classfit6$predicted.match, 1, all))  #Exactly right about 33%
```

```
[1] 0.3299566
```

```
mean(apply(classfit6$predicted.match, 1, any))  #Get one party exactly right 52%
```

```
[1] 0.5166425
```

```
colMeans(classfit6$predicted.match)  #Better at sender than receiver!
```

```
   FromId      ToId
0.5050651 0.3415340
```

```
classfit6$observed.rank
```

```
  [1]   1   1  58   1  77   1  39   1   3   4   4   4   4   4  19   4   4   4
 [19]   4   3   3   4   4   1   3   4   4  54   3   3   3   3   3   3   3   3
 [37]   3   3   3   2   2   3   3   3   3   3  92   1   5   1   4   1   4   1
 [55]   4   1  40   1  65   6  11   9  20   7   8   4  10  99   1   1   1  59
 [73]   3   3  14  14   3   3   3  12   3  10   3   2   2   3   3   3   3   3
 [91]  23   1   3   4   4   4   4   4   4   4   4   4   4   4   3   3   4   1
[109]   3   4   4 374   1 110   1  59   4   4   4   4   4   4   4   4   4   4
[127]   4   3   3   4   1   3   4   4 106   1 123   1  60   4   4   4   4   4
[145]   4   4   4   4   4   4   3   3   4   1   3   4   4 111   8   8   5   7
[163]   2  10   8   6   7   3   7   7  14   8   8   8   8   8 122   1 125   1
[181] 115   1 130   1  86   2  10   1  29   1  47   1 115   1 110   1  48   1
[199]   1   1  61   1  36   1   8   1   8   1 130   1   1   1 133   1  50   1
```

```
[217]  89   1  98   2   4   2  33   1 119   1  94   1  51   1 122   1   2   1
[235]   2   1  50   1 117   1  51   1   3   1 131   1  88   1  27   1  32   1
[253]  74   1  53   1 119   1 121   1  52   1  52   1  91   1 128   1  54   1
[271]  92   1   2   1  94   1  92   1 120   1 128   1  53   1  52   1  25   1
[289]  24   1  94   1  92   1  52   1  92   1  52   1 113   1  70   9   9   8
[307]   8   3   8   9   9   1   8   9  19   4   4   4   8   9   6  55  10  10
[325]   4   4  10   2  10   7   3   9   6  10  11  11  10   9   9   9 113   1
[343] 129   1 368   1 120   1 371   1 110   1  24   1  51   1   9   2  10   2
[361]  49   1 119   1 123   1 108   1 120   1 118   1 129   1   4   1  55   1
[379]  94  89   1  13   1  32   1  74   1   4   1  58   1 111   1 103  92   1
[397]   4   1  97   1  58   1  56   1 128   1   1   1  19   1   2   2   4   2
[415] 379   1 369   1  92   1 127   1  52   1 112   1  79   1  32   1  30   1
[433]  72   1  53   1   2   1 133   1  92   1 113   1  52   1  53   1  54   1
[451]  54   1  73   1  31   1  34   1  70   1 138   1 116   1 127   1  55   1
[469]   3   1   2   1  91   1  55   1  91   1  83   1  15   1  35   1  70   1
[487] 116   1  54   1  89   1 114   1   3   1 114   1  83   1  17   1  21   1
[505]  20   1  64   1  12   2  20   1  98   1  53   1  53   1  44   2   8   2
[523]   9   2  66   1 114   1  86   1  88   1 109   1 112   1  78   2  19   1
[541]  19   1  47   1  88   1 110   1  89   1 119   1  77   2  19   2  19   2
[559]  94   1   2   1 112  52   1 121   1   2   1 134   1  87   1 125   1  53
[577]   1  72   1  18   1  17   1  18   1   5   1  53   1 125 123   1  90   1
[595] 111   1  77   2  19   1  20   2  48   1  86   1  54   1  88   1 112   1
[613]   1   1  89   1 114   1 132   2   1  80   1  14   2  19   2  72   1  89
[631]   1  53   1  54   1  54   1  89   1   3   1  45   2   8   2   8   2  10
[649]   2 106   1 122  52   1  91   1  53   1 115   1  53   1  91   1  47   2
[667]   2  17  17  13  12   6  13   5   9   5  10  10  19   2   8   7   4  15
[685]   8  59  22   1 142 380   1
```

```
cbind(Class, c(classfit6$observed.rank, NA))
```

```
    StartTime FromId ToId c(classfit6$observed.rank, NA)
1       0.135     14   12                              1
2       0.270     12   14                              1
3       0.405     18   12                             58
4       0.540     12   18                              1
5       0.675      1   12                             77
6       0.810     12    1                              1
7       0.945     14   17                             39
8       1.080     17   14                              1
9       1.257     14    1                              3
10      1.267     14    2                              4
11      1.277     14    3                              4
12      1.287     14    4                              4
13      1.297     14    5                              4
14      1.307     14    6                              4
15      1.317     14    7                             19
16      1.327     14    8                              4
17      1.337     14    9                              4
18      1.347     14   10                              4
19      1.357     14   11                              4
20      1.367     14   12                              3
21      1.377     14   13                              3
22      1.387     14   15                              4
23      1.397     14   16                              4
24      1.407     14   17                              1
```

25	1.417	14	18	3
26	1.427	14	19	4
27	1.437	14	20	4
28	1.613	7	1	54
29	1.623	7	2	3
30	1.633	7	3	3
31	1.643	7	4	3
32	1.653	7	5	3
33	1.663	7	6	3
34	1.673	7	8	3
35	1.683	7	9	3
36	1.693	7	10	3
37	1.703	7	11	3
38	1.713	7	12	3
39	1.723	7	13	3
40	1.733	7	14	2
41	1.743	7	15	2
42	1.753	7	16	3
43	1.763	7	17	3
44	1.773	7	18	3
45	1.783	7	19	3
46	1.793	7	20	3
47	1.970	4	12	92
48	2.147	12	4	1
49	2.323	12	10	5
50	2.500	10	12	1
51	2.677	10	4	4
52	2.853	4	10	1
53	3.030	4	5	4
54	3.207	5	4	1
55	3.383	5	10	4
56	3.560	10	5	1
57	3.737	5	12	40
58	3.913	12	5	1
59	4.090	7	4	65
60	4.267	7	5	6
61	4.443	7	12	11
62	4.620	7	10	9
63	4.797	14	7	20
64	4.973	14	4	7
65	5.150	14	5	8
66	5.327	14	12	4
67	5.503	14	10	10
68	5.680	16	17	99
69	5.857	17	16	1
70	6.033	16	17	1
71	6.210	17	16	1
72	6.387	7	1	59
73	6.397	7	2	3
74	6.407	7	3	3
75	6.417	7	4	14
76	6.427	7	5	14
77	6.437	7	6	3
78	6.447	7	8	3

79	6.457	7	9	3
80	6.467	7	10	12
81	6.477	7	11	3
82	6.487	7	12	10
83	6.497	7	13	3
84	6.507	7	14	2
85	6.517	7	15	2
86	6.527	7	16	3
87	6.537	7	17	3
88	6.547	7	18	3
89	6.557	7	19	3
90	6.567	7	20	3
91	6.743	17	7	23
92	6.920	7	17	1
93	7.037	7	1	3
94	7.047	7	2	4
95	7.057	7	3	4
96	7.067	7	4	4
97	7.077	7	5	4
98	7.087	7	6	4
99	7.097	7	8	4
100	7.107	7	9	4
101	7.117	7	10	4
102	7.127	7	11	4
103	7.137	7	12	4
104	7.147	7	13	4
105	7.157	7	14	3
106	7.167	7	15	3
107	7.177	7	16	4
108	7.187	7	17	1
109	7.197	7	18	3
110	7.207	7	19	4
111	7.217	7	20	4
112	7.334	10	5	374
113	7.451	5	10	1
114	7.569	4	12	110
115	7.686	12	4	1
116	7.803	7	1	59
117	7.813	7	2	4
118	7.823	7	3	4
119	7.833	7	4	4
120	7.843	7	5	4
121	7.853	7	6	4
122	7.863	7	8	4
123	7.873	7	9	4
124	7.883	7	10	4
125	7.893	7	11	4
126	7.903	7	12	4
127	7.913	7	13	4
128	7.923	7	14	3
129	7.933	7	15	3
130	7.943	7	16	4
131	7.953	7	17	1
132	7.963	7	18	3

133	7.973	7	19	4
134	7.983	7	20	4
135	8.100	18	1	106
136	8.217	1	18	1
137	8.334	20	17	123
138	8.451	17	20	1
139	8.569	7	1	60
140	8.579	7	2	4
141	8.589	7	3	4
142	8.599	7	4	4
143	8.609	7	5	4
144	8.619	7	6	4
145	8.629	7	8	4
146	8.639	7	9	4
147	8.649	7	10	4
148	8.659	7	11	4
149	8.669	7	12	4
150	8.679	7	13	4
151	8.689	7	14	3
152	8.699	7	15	3
153	8.709	7	16	4
154	8.719	7	17	1
155	8.729	7	18	3
156	8.739	7	19	4
157	8.749	7	20	4
158	8.866	4	1	111
159	8.876	4	2	8
160	8.886	4	3	8
161	8.896	4	5	5
162	8.906	4	6	7
163	8.916	4	7	2
164	8.926	4	8	10
165	8.936	4	9	8
166	8.946	4	10	6
167	8.956	4	11	7
168	8.966	4	12	3
169	8.976	4	13	7
170	8.986	4	14	7
171	8.996	4	15	14
172	9.006	4	16	8
173	9.016	4	17	8
174	9.026	4	18	8
175	9.036	4	19	8
176	9.046	4	20	8
177	9.163	16	20	122
178	9.280	20	16	1
179	9.397	9	18	125
180	9.514	18	9	1
181	9.631	20	17	115
182	9.749	17	20	1
183	9.866	13	3	130
184	9.983	3	13	1
185	10.100	14	18	86
186	10.217	18	14	2

187	10.334	14	1	10
188	10.451	1	14	1
189	10.569	14	9	29
190	10.686	9	14	1
191	10.803	10	4	47
192	10.920	4	10	1
193	11.037	18	1	115
194	11.154	1	18	1
195	11.271	4	5	110
196	11.389	5	4	1
197	11.506	18	1	48
198	11.623	1	18	1
199	11.740	18	1	1
200	11.857	1	18	1
201	11.974	14	12	61
202	12.091	12	14	1
203	12.209	14	5	36
204	12.326	5	14	1
205	12.443	14	4	8
206	12.560	4	14	1
207	12.677	4	12	8
208	12.794	12	4	1
209	12.911	11	15	130
210	13.029	15	11	1
211	13.146	11	15	1
212	13.263	15	11	1
213	13.380	8	13	133
214	13.497	13	8	1
215	13.614	20	17	50
216	13.731	17	20	1
217	13.849	14	10	89
218	13.966	10	14	1
219	14.083	7	20	98
220	14.200	20	7	2
221	14.317	7	17	4
222	14.434	17	7	2
223	14.551	7	16	33
224	14.669	16	7	1
225	14.786	1	9	119
226	14.903	9	1	1
227	15.020	13	3	94
228	15.137	3	13	1
229	15.254	11	15	51
230	15.371	15	11	1
231	15.489	12	10	122
232	15.606	10	12	1
233	15.723	10	4	2
234	15.840	4	10	1
235	15.957	4	12	2
236	16.074	12	4	1
237	16.191	10	4	50
238	16.309	4	10	1
239	16.426	17	16	117
240	16.543	16	17	1

241	16.660	1	9	51
242	16.777	9	1	1
243	16.894	9	18	3
244	17.011	18	9	1
245	17.129	3	8	131
246	17.246	8	3	1
247	17.363	7	15	88
248	17.480	15	7	1
249	17.597	7	6	27
250	17.714	6	7	1
251	17.831	7	11	32
252	17.949	11	7	1
253	18.066	12	10	74
254	18.183	10	12	1
255	18.300	17	16	53
256	18.417	16	17	1
257	18.534	4	5	119
258	18.651	5	4	1
259	18.769	12	5	121
260	18.886	5	12	1
261	19.003	4	5	52
262	19.120	5	4	1
263	19.237	13	3	52
264	19.354	3	13	1
265	19.471	18	1	91
266	19.589	1	18	1
267	19.706	6	11	128
268	19.823	11	6	1
269	19.940	13	3	54
270	20.057	3	13	1
271	20.174	10	4	92
272	20.291	4	10	1
273	20.409	4	5	2
274	20.526	5	4	1
275	20.643	3	8	94
276	20.760	8	3	1
277	20.877	20	17	92
278	20.994	17	20	1
279	21.111	5	10	120
280	21.229	10	5	1
281	21.346	15	6	128
282	21.463	6	15	1
283	21.580	12	5	53
284	21.697	5	12	1
285	21.814	4	5	52
286	21.931	5	4	1
287	22.049	4	12	25
288	22.166	12	4	1
289	22.283	4	5	24
290	22.400	5	4	1
291	22.517	8	13	94
292	22.634	13	8	1
293	22.751	17	16	92
294	22.869	16	17	1

295	22.986	3	8	52
296	23.103	8	3	1
297	23.220	4	12	92
298	23.337	12	4	1
299	23.454	3	8	52
300	23.571	8	3	1
301	23.689	5	10	113
302	23.806	10	5	1
303	23.923	7	1	70
304	23.933	7	2	9
305	23.943	7	3	9
306	23.953	7	4	8
307	23.963	7	5	8
308	23.973	7	6	3
309	23.983	7	8	8
310	23.993	7	9	9
311	24.003	7	10	9
312	24.013	7	11	1
313	24.023	7	12	8
314	24.033	7	13	9
315	24.043	7	14	19
316	24.053	7	15	4
317	24.063	7	16	4
318	24.073	7	17	4
319	24.083	7	18	8
320	24.093	7	19	9
321	24.103	7	20	6
322	24.220	14	1	55
323	24.230	14	2	10
324	24.240	14	3	10
325	24.250	14	4	4
326	24.260	14	5	4
327	24.270	14	6	10
328	24.280	14	7	2
329	24.290	14	8	10
330	24.300	14	9	7
331	24.310	14	10	3
332	24.320	14	11	9
333	24.330	14	12	6
334	24.340	14	13	10
335	24.350	14	15	11
336	24.360	14	16	11
337	24.370	14	17	10
338	24.380	14	18	9
339	24.390	14	19	9
340	24.400	14	20	9
341	24.480	10	4	113
342	24.560	4	10	1
343	24.639	8	13	129
344	24.719	13	8	1
345	24.799	18	1	368
346	24.879	1	18	1
347	24.958	12	18	120
348	25.038	18	12	1

349	25.118	3	8	371
350	25.198	8	3	1
351	25.277	20	16	110
352	25.357	16	20	1
353	25.437	20	4	24
354	25.517	4	20	1
355	25.597	14	9	51
356	25.676	9	14	1
357	25.756	14	1	9
358	25.836	1	14	2
359	25.916	14	18	10
360	25.995	18	14	2
361	26.075	10	4	49
362	26.155	4	10	1
363	26.235	1	9	119
364	26.314	9	1	1
365	26.394	4	12	123
366	26.474	12	4	1
367	26.554	18	1	108
368	26.633	1	18	1
369	26.713	12	10	120
370	26.793	10	12	1
371	26.873	9	18	118
372	26.953	18	9	1
373	27.032	16	17	129
374	27.112	17	16	1
375	27.192	17	20	4
376	27.272	20	17	1
377	27.351	3	8	55
378	27.431	8	3	1
379	27.511	20	4	94
380	27.591	7	18	89
381	27.670	18	7	1
382	27.750	7	1	13
383	27.830	1	7	1
384	27.910	7	9	32
385	27.990	9	7	1
386	28.069	8	13	74
387	28.149	13	8	1
388	28.229	13	3	4
389	28.309	3	13	1
390	28.388	16	17	58
391	28.468	17	16	1
392	28.548	12	18	111
393	28.628	18	12	1
394	28.707	17	4	103
395	28.787	9	18	92
396	28.867	18	9	1
397	28.947	18	1	4
398	29.027	1	18	1
399	29.106	10	4	97
400	29.186	4	10	1
401	29.266	18	1	58
402	29.346	1	18	1

403	29.425	13	3	56
404	29.505	3	13	1
405	29.585	6	11	128
406	29.665	11	6	1
407	29.744	6	11	1
408	29.824	11	6	1
409	29.904	14	18	19
410	29.984	18	14	1
411	30.063	14	1	2
412	30.143	1	14	2
413	30.223	14	9	4
414	30.303	9	14	2
415	30.383	17	4	379
416	30.462	4	17	1
417	30.542	15	6	369
418	30.622	6	15	1
419	30.702	12	10	92
420	30.781	10	12	1
421	30.861	4	12	127
422	30.941	12	4	1
423	31.021	13	3	52
424	31.100	3	13	1
425	31.180	20	16	112
426	31.260	16	20	1
427	31.340	14	11	79
428	31.420	11	14	1
429	31.499	14	15	32
430	31.579	15	14	1
431	31.659	14	6	30
432	31.739	6	14	1
433	31.818	15	6	72
434	31.898	6	15	1
435	31.978	4	12	53
436	32.058	12	4	1
437	32.137	12	10	2
438	32.217	10	12	1
439	32.297	5	4	133
440	32.377	4	5	1
441	32.457	3	8	92
442	32.536	8	3	1
443	32.616	17	20	113
444	32.696	20	17	1
445	32.776	15	6	52
446	32.855	6	15	1
447	32.935	3	8	53
448	33.015	8	3	1
449	33.095	13	3	54
450	33.174	3	13	1
451	33.254	17	20	54
452	33.334	20	17	1
453	33.414	7	3	73
454	33.493	3	7	1
455	33.573	7	8	31
456	33.653	8	7	1

457	33.733	7	13	34
458	33.813	13	7	1
459	33.892	16	17	70
460	33.972	17	16	1
461	34.052	5	10	138
462	34.132	10	5	1
463	34.211	6	11	116
464	34.291	11	6	1
465	34.371	1	9	127
466	34.451	9	1	1
467	34.530	6	11	55
468	34.610	11	6	1
469	34.690	11	15	3
470	34.770	15	11	1
471	34.850	15	6	2
472	34.929	6	15	1
473	35.009	20	16	91
474	35.089	16	20	1
475	35.169	11	15	55
476	35.248	15	11	1
477	35.328	5	4	91
478	35.408	4	5	1
479	35.488	14	16	83
480	35.567	16	14	1
481	35.647	14	17	15
482	35.727	17	14	1
483	35.807	14	20	35
484	35.887	20	14	1
485	35.966	5	10	70
486	36.046	10	5	1
487	36.126	16	17	116
488	36.206	17	16	1
489	36.285	12	10	54
490	36.365	10	12	1
491	36.445	20	16	89
492	36.525	16	20	1
493	36.604	10	4	114
494	36.684	4	10	1
495	36.764	4	12	3
496	36.844	12	4	1
497	36.923	17	20	114
498	37.003	20	17	1
499	37.083	7	12	83
500	37.163	12	7	1
501	37.243	7	4	17
502	37.322	4	7	1
503	37.402	7	5	21
504	37.482	5	7	1
505	37.562	7	10	20
506	37.641	10	7	1
507	37.721	14	13	64
508	37.801	13	14	1
509	37.881	14	1	12
510	37.960	1	14	2

511	38.040	14	3	20
512	38.120	3	14	1
513	38.196	4	10	98
514	38.272	10	4	1
515	38.347	10	4	53
516	38.423	4	10	1
517	38.499	11	15	53
518	38.575	15	11	1
519	38.651	14	20	44
520	38.726	20	14	2
521	38.802	14	17	8
522	38.878	17	14	2
523	38.954	14	16	9
524	39.029	16	14	2
525	39.105	5	10	66
526	39.181	10	5	1
527	39.257	13	3	114
528	39.333	3	13	1
529	39.408	17	20	86
530	39.484	20	17	1
531	39.560	6	11	88
532	39.636	11	6	1
533	39.712	4	12	109
534	39.787	12	4	1
535	39.863	9	18	112
536	39.939	18	9	1
537	40.015	7	16	78
538	40.091	16	7	2
539	40.166	7	20	19
540	40.242	20	7	1
541	40.318	7	17	19
542	40.394	17	7	1
543	40.469	6	11	47
544	40.545	11	6	1
545	40.621	1	9	88
546	40.697	9	1	1
547	40.773	5	4	110
548	40.848	4	5	1
549	40.924	15	6	89
550	41.000	6	15	1
551	41.076	3	8	119
552	41.152	8	3	1
553	41.227	7	11	77
554	41.303	11	7	2
555	41.379	7	15	19
556	41.455	15	7	2
557	41.531	7	6	19
558	41.606	6	7	2
559	41.682	18	1	94
560	41.758	1	18	1
561	41.834	1	9	2
562	41.909	9	1	1
563	41.985	12	10	112
564	42.061	4	5	52

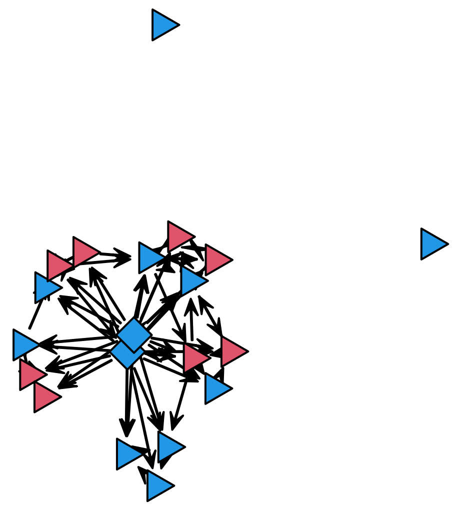
565	42.137	5	4	1
566	42.213	12	18	121
567	42.288	18	12	1
568	42.364	18	1	2
569	42.440	1	18	1
570	42.516	12	10	134
571	42.592	10	12	1
572	42.667	4	12	87
573	42.743	12	4	1
574	42.819	20	16	125
575	42.895	16	20	1
576	42.971	18	1	53
577	43.046	1	18	1
578	43.122	14	4	72
579	43.198	4	14	1
580	43.274	14	10	18
581	43.349	10	14	1
582	43.425	14	5	17
583	43.501	5	14	1
584	43.577	14	12	18
585	43.653	12	14	1
586	43.728	12	10	5
587	43.804	10	12	1
588	43.880	20	16	53
589	43.956	16	20	1
590	44.032	4	10	125
591	44.107	16	17	123
592	44.183	17	16	1
593	44.259	15	6	90
594	44.335	6	15	1
595	44.411	17	20	111
596	44.486	20	17	1
597	44.562	7	9	77
598	44.638	9	7	2
599	44.714	7	18	19
600	44.789	18	7	1
601	44.865	7	1	20
602	44.941	1	7	2
603	45.017	16	17	48
604	45.093	17	16	1
605	45.168	4	12	86
606	45.244	12	4	1
607	45.320	13	3	54
608	45.396	3	13	1
609	45.472	20	16	88
610	45.547	16	20	1
611	45.623	6	11	112
612	45.699	11	6	1
613	45.775	6	11	1
614	45.851	11	6	1
615	45.926	12	10	89
616	46.002	10	12	1
617	46.078	9	18	114
618	46.154	18	9	1

619	46.229	4	10	132
620	46.305	4	12	2
621	46.381	12	4	1
622	46.457	14	18	80
623	46.533	18	14	1
624	46.608	14	1	14
625	46.684	1	14	2
626	46.760	14	9	19
627	46.836	9	14	2
628	46.912	3	8	72
629	46.987	8	3	1
630	47.063	16	17	89
631	47.139	17	16	1
632	47.215	13	3	53
633	47.291	3	13	1
634	47.366	15	6	54
635	47.442	6	15	1
636	47.518	16	17	54
637	47.594	17	16	1
638	47.669	3	8	89
639	47.745	8	3	1
640	47.821	8	13	3
641	47.897	13	8	1
642	47.973	14	12	45
643	48.048	12	14	2
644	48.124	14	5	8
645	48.200	5	14	2
646	48.276	14	10	8
647	48.352	10	14	2
648	48.427	14	4	10
649	48.503	4	14	2
650	48.579	18	1	106
651	48.655	1	18	1
652	48.731	12	18	122
653	48.806	3	8	52
654	48.882	8	3	1
655	48.958	13	3	91
656	49.034	3	13	1
657	49.109	15	6	53
658	49.185	6	15	1
659	49.261	11	15	115
660	49.337	15	11	1
661	49.413	20	16	53
662	49.488	16	20	1
663	49.564	8	13	91
664	49.640	13	8	1
665	49.761	7	15	47
666	49.882	15	7	2
667	50.003	7	1	2
668	50.013	7	2	17
669	50.023	7	3	17
670	50.033	7	4	13
671	50.043	7	5	12
672	50.053	7	6	6

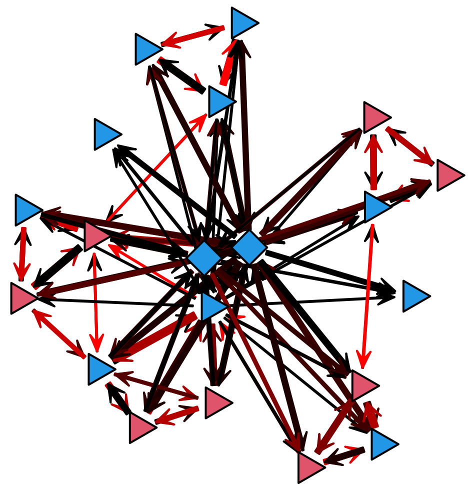
673	50.063	7	8	13
674	50.073	7	9	5
675	50.083	7	10	9
676	50.093	7	11	5
677	50.103	7	12	10
678	50.113	7	13	10
679	50.123	7	14	19
680	50.133	7	15	2
681	50.143	7	16	8
682	50.153	7	17	7
683	50.163	7	18	4
684	50.173	7	19	15
685	50.183	7	20	8
686	50.304	1	7	59
687	50.426	1	3	22
688	50.547	3	1	1
689	50.668	6	17	142
690	50.789	6	17	380
691	50.910	17	6	1
692	50.920	NA	NA	NA

It looks like there is some structure in the errors: we aren't able to capture certain kinds of intrusive events. Does looking at the "surprising" events (say, those for which the observed event is not in the top 5% of those predicted) in time-aggregate form help?

```
# Get the surprising events, and display as a network
surprising <- as.sociomatrix.eventlist(Class[classfit6$observed.rank >
   19, ], 20)
gplot(surprising, vertex.col = 4 - 2 * ClassIsFemale, vertex.sides = 3 +
   ClassIsTeacher, vertex.cex = 2)
```



```
# Show how the 'surprising' events fit into the broader
# communication structure
edgecol <- matrix(rgb(surprising/(ClassNet + 0.01), 0, 0), 20,
   20)  #Color me surprised
gplot(ClassNet, edge.col = edgecol, edge.lwd = ClassNet^0.75,
   vertex.col = 4 - 2 * ClassIsFemale, vertex.sides = 3 + ClassIsTeacher,
   vertex.cex = 2)
```



The visualization gives us more of a clue about what we're missing: various side discussions occur that are not well-captured by the current model. This could be due to the fact that things like P-shift effects fail to capture simultaneous side conversations (each of which may have its own set of turn-taking patterns), or to a lack of covariates to capture the enhanced propensity of subgroup members to address each other. Further elaboration could be helpful here. On the other hand, we seem to be doing reasonably well at capturing the main line of discussion within the classroom, particularly vis a vis the instructors. Whether or not this is adequate depends on the purpose to which the model is to be put; as always, adequacy must be considered in light of specific scientific goals.

2.5 Simulating from the fitted model

Simulation from fitted models with exact timing proceeds exactly as in the ordinal timing case: we can use the `simulate` method for `rem.dyad` to generate trajectories from the fitted model object.

For instance, to generate a new trajectory from the final classroom model, we would use the code

```
set.seed(1331)
ClassSim <- simulate(classfit6, covar = list(CovSnd = cbind(ClassIntercept,
   ClassIsTeacher), CovRec = ClassIsTeacher))

ClassSim  #Examine the resulting trajectory
```

```
             [,1] [,2] [,3]
  [1,]  0.2452372    7    1
  [2,]  0.2648195    1    7
  [3,]  0.2823672    7   18
  [4,]  0.3090297   13   19
  [5,]  0.3387736    7    2
```

```
 [6,] 0.4176939  15  19
 [7,] 0.4333086  14  16
 [8,] 0.4465340  14  18
 [9,] 0.5291581  14   9
[10,] 0.6368619  14  10
[11,] 0.7403585  10  14
[12,] 0.8529512  14  18
[13,] 0.8781311  14  10
[14,] 0.9604370  14  12
[15,] 0.9896636  16  14
[16,] 0.9930088  14  16
[17,] 1.1128288   7   6
[18,] 1.1441737  14   7
[19,] 1.1651333  14   8
[20,] 1.1939697   8  14
[21,] 1.2091901  14   6
[22,] 1.2305970  14  17
[23,] 1.3323752  14  10
[24,] 1.3605140  10   5
[25,] 1.4054487   5  10
[26,] 1.4482442  10   5
[27,] 1.5010568  14   8
[28,] 1.5663818  14  16
[29,] 1.8808446  14  13
[30,] 2.0099539  14  19
[31,] 2.0260119  14  15
[32,] 2.1178422  14  18
[33,] 2.1972077  14  10
[34,] 2.2516703   7  14
[35,] 2.2757559   7  18
[36,] 2.2767436  17  19
[37,] 2.3050881  17  14
[38,] 2.3331853   7  20
[39,] 2.3480431   7  15
[40,] 2.4195393  15   7
[41,] 2.4486735   7  15
[42,] 2.5661399   7  19
[43,] 2.6034453   7  14
[44,] 2.6443936  14   7
[45,] 2.7902113   7  14
[46,] 2.8476409   7   4
[47,] 2.8892588   7   3
[48,] 2.8958807  13  16
[49,] 2.9141100  16  13
[50,] 3.2577642  13  16
[51,] 3.3103058  18   6
[52,] 3.6363654  17  20
[53,] 3.6689292  20  17
[54,] 3.6927953  20  10
[55,] 3.7176316  10  20
[56,] 3.8541570  20  10
[57,] 3.8883449  20   6
[58,] 3.9300268   6  20
[59,] 4.1380025  20   6
```

```
 [60,] 4.2587120  2  6
 [61,] 4.2860588  2  1
 [62,] 4.3804901 16  9
 [63,] 4.4430955 16 12
 [64,] 4.5267698 12 16
 [65,] 4.5634829 16 12
 [66,] 4.6236388 16  8
 [67,] 4.6471144  8 16
 [68,] 4.7473854 10 11
 [69,] 4.7938856 11 10
 [70,] 4.7961302  2 13
 [71,] 4.8091500 13  2
 [72,] 4.8247462 14 12
 [73,] 4.8955373 14 17
 [74,] 4.9081383 14  9
 [75,] 4.9153592 14 17
 [76,] 4.9805986 14  7
 [77,] 5.0241813 14  4
 [78,] 5.0613228  4 14
 [79,] 5.0644384 14  4
 [80,] 5.2103329 14  3
 [81,] 5.2849561 14  4
 [82,] 5.3802349 14 19
 [83,] 5.4722991 14  8
 [84,] 5.4730600 14  4
 [85,] 5.4855860  4 14
 [86,] 5.5160426 14  4
 [87,] 5.5235733  4 14
 [88,] 5.5503236  4 18
 [89,] 5.6166152 14 11
 [90,] 5.6329490 11 14
 [91,] 5.6923629 11  8
 [92,] 5.6941925  8 11
 [93,] 5.7328254 11  8
 [94,] 6.0435956 19 20
 [95,] 6.1846005  3 12
 [96,] 6.3465232  3 10
 [97,] 6.3948629 10  3
 [98,] 6.4390538 11 19
 [99,] 6.5775315 11  8
[100,] 6.7374124 11 14
[101,] 6.7471360 14 11
[102,] 6.9234757 14  2
[103,] 6.9320873 14 11
[104,] 7.0017608 14  7
[105,] 7.0218169 14  4
[106,] 7.1118122 10  2
[107,] 7.2821661  2 10
[108,] 7.2954076  2  5
[109,] 7.4197045  2 13
[110,] 7.5067671 19 13
[111,] 7.5148894 13 19
[112,] 7.5259201 13 17
[113,] 7.5536243 17 13
```

```
[114,]  7.6412278  14   9
[115,]  7.6970344  14  11
[116,]  7.7208712  14   4
[117,]  7.7972453  14  17
[118,]  7.8557040  14  11
[119,]  7.8695206  14  13
[120,]  7.8987646   4   8
[121,]  7.9182257  20  15
[122,]  7.9187713  20  19
[123,]  7.9489062  19  20
[124,]  8.1251780  20  19
[125,]  8.1252785  12   1
[126,]  8.1870581   7  14
[127,]  8.2255085   6  13
[128,]  8.2850134  13   6
[129,]  8.3191157  13   1
[130,]  8.3472196  13  11
[131,]  8.4147432  11  13
[132,]  8.5221297  13  15
[133,]  8.5532070  15  13
[134,]  8.6038985   9   2
[135,]  8.6043392   2   9
[136,]  8.6103288  15  13
[137,]  9.1065400  10  11
[138,]  9.1889692  11   9
[139,]  9.1972623  10  20
[140,]  9.2938692  14   3
[141,]  9.4185938   3  14
[142,]  9.4677391  14   3
[143,]  9.5318032  11  10
[144,]  9.6027228   8   4
[145,]  9.6083322  14  10
[146,]  9.6510630  10  14
[147,]  9.6584675  14  10
[148,]  9.6731079  14   3
[149,]  9.7398108  14  10
[150,]  9.7956136  14   3
[151,]  9.9223679  17  10
[152,]  9.9842716  10  17
[153,] 10.2349158   6   2
[154,] 10.2442451   4   8
[155,] 10.5105560   4  14
[156,] 10.5192202  14   4
[157,] 10.5904550  14   5
[158,] 10.6502211   4   8
[159,] 10.7461410  11  17
[160,] 10.7515991  17  11
[161,] 10.7688993   2   6
[162,] 10.8838438   2  17
[163,] 11.1188273  19  11
[164,] 11.1589938  11  19
[165,] 11.1621731  12   3
[166,] 11.1632288   3  12
[167,] 11.1641170  12   3
```

```
[168,] 11.3015630   14    1
[169,] 11.3321034   14   11
[170,] 11.3688719   11   14
[171,] 11.3762782   14   11
[172,] 11.4384228    5    3
[173,] 11.4401584    3    5
[174,] 11.4723941    5    3
[175,] 11.5888687   19    4
[176,] 11.6020910   19   10
[177,] 11.6070507   10   20
[178,] 11.9241856   20   10
[179,] 12.0264030   20   13
[180,] 12.1623650   20   15
[181,] 12.2223621    6   20
[182,] 12.2440493   20    6
[183,] 12.3746770   20   10
[184,] 12.3943923    4   14
[185,] 12.4116423   14    4
[186,] 12.4162427   16   12
[187,] 12.4301145   12   16
[188,] 12.5619434   16   11
[189,] 12.5627711   16   15
[190,] 12.6720149    9    4
[191,] 12.6735886    9   12
[192,] 12.6891810    7   19
[193,] 12.7220991    7    9
[194,] 12.8534795    7    4
[195,] 12.8591847    4    7
[196,] 12.9332132    3   18
[197,] 12.9885167   13   20
[198,] 13.1496426   14    1
[199,] 13.1568001   14   11
[200,] 13.2272979   14    6
[201,] 13.3043678    7   20
[202,] 13.3308943    7    6
[203,] 13.3764960    7    9
[204,] 13.4877359    9    7
[205,] 13.4935416    7    9
[206,] 13.6490008    9    7
[207,] 13.7357452    7    9
[208,] 13.7451055   13   11
[209,] 13.8765810   13   17
[210,] 13.8839156   18    3
[211,] 14.4086862    3   18
[212,] 14.4922354    3   19
[213,] 14.5186833   19    3
[214,] 14.5962353   19    2
[215,] 14.6480218    2   19
[216,] 14.6541635    2    9
[217,] 14.6682621    9    2
[218,] 14.6687569    2    9
[219,] 14.7126368    7    4
[220,] 14.7253559    7   15
[221,] 14.7994312    7    9
```

```
[222,] 14.8052581    7    1
[223,] 14.8206612    7    4
[224,] 14.9569104    7   14
[225,] 14.9714722    7    9
[226,] 15.1169674    7   20
[227,] 15.1287157    7    3
[228,] 15.2600147   19    9
[229,] 15.2657333    9   19
[230,] 15.6325196   19    9
[231,] 15.7465865   19   13
[232,] 15.7480988   13   19
[233,] 15.9444435   13    1
[234,] 16.0255688    1   13
[235,] 16.0931371    9   19
[236,] 16.0985941   19    9
[237,] 16.5888540    3   11
[238,] 16.6231550   11    3
[239,] 16.9295861   11   13
[240,] 17.0471600   17   16
[241,] 17.0910953   17    3
[242,] 17.1266546    3   17
[243,] 17.1827762    5   14
[244,] 17.2051254    5    7
[245,] 17.2291613    7    5
[246,] 17.2316348    7    1
[247,] 17.2980657   13   11
[248,] 17.3476952   11   13
[249,] 17.5733305   11   19
[250,] 17.6935952   19   11
[251,] 17.7786883   11   19
[252,] 17.8078964   19   15
[253,] 17.8135194   19   14
[254,] 17.8700523   14   19
[255,] 18.0942507   14    4
[256,] 18.2632841   14   19
[257,] 18.2812178   14    3
[258,] 18.3059446    7    9
[259,] 18.3070388   13    6
[260,] 18.4394049   16   20
[261,] 18.4540614   14   16
[262,] 18.4687117   14    5
[263,] 18.4897639    5   14
[264,] 18.5399921   14    5
[265,] 18.5502284   14   19
[266,] 18.7682225   14   16
[267,] 18.9536949   14    5
[268,] 18.9650179   14    6
[269,] 18.9729763    6   14
[270,] 18.9750813   14    6
[271,] 18.9807737    1    6
[272,] 19.0750653    6    1
[273,] 19.1589048    6   14
[274,] 19.2729113    6    1
[275,] 19.2987954    6   11
```

```
[276,] 19.3287846   11   20
[277,] 19.4770884   11    6
[278,] 19.5030839   11   17
[279,] 19.5129482   11    8
[280,] 19.5886208   15    4
[281,] 19.5989402    4   15
[282,] 19.7559520    4    9
[283,] 19.8207397    4   15
[284,] 19.8528424   15    4
[285,] 19.9563474   15   12
[286,] 19.9867124   12   15
[287,] 19.9880540    9   15
[288,] 20.1749546   13    9
[289,] 20.2484957   13    1
[290,] 20.2722347   13   11
[291,] 20.2909075   11   13
[292,] 20.3048958   13   11
[293,] 20.3518038   13    3
[294,] 20.4587377   14   15
[295,] 20.4748312    8    4
[296,] 20.6330859    4    8
[297,] 20.6587722   18    4
[298,] 20.7266679    4   18
[299,] 20.9256894    4   17
[300,] 20.9439174   17    4
[301,] 21.1166487   14    5
[302,] 21.1223181   14    6
[303,] 21.1269821   14   11
[304,] 21.1494679   14    6
[305,] 21.1527868    6   14
[306,] 21.1716312   20    4
[307,] 21.5400175    2    5
[308,] 21.5617747    5    2
[309,] 21.5820643   11   16
[310,] 21.5830246   16   11
[311,] 21.6441086   19   10
[312,] 21.6464329   10   11
[313,] 21.6933296   10    2
[314,] 21.7285253   20   11
[315,] 21.7582696   11   20
[316,] 21.7998642   20   19
[317,] 21.9374925   20   18
[318,] 21.9607010   20   11
[319,] 21.9671142   20    1
[320,] 21.9844282    1   20
[321,] 22.0354730    7    6
[322,] 22.1758811    6    7
[323,] 22.2489652    7    6
[324,] 22.3665573    7    5
[325,] 22.4025380    7    6
[326,] 22.4088482   19   20
[327,] 22.4804943   10   11
[328,] 22.5406789   11   10
[329,] 22.5627199   10   11
```

```
[330,] 22.6057460   10   14
[331,] 22.6091426   14   10
[332,] 22.6341038   14   12
[333,] 22.6471486   14    3
[334,] 22.6820063    2    7
[335,] 22.7124162    2   16
[336,] 22.8276142    7   10
[337,] 22.8519107    7   16
[338,] 22.8864356    7   17
[339,] 22.9235805    7    2
[340,] 22.9242318    7    9
[341,] 23.1397274   14   10
[342,] 23.1551784   14    6
[343,] 23.2385703   14    5
[344,] 23.2897788    1   16
[345,] 23.3228644   18   20
[346,] 23.3905634   20   18
[347,] 23.3950810   18   14
[348,] 23.4083511   14   18
[349,] 23.4152152    4    3
[350,] 23.7223733    3    4
[351,] 23.8675070    4    3
[352,] 23.9109710    4   11
[353,] 24.0927904    9    5
[354,] 24.0999334    5    9
[355,] 24.1244272    5   14
[356,] 24.1409506   14   15
[357,] 24.3240890   14   11
[358,] 24.4494694   11   14
[359,] 24.5152849   14   11
[360,] 24.7184901   14   19
[361,] 24.7985744    8   11
[362,] 24.9027805   11    8
[363,] 25.0739456   14    2
[364,] 25.1534920    2   14
[365,] 25.2351461    2    8
[366,] 25.2433560    3    4
[367,] 25.3300615   11    1
[368,] 25.3626113    1   11
[369,] 25.3696255    1    3
[370,] 25.3703841    1   11
[371,] 25.4900025    7    3
[372,] 25.5203766    7    2
[373,] 25.5865172   17   11
[374,] 25.6118047   17    5
[375,] 25.6712600    8   11
[376,] 25.7244851   11    8
[377,] 25.7783775   11    4
[378,] 25.8277442    4   12
[379,] 25.9006079   12    4
[380,] 25.9204849    4   12
[381,] 26.1071106    5   20
[382,] 26.1105864   17    4
[383,] 26.1491487    7    2
```

[384,]	26.1590358	7	6
[385,]	26.1899565	6	1
[386,]	26.2068331	1	6
[387,]	26.2516567	14	18
[388,]	26.2939241	14	2
[389,]	26.2979687	14	6
[390,]	26.3107718	14	8
[391,]	26.3209816	14	11
[392,]	26.3994654	11	14
[393,]	26.4001341	14	11
[394,]	26.4715209	14	13
[395,]	26.5009230	14	12
[396,]	26.5017107	12	14
[397,]	26.5464784	14	12
[398,]	26.5799750	14	18
[399,]	26.6242580	4	17
[400,]	26.7884379	4	9
[401,]	26.8852492	4	11
[402,]	26.8948308	15	14
[403,]	26.9037410	14	15
[404,]	26.9183792	14	9
[405,]	27.2077154	14	8
[406,]	27.2401391	14	11
[407,]	27.2716196	14	15
[408,]	27.2869494	14	20
[409,]	27.2947321	14	1
[410,]	27.3145867	1	14
[411,]	27.3323034	14	1
[412,]	27.3824430	2	7
[413,]	27.3918433	7	17
[414,]	27.4192787	8	16
[415,]	27.4605148	16	8
[416,]	27.4803240	11	16
[417,]	27.6083357	16	17
[418,]	27.6483634	16	9
[419,]	27.6509434	9	16
[420,]	27.6532352	16	9
[421,]	27.7052963	6	15
[422,]	27.7181538	20	5
[423,]	27.8117664	20	16
[424,]	27.8161879	20	1
[425,]	27.9489749	20	6
[426,]	27.9527887	6	20
[427,]	27.9656538	6	3
[428,]	28.0367301	20	10
[429,]	28.0556194	10	20
[430,]	28.1705236	1	17
[431,]	28.1831938	17	1
[432,]	28.3132621	17	19
[433,]	28.4082253	19	17
[434,]	28.4252073	17	19
[435,]	28.4574575	6	7
[436,]	28.4740184	7	6
[437,]	28.5847768	7	2

[438,]	28.6314858	7	16
[439,]	28.6455631	7	15
[440,]	28.6835829	7	5
[441,]	28.6893573	7	2
[442,]	28.7736725	7	9
[443,]	28.8442967	7	12
[444,]	28.9420232	7	5
[445,]	28.9677329	7	13
[446,]	28.9684016	7	6
[447,]	29.0332366	7	8
[448,]	29.0729869	7	18
[449,]	29.0885583	3	6
[450,]	29.1176373	6	3
[451,]	29.3609658	14	3
[452,]	29.4396326	14	6
[453,]	29.5425672	6	14
[454,]	29.5734727	9	15
[455,]	29.7459300	15	9
[456,]	29.9428882	9	15
[457,]	30.0583443	15	4
[458,]	30.0658484	4	15
[459,]	30.1389912	15	4
[460,]	30.2240237	4	15
[461,]	30.3287143	4	6
[462,]	30.3533487	6	4
[463,]	30.3790072	14	13
[464,]	30.4014991	14	3
[465,]	30.4216776	14	1
[466,]	30.5114485	1	14
[467,]	30.8027570	1	20
[468,]	30.9267666	20	1
[469,]	31.0252790	6	16
[470,]	31.1076660	12	7
[471,]	31.2136179	6	4
[472,]	31.2565079	6	18
[473,]	31.3425495	6	2
[474,]	31.3755728	2	6
[475,]	31.5430522	6	2
[476,]	31.6464642	2	6
[477,]	31.7037192	2	8
[478,]	31.7944204	6	2
[479,]	31.9600930	2	6
[480,]	32.0557958	19	17
[481,]	32.0730092	14	20
[482,]	32.0969157	16	2
[483,]	32.2296514	7	12
[484,]	32.2930153	15	17
[485,]	32.3122852	17	15
[486,]	32.4043979	7	14
[487,]	32.4761797	14	7
[488,]	32.4962005	6	4
[489,]	32.5985593	20	18
[490,]	32.6003081	18	20
[491,]	32.6328173	14	6

```
[492,] 32.6614003    6   11
[493,] 32.7049714   11    6
[494,] 32.7359335    6   11
[495,] 32.7724468   11    6
[496,] 32.7807211    6   11
[497,] 32.8309900   14    1
[498,] 32.9258172   14    8
[499,] 32.9415915   14    7
[500,] 32.9795996    7   14
[501,] 33.0389313    7    4
[502,] 33.0560944    7    8
[503,] 33.0637281    7   15
[504,] 33.1194538    7   20
[505,] 33.2014479    7    2
[506,] 33.2276010    7    9
[507,] 33.2528512    7   13
[508,] 33.3884686    7    5
[509,] 33.4127552    7    1
[510,] 33.4690361    1    7
[511,] 33.5092166    7    1
[512,] 33.5246947    7   16
[513,] 33.5461316    7   20
[514,] 33.5462632   14    7
[515,] 33.5638235   14   13
[516,] 33.5936392   14   19
[517,] 33.5967103   19   11
[518,] 33.5997384   11   19
[519,] 33.6167387    4    7
[520,] 33.6188041    2    7
[521,] 33.7008748    7    2
[522,] 33.8048819    7    1
[523,] 33.8159516    1    7
[524,] 33.8243336    7    1
[525,] 33.8258590    7    3
[526,] 33.8613576    7    2
[527,] 33.9526036    7    4
[528,] 33.9871033    7    1
[529,] 33.9965713    7    6
[530,] 34.0474414   11    6
[531,] 34.1840120    6   20
[532,] 34.2327342   20   12
[533,] 34.2845849   17    6
[534,] 34.3218019    6   17
[535,] 34.4210319   16    5
[536,] 34.4392404    5   16
[537,] 34.6412442    5   20
[538,] 34.6707769   20    5
[539,] 34.6808667    3    8
[540,] 34.6928298   13   14
[541,] 34.7119419   14   13
[542,] 34.8117361   14    1
[543,] 34.8259857   14   10
[544,] 34.9669774   15    9
[545,] 34.9828878    9    7
```

```
[546,] 35.0237463    7    9
[547,] 35.0540788    7    8
[548,] 35.0965099    7    4
[549,] 35.1386016    7   13
[550,] 35.2203470    7    9
[551,] 35.2440522    7   16
[552,] 35.2899990    7    9
[553,] 35.4144232    7    1
[554,] 35.7228047    7   15
[555,] 35.9513281   19   10
[556,] 35.9872415   10   19
[557,] 36.0310719   10    6
[558,] 36.0743168    6   10
[559,] 36.1376287   10    6
[560,] 36.1451585    6   11
[561,] 36.1503157   11    6
[562,] 36.1541586    6    4
[563,] 36.2547491    5   20
[564,] 36.3573465   12    5
[565,] 36.5288562    5   12
[566,] 36.5394303    7    9
[567,] 36.5524685    9    7
[568,] 36.5757749    7    9
[569,] 36.6115378    7   17
[570,] 36.6135565    7    5
[571,] 36.6423432   11   17
[572,] 36.6445978    9   15
[573,] 36.7032633   15    6
[574,] 36.7057953    6   15
[575,] 36.7655722   15    6
[576,] 36.7988189   15    5
[577,] 37.0673646   13    7
[578,] 37.0807186    7   13
[579,] 37.0813209    7    1
[580,] 37.1054444    7    5
[581,] 37.1184758   14   13
[582,] 37.2678427    7    1
[583,] 37.2959551    7   20
[584,] 37.3676302    4    8
[585,] 37.4023607    8    4
[586,] 37.5068382    4    6
[587,] 37.5070893   16   15
[588,] 37.5127231   15   16
[589,] 37.6102426   15    6
[590,] 37.6523134    6   15
[591,] 37.6834540    7   13
[592,] 37.7863716    2    9
[593,] 37.8187649    9    2
[594,] 37.8916086    7    9
[595,] 37.9512188    7    8
[596,] 37.9978851    7   13
[597,] 38.1545412   14    8
[598,] 38.3169869    3   15
[599,] 38.3435826   15    3
```

[600,]	38.3961775	3	19
[601,]	38.4507204	19	3
[602,]	38.5616126	19	2
[603,]	38.5672796	19	3
[604,]	38.5778039	3	19
[605,]	38.6003767	14	13
[606,]	38.6034308	14	5
[607,]	38.6234746	14	16
[608,]	38.7537680	14	13
[609,]	38.7694516	14	9
[610,]	38.7846001	8	14
[611,]	38.8139759	14	10
[612,]	38.8545082	10	14
[613,]	38.8855406	3	12
[614,]	38.8970549	12	3
[615,]	38.8984521	7	13
[616,]	39.0301539	7	19
[617,]	39.1841232	2	19
[618,]	39.1901459	19	2
[619,]	39.2848848	19	7
[620,]	39.4800567	7	19
[621,]	39.5655884	7	15
[622,]	39.5744302	12	3
[623,]	39.5890979	12	5
[624,]	39.5988313	12	3
[625,]	39.8598232	11	6
[626,]	39.8980221	12	3
[627,]	40.0774314	17	11
[628,]	40.2292548	4	17
[629,]	40.2486764	4	6
[630,]	40.2913500	4	15
[631,]	40.3511512	15	4
[632,]	40.4005956	4	13
[633,]	40.4153557	13	4
[634,]	40.4344204	2	19
[635,]	40.4469573	12	4
[636,]	40.4762805	12	5
[637,]	40.5126905	3	17
[638,]	40.5239256	17	3
[639,]	40.5358073	17	2
[640,]	40.5954386	9	7
[641,]	40.6046174	7	9
[642,]	40.9341003	7	8
[643,]	41.1042480	8	7
[644,]	41.1303095	7	12
[645,]	41.1318759	7	8
[646,]	41.2789211	8	13
[647,]	41.3158808	13	8
[648,]	41.5101403	8	13
[649,]	41.6070875	13	8
[650,]	41.7517214	13	6
[651,]	41.8730501	6	13
[652,]	42.0840672	17	3
[653,]	42.4130248	3	1

```
[654,] 42.6918269   16   17
[655,] 42.7740917   17   16
[656,] 42.8747921   11    4
[657,] 42.9345981    4   11
[658,] 43.0039277    5   12
[659,] 43.1982366    9    5
[660,] 43.2059364   19   17
[661,] 43.2923849   17   19
[662,] 43.3369606   19   17
[663,] 43.3499270   14    1
[664,] 43.5716572   14   10
[665,] 43.5899311   14    5
[666,] 43.7440393    5   14
[667,] 43.8433473   14   20
[668,] 43.9929849   14   11
[669,] 44.0049355   14    4
[670,] 44.1510948   14    5
[671,] 44.2429583    1   13
[672,] 44.2577970    1   10
[673,] 44.4451469    1    2
[674,] 44.4917046    2    1
[675,] 44.5411244    1    2
[676,] 44.5798632    1   18
[677,] 44.5896300   18    1
[678,] 44.6074224   10    6
[679,] 44.6648673   10    1
[680,] 44.6762521    1   10
[681,] 44.7266581    7    2
[682,] 44.8004509    7   11
[683,] 44.8936062   11    7
[684,] 44.9190784    7   11
[685,] 45.1130771    7    5
[686,] 45.1782824    5    7
[687,] 45.2121327    7    5
[688,] 45.2140466    7    2
[689,] 45.2353393    7    1
[690,] 45.2422281    7    6
[691,] 45.3342377    7   12
[692,] 45.5329383    7    5
attr(,"n")
[1] 20
```

As we saw in section 1.5, running the `simulate` command on the fitted model object produces a new trajectory of identical length to the original, with the same coefficients. Note that the new trajectory is identical in terms of the number of *realized events* it contains, and it will not in general cover the same *time period.* Some disparity between the two is normal (and, indeed, will happen with probability 1); however, when the total mean time period of the replicate sequences is substantially different from that of the original data, this suggests that the pacing of the model is off.

In section 1.5, we showed how an *in silico* knock-out study could be used to gain insights into model behavior. Another useful strategy can be to simulate trajectories from a fitted model with alternative choices of covariates. For instance, what might we expect if we replaced the teachers in our classroom with students? This anarchic state of affairs can be probed by conditional simulation with a different set of covariates:

```r
set.seed(1331)
AnarchSim <- simulate(classfit6, covar = list(CovSnd = cbind(ClassIntercept,
   rep(0, 20)), CovRec = rep(0, 20)))

AnarchSim  #Examine the trajectory
```

```
            [,1] [,2] [,3]
  [1,] 0.2923476   10    2
  [2,] 0.3138169    2   10
  [3,] 0.3510402   10   11
  [4,] 0.3835927   16   10
  [5,] 0.4142638   10    9
  [6,] 0.5083167   17   10
  [7,] 0.5244295    8    2
  [8,] 0.5389847    2    8
  [9,] 0.6802120    2    9
 [10,] 0.7892631    2   10
 [11,] 0.9134020   10    2
 [12,] 1.1401766   12    9
 [13,] 1.1722723    9   12
 [14,] 1.2860364    9    5
 [15,] 1.3204401    5    7
 [16,] 1.3273566    7    5
 [17,] 1.5074910   20    8
 [18,] 1.5491034    1   17
 [19,] 1.5938873   17   14
 [20,] 1.6298776   14   17
 [21,] 1.6616822    8   20
 [22,] 1.7067478    8   12
 [23,] 1.8390356   12    8
 [24,] 1.8913117   14    9
 [25,] 1.9369458    9   14
 [26,] 1.9811527   14    9
 [27,] 2.0370493    9   14
 [28,] 2.1335794   14    9
 [29,] 2.6729667   14   18
 [30,] 2.8382370   14   16
 [31,] 2.8590460   14    9
 [32,] 3.0556402    9    3
 [33,] 3.1590035    3    9
 [34,] 3.2669413   16   12
 [35,] 3.3078785   12    8
 [36,] 3.3095245    9   13
 [37,] 3.3376740    9    3
 [38,] 3.4362942   10   15
 [39,] 3.4532597   15   10
 [40,] 3.5902631   10   15
 [41,] 3.6483523   10    7
 [42,] 3.7520920    7   10
 [43,] 3.8268862    7   17
 [44,] 3.9045994   17    7
 [45,] 4.2060566    7   17
 [46,] 4.3229867    7   15
 [47,] 4.3705204    7   17
```

```
 [48,]  4.3828970  15  18
 [49,]  4.4004880  18  15
 [50,]  4.7473149  15  18
 [51,]  4.8002721   6   7
 [52,]  5.1515952  14   2
 [53,]  5.1837837   2  14
 [54,]  5.2083843   2  12
 [55,]  5.2336125  12   2
 [56,]  5.3725548   2  12
 [57,]  5.4078890   2   1
 [58,]  5.4506936   1   2
 [59,]  5.6765250   2   1
 [60,]  5.8028868  18   9
 [61,]  5.8301328  18  11
 [62,]  5.9230796  19  12
 [63,]  5.9907164  19  18
 [64,]  6.0822529  18  19
 [65,]  6.1201552  19  18
 [66,]  6.1839281  19   2
 [67,]  6.2083454   2  19
 [68,]  6.3104230   6   3
 [69,]  6.3612375   3   6
 [70,]  6.3635596  18  20
 [71,]  6.3764722  20  18
 [72,]  6.3926626   9   2
 [73,]  6.4801215   2   9
 [74,]  6.5008767   2  16
 [75,]  6.5093808  16   2
 [76,]  6.6197236   2  16
 [77,]  6.7288420   2   6
 [78,]  6.7719947   6   2
 [79,]  6.7782700   2   6
 [80,]  6.9902600   2  16
 [81,]  7.1294971  16   2
 [82,]  7.2700611  16  10
 [83,]  7.3857314  10  16
 [84,]  7.3871497  16  10
 [85,]  7.4055389  10  16
 [86,]  7.4659056  10  15
 [87,]  7.4745699  15  10
 [88,]  7.5276088  10   3
 [89,]  7.5931173   5   7
 [90,]  7.6173192   7   5
 [91,]  7.7344420  17   7
 [92,]  7.7368855   7  17
 [93,]  7.7765890  17   7
 [94,]  8.0974308  14  11
 [95,]  8.2430388   3  18
 [96,]  8.4024232   3   6
 [97,]  8.4627261   3  10
 [98,]  8.5094596   8   5
 [99,]  8.6521118   8  12
[100,]  8.7905033  12   8
[101,]  8.8102299  12  16
```

```
[102,]  9.0012077  16  12
[103,]  9.0185906  12  16
[104,]  9.1225491  12  11
[105,]  9.1571843  11  12
[106,]  9.3197623  18  13
[107,]  9.4919514  13  18
[108,]  9.5060038  13   4
[109,]  9.6355539  13  18
[110,]  9.7350181  10   6
[111,]  9.7425661   6  10
[112,]  9.7545220   6  14
[113,]  9.7835823  14   6
[114,]  9.8766565  12   8
[115,]  9.9864765   8  12
[116,] 10.0228536  12   8
[117,] 10.1605709  12  11
[118,] 10.2659908  11  12
[119,] 10.2862677  11   8
[120,] 10.3230630  18   1
[121,] 10.3425045  16   4
[122,] 10.3430870  16  12
[123,] 10.3727043  12  16
[124,] 10.5623821  16  12
[125,] 10.5624894  10  12
[126,] 10.6277419  10  18
[127,] 10.6907312  15  12
[128,] 10.7539802  12  15
[129,] 10.7890927  12   2
[130,] 10.8201173  12   3
[131,] 10.8885994   3  12
[132,] 10.9998222  12   8
[133,] 11.0466308  12  16
[134,] 11.0947486  14   3
[135,] 11.0951965   3  14
[136,] 11.1014127  14  18
[137,] 11.4259938  18  11
[138,] 11.4991777   4  16
[139,] 11.5106850   3  20
[140,] 11.5998214  18  10
[141,] 11.8147666  10  18
[142,] 11.9101625  18  10
[143,] 12.0030597   4  16
[144,] 12.0763611   4  10
[145,] 12.0799917   4  18
[146,] 12.1311849  18   4
[147,] 12.1456216  18   7
[148,] 12.1590339   7  18
[149,] 12.2702760  18   7
[150,] 12.3513811   7  18
[151,] 12.5627726   4   9
[152,] 12.6246743   9   4
[153,] 12.8824258  11  14
[154,] 12.8920507   6  14
[155,] 13.1055420  14   6
```

```
[156,] 13.1348553    6   14
[157,] 13.2410002    6    1
[158,] 13.3166928    5    4
[159,] 13.3782782    3   16
[160,] 13.3837846   16    3
[161,] 13.4015381   20    3
[162,] 13.5026941    3   20
[163,] 13.8862291   19   10
[164,] 13.9253554   10   19
[165,] 13.9286514   10    5
[166,] 13.9295118    5   10
[167,] 13.9304198   10    5
[168,] 14.0702082   14   11
[169,] 14.1301796   14    4
[170,] 14.1764226    4   14
[171,] 14.1910475   14    4
[172,] 14.2846297   20    1
[173,] 14.2864105    1   20
[174,] 14.3198832   20    1
[175,] 14.4427156   19    4
[176,] 14.4565313   19    8
[177,] 14.4617283    8   10
[178,] 14.7527743   10    8
[179,] 14.8584625    8   19
[180,] 15.0702672   19   15
[181,] 15.1330639    4   14
[182,] 15.1622000    4   19
[183,] 15.2693648   19    4
[184,] 15.2914344   13   14
[185,] 15.3272240   14   13
[186,] 15.3341290   19    3
[187,] 15.3461867    3   19
[188,] 15.4825263   19   14
[189,] 15.4833777   19    2
[190,] 15.6103139    9   19
[191,] 15.6119796    9   12
[192,] 15.6295171    7   10
[193,] 15.6695265   10    7
[194,] 15.9102795    7   13
[195,] 15.9169702   13    7
[196,] 16.0641202    1   16
[197,] 16.1209013   15   19
[198,] 16.2847947    9    3
[199,] 16.2942853    3    9
[200,] 16.4172726    3    7
[201,] 16.5178685    2   14
[202,] 16.5520249    2   18
[203,] 16.6079254    7    3
[204,] 16.8327095    3    7
[205,] 16.8441731    7    3
[206,] 17.0728833    3    7
[207,] 17.2441642    3   16
[208,] 17.2569613    6   13
[209,] 17.3824748    6   14
```

```
[210,] 17.3904017   13    6
[211,] 17.8591274    6   13
[212,] 17.9480207    6    9
[213,] 17.9755122    9    6
[214,] 18.0574950    9    7
[215,] 18.1109672    7    9
[216,] 18.1174337    7   13
[217,] 18.1336749   13    7
[218,] 18.1341960    7   13
[219,] 18.1802639   12    9
[220,] 18.1960324    9   12
[221,] 18.3310178   12    9
[222,] 18.3398928    9   12
[223,] 18.3660590    9    7
[224,] 18.5502745    9   11
[225,] 18.5777781   11    9
[226,] 18.8005435    9   10
[227,] 18.8156050    9    5
[228,] 18.9796911   20    9
[229,] 18.9856026    9   20
[230,] 19.3622465   20    9
[231,] 19.4821597   20    6
[232,] 19.4836223    6   20
[233,] 19.6906986    6    7
[234,] 19.7745441    7    6
[235,] 19.8447431   11    9
[236,] 19.8497741    9   11
[237,] 20.3587531   13    1
[238,] 20.3936526    1   13
[239,] 20.7142745   13    1
[240,] 20.8830258    5    1
[241,] 20.9276720    5    7
[242,] 20.9663152    7    5
[243,] 21.0250251   19   10
[244,] 21.0761644    3    4
[245,] 21.1336628    3    5
[246,] 21.1359252    5    3
[247,] 21.2566208    1   20
[248,] 21.2941176   20    1
[249,] 21.5312827   20    8
[250,] 21.6591883    8   20
[251,] 21.7491764   20    8
[252,] 21.7800263    8   12
[253,] 21.7859016   18    2
[254,] 21.9923980    2   12
[255,] 22.2277790   12    2
[256,] 22.5365273    2   12
[257,] 22.5633974   12    2
[258,] 22.6113058   13    9
[259,] 22.6126394   14   11
[260,] 22.7569517    2   13
[261,] 22.7723499   20    8
[262,] 22.7970899    8   20
[263,] 22.8360553   20    8
```

```
[264,] 22.9347997   20    4
[265,] 22.9444485    4   20
[266,] 23.3083117   20   17
[267,] 23.5411130   17   20
[268,] 23.5581578   20   19
[269,] 23.5682095   19   20
[270,] 23.5723161   19    8
[271,] 23.5785719   15    3
[272,] 23.6733110    3   15
[273,] 23.7594996    3    1
[274,] 23.9044456    3   15
[275,] 23.9308584    3   20
[276,] 23.9632351   20   13
[277,] 24.1162688   20    3
[278,] 24.1431824   20    1
[279,] 24.1559287   20   10
[280,] 24.2183264    2    3
[281,] 24.2288747    3    2
[282,] 24.3900974    2    3
[283,] 24.4835569    2    5
[284,] 24.5043925    5    2
[285,] 24.6128313    5    6
[286,] 24.6441341    6    5
[287,] 24.6455343   20   15
[288,] 24.8365782   13   19
[289,] 24.9104558   13   20
[290,] 24.9332760   20   13
[291,] 24.9597418   13   20
[292,] 24.9742885   20   13
[293,] 25.0230139   20    1
[294,] 25.1838857   15   18
[295,] 25.2048109    2   14
[296,] 25.3594422   14    2
[297,] 25.3858380    1    5
[298,] 25.4556166    5    1
[299,] 25.6626715    5   16
[300,] 25.6813944   16    5
[301,] 25.8620912    5    1
[302,] 25.8714207    1    5
[303,] 25.8789697    1    3
[304,] 25.9103893    3    1
[305,] 25.9157802    1    3
[306,] 25.9528417   12   14
[307,] 26.3408947    9   10
[308,] 26.3634775   10    9
[309,] 26.3839404   18    4
[310,] 26.3848417    4   18
[311,] 26.4479034   10   14
[312,] 26.4502289   15   20
[313,] 26.5088135   15   17
[314,] 26.5438637   15    1
[315,] 26.5707329    1   15
[316,] 26.6147561   10    9
[317,] 26.8250660    9    3
```

```
[318,] 26.8505573    9   10
[319,] 26.8572759    9   14
[320,] 26.8762539   14    9
[321,] 26.9294912   15   19
[322,] 27.1229950   19   15
[323,] 27.2655738   19   18
[324,] 27.3818236   18   19
[325,] 27.4469673   19   18
[326,] 27.4562683    9   14
[327,] 27.5543539    8    4
[328,] 27.6147748    4    8
[329,] 27.6379374    8    4
[330,] 27.6836848   17   15
[331,] 27.6932950   15   17
[332,] 27.7312181   15    2
[333,] 27.7479134   15   17
[334,] 27.8183864    1   16
[335,] 27.8882129    4    5
[336,] 28.0179240   10   19
[337,] 28.0538150   10   12
[338,] 28.0997399   10    5
[339,] 28.1525601    5   10
[340,] 28.1535859   10    5
[341,] 28.5548566   13    1
[342,] 28.5709198    1   13
[343,] 28.7196746    1    3
[344,] 28.8084685   13   15
[345,] 28.8425632   12   10
[346,] 28.9169140   10   12
[347,] 28.9215957    6    5
[348,] 28.9580666    6    2
[349,] 28.9650163    4    2
[350,] 29.2835217    2    4
[351,] 29.4319091    4    2
[352,] 29.4769153    4   15
[353,] 29.6630762   16   19
[354,] 29.6701225   19   16
[355,] 29.6950766   19   13
[356,] 29.7266264    8   20
[357,] 29.9811693    8    4
[358,] 30.1749085    4    8
[359,] 30.3042077    8    4
[360,] 30.6135395    4    8
[361,] 30.7648989   10    2
[362,] 30.8647540    2   10
[363,] 31.0413652    6    7
[364,] 31.1729317    7    6
[365,] 31.3340448   11   18
[366,] 31.3429703    6    7
[367,] 31.4502756   19   17
[368,] 31.4834029   17   19
[369,] 31.4908001   17   13
[370,] 31.4915844   17   19
[371,] 31.6177319   10   20
```

```
[372,] 31.6618316   20   10
[373,] 31.7698939   14    3
[374,] 31.7947628   14   19
[375,] 31.8571037   13   18
[376,] 31.9030162   18   13
[377,] 31.9590203   18   15
[378,] 31.9968510   18   17
[379,] 32.0710023   17   18
[380,] 32.0916161   18   17
[381,] 32.2855277    3    8
[382,] 32.2890891    1    5
[383,] 32.3317549    1    8
[384,] 32.3419710    8    1
[385,] 32.3936127    2    6
[386,] 32.4127757    2   10
[387,] 32.4518540   10    9
[388,] 32.5102836    9   10
[389,] 32.5167872    9   13
[390,] 32.5336912    9   10
[391,] 32.5539176    9   13
[392,] 32.6595291   13    9
[393,] 32.6608488   13   14
[394,] 32.7358451   13   19
[395,] 32.7827946   19    3
[396,] 32.7838726    3   19
[397,] 32.8712506    3   14
[398,] 32.9125276   14    3
[399,] 32.9914511    9   11
[400,] 33.1294328    9   10
[401,] 33.2680131    9   17
[402,] 33.2760206    9   19
[403,] 33.2952151    9    1
[404,] 33.3092158    9   10
[405,] 33.8506090   10   16
[406,] 33.8950125   16   10
[407,] 33.9527831   10   16
[408,] 33.9761946   10   14
[409,] 33.9862128   10    1
[410,] 34.0120787    1   10
[411,] 34.0462638    1    6
[412,] 34.0979760   18    2
[413,] 34.1143814    3    1
[414,] 34.1527231    6    4
[415,] 34.1891149    4    6
[416,] 34.2093612    8    2
[417,] 34.3148677    8   16
[418,] 34.3515881    8    9
[419,] 34.3541918    9    8
[420,] 34.3565100    8    9
[421,] 34.4095189   14   20
[422,] 34.4219504    2   18
[423,] 34.5129716    2   15
[424,] 34.5178496    2   19
[425,] 34.6439730    2    4
```

[426,]	34.6478736	4	2
[427,]	34.6609712	4	11
[428,]	34.7330393	1	2
[429,]	34.7504841	2	1
[430,]	34.8679105	5	18
[431,]	34.8808934	18	5
[432,]	35.0152877	18	16
[433,]	35.1121269	16	18
[434,]	35.1294240	18	16
[435,]	35.1626873	9	17
[436,]	35.1908011	17	9
[437,]	35.3546903	9	17
[438,]	35.4320712	17	4
[439,]	35.4500504	17	9
[440,]	35.5245910	17	3
[441,]	35.5317252	3	17
[442,]	35.6771926	3	9
[443,]	35.7698788	3	20
[444,]	35.9084439	3	17
[445,]	35.9584064	3	20
[446,]	35.9593700	20	3
[447,]	36.0531370	20	17
[448,]	36.1055607	20	13
[449,]	36.1265833	5	18
[450,]	36.1562300	18	5
[451,]	36.4026318	16	8
[452,]	36.5276642	16	20
[453,]	36.6578454	20	16
[454,]	36.7170953	17	4
[455,]	36.8870594	4	17
[456,]	37.0889044	17	4
[457,]	37.2070882	4	6
[458,]	37.2176555	4	17
[459,]	37.2932600	17	4
[460,]	37.3809633	4	17
[461,]	37.4891774	4	9
[462,]	37.5152103	9	4
[463,]	37.5413855	18	16
[464,]	37.5757743	16	13
[465,]	37.6015938	13	16
[466,]	37.7704117	13	20
[467,]	38.2290438	13	10
[468,]	38.3332991	10	13
[469,]	38.4341419	6	17
[470,]	38.5177878	17	7
[471,]	38.7448140	4	13
[472,]	38.7739227	13	4
[473,]	38.8979550	7	17
[474,]	38.9489368	7	16
[475,]	39.0582643	16	7
[476,]	39.1648202	7	16
[477,]	39.2238019	7	15
[478,]	39.3158731	17	6
[479,]	39.5081292	17	7

```
[480,] 39.5854449    7   13
[481,] 39.5971942   13   20
[482,] 39.6349884    3   12
[483,] 39.7717400    3   20
[484,] 39.8424554    2   20
[485,] 39.8620605   20    2
[486,] 39.9568023   10    9
[487,] 40.1125780    9   10
[488,] 40.1535260   18   10
[489,] 40.2254364    7    6
[490,] 40.2270528    6    7
[491,] 40.2597972   13    7
[492,] 40.3025721    2    1
[493,] 40.3634377    2   20
[494,] 40.3946396   20    2
[495,] 40.4311074    2   20
[496,] 40.4394458   20    2
[497,] 40.4896520   14   10
[498,] 40.6098875   10   18
[499,] 40.6363165   18   10
[500,] 40.7142749   10   18
[501,] 40.8327413   10    4
[502,] 40.8528630   15    7
[503,] 40.8639700   15    4
[504,] 40.9325118   15   20
[505,] 41.0358711   20   15
[506,] 41.0816727   20    4
[507,] 41.1120374   20    5
[508,] 41.2731126    5   20
[509,] 41.3169948    5   14
[510,] 41.3830107   14    5
[511,] 41.4599402   14   12
[512,] 41.4776352   14    4
[513,] 41.5048913    7   15
[514,] 41.5050971    1    3
[515,] 41.5395413    3   16
[516,] 41.5767481    3   19
[517,] 41.5818210   15    2
[518,] 41.5849502    2   15
[519,] 41.6022013   15   16
[520,] 41.6066714    2    3
[521,] 41.7871649    3    2
[522,] 41.9381163    2    3
[523,] 41.9568132    3    2
[524,] 41.9727164    2    3
[525,] 41.9749099    2    9
[526,] 42.0202894    9    2
[527,] 42.1749248    9    3
[528,] 42.2159526    3    9
[529,] 42.2295776    9    3
[530,] 42.3254337    5    3
[531,] 42.4226282    5   10
[532,] 42.4675178   14    5
[533,] 42.5327724   11   13
```

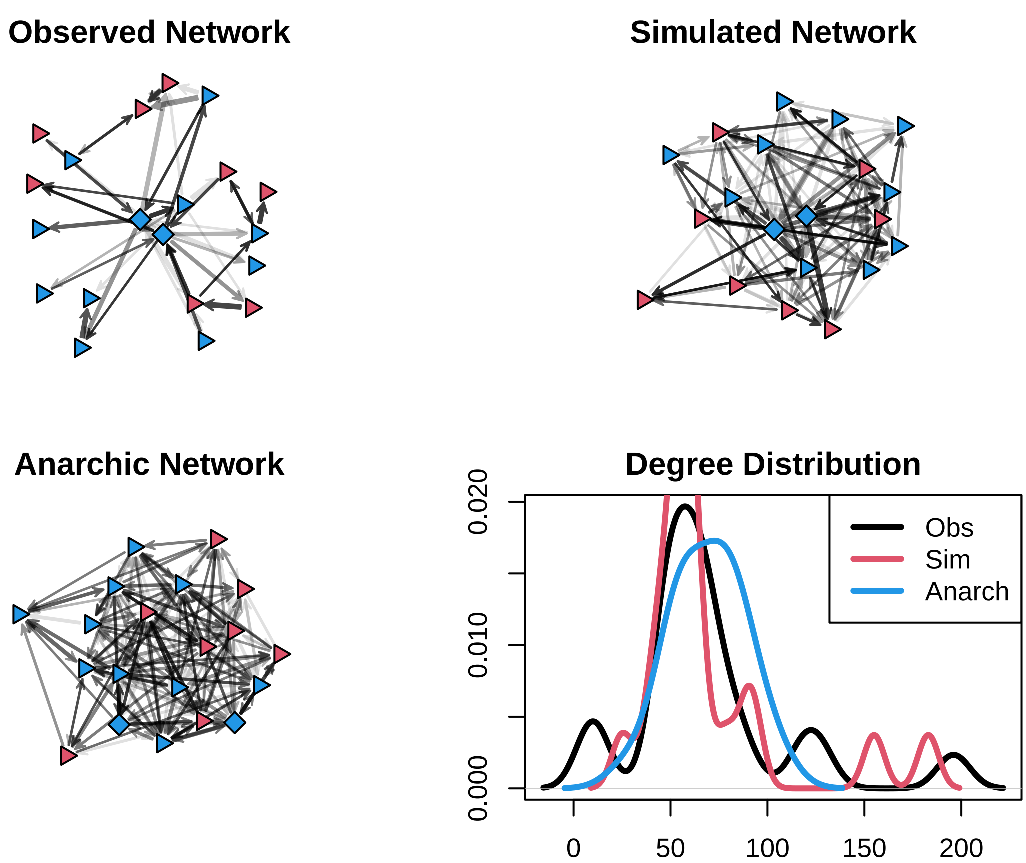
```
[534,] 42.5699451   13   11
[535,] 42.6715205    8    7
[536,] 42.6898927    7    8
[537,] 42.8974270    7   13
[538,] 42.9323569   13    7
[539,] 42.9427113    6   16
[540,] 42.9547196    5   14
[541,] 43.0041139    5   11
[542,] 43.0990994   11    5
[543,] 43.1255727   16    6
[544,] 43.3890319    4   16
[545,] 43.4001503    4    2
[546,] 43.4953514    2    4
[547,] 43.5397988    2   19
[548,] 43.5956860   19    2
[549,] 43.6700460    2   18
[550,] 43.7900404   18    2
[551,] 43.8252859   18    9
[552,] 43.8832906    9   18
[553,] 44.0694248   18    9
[554,] 44.5834657   18    4
[555,] 44.8794815    3    8
[556,] 44.9153291    8    3
[557,] 44.9605086    8   12
[558,] 45.0083699   12    8
[559,] 45.0743488    8   12
[560,] 45.0821859   16    4
[561,] 45.0869696    4   16
[562,] 45.0909540   15    2
[563,] 45.1865309    3    8
[564,] 45.2767106    4    6
[565,] 45.4788075    6    4
[566,] 45.4898950    6    7
[567,] 45.5038118    7    6
[568,] 45.5496031    6    7
[569,] 45.6028214    6   10
[570,] 45.6054561    6    4
[571,] 45.6586879   18    7
[572,] 45.6609304   15   13
[573,] 45.7025531   15   18
[574,] 45.7051613   18   15
[575,] 45.7672509   15   18
[576,] 45.8018652   15    5
[577,] 46.0776347    5    6
[578,] 46.1069861    6    5
[579,] 46.1078829    5    6
[580,] 46.1488675   10    6
[581,] 46.1689557   19    2
[582,] 46.3701996    8    7
[583,] 46.4069460    8   10
[584,] 46.5028454   16    8
[585,] 46.5361385    8   16
[586,] 46.6440341    8    5
[587,] 46.6442387   20    7
```

```
[588,] 46.6499419    7   20
[589,] 46.7509533   20    7
[590,] 46.7965984    7   20
[591,] 46.8287198    7    8
[592,] 46.9393723    2    1
[593,] 46.9757491    1    2
[594,] 47.0498693    1   16
[595,] 47.1178208    1    5
[596,] 47.1804617    5    1
[597,] 47.4090311    8   19
[598,] 47.6156027   17    8
[599,] 47.6424209    8   17
[600,] 47.6963035   17    4
[601,] 47.7538708    4   17
[602,] 47.8676862    4   10
[603,] 47.8744255    4    6
[604,] 47.8830799    6    4
[605,] 47.9065021    6    7
[606,] 47.9101331    6    4
[607,] 47.9484248    4    2
[608,] 48.1363401    2    4
[609,] 48.1613946    4    3
[610,] 48.1814661   16    4
[611,] 48.2443519   11    5
[612,] 48.3088535    5   11
[613,] 48.3687629   19    6
[614,] 48.3794947    6   19
[615,] 48.3809271    6    8
[616,] 48.5045082    8    6
[617,] 48.8088029    9   10
[618,] 48.8140227   10    9
[619,] 48.9101525   10    4
[620,] 49.1903396    4   10
[621,] 49.3157943    4   20
[622,] 49.3285007   13   10
[623,] 49.3388370   10   13
[624,] 49.3493566   10    8
[625,] 49.5349245   10   14
[626,] 49.5700779   14    7
[627,] 49.6858072   14    5
[628,] 49.8165182    7   18
[629,] 49.8385383    7   13
[630,] 49.8803109    7   14
[631,] 49.9557411   14    7
[632,] 50.0061081    7   18
[633,] 50.0248773   18    7
[634,] 50.0442816   14    7
[635,] 50.0569946    8   11
[636,] 50.0870596    8    6
[637,] 50.1237581    9    6
[638,] 50.1350640    6    9
[639,] 50.1473599    6   10
[640,] 50.2098620   12   16
[641,] 50.2294310   16   12
```

```
[642,] 50.7116804    7   14
[643,] 51.0497463   14    7
[644,] 51.0999038    6    9
[645,] 51.1028667    9    6
[646,] 51.3223849    8   16
[647,] 51.3642178   16    8
[648,] 51.5619668    8   16
[649,] 51.6603858   16    8
[650,] 51.8076178   16    6
[651,] 51.9340250    6   16
[652,] 52.1514138   10    6
[653,] 52.4182965   10   20
[654,] 52.6960139   16    7
[655,] 52.7797109    7   16
[656,] 52.8840768   15   17
[657,] 52.9438589   17   15
[658,] 53.0163134    7   16
[659,] 53.2151139   17   18
[660,] 53.2224524    3   20
[661,] 53.3074278   20    3
[662,] 53.3531059    3   20
[663,] 53.3663636   20    6
[664,] 53.6461119    6   20
[665,] 53.6740408   18   17
[666,] 53.9750758   17   18
[667,] 54.1708750    9    2
[668,] 54.3843430    9   18
[669,] 54.4011703    9   12
[670,] 54.6074988   12    9
[671,] 54.7496706    5   16
[672,] 54.7640188    5    6
[673,] 54.9591427    5   20
[674,] 55.0083999   20    5
[675,] 55.0590682    5   20
[676,] 55.0983340    5    9
[677,] 55.1084875    9    5
[678,] 55.1266379   11    9
[679,] 55.1824950    9   11
[680,] 55.1972862   11    9
[681,] 55.2491901    9   15
[682,] 55.3383389    9    6
[683,] 55.4784159    6    9
[684,] 55.5275188    9    6
[685,] 55.8170333    9    4
[686,] 55.9022600    4    9
[687,] 55.9677155    4    6
[688,] 55.9696607    4    9
[689,] 56.0099200    4    6
[690,] 56.0191738    4   16
[691,] 56.1500782    4   18
[692,] 56.4314021   18    4
attr(,"n")
[1] 20
```

```
# Plot the network structure of the simulations, and the
# observed data
par(mfrow = c(2, 2), mar = c(2, 2, 2, 2))
gplot(ClassNet, vertex.col = 4 - 2 * ClassIsFemale, vertex.sides = 3 +
   ClassIsTeacher, vertex.cex = 2, edge.lwd = ClassNet^0.75,
   main = "Observed Network", edge.col = rgb(0, 0, 0, (1 - 1/(1 +
      ClassNet))^3))
SimNet <- as.sociomatrix.eventlist(ClassSim, 20)  #Create a network from the fitted sim
gplot(SimNet, vertex.col = 4 - 2 * ClassIsFemale, vertex.sides = 3 +
   ClassIsTeacher, vertex.cex = 2, edge.lwd = SimNet^0.75, main = "Simulated Network",
   edge.col = rgb(0, 0, 0, (1 - 1/(1 + SimNet))^3))
AnarchNet <- as.sociomatrix.eventlist(AnarchSim, 20)  #Create a network from the anarchy sim
gplot(AnarchNet, vertex.col = 4 - 2 * ClassIsFemale, vertex.sides = 3 +
   ClassIsTeacher, vertex.cex = 2, edge.lwd = AnarchNet^0.75,
   main = "Anarchic Network", edge.col = rgb(0, 0, 0, (1 - 1/(1 +
      AnarchNet))^3))

# Plot the valued degree distributions
plot(density(degree(ClassNet), bw = "SJ"), lwd = 3, main = "Degree Distribution")
lines(density(degree(SimNet), bw = "SJ"), lwd = 3, col = 2)
lines(density(degree(AnarchNet), bw = "SJ"), lwd = 3, col = 4)
legend("topright", legend = c("Obs", "Sim", "Anarch"), lwd = 3,
   col = c(1, 2, 4))
```



Comparing the plots, we can see several things. First, we note some limitations of our fitted model: while it does relatively well at ensuring that the teachers are central, enduring that many of the strongest interactions are student-teacher interactions, creating a network in which strong interactions are localized to a fairly small number of (highly reciprocal) dyads, and reproducing the overall valued degree distribution, it also produces a large "halo" of weak side-interactions among the students that is not seen in the observed network. This suggests the potential for further model improvement.

Turning to our "anarchy in the classroom" model, however, we see that the effect of removing teachers is substantively reasonable. The nodes that were formerly teachers no longer have any particular significance, and are now well-mixed with their peers; likewise, without the teachers to focus attention, the network is as a whole much less centralized. Thus, the model does plausibly produce many of the effects one would expect to see from such a change in group composition. Such scenario-based probes can be a useful tool for assessing model behavior, as well as being of possible substantive interest in and of themselves.

Section 3. Simulating *De Novo* Dyadic Relational Event Models

We have seen how the `simulate` command can be used to simulate draws from fitted `rem.dyad` objects, and even how these may be modified by switching coefficients or covariates for particular purposes. What if we want to create a *de novo* simulation? This can also be done, using `rem.dyad` to create a *model skeleton* that can subsequently be used for simulation.

3.0 Creating a model skeleton

To set up a REM for simulation, we need to create an object that records the system size (i.e., number of vertices), effects involved, and other critical information. When we fit models using 'rem.dyad'', this information was encoded in the model object. In the *de novo* case, we use the same approach - except that we simply omit the data!

To see how this is done, let's consider an example. Let us say that we want to create a model for a 25-node REM with a baseline intercept, an AB-BA P-shift, and a recency effect of sending on future sending (`RSndSnd`). We then proceed by creating a model just as we would normally, but with `NULL` where the data should be:

```
ModInt <- rep(1, 25)
modskel <- rem.dyad(NULL, n = 25, effects = c("CovSnd", "PSAB-BA",
   "RSndSnd"), covar = list(CovSnd = ModInt))
```

```
NULL edgelist passed to rem.dyad - creating model skeleton.
Checking/prepping covariates.
```

```
modskel
```

```
Relational Event Model
    Model skeleton (not fit)

Embedded coefficients:
      RSndSnd      CovSnd.1       PSAB-BA
 0.0004960916  0.0007342584 -0.0007378757
```

Note that the model is correctly identified as a skeleton, with a reminder that it was not fit to data. It also comes equipped with "default" coefficients, but these are not very useful: if a seed coefficient is not passed, `rem.dyad` always initializes with perturbed coefficients near zero. If one knows what coefficients one wants to embed in the skeleton, one can set them using the `coef.seed` argument.

Note that none of the inferential or other arguments to `rem.dyad` are needed here, since no fitting is done. Perhaps less obviously, we do not need to set the `ordinal` variable, since all REM simulation is done in continuous time. (The resulting trajectories can, of course, be interpreted ordinally, if the pacing constant used was arbitrary.)

3.1 Simulating from the model skeleton

Simulation from the model skeleton is then performed just as simulation with fitted model objects, except that one needs to pass the number of draws to take (`nsim`, which was optional before) and `coef` (unless one already embedded the coefficients one wants in the model object). Be sure to enter your coefficients in the order stored in the skeleton, which may not be the order you initially specified the effects! Let's see how this works, using our example:

```
set.seed(1331)
modsim <- simulate(modskel, nsim = 100, coef = c(0.25, -1, 4),
   covar = list(CovSnd = ModInt))
head(modsim)  #See the trajectory
```

```
            [,1] [,2] [,3]
[1,] 0.003446229   20    2
[2,] 0.004616100    9    6
[3,] 0.005879407   25   17
[4,] 0.007751835   10   23
[5,] 0.009434835    9   20
[6,] 0.015124865   14   23
```

```
grecip(as.sociomatrix.eventlist(modsim, 25), measure = "edgewise")  #Relatively reciprocal
```

```
      Mut
0.2444444
```

Any number of events may be simulated in this way.

3.2 Simulation with time-varying covariates

Time-varying covariates must, by definition, be specified at each time step. `rem.dyad` understands several covariate formats (see `?rem.dyad`):

- Single covariate, time invariant: For `CovSnd`, `CovRec`, or `CovInt`, a vector or single-column matrix/array. For `CovEvent`, an `n` by `n` matrix or array.
- Multiple covariates, time invariant: For `CovSnd`, `CovRec`, or `CovInt`, a two-dimensional `n` by `p` matrix/array whose columns contain the respective covariates. For `CovEvent`, a `p` by `n` by `n` array, whose first dimension indexes the covariate matrices.
- Single or multiple covariates, time varying: For `CovSnd`, `CovRec`, or `CovInt`, an `m` by `n` by `p` array whose respective dimensions index time (i.e., event number), covariate, and actor. For `CovEvent`, a `m` by `p` by `n` by `n` array, whose dimensions are analogous to the previous case.

Thus, in the time-varying case, the dimensions of the covariate object must be consistent with `nsim`. Let's see a simple example, involving a 10-person group with an initial activity covariate that decays with time. We will simulate for 100 time steps, so need to create a 100 by 1 by 10 matrix to hold the covariate (the ith slice containing the covariate values "going into'' the ith event). When creating the skeleton, it is currently necessary to pass covariates as if they are static, since there are not yet multiple time points; the checks that are performed to ensure that the covariates are legal will object if too many time points are given. (This will probably change in the future.) The time-varying version is then passed to the simulator.

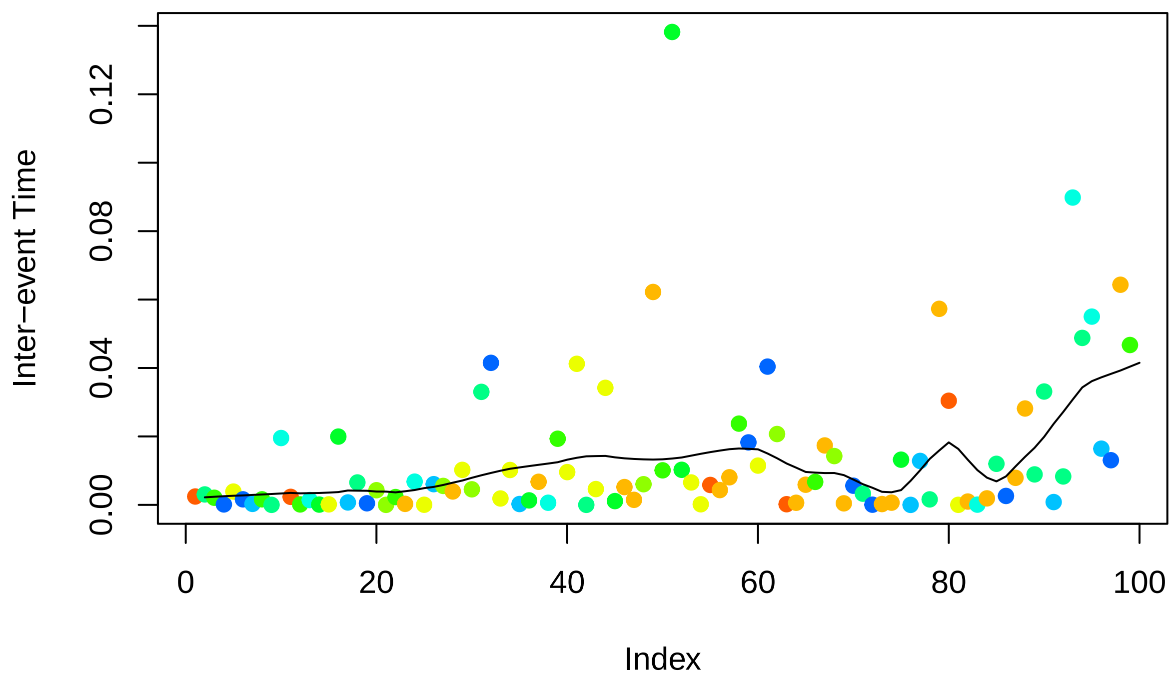
```
set.seed(1331)
# Set up the model
tcovar <- array(sweep(sapply(1:10, rep, 100), 1, 1/1.05^(0:99),
   "*"), dim = c(100, 10, 1))
SndInt <- rep(1, 10)
# Note that, in making the skeleton, we need to pass the
# covariates as if they are static - that's because the
# model doesn't contain time points yet.
```

```r
modskel2 <- rem.dyad(NULL, n = 10, effects = c("CovSnd", "CovInt"),
   coef.seed = c(-1, 1), covar = list(CovSnd = SndInt, CovInt = tcovar[1,
      , 1]))
```

```
NULL edgelist passed to rem.dyad - creating model skeleton.
Checking/prepping covariates.
```

```r
# Simulate draws
modsim2 <- simulate(modskel2, nsim = 100, covar = list(CovSnd = SndInt,
   CovInt = tcovar))

# Note that dynamics slow down, and participation evens out
plot(diff(modsim2[, 1]), col = hsv(modsim2[, 2]/10 * 0.6), pch = 19,
   ylab = "Inter-event Time")
lines(supsmu(x = 2:100, y = diff(modsim2[, 1])))
```



On average, dynamics slow down, as we would expect, and more low-numbered (redder) vertices interact after the initial period.

References

Butts, Carter T. (2008). "A Relational Event Framework for Social Action." *Sociological Methodology*, 38(1), 155-200.

Butts, Carter T. and Marcum, Christopher S. (2017). "A Relational Event Approach to Modeling Behavioral Dynamics." In Andrew Pilney and Marshall Scott Poole (Eds.), *Group Processes: Data-Driven Computational Approaches*. Springer.

Butts, Carter T.; Petrescu-Prahova, Miruna; and Cross, B. Remy. (2007). “Responder Communication Networks in the World Trade Center Disaster: Implications for Modeling of Communication Within Emergency Settings.” *Journal of Mathematical Sociology,* 31(2), 121-147.

Marcum, Christopher S. and Butts, Carter T. (2015). “Constructing and Modifying Sequence Statistics for `relevent` using `informR` in `R`.” *Journal of Statistical Software*, 64(5). [https://doi.org/10.18637/jss.v064.i05]

Bender-deMoll, Skye and McFarland, Daniel A. (2006). “The Art and Science of Dynamic Network Visualization.’ ’ *Journal of Social Structure*, 7. [https://www.cmu.edu/joss/content/articles/volume7/deMollMcFarland/]