Siena Advanced Users' Meeting 2014

Tom A.B. Snijders

University of Oxford
University of Groningen

Siena Advanced Users' Meeting Sunbelt, 2014

Overview

1. Goodness of fit: sienaTimeTest(), sienaGOF()
2. Model specification
3. ML
4. New effects
5. New features
6. Algorithmic issues
7. Documentation
8. Multilevel: sienaBayes()

Where to look?

Siena is an evolving endeavour, which may be hard to follow.

Where to look?

Siena is an evolving endeavour, which may be hard to follow.

- New versions are announced at the Siena/Stocnet discussion list,
  and at the *News* and *downloads* pages of
  `http://www.stats.ox.ac.uk/~snijders/siena/`
  The *News* page, and Appendix B in the manual,
  give description of changes in the new versions.

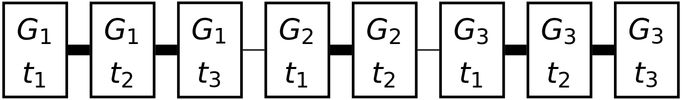
Where to look?

Siena is an evolving endeavour, which may be hard to follow.

- New versions are announced at the Siena/Stocnet discussion list,
  and at the *News* and *downloads* pages of
  `http://www.stats.ox.ac.uk/~snijders/siena/`
  The *News* page, and Appendix B in the manual,
  give description of changes in the new versions.
- Manual and help pages are sometimes updated.
  During 2013 there were some major improvements.
  Manual and help pages are complementary, use both!

Goodness of fit: timeTest

- sienaTimeTest() test for time homogeneity across periods.
- Since multi-group data are organized as multiple periods, homogeneity across groups can also be tested by sienaTimeTest().



Example: waves for three groups strung in a sequence,
groups 1 and 3 have 3 waves, group 2 has 2 waves;
the analyzed 2+1+2 periods are indicated as bold lines.

This can also be used for network rearrangements.

Goodness of fit: timeTest (contd.)

- sienaTimeTest() updated April 2013; now also contains effect-wise tests, groupwise tests (for group objects), automatic exclusion of collinear effects,
  and has prettier output and improved summary.
- Dummy variables can be added automatically by sienaTimeTest().
  The user has more control by including self-defined time variables (trends or jumps or dummies or whatever).

Goodness of fit: sienaGOF

- Use of auxiliary functions totally redone early 2013.
  The currently available set addresses distributions of degrees and behaviors.
  This is not supposed to be complete.
  You may add functions yourself!
  Note that the help page of sienaGOF-auxiliary has important further auxiliary functions and examples.
- Example: geodesic distances.
  These are now given as non-directed.

- If you have missing or structurally determined values, it is important to know how these are handled by sienaGOF().
  See the help page for sienaGOF():
  Tie variables that are structurally determined at the beginning of a period are used to replace observed values at the end of the period;
  tie variables that are structurally determined at the end, but not the beginning, of a period are used to replace simulated values at the end of the period.

- If you have missing or structurally determined values, it is important to know how these are handled by sienaGOF().
  See the help page for sienaGOF():
  Tie variables that are structurally determined at the beginning of a period are used to replace observed values at the end of the period;
  tie variables that are structurally determined at the end, but not the beginning, of a period are used to replace simulated values at the end of the period.
- descriptives.sienaGOF() gives numerical values of the sienaGOF plots (percentiles etc.).
- Improved plots if observations outside of simulated range.

Model specification

Since recently we are moving to a new standard for publications using Siena, where the fit for the degree and behavior distributions should be adequate.

It may not always be possible to achieve a fit with $p > 0.05$
for the Mahalanobis combination
of all statistics under consideration.
But it should be attempted,
and in my experience it will usually be possible,
to have the data within the confidence band of *plot.sienaGOF*.

Model specification (cont.)

- For large networks, try to find information about contact opportunities (same class, same department, distance, ...) and use this;
  perhaps also in interaction with reciprocity.

Model specification (cont.)

- For large networks, try to find information about contact opportunities (same class, same department, distance, ...) and use this;
  perhaps also in interaction with reciprocity.
- There are a large number of degree effects (see below).
  Pay attention to in- and out-isolates!
- For dyadic covariates, there are not so many effects, but you can transform dyadic covariates before creating the data set
  (e.g., reflection ego $\longleftrightarrow$ alter; twopaths; etc.)

Model specification (cont.)

- There may be actor covariates that have a strong effect in *some* ego-alter combination.
  How to find a good combination?
  E.g.: plot observed tie frequencies depending on ego and alter covariate values.
  Filter out missings and structurally determined values.

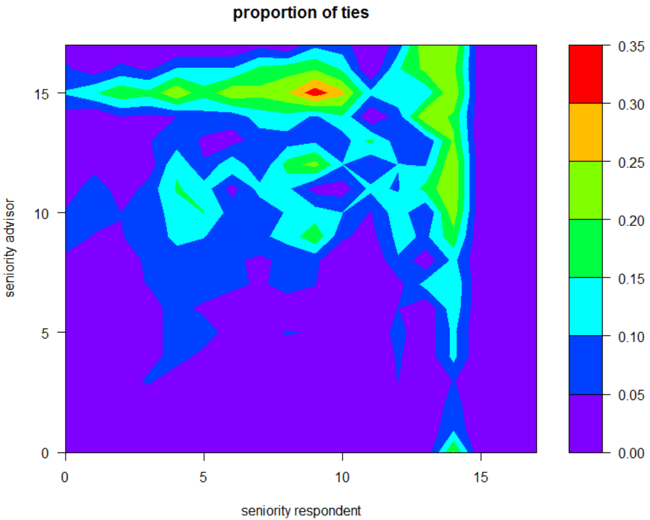
```
# Example of script for the main idea. I did not check that this is exactly right.

# Actor variable sen, adjacency matrix mat
# sen has values 0-20, but 16 and 19 are not observed
# throw together the two highest values 18 and 20
# Make a list with 18 categories of values for sen
scodes <- list()
for (i in 1:16) {scodes[[i]] <- i-1}
scodes[[17]] <- 17
scodes[[18]] <- c(18,20)
# check the list
vs <- sapply(1:18,function(i){sum(sen %in% scodes[[i]])})
# Construct the matrix with frequency of values of (sen[ego], sen[alter])
all.sen <- matrix(NA, 18, 18)
for (i in (1:18)){
   for (j in (1:18)){
      all.sen[i,j] <-
  sum(((1*(sen %in% scodes[[i]])) %o% (1*(sen %in% scodes[[j]]))), na.rm=TRUE)
  }}
# Construct the matrix with tie frequencies for values of (sen[ego], sen[alter])
ties.sen <- matrix(NA, 18, 18)
for (i in (1:18)){
   for (j in (1:18)){
      ties.sen[i,j] <-
    sum(vw2*((1*(sen %in% scodes[[i]])) %o% (1*(sen %in% scodes[[j]]))), na.rm=TRUE)
  }}
```

```
# Calculate the proportion; add 1e-10 to avoid division by 0.
prop.sen <- ties.sen/(all.sen + 1e-10)
contour(x=0:17, y= 0:17, z=all.sen,
levels= c(10,50*(0:8)),
xlab="seniority respondent", ylab="seniority advisor",
main="combinations")

bowrain <- function(n){rainbow(n, start=0, end=3/4)[n:1]}
# invert order of colors, do not use beyond blue

filled.contour(x=0:17, y= 0:17, z=prop.sen, color.palette=bowrain,
             nlevels=10, xlab="seniority respondent", ylab="seniority advisor",
main="proportion of ties") # saved as sen_ties.png
# Not sure this is exactly the script to produce sen_ties.png
```

This gave the plot on the following page.

As a conclusion the added covariate effects were:

higher, sen[ego], sen[alter], their squares, and interaction.


proportion of ties
seniority advisor
seniority respondent
0
5
10
15
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00

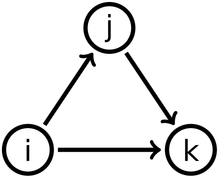
Model specification (cont.)

For SAOMs, the hierarchy principle in model specification also should be followed (unless there are good reasons not to):

1. If a product interaction of two actor variables is included, then also the ego and alter effects should be included.
2. If a structural effect is included based on a subgraph count (e.g., a triadic effect), then also the counts of sub-configurations of this subgraph should be included.

E.g., for a test of transitivity by means of transitive triplets ($i \rightarrow j \rightarrow k,\ i \rightarrow k$),

this means that the counts of
in-2-stars ($j \rightarrow k,\ i \rightarrow k$)
(indegree-popularity),
out-twostars ($i \rightarrow j,\ i \rightarrow k$)
(outdegree activity),



and two paths ($i \rightarrow j \rightarrow k$) (indegree activity or outdegree popularity) should be included.

We can be flexible and replace this by the square root versions; but if degree-related effects are excluded altogether then a large number of transitive triplets could be "caused" by, e.g., outdegree popularity and some random transitive closures without a systematic tendency toward transitivity.

This issue with transitivity was noted already by Feld and Elmore (*Soc Psy Quarterly*, 1982).

For mixed triadic effects this hierarchy principle is very important in practice.
There have been various cases with unexpected results (e.g., a negative parameter for mixed transitive closure) because the control for lower-order configurations was inadequate.

For example (see the manual for the definition of effects), the mixed $WW \Rightarrow X$ closure ("closure") has as lower-order configurations the outdegree popularity or indegree activity for W; the effect of out-degree in W on X-activity (“outActIntn"); and the effect of in-degree in W on X-popularity (“inPopIntn").

It is evident that this can lead to very elaborate and heavy models. If the lower-order effects are non-significant they might be dropped in a final model. But they should at least be checked, and in a publication it should be mentioned that this was done.

Maximum Likelihood

I have mentioned before that the Method of Moments (MoM) is not much less efficient than Maximum Likelihood (ML).

For data sets with more than one dependent variable (networks and behavior; multiple networks), however, ML may have clearly higher efficiency.

Therefore, for small data sets with multiple dependent variables, consider ML!

There are plans to improve MoM for multiple dependent variables, and for some speed improvements for ML.

Parallelization for ML operates only by periods (note that multiple groups, such as in sienaBayes(), are implemented by multiple periods.)

Diffusion of innovations

(Charlotte Greenan, *JRSS-A*, to appear; also see *J. Res. Adolesc.*, 2013.)

Behavior effects of covariates and of influence put in rate function to combine proportional odds model with endogenous network dynamics.

- average exposure effect on rate xxxxxx: avExposure
- susceptibility to av. exp. by indegree effect on rate xxxxxx: suscptAvIn
- total exposure effect on rate xxxxxx: totExposure
- infection by indegree effect on rate xxxxxx: infectIn
- infection by outdegree effect on rate xxxxxx: infectOut
- susceptibility to av. exp. by zzzzzz effect on rate xxxxxx: suscptAvCovar
- infection by zzzzzz effect on rate xxxxxx: infectCovar
- more to be added.

New rate effect

- log of outdegree effect on rate, outRateLog
  (since the log-link function is used $\Rightarrow$
  rate itself depends on power of (outdegree + 1))

Isolation and low degrees

Some effects are defined for isolation, others against isolatio; some for in-, some for out-, some for total isolation.

- out-isolate: outTrunc (parameter = 1)
- in-isolate: antiInIso (others' indegree ≥ 1)
- network-isolate: isolateNet (ego isolation)
- anti-isolates: antiIso (against total isolation of others)
- anti-in-isolates: antiInIso (others' indegree ≥ 1)

Then some variations:

- outTrunc2: duplication of outTrunc (different par.)
- anti-in-near-isolates: antiInIso2 (others' indegree ≥ 2)
- in-isolate Outdegree: inIsDegree (interaction in-isolate $\times$ outdegree)

Isolation and low degrees: two-mode networks

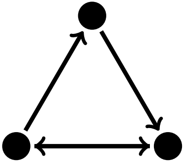
For two-mode networks:

- Degree effects for two-mode networks were added.
- In-isolation and near in-isolation (antiInIso, antiInIso2) particularly important effects for two-mode networks.
- Further: bug in starting values for two-mode corrected.

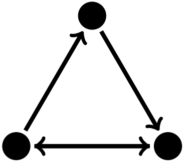
Note that for two-mode networks,
avoiding low indegrees may be especially meaningful.

Fancy triadic effects

Theoretical elaboration by Per Block:
reciprocity × transitivity (transRecTrip)
often better than three-cycles.



Fancy triadic effects



Theoretical elaboration by Per Block:
reciprocity × transitivity (transRecTrip)
often better than three-cycles.

- transitive reciprocated triplets, transRecTrip (3-cycles ↓)
- transitive triplets xxxxxx similarity, simXTransTrip
- transitive triplets same xxxxxx, sameXTransTrip
- transitive triplets jumping xxxxxx, jumpXTransTrip
- homogeneous covariate × transitive triplets, homXTransTrip
- mixed WX ⇒ X closure, homogeneous on V, homWXClosure

GWESP

GWESP (*geometrically weighted edgewise shared partners*) (cf. ERGM) is intermediate between transTrip and transTies.

Was redone with the help of Nynke Niezink for 1.1-254: parameters brought in line with usual definition.

$$\mathrm{GWESP}(i,\alpha) = \sum_j x_{ij}\, e^{\alpha} \left\{ 1 - \left(1 - e^{-\alpha}\right)^{\sum_h x_{ih} x_{hj}} \right\} .$$

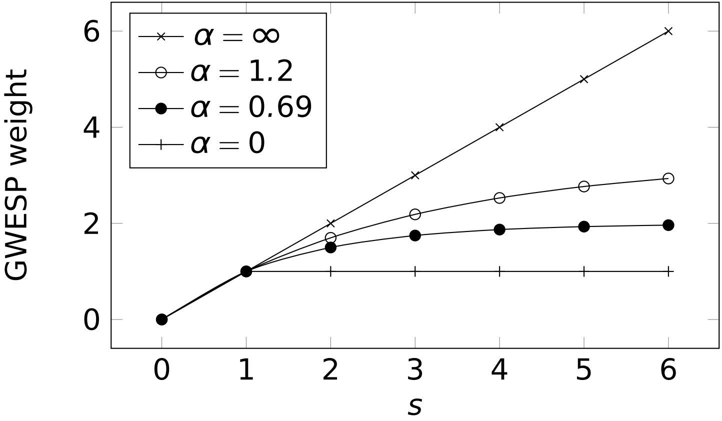
for $\alpha \geq 0$ (effect parameter = $100 \times \alpha$).

Correspondence between old (α') and new (α) parameters:

$$e^{-\alpha} + e^{-\alpha'} = 1$$

For default $\alpha = \log(2)$ this means no change!

GWESP (contd.)



Weight of tie $i \rightarrow j$ for $s = \sum_h x_{ih} x_{hj}$ two-paths.

The current implementation of GWESP is not a true evaluation effect:

For creation of a new tie,
only its role as $i \to j$ in the formula is counted,
not its role as $i \to h$.

Same target statistic, other parameter estimate.
GWESP sometimes yields better fit than transTrip or transTies.

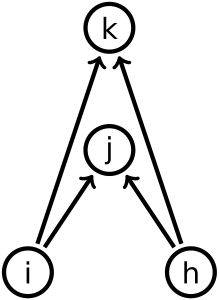
Further new network effects

A number of effects for directed one-mode networks have been extended to non-directed and to two-mode networks.

Further new network effects

A number of effects for directed one-mode networks have been extended to non-directed and to two-mode networks.

E.g., 4-cycles;
but for directed, only as sharedPop.



Note that the list of all implemented effects (for the various types of dependent variables) can be obtained from
effectsDocumentation()

and a list of all effects in object myeff from
effectsDocumentation(myeff)

New behavior effects

There were already interactions of influence with ego characteristics;
now interactions are also implemented with alter characteristics.

Ego: avSimEgoX, totSimEgoX, avAltEgoX

Alter: avSimAltX, totSimAltX, avAltAltX

New behavior effects

There were already interactions of influence with ego characteristics;
now interactions are also implemented with alter characteristics.

Ego: avSimEgoX, totSimEgoX, avAltEgoX

Alter: avSimAltX, totSimAltX, avAltAltX

For these interactions, it may be preferable *not to center* the actor covariates.

This can be achieved by

coCovar(..., centered=FALSE)
varCovar(..., centered=FALSE)

New features

- Possibility not to center actor covariates.

New features

- Possibility not to center actor covariates.
- setEffect() and prevAns in siena07() now handle user-defined interactions better.

New features

- Possibility not to center actor covariates.
- setEffect() and prevAns in siena07() now handle user-defined interactions better.
- Note the possibility of allowOnly=FALSE in sienaDependent(), to turn off automatic upOnly or downOnly simulations.

New features

- Possibility not to center actor covariates.
- setEffect() and prevAns in siena07() now handle user-defined interactions better.
- Note the possibility of allowOnly=FALSE in sienaDependent(), to turn off automatic upOnly or downOnly simulations.
- Note the possibility of simOnly=TRUE in sienaAlgorithmCreate() for simulation without estimation.

New features

- Possibility not to center actor covariates.
- setEffect() and prevAns in siena07() now handle user-defined interactions better.
- Note the possibility of allowOnly=FALSE in sienaDependent(), to turn off automatic upOnly or downOnly simulations.
- Note the possibility of simOnly=TRUE in sienaAlgorithmCreate() for simulation without estimation.
- We are thinking about changes in terminology: e.g., ‘maintenance’ instead of ‘endowment’; ‘sienaEstimate()’ instead of ‘siena07()’ ...?

Algorithm; siena07(), sienaAlgorithmCreate()

- Operation of siena07() with MaxDegree option corrected.

Algorithm; siena07(), sienaAlgorithmCreate()

- Operation of siena07() with MaxDegree option corrected.
- Some bugs corrected that occurred mainly for two-mode networks with a lot of structural values.
- Better avoidance of execution errors in case of linearly dependent results.
- Composition change now forces non-conditional estimation.
- Dolby
- diagonalize
- reduceg
- tconv added to sienaFit objects, allowing, e.g., the following:

```
siena07ToConvergence <- function(alg, dat, eff){
 numr <- 0
 ans <- siena07(alg, data=dat, effects=eff)
 repeat {
   numr <- numr+1
   maxt <- max(abs(ans$tconv[!eff$fix[eff$include]]))
   # convergence indicator, excluding fixed effects
   cat(numr, maxt,"\n")
   if (maxt < 0.10) {break} # success
   if (maxt > 5)    {break} # divergence
   if (numr > 100)  {break} # too many trials
   ans <- siena07(alg, data=dat, effects=eff,
                                    prevAns=ans) }
 ans
}
```

Documentation and help

Many help pages improved (not yet all).

Start with reorganization of manual
(with help from Zsófia Boda and András Vörös)

Function effectsDocumentation(myeff)
makes a list of all effects provided in myeff.

Documentation and help

Many help pages improved (not yet all).

Start with reorganization of manual
(with help from Zsófia Boda and András Vörös)

Function effectsDocumentation(myeff)
makes a list of all effects provided in myeff.

In the manual, Section 6.4 (p. 57), an explanation is given of how to import tables produced by siena.table() into MS-Word.

Various small changes in output
(print functions for Siena data objects; better printing of results for simulation-only use of siena07(); etc.)

Names *sienaNet* changed to *sienaDependent*;
sienaModelCreate changed to *sienaAlgorithmCreate*.

sienaBayes()

(Koskinen & Snijders)

Multi-group random effects multilevel longitudinal network analysis.

- It works.

sienaBayes()

(Koskinen & Snijders)

Multi-group random effects multilevel longitudinal network analysis.

- It works. Paper still to be finished.

sienaBayes()

(Koskinen & Snijders)

Multi-group random effects multilevel longitudinal network analysis.

- It works. Paper still to be finished.
- Takes a long time.

sienaBayes()

(Koskinen & Snijders)

Multi-group random effects multilevel longitudinal network analysis.

- It works. Paper still to be finished.
- Takes a long time. Parallellization helps.

sienaBayes()

(Koskinen & Snijders)

Multi-group random effects multilevel longitudinal network analysis.

- It works. Paper still to be finished.
- Takes a long time. Parallellization helps.
- Further experience is needed.

sienaBayes()

(Koskinen & Snijders)

Multi-group random effects multilevel longitudinal network analysis.

- It works. Paper still to be finished.
- Takes a long time. Parallellization helps.
- Further experience is needed.
- Presentation on Friday afternoon.