

1 **Conducting social network analysis with animal** 2 **telemetry data: applications and methods using `spatsoc`**

3 Alec L. Robitaille ^a, Quinn M.R. Webber ^b & Eric Vander Wal ^{a,b}

4 ^a Department of Biology, Memorial University of Newfoundland, St. John's, NL, Canada

5 ^b Cognitive and Behavioural Ecology Interdisciplinary Program, Memorial University of
6 Newfoundland, St. John's, NL, Canada

7 **Summary**

- 8 1. We present `spatsoc`: an R package for conducting social network analysis with
9 animal telemetry data.
- 10 2. Animal social network analysis is a method for measuring relationships between
11 individuals to describe social structure. Using animal telemetry data for social
12 network analysis requires functions to generate proximity-based social networks that
13 have flexible temporal and spatial grouping. Data can be complex and relocation
14 frequency can vary so the ability to provide specific temporal and spatial thresholds
15 based on the characteristics of the species and system is required.
- 16 3. `spatsoc` fills a gap in R packages by providing flexible functions, explicitly for
17 animal telemetry data, to generate gambit-of-the-group data, perform data-stream
18 randomization and generate group by individual matrices.

19 4. The implications of spatSOC are that current users of large animal telemetry or
20 otherwise georeferenced data for movement or spatial analyses will have access to
21 efficient and intuitive functions to generate social networks.

22 **Introduction**

23 Animal social network analysis is a method for measuring the relationships between
24 individuals to describe social structure (Wey et al. 2008; Croft, James, and Krause 2008).
25 Association networks are built from a set of observed elements of social community
26 structure and are useful to understand a variety of ecological and behavioural processes,
27 including disease transmission, interactions between individuals and community structure
28 (Pinter-Wollman et al. 2014). Among the most common types of social network data
29 collection is gambit-of-the-group, where individuals observed in the same group are
30 assumed to be associating or interacting (Franks, Ruxton, and James 2010). Similar to
31 gambit-of-the-group, proximity based social networks (PBSNs) are association networks
32 based on close proximity between individuals (Spiegel et al. 2016). PBSNs rely on spatial
33 location datasets that are typically acquired by georeferenced biologging methods such as
34 radio-frequency identification tags, radiotelemetry, and Global Positioning System (GPS)
35 devices (hereafter, animal telemetry).

36 Biologging using GPS devices allow simultaneous spatiotemporal sampling of multiple
37 individuals in a group or population, thus generating large datasets which may otherwise
38 be challenging to collect. The advent of biologging technology allows researchers to
39 study individuals of species that range across large areas, migrate long distances, or
40 spend time in inaccessible areas (Cagnacci et al. 2010; Cooke et al. 2013; Hebblewhite

41 and Haydon 2010). Moreover, the recent increase in the number of studies using GPS
42 telemetry to study movement ecology (Kays et al. 2015; Tucker et al. 2018) indicates the
43 potential for a large number of existing datasets that may be retro-actively analyzed to
44 test a priori hypotheses about animal social structure. As animal telemetry data have
45 become more accessible and finer scaled, a number of techniques and methods have been
46 developed to quantify various aspects of animal social structure. These include dynamic
47 interaction networks (Long et al. 2014), PBSNs (Spiegel et al. 2017) and the development
48 of traditional randomization techniques to assess non-random structure of PBSNs
49 constructed using animal telemetry data (Spiegel et al. 2016). Despite the recent increase
50 in the number of studies using animal telemetry data and GPS relocation data (Webber
51 and Vander Wal 2018), there is no comprehensive R package that generates PBSNs using
52 animal telemetry data.

53 Here, we present `spatsoc`, a package developed for the R programming language (R
54 Core Team 2018) to (i) convert animal telemetry data into gambit-of-the-group format to
55 build PBSNs, (ii) implement data-stream social network randomization methods of
56 animal telemetry data (Farine and Whitehead 2015; Spiegel et al. 2016), and (iii) provide
57 flexible spatial and temporal grouping of individuals from large datasets. Animal
58 telemetry data can be complex both temporally (e.g., data can be partitioned into
59 monthly, seasonal or yearly segments) and spatially (e.g., subgroups, communities or
60 populations). Functions in `spatsoc` were developed taking these complexities into
61 account and provide users with flexibility to select relevant parameters based on the
62 biology of their study species and systems and test the sensitivity of results across spatial
63 and temporal scales.

64 **Functions**

65 The `spatsoc` package provides functions for using animal telemetry data to generate
66 PBSNs. Relocations are converted to `gambit-of-the-group` using grouping functions
67 which can be used to build PBSNs. Raw data streams can be randomized where animal
68 telemetry data is swapped between individuals at hourly or daily scales (Farine and
69 Whitehead 2015), or within individuals using a daily trajectory method (Spiegel et al.
70 2016).

71 **Grouping**

72 `Gambit-of-the-group` data is generated from animal telemetry data where individuals are
73 grouped based on temporal and spatial overlap (Figure 1). The `spatsoc` package provides
74 one temporal grouping function:

75 1. `group_times` groups animal telemetry relocations into time groups. The function
76 accepts date time formatted data and a temporal threshold argument. The temporal
77 threshold argument allows users to specify a time window within which relocations
78 are grouped, for example 5 minutes, 2 hours or 10 days. We recommend this
79 temporal threshold is based on the nuances of the animal telemetry data, study
80 species and system.

81 The `spatsoc` package provides three spatial grouping functions:

82 1. `group_pts` compares the Euclidean distance between animal telemetry relocations
83 (Figure 2 - Panel A). Relocations for all individuals within each time group will be

84 grouped based on spatial proximity. Spatial proximity is defined by the user-
85 specified distance threshold.

86 2. `group_lines` groups overlapping movement trajectories generated from animal
87 telemetry data. Movement trajectories for each individual within each time group,
88 e.g. 8 hours, 1 day or 20 days, are generated and grouped based on spatial overlap,
89 optionally within the user-specified distance threshold buffers around each trajectory
90 (Figure 2 - Panel B).

91 3. `group_polys` generates and groups overlapping home ranges using kernel
92 utilization distributions or minimum convex polygons generated in `adehabitatHR`
93 of individuals and optionally returns a measure of proportional area overlap.

94 For all three spatial grouping functions, individuals that are not within the user-specified
95 distance threshold or that do not overlap with any other individuals are assigned to a
96 group on their own. Functions provided by `spatsoc` emphasize flexibility to allow users
97 the ability to modify functions to better suit their study systems. The temporal threshold
98 argument of `group_times` accepts units of minutes, hours and days to consider spatial
99 grouping at different temporal scales. For example, grouping trajectories with
100 `group_lines` is compatible with daily or weekly time group thresholds while point
101 grouping with `group_pts` is compatible with minute or hourly time group thresholds.

102 **Randomizations**

103 Randomization procedures in social network analysis are important to test assumptions of
104 spatial and temporal non-independence of social association data (Farine and Whitehead
105 2015). Data-stream randomization is the recommended randomization technique for

106 social network users (Farine and Whitehead 2015) and involves swapping individuals and
107 group observations within or between temporal groups and individuals. Animal telemetry
108 data has inherent temporal structure and is well suited to randomization methods. The
109 `spatsoc` package provides three data-stream randomization methods:

- 110 1. Step - randomizes identities of animal telemetry relocations between individuals
111 within each time step.
- 112 2. Daily - randomizes daily animal telemetry relocations between individuals,
113 preserving the order of time steps.
- 114 3. Trajectory - randomizes daily trajectories generated from animal telemetry
115 relocations within individuals (Spiegel et al. 2016).

116 The randomizations functions return the input data with random fields appended, ready
117 to use by the grouping functions or to build social networks. Step and daily methods
118 return a “randomID” field that can be used in place of the ID field and the trajectory
119 method returns a “randomDatetime” that can be used in place of the datetime field. The
120 randomizations function in `spatsoc` allow users to split randomizations between
121 spatial or temporal subgroups to ensure that relocations are only swapped between or
122 within relevant individuals.

123 **Using `spatsoc` in social network analysis**

124 `spatsoc` is integrated with social network analysis in R in three main steps: 1) generate
125 gambit-of-the-group data by temporal and spatial grouping, 2) generate group by
126 individual matrices and 3) PBSN data-stream randomization. Users should first determine

127 relevant temporal and spatial grouping thresholds based on details from their study
128 species and systems. These thresholds depend on the fix rate of animal telemetry devices,
129 movement rates of study species, and other biological details of each species and system
130 as well as the questions or hypotheses of interest. In addition, thresholds selected for
131 temporal and spatial grouping must be relevant to each other. For example, point based
132 spatial grouping with `group_times` may only be relevant with temporal thresholds in
133 units of hours or minutes while line and polygon based spatial grouping with
134 `group_lines` and `group_polys` may only be relevant with temporal thresholds in units
135 of hours or days.

136 **Generating networks**

137 Here, we will provide an example of point based spatial grouping with `spatsoc`'s
138 "Newfoundland Bog Cow" example animal telemetry data. The data has a relocation rate
139 of 2 hours but in this case, we consider relocations that occur within 5 minutes of each
140 other. The columns in this data are "id" (character type), "datetime" (character type), "X"
141 and "Y" Universal Transverse Mercator (UTM) coordinates (numeric type). The
142 character type "datetime" will be converted to POSIXct, R's date time format, the
143 required type for `spatsoc`'s temporal grouping function `group_times`. The coordinates
144 "X" and "Y" must be in a projected coordinate system with units of meters. In this case,
145 the coordinate system is UTM Zone 21 N. We will use a spatial distance threshold of 50
146 m given the size and behaviour of the study species. The combination of spatial and
147 temporal thresholds means that any individuals within 50 m of each other within 5
148 minutes will be assigned to the same group.

```
149 # Load packages
150 libs <- c('spatsoc', 'data.table', 'asnipe', 'igraph')
151 lapply(libs, require, character.only = TRUE)
152
153 # Read spatsoc's example data as a data.table
154 DT <- fread(system.file("extdata", "DT.csv", package = "spatsoc"))
155
156 # Cast datetime column to POSIXct
157 DT[, datetime := as.POSIXct(datetime)]
158
159 # Temporal groups
160 group_times(DT, datetime = 'datetime', threshold = '5 minutes')
161
162 # Spatial groups using 'timegroup' generated from group_times
163 group_pts(
164   DT,
165   threshold = 50,
166   id = 'ID',
167   coords = c('X', 'Y'),
168   timegroup = 'timegroup'
169 )
```

170 After the temporal and spatial grouping is completed with `group_times` and `group_pts`,
171 a group by individual matrix is generated (described by Farine and Whitehead (2015)). A

172 group by individual matrix forms columns of individuals and rows of groups and a
173 boolean will indicate membership of each individual to a group.

```
174 gbi <- get_gbi(DT, group = 'group', id = 'ID')
```

175 After generating the group by individual matrix, it is passed directly to `asnipe`, the
176 animal social network package (Farine 2013), to generate a proximity based social
177 network. Note, in this example we use the simple ratio index (SRI) as an association
178 index because all individuals are correctly identified and observed at each relocation
179 event (i.e. the equivalent to an observational period for networks generated using focal
180 observations).

```
181 net <- get_network(gbi, data_format = "GBI", association_index = "SRI")
```

182 **Data-stream randomization**

183 To perform network data-stream permutations, the `randomizations` function is used to
184 permute spatial and temporal groupings and rebuild PBSNs at each iteration. In this
185 example, we use the “step” method to randomize between individuals at each time step
186 for 500 iterations. The output `randStep` contains the observed and randomized data and
187 can subsequently be used to generate group by individual matrices, networks and
188 calculate network metrics.

```
189 # Calculate year column  
190 DT[, yr := year(datetime)]  
191  
192 # Step type randomizations
```

```
193 # providing 'timegroup' (from group_times) as datetime
194 # splitBy = 'yr' to randomize only within year
195 randStep <- randomizations(
196   DT,
197   type = 'step',
198   id = 'ID',
199   group = 'group',
200   datetime = 'timegroup',
201   iterations = 500,
202   splitBy = 'yr'
203 )
204
205 # Create a data.table of unique combinations of iteration and year,
206 # including observed and random rows
207 iterYearLs <- unique(randStep[, .(iteration, yr)])
208
209 # Generate group by individual matrix
210 # for each combination of iteration and year
211 # 'group' generated by spatsoc::group_pts
212 # 'randomID' used instead of observed ID (type = 'step')
213 gbiLs <- mapply(FUN = function(i, y) {
214   get_gbi(randStep[iteration == i & yr == y],
215           'group', 'randomID')
216 },
217 i = iterYearLs$iter,
```

```
218   y = iterYearLs$yr
219 )
220
221 # Generate a List of random networks
222 netLs <- lapply(gbiLs, FUN = get_network,
223               data_format = "GBI", association_index = "SRI")
224
225 # Generate graph and calculate network metrics
226 mets <- lapply(seq_along(netLs), function(n) {
227   g <- graph.adjacency(netLs[[n]], 'undirected',
228                       diag = FALSE, weighted = TRUE)
229
230   data.table(
231     centrality = evcent(g)$vector,
232     strength = graph.strength(g),
233     ID = names(degree(g)),
234     iteration = iterYearLs$iter[[n]],
235     yr = iterYearLs$yr[[n]]
236   )
237 })
238
239 # Observed and random for all individuals across all iterations and
240 years
241 out <- rbindlist(mets)
242
```

```
243 # Split observed and random
244 out[, observed := ifelse(iteration == 0, TRUE, FALSE)]
245
246 # Mean values for each individual and year, by observed/random
247 meanMets <- out[, lapply(.SD, mean), by = .(ID, yr, observed),
248                      .SDcols = c('centrality', 'strength')]
```

249 The `splitBy` argument can be used in randomizations and grouping functions to
250 specify spatial, e.g. groups or populations, or temporal groups, e.g. weekly, monthly,
251 yearly, by which PBSNs will be generated. For example, in large datasets with
252 individuals in two distinct populations with data over many years, users may use the
253 `splitBy` argument to generate PBSNs for each population by year combination as opposed
254 to either generating each PBSN separately or using loops.

255 **Implications**

256 `spatsoc` represents a novel integration of tools for generating PBSNs from animal
257 telemetry data. The grouping and randomization functions allow users to efficiently and
258 rapidly generate a large number PBSNs within the `spatsoc` environment. `spatsoc` will be
259 of interest and use to a wide range of behavioural ecologists who either already use social
260 network analysis or those who typically work with GPS relocation but are interested in
261 becoming social network users. We advocate for the use of `spatsoc` in conjunction with
262 other R packages, such as `asnipe` (Farine 2013), `igraph` (Csardi and Nepusz 2006), to
263 facilitate greater sharing of computational and statistical efficiencies and ideas for users
264 of social network analysis.

265 **Resources**

266 `spatsoc` is a free and open source software available on CRAN (stable release) and at
267 (development version). It is licensed under the GNU General Public License 3.0.
268 `spatsoc` depends on other R packages: `data.table` (Dowle and Srinivasan 2018),
269 `igraph` (Csardi and Nepusz 2006), `rgeos` (Bivand and Rundel 2018), `sp` (Bivand,
270 Pebesma, and Gomez-Rubio 2013) and `adehabitatHR` (Calenge 2006). Documentation
271 of all functions and detailed vignettes can be found on the companion website at
272 spatsoc.gitlab.io. Development of `spatsoc` welcomes contribution of feature requests,
273 bug reports and suggested improvements through the issue board at
274 <https://gitlab.com/robit.a/spatsoc/issues> }.

275 **Acknowledgements**

276 We thank all members of the Wildlife Evolutionary Ecology Lab, including Juliana
277 Balluffi-Fry, Sana Zabihi-Seissan, Erin Koen, Mike Laforge, Christina Prokopenko, Julie
278 Turner, Levi Newediuk and Chris Hart for their comments on previous versions of this
279 manuscript. We thank Tyler Bonnell, Martin Leclerc and Shane Frank for testing the
280 package ahead of its release. We also thank the rOpenSci organization for their package
281 onboarding process including rOpenSci reviewers for their code review, which
282 contributed to improving this package. Funding for this study was provided by a Vanier
283 Canada Graduate Scholarship to QMRW and a NSERC Discovery Grant to EVW.

284 **Author contributions**

285 ALR, QMRW, and EVW conceived of the original package concept. ALR developed the
286 package. ALR and QMRW drafted the manuscript and all co-authors contributed
287 critically to the drafts and gave final approval for publication.

288 **Data accessibility**

289 The data used for illustration is distributed with the package as example data. After
290 installing the package, the data can be viewed with:

```
291 data.table::fread(system.file("extdata", "DT.csv", package =  
292 "spatsoc"))`
```

293 **Citation**

294 Users of spatsoc should cite this article directly. A formatted citation and BibTex entry
295 is provided in R:

```
296 citation('spatsoc')
```

297

298 **References**

- 299 Bivand, Roger, and Colin Rundel. 2018. *rgeos: Interface to Geometry Engine - Open*
300 *Source ('GEOS')*. <https://cran.r-project.org/package=rgeos>.
- 301 Bivand, Roger S, Edzer Pebesma, and Virgilio Gomez-Rubio. 2013. *Applied spatial data*
302 *analysis with R, Second edition*. Springer, NY. <http://www.asdar-book.org/>.
- 303 Cagnacci, Francesca, Luigi Boitani, Roger A. Powell, and Mark S. Boyce. 2010. “Animal
304 ecology meets GPS-based radiotelemetry: A perfect storm of opportunities and
305 challenges.” *Philosophical Transactions of the Royal Society B: Biological Sciences* 365
306 (1550): 2157–62. <https://doi.org/10.1098/rstb.2010.0107>.
- 307 Calenge, C. 2006. “The package adehabitat for the R software: tool for the analysis of
308 space and habitat use by animals.” *Ecological Modelling* 197: 1035.
- 309 Cooke, Steven J., Jonathan D. Midwood, Jason D. Thiem, Peter Klimley, Martyn C.
310 Lucas, Eva B. Thorstad, John Eiler, Chris Holbrook, and Brendan C. Ebner. 2013.
311 “Tracking animals in freshwater with electronic tags: Past, present and future.” *Animal*
312 *Biotelemetry* 1 (1): 1–19. <https://doi.org/10.1186/2050-3385-1-5>.
- 313 Croft, D P, R James, and J Krause. 2008. *Exploring Animal Social Networks*. Princeton
314 University Press. <https://books.google.ca/books?id=CViwRbohZkC>.
- 315 Csardi, Gabor, and Tamas Nepusz. 2006. “The igraph software package for complex
316 network research.” *InterJournal Complex Systems*: 1695. <http://igraph.org>.
- 317 Dowle, Matt, and Arun Srinivasan. 2018. *data.table: Extension of 'data.frame'*.
318 <https://cran.r-project.org/package=data.table>.

- 319 Farine, Damien R. 2013. “Animal social network inference and permutations for
320 ecologists in R using asnipe.” *Methods in Ecology and Evolution* 4 (12): 1187–94.
321 <https://doi.org/10.1111/2041-210X.12121>.
- 322 Farine, Damien R., and Hal Whitehead. 2015. “Constructing, conducting and interpreting
323 animal social network analysis.” *Journal of Animal Ecology* 84 (5): 1144–63.
324 <https://doi.org/10.1111/1365-2656.12418>.
- 325 Franks, Daniel W., Graeme D. Ruxton, and Richard James. 2010. “Sampling animal
326 association networks with the gambit of the group.” *Behavioral Ecology and*
327 *Sociobiology* 64 (3): 493–503. <https://doi.org/10.1007/s00265-009-0865-8>.
- 328 Hebblewhite, Mark, and Daniel T. Haydon. 2010. “Distinguishing technology from
329 biology: A critical review of the use of GPS telemetry data in ecology.” *Philosophical*
330 *Transactions of the Royal Society B: Biological Sciences* 365 (1550): 2303–12.
331 <https://doi.org/10.1098/rstb.2010.0087>.
- 332 Kays, Roland, Margaret C. Crofoot, Walter Jetz, and Martin Wikelski. 2015. “Terrestrial
333 animal tracking as an eye on life and planet.” *Science* 348 (6240): aaa2478.
334 <https://doi.org/10.1126/science.aaa2478>.
- 335 Long, Jed A., Trisalyn A. Nelson, Stephen L. Webb, and Kenneth L. Gee. 2014. “A
336 critical examination of indices of dynamic interaction for wildlife telemetry studies.”
337 *Journal of Animal Ecology* 83 (5): 1216–33. <https://doi.org/10.1111/1365-2656.12198>.
- 338 Pinter-Wollman, Noa, Elizabeth A. Hobson, Jennifer E. Smith, Andrew J. Edelman,
339 Daizaburo Shizuka, De SilvaShermin, James S. Waters, et al. 2014. “The dynamics of

- 340 animal social networks: Analytical, conceptual, and theoretical advances.” *Behavioral*
341 *Ecology* 25 (2): 242–55. <https://doi.org/10.1093/beheco/art047>.
- 342 R Core Team. 2018. *R: A Language and Environment for Statistical Computing*. Vienna,
343 Austria: R Foundation for Statistical Computing. <https://www.r-project.org/>.
- 344 Spiegel, Orr, Stephan T. Leu, Andrew Sih, and C. Michael Bull. 2016. “Socially
345 interacting or indifferent neighbours? Randomization of movement paths to tease apart
346 social preference and spatial constraints.” *Methods in Ecology and Evolution* 7 (8): 971–
347 79. <https://doi.org/10.1111/2041-210X.12553>.
- 348 Spiegel, Orr, Andrew Sih, Stephan T. Leu, and C. Michael Bull. 2017. “Where should we
349 meet? Mapping social network interactions of sleepy lizards shows sex-dependent social
350 network structure.” *Animal Behaviour* 136: 207–15.
351 <https://doi.org/10.1016/j.anbehav.2017.11.001>.
- 352 Tucker, Marlee A., Katrin Böhning-Gaese, William F. Fagan, John M. Fryxell, Van
353 MoorterBram, Susan C. Alberts, Abdullahi H. Ali, et al. 2018. “Moving in the
354 Anthropocene: Global reductions in terrestrial mammalian movements.” *Science* 359
355 (6374): 466–69. <https://doi.org/10.1126/science.aam9712>.
- 356 Webber, Quinn MR, and Vander WalEric. 2018. “Trends and Perspectives on the Use of
357 Social Network Analysis in Behavioural Ecology: A Bibliometric Approach.” *bioRxiv*,
358 379008.
- 359 Wey, Tina, Daniel T. Blumstein, Weiwei Shen, and Ferenc Jordán. 2008. “Social
360 network analysis of animal behaviour: a promising tool for the study of sociality.” *Animal*
361 *Behaviour* 75 (2): 333–44. <https://doi.org/10.1016/j.anbehav.2007.06.020>.

363 Figure 1: Broad overview flowchart of recommended usage of grouping functions. A
364 temporal grouping function (`group_times`) is followed by one of three spatial grouping
365 functions. Panel A) `group_pts` compares the Euclidean distance between animal
366 telemetry relocations and groups points based on a threshold distance with an output
367 containing temporal and spatial groups for each individual at each relocation; Panel B)
368 `group_lines` groups overlapping movement trajectories generated from GPS relocations
369 with an output containing temporal and spatial groups for each individual at each
370 relocation; or Panel C) `group_polys` generates and groups overlapping home ranges of
371 individuals and optionally returns a measure of proportional area overlap with an output
372 containing the area of overlap (m^2) and proportion of overlap among individuals. Note,
373 coordinates for each panel are projected in Universal Transverse Mercator and the unit
374 for distance thresholds is in meters.

375

376 Figure 2: Detailed overview flowchart of `group_pts` and `group_lines`. Panel A)
377 `group_pts` details showing a single spatial relocation for each of seven individuals at the
378 same time, each indicated by a different colour. Based on the distance among spatial
379 relocations, a distance matrix is generated. Distances are compared to the user-provided
380 spatial distance threshold and individuals within this distance are grouped together. The
381 distance threshold in this example is 15 m and distances less than this threshold are
382 highlighted in yellow. Note, *individual J* was not within 15 meters of any other
383 individuals, but is still assigned a group number as a solitary individual. Panel B)
384 `group_lines` details showing buffered lines for three individuals and resulting spatial
385 groups. Within each time group generated by `group_times`, line segments are generated
386 and buffered by the distance threshold and individuals with overlapping buffers, indicated
387 by the dark grey shaded areas, are grouped together. Note, coordinates for each panel are
388 projected in Universal Transverse Mercator and the unit for distance thresholds is in
389 meters.

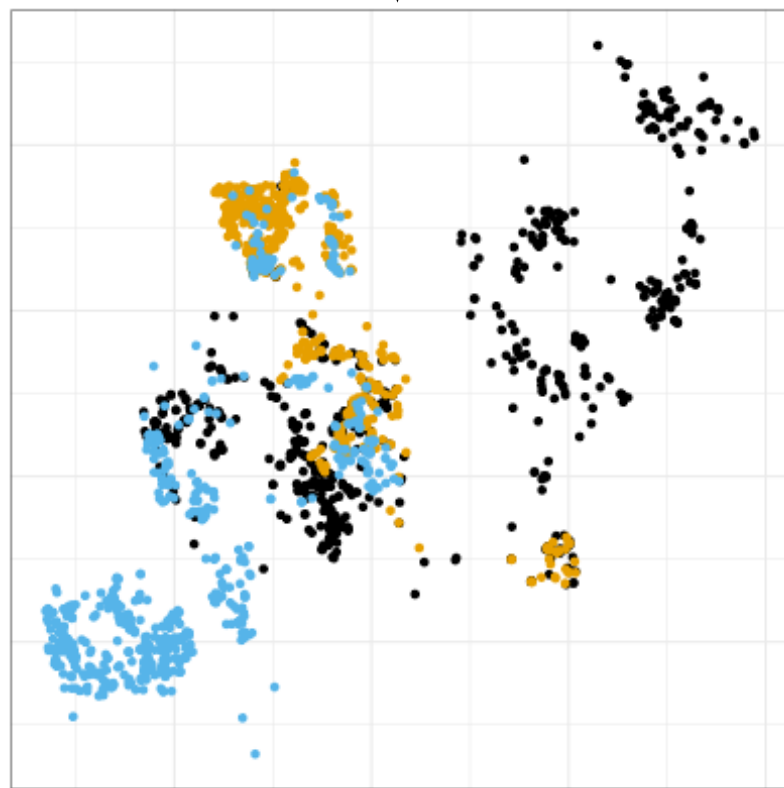
id	datetime	x	y
G	2017-01-01 00:00:10	698534	5511934
H	2017-01-01 00:00:41	699746	5508508
C	2017-01-01 00:00:47	700281	5508670
H	2017-01-01 02:00:41	699736	5508505
G	2017-01-01 02:00:48	698617	5511800
C	2017-01-01 02:01:08	700282	5508654
C	2017-01-01 04:00:42	700319	5508524
H	2017-01-01 04:00:47	699788	5508434
G	2017-01-01 04:00:54	698376	5511762
...

Panel A

bioRxiv preprint first posted online Oct. 18, 2018; doi: <http://dx.doi.org/10.1101/447284>. The copyright holder for this preprint (which was not peer-reviewed) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. All rights reserved. No reuse allowed without permission.

group_times(DT, threshold = "5 minutes")

group_pts(DT, threshold = 50)

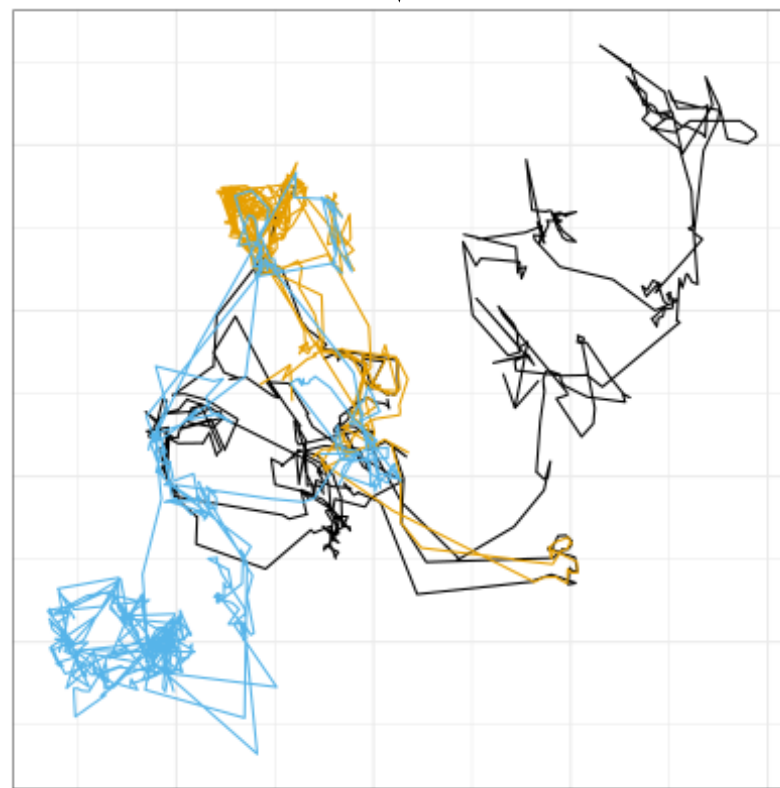


id	x	y	timegroup	group
C	700281	5508670	1	1
G	698534	5511934	1	2
H	699746	5508508	1	3
C	699785	5508651	2	4
H	699798	5508631	2	4
C	700282	5508654	3	5
G	698617	5511800	3	6
C	699920	5507582	4	7
H	699929	5507552	4	7
...

Panel B

group_times(DT, threshold = "1 day")

group_lines(DT, threshold = 100)

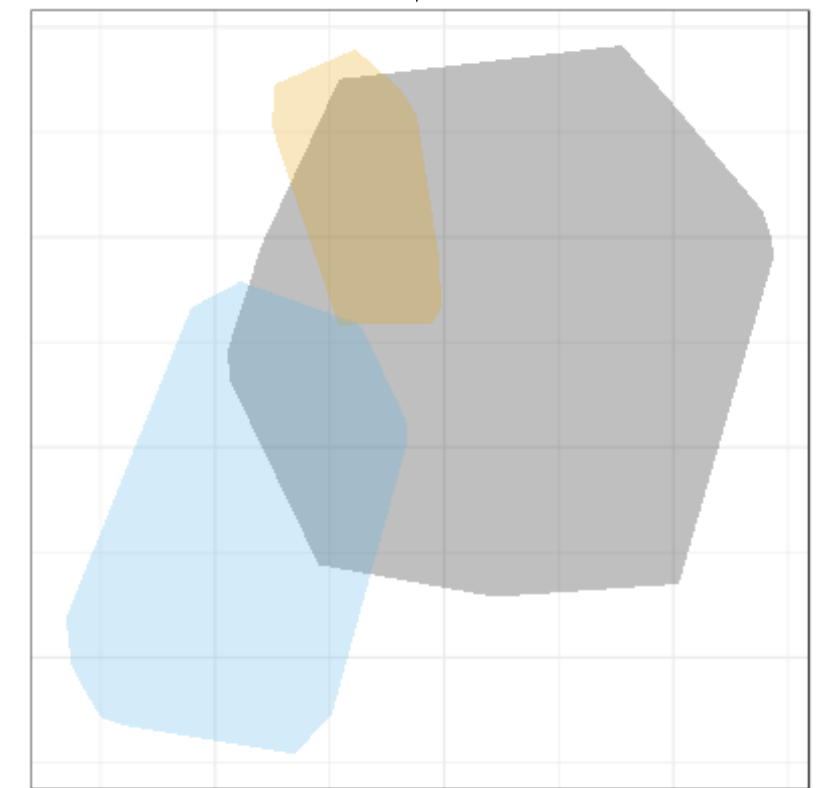


id	date	timegroup	group
C	2017-01-01	1	1
H	2017-01-01	1	1
G	2017-01-01	1	2
C	2017-01-02	2	3
H	2017-01-02	2	3
G	2017-01-02	2	4
C	2017-01-03	3	5
H	2017-01-03	3	5
G	2017-01-03	3	6
...

Panel C

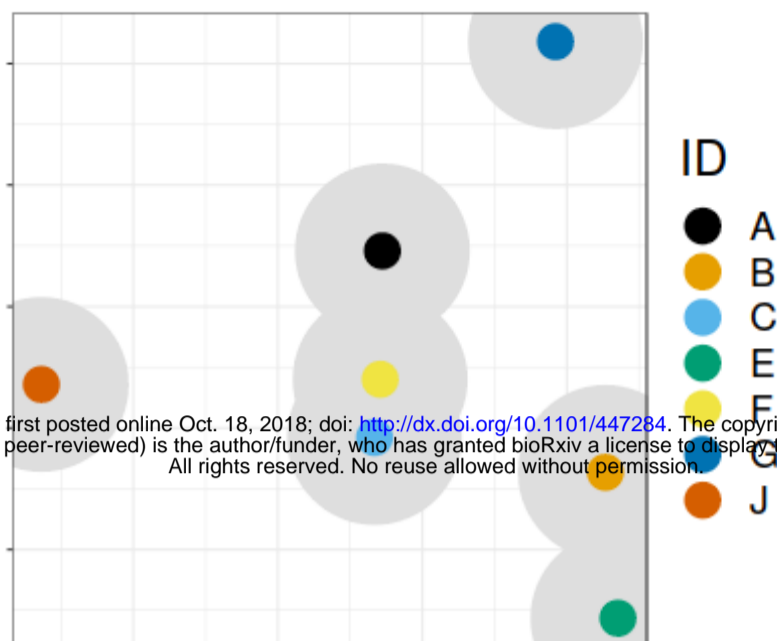
group_times(DT, threshold = "90 days")

group_polys(DT, area = TRUE)



id1	id2	area	proportion
C	C	67526629	1.000
C	G	18634367	0.276
C	H	20219692	0.299
G	C	18634367	0.946
G	G	19692803	1.000
G	H	9373907	0.476
H	C	20219692	0.526
H	G	9373907	0.244
H	H	38472926	1.000

Panel A



bioRxiv preprint first posted online Oct. 18, 2018; doi: <http://dx.doi.org/10.1101/447284>. The copyright holder for this preprint (which was not peer-reviewed) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. All rights reserved. No reuse allowed without permission.

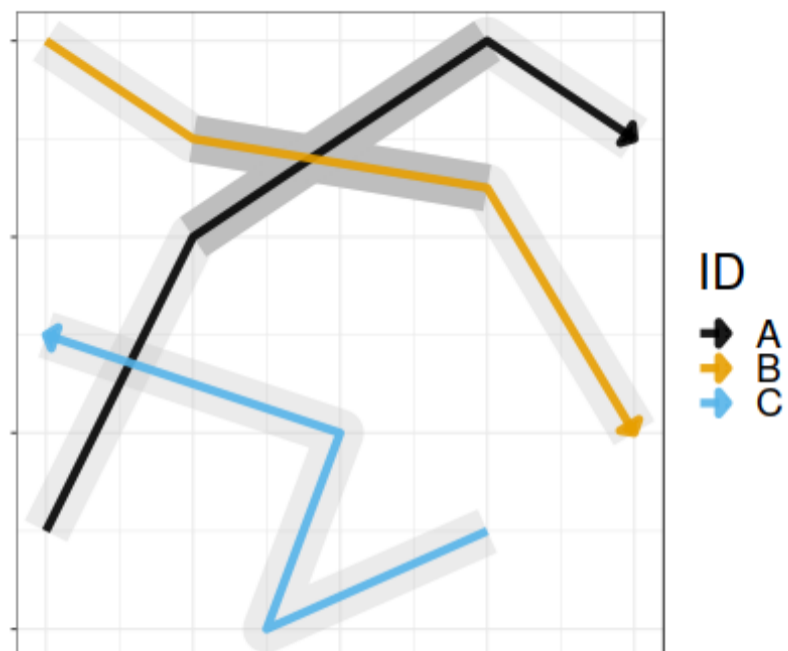


	A	B	C	E	F	G	J
A	0	17.92	7.7	22.23	5.29	14.75	24.23
B	17.92	0	16.04	6.07	16.05	18.06	39.19
C	7.7	16.04	0	18.41	2.43	20.55	23.15
E	22.23	6.07	18.41	0	19.17	24.12	41.03
F	5.29	16.05	2.43	19.17	0	18.45	23.44
G	14.75	18.06	20.55	24.12	18.45	0	38.27
J	24.23	39.19	23.15	41.03	23.44	38.27	0



ID	X	Y	timegroup	group
A	697837.2	5510832	1	1
C	697836.7	5510825	1	1
F	697837.1	5510827	1	1
B	697852.7	5510823	1	2
E	697853.5	5510817	1	2
G	697849.2	5510841	1	3
J	697813.6	5510827	1	4

Panel B



ID	timegroup	group
A	1	1
B	1	2
C	1	3
A	2	4
B	2	4
C	2	5
A	3	6
B	3	7
C	3	8
A	4	9
B	4	10
C	4	11