1 **The forestecology R package for fitting and**

2 **assessing neighborhood models of the effect**

3 **of interspecific competition on the growth of**

4 **trees**

5 Albert Y. Kim ∗

Program in Statistical & Data Sciences, Smith College and

David N. Allen

Biology Department, Middlebury College and

Simon P. Couch Mathematics Department, Reed College

6 August 11, 2021

7 **Abstract**

8 1. Neighborhood competition models are powerful tools to measure the effect of

9 interspecific competition. Statistical methods to ease the application of these models

10 are currently lacking.

11 2. We present the forestecology package providing methods to i) specify neighbor-

12 hood competition models, ii) evaluate the effect of competitor species identity using

13 permutation tests, and iii) measure model performance using spatial cross-validation.

14 Following [Allen & Kim (2020),](#_bookmark21) we implement a Bayesian linear regression neighbor-

15 hood competition model.

16 3. We demonstrate the package’s functionality using data from the Smithsonian Con-

17 servation Biology Institute’s large forest dynamics plot, part of the ForestGEO global

18 network of research sites. Given ForestGEO’s data collection protocols and data for-

19 matting standards, the package was designed with cross-site compatibility in mind.

20 We highlight the importance of spatial cross-validation when interpreting model re-

21 sults.

22 4. The package features i) tidyverse-like structure whereby verb-named functions

∗Assistant Professor, Statistical & Data Sciences, Smith College, Northampton, MA 01063 (e-mail:

[akim04@smith.edu](mailto:akim04@smith.edu)).

23 can be modularly “piped” in sequence, ii) functions with standardized inputs/outputs

24 of simple features sf package class, and iii) an S3 object-oriented implementation of

25 the Bayesian linear regression model. These three facts allow for clear articulation of

26 all the steps in the sequence of analysis and easy wrangling and visualization of the

27 geospatial data. Furthermore, while the package only has Bayesian linear regression

28 implemented, the package was designed with extensibility to other methods in mind.

29 *Keywords:* forest ecology, interspecific competition, neighborhood competition, tree growth,

30 R, ForestGEO, spatial cross-validation

31 **1 Introduction**

32 Repeat-censused forest plots offer excellent opportunities to test neighborhood models of

33 the effect of competition on the growth of trees [(Canham et al. 2004).](#_bookmark26) Neighborhood mod-

34 els of competition have been used to: test whether the species identity of a competitor

35 matters [[Uriarte et al. (2004);](#_bookmark44) measure species-specific competition coefficients [(Das 2012,](#_bookmark28)

36 [Tatsumi et al. (2016));](#_bookmark42) test competing models to see what structures competitive interac-

37 tions, e.g. traits or phylogeny [(Allen & Kim 2020,](#_bookmark21) [Uriarte et al. 2010);](#_bookmark45) and inform selective

38 logging practices [(Canham et al.](#_bookmark27) [2006).](#_bookmark27) Although these are well-described methods, few

39 methods are currently available for easy application.

40 We address this shortcoming with the forestecology R package providing methods

41 and data for forest ecology model fitting and assessment, available on CRAN ([https://](https://cran.r-project.org/package%3Dforestecology)

42 [cran.r-project.org/package=forestecology](https://cran.r-project.org/package%3Dforestecology)) and on GitHub ([https://github.com/](https://github.com/rudeboybert/forestecology)

43 [rudeboybert/forestecology](https://github.com/rudeboybert/forestecology)). The package is written to model stem diameter growth

44 between two censuses based on neighborhood competition, largely following the methods

45 in [Allen & Kim (2020).](#_bookmark21)

46 Let *i* = 1*, . . . , nj* index all *nj* trees of “focal” species *j*; let *j* = 1*, . . . , J* index all *J* focal

47 species; and let *k* = 1*, . . . , K* index all *K* “competitor” species. The average annual growth

48 in diameter at breast height (DBH) *yij* (in centimeters/year) of the *ith* tree of focal species

49 *j* is modeled as

*K*

*yij* = *β*0*,j* + *β*dbh*,j ·* dbh*ij* + *λjk · x*comp + *ij* (1)

*k*=1

*ijk*

50 where *β*0*,j* is the diameter-independent growth rate of species *j*; dbh*ij* is the DBH of

51 the focal tree at the earlier census and *β*dbh*,j* the slope of that species’s diameter-growth

52 relationship; *x*comp is the sum of some numerical explanatory variable of all trees of com-

*ijk*

53 petitor species *k*, and *λjk* quantifies the corresponding change in growth for individuals of

54 species *j* from these competitors; and *ij* is a random error term distributed Normal(0*, σ*2).

55 [Allen & Kim](#_bookmark21) [(2020)](#_bookmark21) use the sum of the basal area of all trees of competitor species

56 *k* as *x*comp. Furthermore, they estimate all parameters via Bayesian linear regression,

*ijk*

57 while exploiting Normal/Inverse Gamma conjugacy to derive closed-form solutions to all

58 posterior distributions[1](#_bookmark0). These closed-form solutions are not as computationally expensive

59 as approximations from Markov Chain Monte Carlo algorithms.

60 To evaluate whether competitor species identity matters, [Allen & Kim (2020)](#_bookmark21) run a per-

61 mutation test where a null hypothesis of no species grouping-specific effects of competition

62 is assumed, thus the species identity of all competitors can be permuted:

*H*0 : *λjk* = *λj* for all *k* = 1*, . . . , K* (2) vs. *HA* : at least one *λjk* is different

63 Furthermore, to account for the spatial autocorrelation in their estimates of out-of-

64 sample model error, [Allen & Kim (2020)](#_bookmark21) use spatial cross-validation. Estimates of model

65 error that do not account for this dependence tend to underestimate the true model error

66 [(Roberts et al. 2017).](#_bookmark38)

67 The package is designed with “tidy” design principles in mind [(Wickham et al. 2019)](#_bookmark50).

68 Much like all tidyverse packages, forestecology has verb-named functions that can be

69 modularly composed using the pipe %*>*% operator to sequentially complete all necessary

70 analysis steps [(Bache & Wickham 2020).](#_bookmark23)

71 Furthermore, the inputs and outputs of most functions use the same “simple features

72 for R” data structures for spatial data from the sf package (P[ebesma 2018).](#_bookmark36) Previously sp

73 package classes were commonly used for storing spatial data and interfacing with geospatial

74 libraries [(Bivand et al. 2013);](#_bookmark24) the sf package aims to improve on the sp package by:

1See S1 Appendix of [Allen & Kim](#_bookmark21) [(2020),](#_bookmark21) available at [https://doi.org/10.1371/journal.pone.](https://doi.org/10.1371/journal.pone.0229930.s004)

[0229930.s004](https://doi.org/10.1371/journal.pone.0229930.s004)

75 1. Using simple feature access as the base standard for representing and encoding spatial

76 data, rather than shapefiles [(Herring 2011)](#_bookmark32).

77 2. Leveraging improvements in external libraries for reading and writing spatial data

78 (GDAL) and for geometrical operations (GEOS) [(Warmerdam 2008,](#_bookmark48) [Team (2017)).](#_bookmark43)

79 3. Integrating closely with the popular tidyverse suite of packages for data science

80 [(Wickham et al. 2019).](#_bookmark50)

81 By using the sf package classes to represent spatial data rather than the sp package,

82 the implementation and use of the forestecology package’s spatial algorithms was greatly

83 simplified.

# 84 2 forestecology workflow: a case study

85 We present a case-study of forestecology’s functionality on data from the Smithsonian

86 Conservation Biology Institute (SCBI) large forest dynamics plot in Front Royal, VA,

87 USA, part of the ForestGEO global network of research sites [(Bourg et al. 2013,](#_bookmark25) [Anderson-](#_bookmark22)

88 [Teixeira et al. (20](#_bookmark22)15), [Davies et al. (2021)).](#_bookmark29) The 25.6 ha (640 x 400 m) plot is located at

89 the intersection of three of the major physiographic provinces of the eastern US—the Blue

90 Ridge, Ridge and Valley, and Piedmont provinces—and is adjacent to the northern end of

91 Shenandoah National Park.

92 The package has the following goals: to evaluate i) the effect of competitor species

93 identity using permutation tests and ii) model performance using spatial cross-validation.

94 We outline the four-step basic analysis sequence:

95 1. Compute the growth of stems based on two censuses.

96 2. Add spatial information:

97 1. Define a buffer region of trees.

98 2. Add spatial cross-validation block information.

99 3. Identify all focal trees and their competitors.

100 4. Apply model, which includes:

101 1. Fit model.

102 2. Compute predicted values.

103 3. Visualize posterior distributions.

104 We start by loading all packages.

library(tidyverse) library(lubridate) library(sf) library(patchwork) library(forestecology) library(blockCV)

*# Resolve conflicting functions*

filter <- dplyr::filter select <- dplyr::select

## 105 2.1 Step 1: Compute the growth of trees based on census data

106 We first compute the growth of trees using data from two censuses. compute growth()

107 computes the average annual growth based on census data that roughly follows ForestGEO

108 standards. Despite such standards, minor variations will still exist between sites, thereby

109 necessitating some data wrangling. For example, the SCBI site records all DBH values in

110 millimeters [(Bourg et al. 2013),](#_bookmark25) whereas the Michigan Big Woods site used in [Allen & Kim](#_bookmark21)

111 [(2020)](#_bookmark21) records them in centimeters [(Allen et al. 2020).](#_bookmark20)

112 We load both 2008 and 2014 SCBI census .csv files as they existed on GitHub on

113 2021/08/02 and perform minor data wrangling [(Gonzalez-Akre, McGregor, Anderson-](#_bookmark30)

114 [Teixeira, Dow, Herrmann, Terrell, Kim, Vinod & Helcoski](#_bookmark30) [2020).](#_bookmark30) We then only consider

115 a 9 ha subsection of the 25.6 ha of the site to speed up computation for this example: gx

116 from 0–300 instead of 0–400 and gy from 300–600 instead of 0–640.

census\_2013\_scbi <- read\_csv("scbi.stem2.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(

*# Convert date from character to date*

date = mdy(date),

*# Convert dbh to be in cm*

dbh = as.numeric(dbh) / 10

) %>%

filter(gx < 300, between(gy, 300, 600))

census\_2018\_scbi <- read\_csv("scbi.stem3.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(

date = mdy(date),

dbh = as.numeric(dbh) / 10

) %>%

filter(gx < 300, between(gy, 300, 600))

117 These two data frames are then used as inputs to compute growth(), along with id

118 specifying the variable that uniquely identifies each tree-stem. We also discard all resprouts

119 with code == R in the later census, since we are only interested in the growth of surviving,

120 and not resprouted, stems.

growth\_scbi <- compute\_growth(census\_1 = census\_2013\_scbi, census\_2 = census\_2018\_scbi filter(!str\_detect(codes, "R")), id = "stemID")

growth\_scbi %>%

select(stemID, sp, dbh1, dbh2, growth, geometry)

*## Simple feature collection with 7954 features and 5 fields ## Geometry type: POINT*

*## Dimension: XY*

*## Bounding box: xmin: 0.2 ymin: 300 xmax: 300 ymax: 600 ## CRS: NA*

*## # A tibble: 7,954 x 6*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *##* | *stemID* | | *sp* | *dbh1* | *dbh2* | *growth* | *geometry* | |
| *##* | *<dbl>* | | *<fct>* | *<dbl>* | *<dbl>* | *<dbl>* | *<POINT>* | |
| *##* | *1* | *4* | *nysy* | *13.6* | *14.2* | *0.103* | *(14.2* | *428)* |
| *##* | *2* | *5* | *havi* | *8.8* | *9.6* | *0.150* | *(9.4* | *436)* |
| *##* | *3* | *6* | *havi* | *3.25* | *4* | *0.140* | *(1.3* | *434)* |
| *##* | *4* | *77* | *qual* | *65.2* | *66* | *0.141* | *(34.7* | *307)* |
| *##* | *5* | *79* | *tiam* | *47.7* | *46.8* | *-0.161* | *(40* | *381)* |
| *## # ... with 7,949 more rows* | | | | | | | | |

121 The output growth scbi is a data frame of class sf that includes among other variables

122 the species variable sp converted to a factor, the average annual growth in DBH (cm *·*

123 y-1) for all stems that were alive at both time points, and the sf package’s encoding of

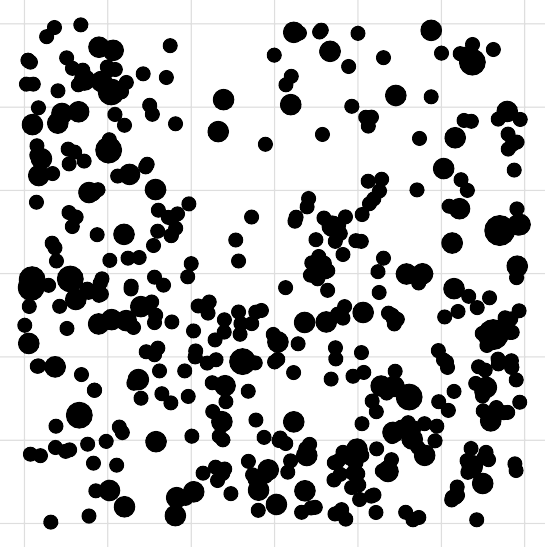
124 geolocations of geometry type *<*POINT*>*. Given that growth scbi is of class sf, it can be

125 easily plotted in ggplot2 using geom sf() as seen in Figure [1.](#_bookmark4)

ggplot() + geom\_sf(data = growth\_scbi %>%

sample\_n(500), aes(size = growth)) + scale\_size\_binned(limits = c(0.1, 1)) + labs(size = expression(paste(Growth, " (cm ", y^*{*

600



550

500

450

400

Growth (cm y1)



0.25

0.50



0.75

350

300

0 50 100 150 200 250 300

Figure 1: Step 1 - Compute growth of trees based on census data. A map of the growth of a random sample of 500 trees from a 9 ha subsection of the Smithsonian Conservation Biology Institute (SCBI) forest plot.

-1

*}*, ")")))

126 We also load species information as it existed on GitHub on 2021/08/02, which in-

127 cludes family, genus, and species information; as well as classifications of the canopy po-

128 sition (canopy, canopy emergent, understory, shrub layer), drought tolerance (intolerant,

129 resistant), and other characteristics of the species.

sp\_info <- read\_csv("SCBI\_ForestGEO\_sp\_ecology.csv") %>%

select(sp = spcode, family, genus, species, canopy\_position, drought\_tolerance) sp\_info

*## # A tibble: 65 x 6*

*## sp family genus species canopy\_position drought\_toleran~ ## <chr> <chr> <chr> <chr> <chr> <chr>*

*## 1 acne Sapindaceae Acer negundo understory drought-resista~*

*## 2 acpl Sapindaceae Acer platanoides canopy drought-resista~ ## 3 acru Sapindaceae Acer rubrum canopy drought-resista~ ## 4 aial Simaroubaceae Ailanthus altissima canopy drought-resista~ ## 5 amar Rosaceae Amelanchier arborea understory drought-resista~ ## # ... with 60 more rows*

130 We join this species information to our growth scbi data frame and convert the species

131 variable to a factor.

growth\_scbi <- growth\_scbi %>% left\_join(sp\_info, by = "sp") %>%

mutate(sp = as.factor(sp))

132 Furthermore, we compute two potential competitor explanatory variables *x*comp from

*ijk*

133 Equation [1.](#_bookmark1) First, the basal area of each tree as a function of its DBH in the earlier

134 census. Second, the above ground biomass as estimated by allometric equations encoded

135 in the get biomass() function from the allodb package [(Gonzalez-Akre, Piponiot, Lepore](#_bookmark31)

136 [& Anderson-Teixeira 202](#_bookmark31)0); this function has DBH, species, and geographic coordinates as

137 arguments.

*# Install development version of allodb using:*

*# remotes::install\_github("forestgeo/allodb")*

library(allodb)

growth\_scbi <- growth\_scbi %>% mutate(

*# Compute basal area:*

basal\_area = 0.0001 \* pi \* (dbh1 / 2)^2,

*# Compute above ground biomass:*

agb = get\_biomass(

dbh = dbh1,

genus = genus, species = species,

coords = c(-78.2, 38.9)

)

)

## 138 2.2 Step 2: Add spatial information

139 We then add spatial information to growth scbi. We first add a “buffer region” to the

140 periphery of the study region. Since some of our model’s explanatory variables are cumula-

141 tive, we must ensure that all trees being modeled are not biased to have different neighbor

142 structures. This is of concern for trees at the boundary of the study region who will not

143 have all their neighbors included in the census stems. To account for such edge effects,

144 only trees that are not part of this buffer region, i.e. are part of the interior of the study

145 region, will have their growth modeled [(Waller & Gotway 2004).](#_bookmark47)

146 Our model of interspecific competition relies on a spatial definition of who competitor

147 trees are: all trees within a distance comp dist of a focal tree. Here we set comp dist to

148 7.5m, a value informed by other studies [(Canham et al. 2004,](#_bookmark26) [Uriarte et al. (2004),](#_bookmark44) [Canham](#_bookmark27)

149 [et al.](#_bookmark27) [(2006)),](#_bookmark27) but the package could also be used to compare multiple distances and see

150 which is best supported (see Appendix [A).](#_bookmark15) We use comp dist and a manually constructed

151 sf representation of the study region’s boundary as inputs to add buffer variable() to

152 add a buffer boolean variable to growth scbi. All trees with buffer equal to FALSE will

153 be our focal trees whose growth will be modeled, whereas those with TRUE will only act as

154 competitor trees.

*# Define competitive distance range*

comp\_dist <- 7.5

*# Manually construct study region boundary*

study\_region\_scbi <- tibble(x = c(0, 300, 300, 0, 0), y = c(300, 300, 600,

600, 300)) %>%

sf\_polygon()

growth\_scbi <- growth\_scbi %>%

add\_buffer\_variable(size = comp\_dist, region = study\_region\_scbi)

155 The second element of spatial information we add are blocks corresponding to folds

156 of a spatial cross-validation algorithm. Conventional cross-validation algorithms assign

157 individual observations to folds by randomly resampling them all while assuming they are

158 statistically independent. In the case of forest census data however, observations exhibit

159 spatial autocorrelation. We therefore incorporate this dependence into the cross-validation

160 algorithm by resampling spatial blocks of trees [(Roberts et al.](#_bookmark38) [2017,](#_bookmark38) [Pohjankukka et al.](#_bookmark37)

161 [(2017)).](#_bookmark37)

162 We first manually define an sf object defining four folds that partition the study region.

163 We then use the output of the spatialBlock() function from the blockCV package to

164 associate each tree in growth scbi to the correct foldID (V[alavi et al. 2019).](#_bookmark46) This foldID

165 variable will be used in Section [2.6.](#_bookmark13)

166 Figure [2](#_bookmark7) illustrates the net effect of adding these two elements of spatial information to

167 growth scbi.

*# Manually define spatial blocks to act as folds*

n\_fold <- 4

fold1 <- cbind(c(0, 150, 150, 0), c(300, 300, 450, 450))

fold2 <- cbind(c(150, 300, 300, 150), c(300, 300, 450, 450))

fold3 <- cbind(c(0, 150, 150, 0), c(450, 450, 600, 600))

fold4 <- cbind(c(150, 300, 300, 150), c(450, 450, 600, 600))

blocks\_scbi <- bind\_rows(sf\_polygon(fold1), sf\_polygon(fold2), sf\_polygon(fold3), sf\_polygon(fold4)) %>%

mutate(folds = c(1:n\_fold) %>% factor())

*# Associate each observation to a fold*

spatial\_block\_scbi <- spatialBlock(speciesData = growth\_scbi, k = n\_fold,

selection = "systematic", blocks = blocks\_scbi, showBlocks = FALSE, verbose = FALSE

growth\_scbi <- growth\_scbi %>%

mutate(foldID = spatial\_block\_scbi$foldID %>% factor())

ggplot() + geom\_sf(data = blocks\_scbi, fill = "transparent", linetype = "dashed") + geom\_sf\_text(data = growth\_scbi %>%

sample\_n(1000), aes(label = foldID, col = buffer))

## 168 2.3 Step 3: Identify all focal and corresponding competitor trees

169 We then identify all focal trees and their corresponding competitor trees. More specifically,

170 identify all trees that are not part of the buffer region, have a valid growth measurement,

171 and have at least one neighbor within 7.5m. We do this using create focal vs comp(),

172 which takes the previously detailed comp dist and id arguments, the sf representation

600

**3** 333 33333 333

3 333333**3**33

3333**3**3333 3 3333 3 3 33

**3**3

3 3

3333

**4**

44 4

4**4**

4

4 44 4

4

3 333

3333 3333**3** 3 333 333

33 3

3 3 33

333 3333

3

3 33**3**

3**3** 34

3 334 4444

444 4444444444**4**4

4

4 **4** 4 4

444

4

4 44

333 333 3 33

33 3 33

3 3 3 **3** 3

**4**

444444

333**33**

33

4 4 **4**4 44 4 44

44444444**4**44 4

4

4**4**4 4

44

4

11 11 1 1 1 1 2 222

111 1 1 1 **1**1 2 2 2

11**3**13 1 3113

3

3 333 3

3 3333 3333 444

4

44 4

4

11 1 11**1**1**1**111

33 3

**11**1 1

3**3** 33

4

444444

4**24** 2 444

44 44 4**4**44

44

44

22 2 22222

4**24**

22442

1 11 11**1**11 111111 11 1

1 11**1**11 **1**

111 **1**111**1** 1 1 1

11 1

1 1112

11 111

22 2 2

2

1 2

2 2222222**2**22 222

2 222 **2**

22

2222 **2**

22 2 222222

2 2

222 2 2222

2 **2**

222**2**2 2

**2**222

222

1 1 11

1111**1**1**1**

1 11 1

1 111

11 11

1

1 11 1

11111

1122 2222 222 22

12 2

2**2**2 2**2**22 **2**

22 2

22 2**2**2**2**

2**22**

22**2** 2

22 **2**2222

**2**

11

**1 1**11111 111 1 111**1** 11 222 22222222222**2** 22

11

1 11 111 1111111**1**1**1**11**1**1**2** 22 2**2**2**2**2**2**22

2222 2

111

1 11 111 11**1** 22

1 12

2 2222

22 2

22**2**

222 **2**2 **2**

44

4

4 444

4

4

44

550

500

buffer

450

y

400

350

300

0 50 100 150 200 250 300

x

a FALSE

a TRUE

Figure 2: Step 2 - Add spatial information. A buffer region and spatial cross-validation blocks 1 through 4. The location of each tree is marked with its fold number where the folds are delineated with solid lines. The color of each digit indicates whether the tree is part of the buffer region (thus will only be considered as a competitor tree) or is part of the interior of the study region (thus is a focal tree whose growth is of modeled interest).

173 of the spatial cross-validation blocks blocks scbi, and a specification comp x var of the

174 basal area variable we use as the competitor explanatory variable *x*comp from Equation [1.](#_bookmark1)

*ijk*

175 This function returns a new data frame focal vs comp scbi.

focal\_vs\_comp\_scbi <- growth\_scbi %>%

create\_focal\_vs\_comp(comp\_dist, blocks = blocks\_scbi, id = "stemID", comp\_x\_var = "basal\_area")

focal\_vs\_comp\_scbi %>%

select(focal\_ID, focal\_sp, geometry, growth, comp)

*## # A tibble: 6,296 x 5*

*## focal\_ID focal\_sp geometry growth comp*

*##*

*<dbl> <fct>*

*<POINT> <dbl> <list>*

*## 1*

*4 nysy*

*(14.2 428) 0.103 <tibble [20 x 4]>*

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *##* | *2* | *5* | *havi* | *(9.4* | *436)* | *0.150* | *<tibble* | *[32* | *x* | *4]>* |
| *##* | *3* | *79* | *tiam* | *(40* | *381)* | *-0.161* | *<tibble* | *[20* | *x* | *4]>* |
| *##* | *4* | *80* | *caca* | *(38.7* | *422)* | *0.253* | *<tibble* | *[12* | *x* | *4]>* |
| *##* | *5* | *96* | *libe* | *(60* | *310)* | *0.262* | *<tibble* | *[14* | *x* | *4]>* |
| *##* | *#* | *... with* | *6,291* | *more rows* |  |  |  |  |  |  |

176 The resulting focal vs comp scbi has 6296 rows, representing the subset of the 7954

177 trees in growth scbi that will be considered as focal trees. The variables focal ID and

178 focal sp relate to tree-stem identification and species information. Most notably however

179 is the variable comp, which contains information on all competitor trees saved in tidyr

180 package list-column format [(Wickham](#_bookmark49) [2020).](#_bookmark49) To inspect this information, we flatten the

181 comp list-column for the tree with focal ID 4 in the first row, here a tibble [20 × 4],

182 into regular columns using unnest() from the tidyr package.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| focal\_vs\_comp\_scbi %>% filter(focal\_ID == 4) %>% select(focal\_ID, dbh, comp) %>%  unnest(cols = "comp") | | | | | |
| *##* | *#* | *A tibble: 20 x 6* |  |  |  |
| *##* |  | *focal\_ID dbh comp\_ID* | *dist* | *comp\_sp* | *comp\_x\_var* |
| *##* |  | *<dbl> <dbl> <dbl>* | *<dbl>* | *<fct>* | *<dbl>* |
| *##* | *1* | *4 13.6 1836* | *7.48* | *tiam* | *0.0176* |
| *##* | *2* | *4 13.6 1847* | *2.81* | *nysy* | *0.00332* |
| *##* | *3* | *4 13.6 1848* | *1.62* | *nysy* | *0.00396* |
| *##* | *4* | *4 13.6 1849* | *2.62* | *nysy* | *0.00535* |
| *##* | *5* | *4 13.6 1850* | *2.98* | *havi* | *0.00472* |
| *##* | *#* | *... with 15 more rows* |  |  |  |

183 We observe 4 variables describing 20 competitor trees: the unique tree-stem ID, the

436

434

432

430

428

y

426

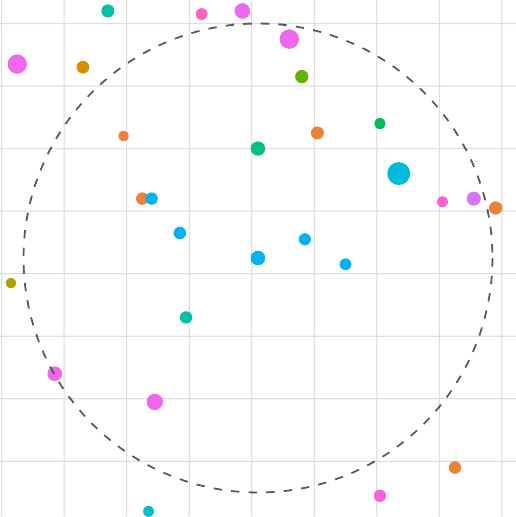
424

acru caca cagl caovl cato ceca ceoc elum fram frni havi

libe litu nysy qual qupr quve saal tiam ulam unk vipr

422

6 8 10 12 14 16 18 20 22



4

x

DBH at first census

 30

 60

90

Figure 3: Step 3 - Identify all focal and corresponding competitor trees. The dashed circle extends 7.5m away from the focal tree 4 while all 20 competitor trees are within this circle.

184 distance to the focal tree (all *≤* 7.5m), the species, and the basal area (in m2) calculated

*π×*(DBH/2)2

185 as 10000 for the DBH in cm from the earlier census. Saving competitor information in

186 list-column format minimizes redundancy since we do not need to repeat information on

187 the focal tree 20 times. We visualize the spatial distribution of these trees in Figure [3.](#_bookmark8)

188 Here we use basal area as the continuous competitor explanatory variable but the pack-

189 age is flexible to allow the user to specify any competitor explanatory variable (basal area,

190 biomass, tree height, a soil nutrient value, . . . ). The package can also be used to compare

191 competitor explanatory variables and see which best explains tree growth, see Appendix

192 [B](#_bookmark17) for an example comparing basal area and above ground biomass. Similarly, the package

193 can use any categorical variable as an explanatory variable and compare between different

194 categorical variables. For example in [Allen & Kim](#_bookmark21) [(2020)](#_bookmark21) we compare grouping individ-

195 uals based on species, family, and based on trait-based groups. In Appendix [C](#_bookmark18) we give

196 another example and compare grouping individuals by species or by potential canopy po-

197 sition (canopy, understory, shrub layer).

198  **2.4 Step 4: Fit model**

199 Lastly, we fit the competition Bayesian linear regression model for tree growth outlined in

200 Equation [1](#_bookmark1) using comp bayes lm(). This function has an option to specify prior distribu-

201 tions of all parameters, chosen here to be the defaults detailed in ?comp bayes lm.

comp\_bayes\_lm\_scbi <- focal\_vs\_comp\_scbi %>%

comp\_bayes\_lm(prior\_param = NULL)

202 The resulting comp bayes lm scbi is an object of S3 class type comp bayes lm con-

203 taining the posterior values of all parameters. Furthermore, this class includes generics for

204 three methods. First, the generic for print() displays the names of all prior and posterior

205 parameters and the model formula:

comp\_bayes\_lm\_scbi

*## Bayesian linear regression model parameters with a multivariate Normal ## likelihood. See ?comp\_bayes\_lm for details:*

*##*

*## parameter\_type prior posterior ## 1 Inverse-Gamma on sigma^2 a\_0 a\_star*

*## 2 Inverse-Gamma on sigma^2 b\_0 b\_star ## 3 Multivariate t on beta mu\_0 mu\_star ## 4 Multivariate t on beta V\_0 V\_star ##*

*## Model formula:*

*## growth ~ sp + dbh + dbh \* sp + acne \* sp + acru \* sp + amar \* sp + astr ## \* sp + caca \* sp + caco \* sp + cade \* sp + cagl \* sp + caovl \* sp + cato ## \* sp + ceca \* sp + ceoc \* sp + chvi \* sp + cofl \* sp + crpr \* sp + crsp ## \* sp + divi \* sp + elum \* sp + fagr \* sp + fram \* sp + frni \* sp + frpe*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *##* | *\* sp* | *+* | *havi* | *\* sp* | *+* | *ilve* | *\* sp* | *+* | *juci* | *\* sp* | *+* | *juni* | *\* sp* | *+* | *libe* | *\* sp* | *+* | *litu* |
| *##* | *\* sp* | *+* | *nysy* | *\* sp* | *+* | *pist* | *\* sp* | *+* | *pivi* | *\* sp* | *+* | *ploc* | *\* sp* | *+* | *prav* | *\* sp* | *+* | *prse* |
| *##* | *\* sp* | *+* | *qual* | *\* sp* | *+* | *quco* | *\* sp* | *+* | *qufa* | *\* sp* | *+* | *qumi* | *\* sp* | *+* | *qupr* | *\* sp* | *+* | *quru* |
| *##* | *\* sp* | *+* | *quve* | *\* sp* | *+* | *rops* | *\* sp* | *+* | *saal* | *\* sp* | *+* | *saca* | *\* sp* | *+* | *tiam* | *\* sp* | *+* | *ulam* |
| *## \* sp + ulru \* sp + unk \* sp + vipr \* sp* | | | | | | | | | | | | | | | | | | |

206 Next, the generic for predict() takes the posterior parameter values in comp bayes lm scbi

207 and a newdata data frame, and outputs a vector growth hat of predicted DBH values *yij*

---

208 computed from the posterior predictive distribution.

focal\_vs\_comp\_scbi <- focal\_vs\_comp\_scbi %>%

mutate(growth\_hat = predict(comp\_bayes\_lm\_scbi, newdata = focal\_vs\_comp\_scbi))

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *##* |  | *focal\_ID* | *focal\_sp* | *dbh* | *growth* | *growth\_hat* |
| *##* |  | *<dbl>* | *<fct>* | *<dbl>* | *<dbl>* | *<dbl>* |
| *##* | *1* | *4* | *nysy* | *13.6* | *0.103* | *0.0809* |
| *##* | *2* | *5* | *havi* | *8.8* | *0.150* | *0.112* |
| *##* | *3* | *79* | *tiam* | *47.7* | *-0.161* | *0.229* |
| *##* | *4* | *80* | *caca* | *5.15* | *0.253* | *0.121* |
| *##* | *5* | *96* | *libe* | *2.3* | *0.262* | *0.142* |
| *## # ... with 6,291 more rows* | | | | | | |

209 We can now compare the observed and predicted growths to compute the root mean

focal\_vs\_comp\_scbi %>%

select(focal\_ID, focal\_sp, dbh, growth, growth\_hat)

*## # A tibble: 6,296 x 5*

210 squared error (RMSE) of our model:

model\_rmse <- focal\_vs\_comp\_scbi %>%

rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

model\_rmse

*## [1] 0.128*

211 Lastly, the generic for ggplot2::autoplot() allows us to visualize all posterior dis-

212 tributions, as seen in Figure [4.](#_bookmark10) Setting type to "intercepts" and "dbh slopes" returns

213 species-specific posterior distributions for *β*0*,j* and *βdbh,j* respectively, while setting type =

214 "competition" returns competition coefficients *λj,k*.

*# Plot posteriors for only a subset of species*

sp\_to\_plot <- c("litu", "quru", "cagl")

plot1 <- autoplot(comp\_bayes\_lm\_scbi, type = "intercepts", sp\_to\_plot = sp\_to\_plot) plot2 <- autoplot(comp\_bayes\_lm\_scbi, type = "dbh\_slopes", sp\_to\_plot = sp\_to\_plot) plot3 <- autoplot(comp\_bayes\_lm\_scbi, type = "competition", sp\_to\_plot = sp\_to\_plot)

*# Combine plots using the patchwork package*

(plot1 | plot2)/plot3

215 For many users the visualizations of *λj,k* will be of particular interest as they provide

216 insight into species-specific competitive interactions, where negative values indicate a com-

217 petitor species which slows the growth of a focal species. Here, for example, we see that

218 tulip poplars (litu) have a strong negative effect on the growth of conspecifics but relatively

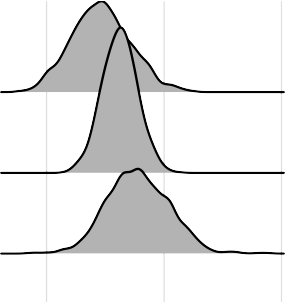
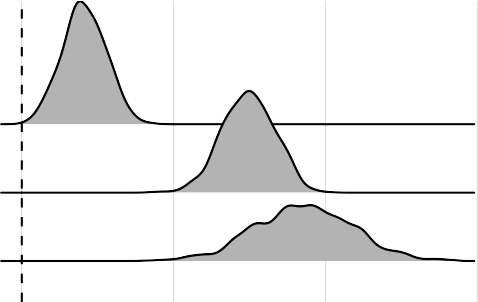
219 lesser effect on pignut hickory (cagl) and red oak (quru) neighbors.

220 Currently the forestecology package can only fit the competition Bayesian linear

221 regression model in Equation [1.](#_bookmark1) However, it can be extended to any model as long as it is

Intercept Slope for DBH

cagl



species

species

cagl

litu

litu

quru

quru

0.0 0.1 0.2 0.3

0 (cm y1)

0.000 0.002 0.004 0.006 0.008

DBH (y1)

Competitor species in rows, focal species in columns

Ex: Top row, second column: competitive effect of cagl on litu

cagl

cagl

litu

quru

litu

species

quru

−0.75 −0.50 −0.25 0.00 0.25 −0.75 −0.50 −0.25 0.00 0.25 −0.75 −0.50 −0.25 0.00 0.25



Figure 4: Step 4 - Fit model. Posterior distributions of all parameters. For compactness we include only three species.

222 implemented in a function similar to comp bayes lm().

## 223 2.5 Evaluate the effect of competitor species identity using per-

224 **mutation tests**

225 To evaluate the effect of competitor species identity, we use the above four steps along with

226 the permutation test in Equation [2.](#_bookmark2) Under a null hypothesis where competitor species

227 identity does not matter, we can permute the competitor species identities within each

228 focal tree, compute the RMSE test statistic, repeat this process several times to construct

229 a null distribution, and compare it to the observed RMSE to assess significance. Going

230 back to our example in Section [2.3](#_bookmark6) of focal tree with focal ID 4 and its 20 competitors,

231 the permutation test only randomly resamples the comp sp variable without replacement,

232 leaving all other variables intact. This resampling is nested within each focal tree in order

233 to preserve neighborhood structure. We perform this permutation test once again using

234 comp bayes lm() but by setting run shuffle = TRUE.

= focal\_vs\_

comp\_bayes\_lm\_scbi\_shuffle <- focal\_vs\_comp\_scbi %>% comp\_bayes\_lm(prior\_param = NULL, run\_shuffle = TRUE)

focal\_vs\_comp\_scbi <- focal\_vs\_comp\_scbi %>%

mutate(growth\_hat\_shuffle = predict(comp\_bayes\_lm\_scbi\_shuffle, newdata

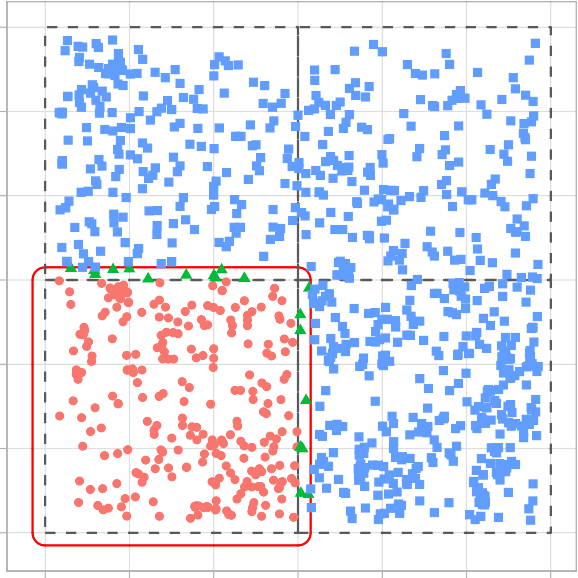
model\_rmse\_shuffle <- focal\_vs\_comp\_scbi %>%

rmse(truth = growth, estimate = growth\_hat\_shuffle) %>% pull(.estimate)

model\_rmse\_shuffle

*## [1] 0.131*

235 The resulting permutation test RMSE of 0.131 is larger than the earlier RMSE of 0.128,

600

550

500

450

400

350

300

0 50 100 150 200 250 300

fold



test

fold buffer training

Figure 5: Schematic of spatial cross-validation. Using the k = 1 fold (bottom-left) as the test set, k = 2 through 4 as the training set, along with a ”fold buffer” extending outwards from the test set to maintain spatial independence between it and the training set.

236 suggesting that models that do incorporate competitor species identity better fit the data.

## 237 2.6 Evaluate model performance using spatial cross-validation

238 To evaluate model performance, we use spatial cross-validation. The model fit in Section

239 [2.4](#_bookmark9) uses the same data to both fit and assess model performance. Given the spatial-

240 autocorrelation of our data, this can potentially lead to overfit models [(Roberts et al.](#_bookmark38)

241 [2017).](#_bookmark38) To mitigate this risk, we use the spatial cross-validation blocking scheme encoded

242 in the foldID variable from Section [2.2](#_bookmark5) and visualized in Figure [2.](#_bookmark7)

243 At each iteration of the cross-validation, one fold acts as the test set and the remaining

244 three act as the training set. We fit the model to all focal trees in the training set, apply

245 the model to all focal trees in the test set, compute predicted values, and compute the

246 RMSE. Furthermore, to maintain spatial independence between the test and training sets,

247 a “fold buffer” that extends 7.5m outwards from the boundary of the test set is considered;

248 all trees within this “fold buffer” are excluded from the training set (see Figure [5).](#_bookmark12)

249 This process is repeated for each of the four folds acting as the test set, then the

250 four RMSE’s are averaged to provide a single estimate of model error. This algorithm is

251 implemented in run cv(), which acts as a wrapper function to both comp bayes lm() that

252 fits the model and predict() that returns predicted values.

focal\_vs\_comp\_scbi <- focal\_vs\_comp\_scbi %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi)

model\_rmse\_cv <- focal\_vs\_comp\_scbi %>%

rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

model\_rmse\_cv

*## [1] 0.14*

253 The resulting RMSE of 0.14 computed using cross-validation is larger than the earlier

254 RMSE of 0.128, suggesting that models that do not account for spatial autocorrelation

255 generate model error estimates that are overly optimistic, i.e. RMSE values that are too

256 low.

# 257 3 Importance of spatial cross-validation

258 run cv() also accepts the run shuffle argument in order to permute competitor species

259 identity as described in Section [2.5.](#_bookmark11) Figure [6](#_bookmark14) compares model performance for 49 permuta-

260 tions of competitor species and RMSE calculations, both with and without cross-validation.

261 Without cross-validation, competitor species identity does matter as the observed RMSE

262 was significantly lower than the permutation null distribution of RMSE. However, once we

263 incorporate spatial cross-validation, this improvement disappears. These results suggest

264 that in this 9 ha subplot of the SCBI plot, competitive interactions do not depend on the

15

10 Cross−validated?

count

FALSE TRUE

5

0

0.128 0.132 0.136 0.140

RMSE (cm y1)

Figure 6: Comparison of root mean squared error of models for standard, permuted, and spatially cross-validated error estimates. The dotted lines show observed RMSE while the histograms show the null distribution of RMSE for 49 permutations under the null hypothesis of no competitor species identity effects. The colors indicate whether spatial cross-validation was used or not.

265 identity of the competitor, which is the opposite of what has been observed in other loca-

266 tions [(Allen & Kim 2020](#_bookmark21), [Uriarte et al. (2004)).](#_bookmark44) This provides a striking example of the

267 importance of cross-validation, as without it the over-fit model gives rise to an incorrect

268 conclusion.

269 **4 Conclusion and future work**

270 The forestecology package provides an accessible way to fit and test models of neighbor-

271 hood competition. The package follows the tidy data design principles, leverages the sf

272 package for spatial data, and S3 open-oriented model implementation structure (P[ebesma](#_bookmark36)

273 [2018).](#_bookmark36) We hope that the package will increase the use of neighborhood competition models

274 to better understand what structures plant competition.

275 While the package is designed with ForestGEO plot data in mind, we envision that it

276 can be modified to work on any single large, mapped forest plot in which at least two mea-

277 surements of each individual have been taken. Furthermore, we hope that future versions of

278 the package will be flexible to other plot layouts, for example inventory plot-structure with

279 many spatially separated plots like the US Forest Service Forest Inventory and Analysis

280 plots [(Smith 2002)](#_bookmark41).

281 We also hope to extend the forestecology package’s functionality to account for a

282 larger variety of models for tree growth. One clear future direction would be to allow

283 competition based on species trait values rather than species identity. There is evidence

284 that traits predict competitive outcomes [(Kunstler et al. 2012,](#_bookmark34) [Lasky et al. (2014),](#_bookmark35) [Uriarte](#_bookmark45)

285 [et al. (2010)).](#_bookmark45) Thus an extension of the model would allow *λ* values from Equation [1](#_bookmark1) to be

286 a function of the traits of competing species.

287 Lastly, the forestecology current uses the blockCV package behind the scenes to cre-

288 ate the spatial blocks acting as folds for our spatial cross-validation algorithm detailed

289 in Sections [2.2](#_bookmark5) and [2.6](#_bookmark13) (V[alavi et al.](#_bookmark46) [2019).](#_bookmark46) This back-end functionality could be substi-

290 tuted with the spatialsample package for spatial resampling infrastructure; a tidymodels

291 package under active development as of 2021 [(Silge 2021,](#_bookmark39) [Kuhn & Wickham (2020)).](#_bookmark33)

292 Lastly, currently the package only implements the Bayesian linear regression model

293 detailed in Equation [1.](#_bookmark1) As we demonstrate in Section [2.4](#_bookmark9) however, the fitting of this

294 model is self-contained in a single function comp bayes lm() which returns an object of

295 S3 class type comp bayes lm. This class has generic methods implemented to print, make

296 predictions, and plot all results. Therefore the package can be modularly extended to fit

297 other models as long as they are coded similarly to comp bayes lm() and have equivalent

298 generic methods implemented.

299 **5 Acknowledgments**

300 We thank Sophie Li for their feedback on the package interface. The authors declare no

301 conflicts of interest.

302 **6 Author’s contributions**

303 AYK and DNA conceived the ideas and coded a draft of the package. AYK wrote an initial

304 manuscript draft. SPC rewrote much of the package’s code to align with R and “tidy” best

305 practices [(Wickham et al.](#_bookmark50) [2019).](#_bookmark50) All authors contributed to subsequent drafts and gave

306 final approval for manuscript.

307 **7 Data accessibility**

308 We intend to archive all data and source code for this manuscript on GitHub at [https://](https://github.com/rudeboybert/forestecology)

309 [github.com/rudeboybert/forestecology](https://github.com/rudeboybert/forestecology) and on Zenodo upon acceptance. The example

310 Smithsonian Conservation Biology Institute census and species information data are avail-

311 able on GitHub at [https://github.com/SCBI-ForestGEO/SCBI-ForestGEO-Data/tree/](https://github.com/SCBI-ForestGEO/SCBI-ForestGEO-Data/tree/master/)

312 [master/](https://github.com/SCBI-ForestGEO/SCBI-ForestGEO-Data/tree/master/) and are archived on Zenodo at <https://doi.org/10.5281/zenodo.2649301>[(Gonzalez-](#_bookmark30)

313 [Akre, McGregor, Anderson-Teixeira, Dow, Herrmann, Terrell, Kim, Vinod & Helcoski](#_bookmark30)

314 [2020).](#_bookmark30)

# 315 A Appendix: Compare different competitive distances

316 For all the above analyses we set the cut off distance (comp dist) for two stems to compete

317 as 7.5m. This distance has been estimated in previous neighborhood competition studies

318 in forests [(Canham et al. 2004,](#_bookmark26) [Uriarte et al. (2004),](#_bookmark44) [Canham et al. (2006)).](#_bookmark27) We used 7.5m

319 in [Allen & Kim](#_bookmark21) [(2020)](#_bookmark21) as an average of the values estimated in other studies. But our

320 package can be used to find which distance is best supported by the data. Here we provide

321 an example using another section of the SCBI plot to provide an additional example of the

322 cross-validation block layout. To speed computation we do not consider species differences

323 in competitive effects and treat all species as the same.

324 We observe in Figure [7](#_bookmark16) that a cut-off distance of approximately 6m minimizes the

325 cross-validation estimated RMSE.

census\_2013\_scbi <- read\_csv("scbi.stem2.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(

date = mdy(date),

dbh = as.numeric(dbh) / 10

) %>%

filter(gx < 400, gy < 400)

census\_2018\_scbi <- read\_csv("scbi.stem3.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(

date = mdy(date),

dbh = as.numeric(dbh) / 10

) %>%

filter(gx < 400, gy < 400)

growth\_scbi <- compute\_growth(

census\_1 = census\_2013\_scbi,

census\_2 = census\_2018\_scbi %>% filter(!str\_detect(codes, "R")),

id = "stemID"

) %>%

*# make all species the same, needs to be factor with at least two levels*

mutate(sp = factor("A", levels = c("A", "B"))) %>%

*# Compute basal area:*

mutate(basal\_area = 0.0001 \* pi \* (dbh1 / 2)^2)

study\_region\_scbi <- tibble( x = c(0, 400, 400, 0, 0),

y = c(0, 0, 400, 400, 0)

) %>%

sf\_polygon()

n\_fold <- 4

fold1 <- cbind(c(0, 200, 200, 0), c(0, 0, 200, 200))

fold2 <- cbind(c(200, 400, 400, 200), c(0, 0, 200, 200))

fold3 <- cbind(c(0, 200, 200, 0), c(200, 200, 400, 400))

fold4 <- cbind(c(200, 400, 400, 200), c(200, 200, 400, 400))

blocks\_scbi <- bind\_rows(

sf\_polygon(fold1), sf\_polygon(fold2), sf\_polygon(fold3), sf\_polygon(fold4)

) %>%

mutate(folds = c(1:n\_fold) %>% factor())

*# Associate each observation to a fold*

spatial\_block\_scbi <- spatialBlock(

speciesData = growth\_scbi, k = n\_fold, selection = "systematic", blocks = blocks\_scbi, showBlocks = FALSE, verbose = FALSE

)

growth\_scbi <- growth\_scbi %>%

mutate(foldID = spatial\_block\_scbi$foldID %>% factor())

mult\_dist\_comp <- tibble(

dist = c(5, 6.25, 7.5, 8.75, 10),

rmse = 0

)

for (i in 1:length(mult\_dist\_comp$dist)) *{*

comp\_dist <- mult\_dist\_comp$dist[i]

growth\_scbi <- growth\_scbi %>%

add\_buffer\_variable(size = comp\_dist, region = study\_region\_scbi)

focal\_vs\_comp\_scbi <- growth\_scbi %>%

create\_focal\_vs\_comp(comp\_dist = comp\_dist, blocks = blocks\_scbi, id = "stemID", co run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi)

mult\_dist\_comp$rmse[i] <- focal\_vs\_comp\_scbi %>%

0.128

0.127

RMSE (cm)

0.127

5 6 7 8 9 10

Distance (m)

Figure 7: Cross-validated RMSE estimates for 5 competitive distances2.

rmse(truth = growth, estimate = growth\_hat) %>%

pull(.estimate)

*}*

# 326 B Appendix: Compare competitor explanatory vari-

327 **ables**

328 In the above code we use the basal area of an individual as a continuous competitor ex-

329 planatory variable. But the package allows the user to specify any competitor explanatory

330 variable in the comp x var argument of create focal vs comp function. Here we use the

331 cross-validated model comparison to see which of two possible competitor explanatory vari-

332 ables computed in Section [2.1,](#_bookmark3) basal area or above ground biomass, best explains growth.

census\_2013\_scbi <- read\_csv("scbi.stem2.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(date = mdy(date), dbh = as.numeric(dbh)/10) %>%

filter(gx < 300, between(gy, 300, 600))

census\_2018\_scbi <- read\_csv("scbi.stem3.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(date = mdy(date), dbh = as.numeric(dbh)/10) %>%

filter(gx < 300, between(gy, 300, 600))

growth\_scbi <- compute\_growth(census\_1 = census\_2013\_scbi, census\_2 = census\_2018\_scbi filter(!str\_detect(codes, "R")), id = "stemID") %>%

left\_join(sp\_info, by = "sp") %>%

mutate(sp = as.factor(sp), basal\_area = 1e-04 \* pi \* (dbh1/2)^2, agb = get\_biomass( genus = genus, species = species, coords = c(-78.2, 38.9)))

study\_region\_scbi <- tibble(x = c(0, 300, 300, 0, 0), y = c(300, 300, 600,

600, 300)) %>%

sf\_polygon()

n\_fold <- 4

fold1 <- cbind(c(0, 150, 150, 0), c(300, 300, 450, 450))

fold2 <- cbind(c(150, 300, 300, 150), c(300, 300, 450, 450))

fold3 <- cbind(c(0, 150, 150, 0), c(450, 450, 600, 600))

fold4 <- cbind(c(150, 300, 300, 150), c(450, 450, 600, 600))

blocks\_scbi <- bind\_rows(sf\_polygon(fold1), sf\_polygon(fold2), sf\_polygon(fold3), sf\_polygon(fold4)) %>%

mutate(folds = c(1:n\_fold) %>% factor())

*# Associate each observation to a fold*

spatial\_block\_scbi <- spatialBlock(speciesData = growth\_scbi, k = n\_fold,

selection = "systematic", blocks = blocks\_scbi, showBlocks = FALSE, verbose = FALSE

growth\_scbi <- growth\_scbi %>%

mutate(foldID = spatial\_block\_scbi$foldID %>% factor())

comp\_dist <- 7.5

growth\_scbi <- growth\_scbi %>%

add\_buffer\_variable(size = comp\_dist, region = study\_region\_scbi)

focal\_vs\_comp\_ba <- growth\_scbi %>%

create\_focal\_vs\_comp(comp\_dist = comp\_dist, blocks = blocks\_scbi, id = "stemID", comp\_x\_var = "basal\_area") %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi)

focal\_vs\_comp\_agb <- growth\_scbi %>%

create\_focal\_vs\_comp(comp\_dist = comp\_dist, blocks = blocks\_scbi, id = "stemID", comp\_x\_var = "agb") %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi)

focal\_vs\_comp\_ba %>%

rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

*## [1] 0.14*

focal\_vs\_comp\_agb %>%

rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

*## [1] 2.08*

333 Here we observe that basal area is a better competitor explanatory variable competitor

334 explanatory variables *x*comp from Equation [1](#_bookmark1) than above ground biomass as suggested by

*ijk*

335 the lower estimated RMSE.

# 336 C Appendix: Compare grouping variables

337 The package also allows the user to specify the categorical explanatory grouping variable.

338 Here we compare two different such variables: species and the potential canopy position of

339 that species. If we had individual-level crown classes (S[mith (1986)](#_bookmark40) dominant, codominant,

340 intermediate and suppressed) that could also be used.

census\_2013\_scbi <- read\_csv("scbi.stem2.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(date = mdy(date), dbh = as.numeric(dbh)/10) %>%

filter(gx < 300, between(gy, 300, 600))

census\_2018\_scbi <- read\_csv("scbi.stem3.csv") %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(date = mdy(date), dbh = as.numeric(dbh)/10) %>%

filter(gx < 300, between(gy, 300, 600))

growth\_scbi\_sp <- compute\_growth(census\_1 = census\_2013\_scbi, census\_2 = census\_2018\_sc filter(!str\_detect(codes, "R")), id = "stemID") %>%

mutate(sp = as.factor(sp), basal\_area = 1e-04 \* pi \* (dbh1/2)^2)

growth\_scbi\_can\_pos <- compute\_growth(census\_1 = census\_2013\_scbi, census\_2 filter(!str\_detect(codes, "R")), id = "stemID") %>%

left\_join(sp\_info, by = "sp") %>%

= census\_20

mutate(canopy\_position = str\_replace(canopy\_position, " ", "\_"), canopy\_position = ",", ""), canopy\_position = ifelse(is.na(canopy\_position), "shrub\_layer", canopy\_position), sp = as.factor(canopy\_position), basal\_area = 1e-04 \*

pi \* (dbh1/2)^2)

study\_region\_scbi <- tibble(x = c(0, 300, 300, 0, 0), y = c(300, 300, 600,

600, 300)) %>%

sf\_polygon()

n\_fold <- 4

fold1 <- cbind(c(0, 150, 150, 0), c(300, 300, 450, 450))

fold2 <- cbind(c(150, 300, 300, 150), c(300, 300, 450, 450))

fold3 <- cbind(c(0, 150, 150, 0), c(450, 450, 600, 600))

fold4 <- cbind(c(150, 300, 300, 150), c(450, 450, 600, 600))

blocks\_scbi <- bind\_rows(sf\_polygon(fold1), sf\_polygon(fold2), sf\_polygon(fold3), sf\_polygon(fold4)) %>%

mutate(folds = c(1:n\_fold) %>% factor())

*# Associate each observation to a fold*

spatial\_block\_scbi <- spatialBlock(speciesData = growth\_scbi, k = n\_fold,

selection = "systematic", blocks = blocks\_scbi, showBlocks = FALSE, verbose = FALSE

growth\_scbi\_sp <- growth\_scbi\_sp %>% mutate(foldID = spatial\_block\_scbi$foldID %>%

factor())

growth\_scbi\_can\_pos <- growth\_scbi\_can\_pos %>% mutate(foldID = spatial\_block\_scbi$foldID %>%

factor()) comp\_dist <- 7.5

growth\_scbi\_sp <- growth\_scbi\_sp %>%

add\_buffer\_variable(size = comp\_dist, region = study\_region\_scbi) growth\_scbi\_can\_pos <- growth\_scbi\_can\_pos %>%

add\_buffer\_variable(size = comp\_dist, region = study\_region\_scbi)

focal\_vs\_comp\_sp <- growth\_scbi\_sp %>%

create\_focal\_vs\_comp(comp\_dist = comp\_dist, blocks = blocks\_scbi, id = "stemID", comp\_x\_var = "basal\_area") %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi)

focal\_vs\_comp\_can\_pos <- growth\_scbi\_can\_pos %>%

create\_focal\_vs\_comp(comp\_dist = comp\_dist, blocks = blocks\_scbi, id = "stemID", comp\_x\_var = "basal\_area") %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi)

focal\_vs\_comp\_sp %>%

rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

*## [1] 0.14*

focal\_vs\_comp\_can\_pos %>%

rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

*## [1] 0.144*

341 We find that species identity has a lower RMSE, so does a better job. We still however

342 plot the competition posteriors for the canopy position groupings in Figure [8.](#_bookmark19) Unsurpris-

343 ingly we see that canopy and canopy emergent competitors generally have negative effects

344 on their neighbors, while shrubs and understory competitors have neutral or even positive

345 effects.

fit\_mod\_can\_pos <- growth\_scbi\_can\_pos %>% create\_focal\_vs\_comp(

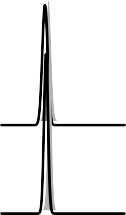
comp\_dist = comp\_dist,

blocks = blocks\_scbi, id = "stemID",

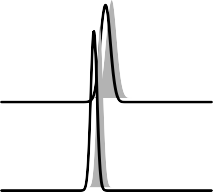
Competitor species in rows, focal species in columns

Ex: Top row, second column: competitive effect of canopy on canopy\_emer

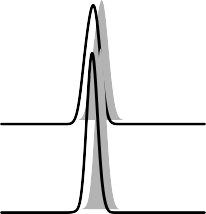
canopy



canopy



canopy\_emergent



shrub\_layer



understory

canopy\_emergent

species

shrub\_layer

understory

−1.0 −0.5 0.0 0.5 1.0−1.0 −0.5 0.0 0.5 1.0−1.0 −0.5 0.0 0.5 1.0−1.0 −0.5 0.0 0.5 1.0



Figure 8: Posterior distributions of all competition parameters.

comp\_x\_var = "basal\_area"

) %>%

comp\_bayes\_lm(prior\_param = NULL)

# 346 D Appendix: Replicate RMSE comparison

347 This code replicates Figure [6](#_bookmark14): A comparison of root mean squared error of models for

348 standard, permuted, and spatial cross-validated error estimates.

library(tidyverse) library(lubridate) library(here) library(sf)

library(viridis)

library(forestecology) library(blockCV) library(tictoc)

*# Compute growth of trees based on census data #*

census\_2013\_scbi <- here("paper/scbi.stem2.csv") %>% read\_csv() %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(date = mdy(date), dbh = as.numeric(dbh)/10) %>%

filter(gx < 300, between(gy, 300, 600))

census\_2018\_scbi <- here("paper/scbi.stem3.csv") %>% read\_csv() %>%

select(stemID, sp, date = ExactDate, gx, gy, dbh, codes, status) %>% mutate(date = mdy(date), dbh = as.numeric(dbh)/10) %>%

filter(gx < 300, between(gy, 300, 600))

growth\_scbi <- compute\_growth(census\_1 = census\_2013\_scbi, census\_2 = census\_2018\_scbi filter(!str\_detect(codes, "R")), id = "stemID") %>%

*# Compute basal area:*

mutate(basal\_area = 1e-04 \* pi \* (dbh1/2)^2)

*# Add spatial information*

*#*

*# using competitive distance range*

comp\_dist <- 7.5

*Define buffer region*

study\_region\_scbi <- tibble(x = c(0, 300, 300, 0, 0), y = c(300, 300, 600,

600, 300)) %>%

sf\_polygon()

growth\_scbi <- growth\_scbi %>%

add\_buffer\_variable(size = comp\_dist, region = study\_region\_scbi)

*# Manually define spatial blocks to act as folds*

fold1 <- rbind(c(0, 300), c(150, 300), c(150, 450), c(0, 450))

fold2 <- rbind(c(150, 300), c(300, 300), c(300, 450), c(150, 450))

fold3 <- rbind(c(0, 450), c(150, 450), c(150, 600), c(0, 600))

fold4 <- rbind(c(150, 450), c(300, 450), c(300, 600), c(150, 600))

n\_fold <- 4

blocks\_scbi <- bind\_rows(sf\_polygon(fold1), sf\_polygon(fold2), sf\_polygon(fold3), sf\_polygon(fold4)) %>%

mutate(folds = c(1:n\_fold) %>% factor())

*# Associate each observation to a fold*

SpatialBlock\_scbi <- spatialBlock(speciesData = growth\_scbi, k = n\_fold,

selection = "systematic", blocks = blocks\_scbi, showBlocks = FALSE, verbose = FALSE

growth\_scbi <- growth\_scbi %>%

mutate(foldID = SpatialBlock\_scbi$foldID %>% factor())

*# Compute focal versus competitor tree information #*

focal\_vs\_comp\_scbi <- growth\_scbi %>%

create\_focal\_vs\_comp(comp\_dist, blocks = blocks\_scbi, id = "stemID", comp\_x\_var = "basal\_area")

*# Fit model and make predictions #*

*# shuffles:*

num\_shuffle <- 49

*# Save results here*

*Number of permutation*

run\_time <- 0

observed\_RMSE <- 0

observed\_RMSE\_CV <- 0 shuffle\_RMSE <- vector("list", 1)

shuffle\_RMSE\_CV <- vector("list", 1)

filename <- here("paper/simulation\_results/") %>% str\_c("2021-03-03\_scbi\_", num\_shuffle, "\_shuffles")

*# Run all simulations 0. Setup simulation for this species type ----*

*# Start clock*

tic()

*# 1. Compute observed test statistic: RMSE with no cross-validation # ---- Fit model (compute posterior parameters)*

comp\_bayes\_lm\_scbi <- focal\_vs\_comp\_scbi %>% comp\_bayes\_lm(prior\_param = NULL, run\_shuffle = FALSE)

*# Make predictions and compute RMSE*

observed\_RMSE <- focal\_vs\_comp\_scbi %>%

mutate(growth\_hat = predict(comp\_bayes\_lm\_scbi, focal\_vs\_comp\_scbi)) %>% rmse(truth = growth, estimate = growth\_hat) %>%

pull(.estimate)

*# 2. Compute observed test statistic: RMSE with cross-validation #*

observed\_RMSE\_CV <- focal\_vs\_comp\_scbi %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi) %>% rmse(truth = growth, estimate = growth\_hat) %>% pull(.estimate)

*# 3. Permutation distribution: RMSE with no cross-validation # ----------- Compute num\_shuffle permutation test statistics* shuffle\_RMSE <- numeric(length = num\_shuffle)

for (j in 1:num\_shuffle) *{*

*# Fit model (compute posterior parameters) with shuffling*

comp\_bayes\_lm\_scbi <- focal\_vs\_comp\_scbi %>% comp\_bayes\_lm(prior\_param = NULL, run\_shuffle = TRUE)

*# Make predictions and compute RMSE*

shuffle\_RMSE[j] <- focal\_vs\_comp\_scbi %>%

mutate(growth\_hat = predict(comp\_bayes\_lm\_scbi, focal\_vs\_comp\_scbi)) %>% rmse(truth = growth, estimate = growth\_hat) %>%

pull(.estimate)

*}*

*# 4. Permutation distribution: RMSE with cross-validation*

*# -------------- Compute num\_shuffle permutation test statistics*

shuffle\_RMSE\_CV <- numeric(length = num\_shuffle)

*# Compute num\_shuffle permutation test statistics*

for (j in 1:num\_shuffle) *{*

*# Compute and save RMSE*

shuffle\_RMSE\_CV[j] <- focal\_vs\_comp\_scbi %>%

run\_cv(comp\_dist = comp\_dist, blocks = blocks\_scbi, run\_shuffle = TRUE) %>% rmse(truth = growth, estimate = growth\_hat) %>%

pull(.estimate)

*# Status update*

str\_c("Shuffle with permutation ", j, " at ", Sys.time()) %>% print()

*}*

*# 5. Save results ----*

clock <- toc(quiet = TRUE) run\_time <- clock$toc - clock$tic

model\_comp\_tbl <- tibble(run\_time = run\_time, observed\_RMSE = observed\_RMSE, observed\_RMSE\_CV = observed\_RMSE\_CV, shuffle\_RMSE = shuffle\_RMSE, shuffle\_RMSE\_CV =

)

save(model\_comp\_tbl, file = filename %>% str\_c(".RData"))

*# Visualize results #*

model\_comp <- bind\_rows(model\_comp\_tbl %>%

select(run\_time, observed = observed\_RMSE, shuffle = shuffle\_RMSE) %>% mutate(CV = FALSE), model\_comp\_tbl %>%

select(run\_time, observed = observed\_RMSE\_CV, shuffle = shuffle\_RMSE\_CV) %>% mutate(CV = TRUE)) %>%

gather(type, RMSE, -c(run\_time, CV))

model\_comp\_observed <- model\_comp %>% filter(type == "observed") %>% unnest(cols = c(RMSE))

model\_comp\_shuffle <- model\_comp %>% filter(type == "shuffle") %>% unnest(cols = c(RMSE))

cv\_plot <- ggplot() + geom\_vline(data = model\_comp\_observed, aes(xintercept = RMSE,

col = CV), linetype = "dashed", show.legend = F) + geom\_histogram(data = model\_comp aes(x = RMSE, fill = CV), bins = 50) + labs(fill = "Cross-validated?",

x = expression(paste("RMSE (cm ", y^*{*

-1

*}*, ")"))) + scale\_color\_viridis(discrete = TRUE, option = "D") + scale\_fill\_viridis theme\_light()

cv\_plot

filename %>%

str\_c(".pdf") %>%

ggsave(plot = cv\_plot, width = 16/2, height = 9/2)

349 **References**

350 Allen, D., Dick, C., Burnham, R. J., Perfecto, I. & Vandermeer, J. (2020), ‘The Michigan

351 Big Woods research plot at the Edwin S. George, Pinckney, MI, USA’, *Miscellaneous*

352 *Publications of the Museum of Zoology, University of Michigan* **207**.

353 **URL:** [*http://hdl.handle.net/2027.42/156251*](http://hdl.handle.net/2027.42/156251)

354 Allen, D. & Kim, A. Y. (2020), ‘A permutation test and spatial cross-validation approach

355 to assess models of interspecific competition between trees’, *PLOS ONE* **15**(3), e0229930.

356 Publisher: Public Library of Science.

357 **URL:** *https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0229930*

358 Anderson-Teixeira, K. J., Davies, S. J., Bennett, A. C., Gonzalez-Akre, E. B., Muller-

359 Landau, H. C., Wright, S. J., Salim, K. A., Zambrano, A. M. A., Alonso, A., Baltzer,

360 J. L., Basset, Y., Bourg, N. A., Broadbent, E. N., Brockelman, W. Y., Bunyavejchewin,

361 S., Burslem, D. F. R. P., Butt, N., Cao, M., Cardenas, D., Chuyong, G. B., Clay, K.,

362 Cordell, S., Dattaraja, H. S., Deng, X., Detto, M., Du, X., Duque, A., Erikson, D. L.,

363 Ewango, C. E. N., Fischer, G. A., Fletcher, C., Foster, R. B., Giardina, C. P., Gilbert,

364 G. S., Gunatilleke, N., Gunatilleke, S., Hao, Z., Hargrove, W. W., Hart, T. B., Hau, B.

365 C. H., He, F., Hoffman, F. M., Howe, R. W., Hubbell, S. P., Inman-Narahari, F. M.,

366 Jansen, P. A., Jiang, M., Johnson, D. J., Kanzaki, M., Kassim, A. R., Kenfack, D.,

367 Kibet, S., Kinnaird, M. F., Korte, L., Kral, K., Kumar, J., Larson, A. J., Li, Y., Li, X.,

368 Liu, S., Lum, S. K. Y., Lutz, J. A., Ma, K., Maddalena, D. M., Makana, J.-R., Malhi,

369 Y., Marthews, T., Serudin, R. M., McMahon, S. M., McShea, W. J., Memiaghe, H. R.,

370 Mi, X., Mizuno, T., Morecroft, M., Myers, J. A., Novotny, V., Oliveira, A. A. d., Ong,

371 P. S., Orwig, D. A., Ostertag, R., Ouden, J. d., Parker, G. G., Phillips, R. P., Sack, L.,

372 Sainge, M. N., Sang, W., Sri-ngernyuang, K., Sukumar, R., Sun, I.-F., Sungpalee, W.,

373 Suresh, H. S., Tan, S., Thomas, S. C., Thomas, D. W., Thompson, J., Turner, B. L.,

374 Uriarte, M., Valencia, R., Vallejo, M. I., Vicentini, A., Vrˇska, T., Wang, X., Wang, X.,

375 Weiblen, G., Wolf, A., Xu, H., Yap, S. & Zimmerman, J. (2015), ‘CTFS-ForestGEO: a

376 worldwide network monitoring forests in an era of global change’, *Global Change Biology*

377 **21**(2), 528–549.

378 **URL:** [*http://onlinelibrary.wiley.com/doi/abs/10.1111/gcb.12712*](http://onlinelibrary.wiley.com/doi/abs/10.1111/gcb.12712)

379 Bache, S. M. & Wickham, H. (2020), *magrittr: A Forward-Pipe Operator for R*. R package

380 version 2.0.1.

381 **URL:** *https://CRAN.R-project.org/package=magrittr*

382 Bivand, R. S., Pebesma, E. & Gomez-Rubio, V. (2013), *Applied spatial data analysis with*

383 *R, Second edition*, Springer, NY.

384 **URL:** *https://asdar-book.org/*

385 Bourg, N. A., McShea, W. J., Thompson, J. R., McGarvey, J. C. & Shen, X. (2013), ‘Initial

386 census, woody seedling, seed rain, and stand structure data for the SCBI SIGEO Large

387 Forest Dynamics Plot’, *Ecology* **94**(9), 2111–2112.

388 **URL:** [*http://esajournals.onlinelibrary.wiley.com/doi/abs/10.1890/13-0010.1*](http://esajournals.onlinelibrary.wiley.com/doi/abs/10.1890/13-0010.1)

389 Canham, C. D., LePage, P. T. & Coates, K. D. (2004), ‘A neighborhood analysis of canopy

390 tree competition: effects of shading versus crowding’, *Canadian Journal of Forest Re-*

391 *search* **34**(4), 778–787. Publisher: NRC Research Press Ottawa, Canada.

392 **URL:** *https://cdnsciencepub.com/doi/abs/10.1139/x03-232*

393 Canham, C. D., Papaik, M. J., Uriarte, M., McWilliams, W. H., Jenkins, J. C.

394 & Twery, M. J. (2006), ‘Neighborhood Analyses Of Canopy Tree Competi-

395 tion Along Environmental Gradients In New England Forests’, *Ecological Applica-*

396 *tions* **16**(2), 540–554. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1890/1051-

397 0761%282006%29016%5B0540%3ANAOCTC%5D2.0.CO%3B2.

398 Das, A. (2012), ‘The effect of size and competition on tree growth rate in old-growth

399 coniferous forests’, *Canadian Journal of Forest Research* **42**, 1983–1995.

400 Davies, S. J., Abiem, I., Abu Salim, K., Aguilar, S., Allen, D., Alonso, A., Anderson-

401 Teixeira, K., Andrade, A., Arellano, G., Ashton, P. S., Baker, P. J., Baker, M. E.,

402 Baltzer, J. L., Basset, Y., Bissiengou, P., Bohlman, S., Bourg, N. A., Brockelman, W. Y.,

403 Bunyavejchewin, S., Burslem, D. F., Cao, M., C´ardenas, D., Chang, L.-W., Chang-Yang,

404 C.-H., Chao, K.-J., Chao, W.-C., Chapman, H., Chen, Y.-Y., Chisholm, R. A., Chu, C.,

405 Chuyong, G., Clay, K., Comita, L. S., Condit, R., Cordell, S., Dattaraja, H. S., de

406 Oliveira, A. A., den Ouden, J., Detto, M., Dick, C., Du, X., A´lvaro Duque, Ediriweera,

407 S., Ellis, E. C., Obiang, N. L. E., Esufali, S., Ewango, C. E., Fernando, E. S., Filip,

408 J., Fischer, G. A., Foster, R., Giambelluca, T., Giardina, C., Gilbert, G. S., Gonzalez-

409 Akre, E., Gunatilleke, I., Gunatilleke, C., Hao, Z., Hau, B. C., He, F., Ni, H., Howe,

410 R. W., Hubbell, S. P., Huth, A., Inman-Narahari, F., Itoh, A., Jan´ık, D., Jansen, P. A.,

411 Jiang, M., Johnson, D. J., Jones, F. A., Kanzaki, M., Kenfack, D., Kiratiprayoon, S.,

412 Kr´al, K., Krizel, L., Lao, S., Larson, A. J., Li, Y., Li, X., Litton, C. M., Liu, Y., Liu,

413 S., Lum, S. K., Luskin, M. S., Lutz, J. A., Luu, H. T., Ma, K., Makana, J.-R., Malhi,

414 Y., Martin, A., McCarthy, C., McMahon, S. M., McShea, W. J., Memiaghe, H., Mi,

415 X., Mitre, D., Mohamad, M., Monks, L., Muller-Landau, H. C., Musili, P. M., Myers,

416 J. A., Nathalang, A., Ngo, K. M., Norden, N., Novotny, V., O’Brien, M. J., Orwig,

417 D., Ostertag, R., Papathanassiou, K., Parker, G. G., P´erez, R., Perfecto, I., Phillips,

418 R. P., Pongpattananurak, N., Pretzsch, H., Ren, H., Reynolds, G., Rodriguez, L. J.,

419 Russo, S. E., Sack, L., Sang, W., Shue, J., Singh, A., Song, G.-Z. M., Sukumar, R., Sun,

420 I.-F., Suresh, H. S., Swenson, N. G., Tan, S., Thomas, S. C., Thomas, D., Thompson,

421 J., Turner, B. L., Uowolo, A., Uriarte, M., Valencia, R., Vandermeer, J., Vicentini, A.,

422 Visser, M., Vrska, T., Wang, X., Wang, X., Weiblen, G. D., Whitfeld, T. J., Wolf, A.,

423 Wright, S. J., Xu, H., Yao, T. L., Yap, S. L., Ye, W., Yu, M., Zhang, M., Zhu, D., Zhu, L.,

424 Zimmerman, J. K. & Zuleta, D. (2021), ‘Forestgeo: Understanding forest diversity and

425 dynamics through a global observatory network’, *Biological Conservation* **253**, 108907.

426 **URL:** [*https://www.scienc*](http://www.sciencedirect.com/science/article/pii/S0006320720309654)*edire*[*ct.com/science/article/pii/S0006320720309654*](http://www.sciencedirect.com/science/article/pii/S0006320720309654)

427 Gonzalez-Akre, E., McGregor, I., Anderson-Teixeira, K., Dow, C., Herrmann, V., Terrell,

428 A., Kim, A. Y., Vinod, N. & Helcoski, R. (2020), ‘SCBI-ForestGEO/SCBI-ForestGEO-

429 Data: 2020 update’.

430 **URL:** *https://doi.org/10.5281/zenodo.4041595*

431 Gonzalez-Akre, E., Piponiot, C., Lepore, M. & Anderson-Teixeira, K. (2020), *allodb: An*

432 *R Database For Biomass Estimation At Extratropical Forest Plots*. R package version

433 1.0.0.9000.

434 **URL:** *https://github.com/forestgeo/allodb*

435 Herring, J. R. (2011), OpenGIS implementation standard for geographic information-simple

436 feature access part 1: Common architecture, Open Geospatial Consortium Inc., p. 211.

437 Kuhn, M. & Wickham, H. (2020), *Tidymodels: a collection of packages for modeling and*

438 *machine learning using tidyverse principles.*

439 **URL:** [*https://www.tidymo*](http://www.tidymodels.org/)*dels.org*

440 Kunstler, G., Lavergne, S., Courbaud, B., Thuiller, W., Vieilledent, G., Zimmermann,

441 N. E., Kattge, J. & Coomes, D. A. (2012), ‘Competitive interactions between forest

442 trees are driven by species’ trait hierarchy, not phylogenetic or functional similarity:

443 implications for forest community assembly’, *Ecology Letters* **15**(8), 831–840. eprint:

444 https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1461-0248.2012.01803.x.

445 **URL:** [*http://onlinelibrary.wiley.com/doi/abs/10.1111/j.1461-0248.2012.01803.x*](http://onlinelibrary.wiley.com/doi/abs/10.1111/j.1461-0248.2012.01803.x)

446 Lasky, J. R., Uriarte, M., Boukili, V. K. & Chazdon, R. L. (2014), ‘Trait-mediated assembly

447 processes predict successional changes in community diversity of tropical forests’, *Pro-*

448 *ceedings of the National Academy of Sciences* **111**(15), 5616–5621. Publisher: National

449 Academy of Sciences Section: Biological Sciences.

450 **URL:** [*https://www.pnas.or*](http://www.pnas.org/content/111/15/5616)*g/c*[*ontent/111/15/5616*](http://www.pnas.org/content/111/15/5616)

451 Pebesma, E. (2018), ‘Simple Features for R: Standardized Support for Spatial Vector Data’,

452 *The R Journal* **10**(1), 439–446.

453 **URL:** *https://journal.r-project.org/archive/2018/RJ-2018-009/index.html*

454 Pohjankukka, J., Pahikkala, T., Nevalainen, P. & Heikkonen, J. (2017), ‘Estimating the

455 prediction performance of spatial models via spatial k-fold cross validation’, *International*

456 *Journal of Geographical Information Science* **31**(10), 2001–2019.

457 Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Arroita, G., Hauen-

458 stein, S., Lahoz-Monfort, J. J., Schr¨oder, B., Thuiller, W., Warton, D. I., Wintle, B. A.,

459 Hartig, F. & Dormann, C. F. (2017), ‘Cross-validation strategies for data with temporal,

460 spatial, hierarchical, or phylogenetic structure’, *Ecography* **40**(8), 913–929.

461 **URL:** [*http://onlinelibrary.wiley.com/doi/abs/10.1111/ecog.02881*](http://onlinelibrary.wiley.com/doi/abs/10.1111/ecog.02881)

462 Silge, J. (2021), *spatialsample: Spatial Resampling Infrastructure*. R package version 0.1.0.

463 **URL:** *https://CRAN.R-project.org/package=spatialsample*

464 Smith, D. M. (1986), ‘The practice of silviculture’.

465 Smith, W. B. (2002), ‘Forest inventory and analysis: a national inventory and monitoring

466 program’, *Environmental pollution* **116**, S233–S242.

467 Tatsumi, S., Owari, T. & Mori, A. S. (2016), ‘Estimating competition coefficients in tree

468 communities: a hierarchical bayesian approach to neighborhood analysis’, *Ecosphere*

469 **7**, e01273.

470 Team, G. D. (2017), GEOS - geometry engine, open source, Open Source Geospatial Foun-

471 dation, p. 211.

472 **URL:** *https://trac.osgeo.org/geos/*

473 Uriarte, M., Condit, R., Canham, C. D. & Hubbell, S. P. (2004), ‘A spa-

474 tially explicit model of sapling growth in a tropical forest: does the iden-

475 tity of neighbours matter?’, *Journal of Ecology* **92**(2), 348–360. eprint:

476 https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.0022-0477.2004.00867.x.

477 **URL:** [*http://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/j.0022-*](http://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/j.0022-)

478 *0477.2004.00867.x*

479 Uriarte, M., Swenson, N. G., Chazdon, R. L., Comita, L. S., Kress, W. J., Erickson, D.,

480 Forero-Montan`a, J., Zimmeran, J. K. & Thompson, J. (2010), ‘Trait similarity, shared

481 ancestry and the structure of neighbourhood interactions in a subtropical wet forest:

482 implications for community assembly’, *Ecology Letters* **13**, 1503–1514.

483 Valavi, R., Elith, J., Lahoz-Monfort, J. J. & Guillera-Arroita, G. (2019), ‘blockCV: An

484 r package for generating spatially or environmentally separated folds for k-fold cross-

485 validation of species distribution models’, *Methods in Ecology and Evolution* **10**(2), 225–

486 232. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/2041-210X.13107.

487 **URL:** [*http://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13107*](http://besjournals.onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13107)

488 Waller, L. A. & Gotway, C. A. (2004), *Applied Spatial Statistics for Public Health Data*,

489 John Wiley & Sons, Incorporated, Hoboken, UNITED STATES.

490 **URL:** [*http://ebookcentral.proquest.com/lib/smith/detail.action?docID=214360*](http://ebookcentral.proquest.com/lib/smith/detail.action?docID=214360)

491 Warmerdam, F. (2008), The Geospatial Data Abstraction Library, *in* G. B. Hall & M. G.

492 Leahy, eds, ‘Open Source Approaches in Spatial Data Handling’, Advances in Geographic

493 Information Science, Springer, Berlin, Heidelberg, pp. 87–104.

494 Wickham, H. (2020), *tidyr: Tidy Messy Data*. R package version 1.1.2.

495 **URL:** *https://CRAN.R-project.org/package=tidyr*

496 Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., Fran¸cois, R., Grole-

497 mund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache,

498 S. M., Mu¨ller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K.,

499 Vaughan, D., Wilke, C., Woo, K. & Yutani, H. (2019), ‘Welcome to the Tidyverse’,

500 *Journal of Open Source Software* **4**(43), 1686.

501 **URL:** *https://joss.theoj.org/papers/10.21105/joss.01686*