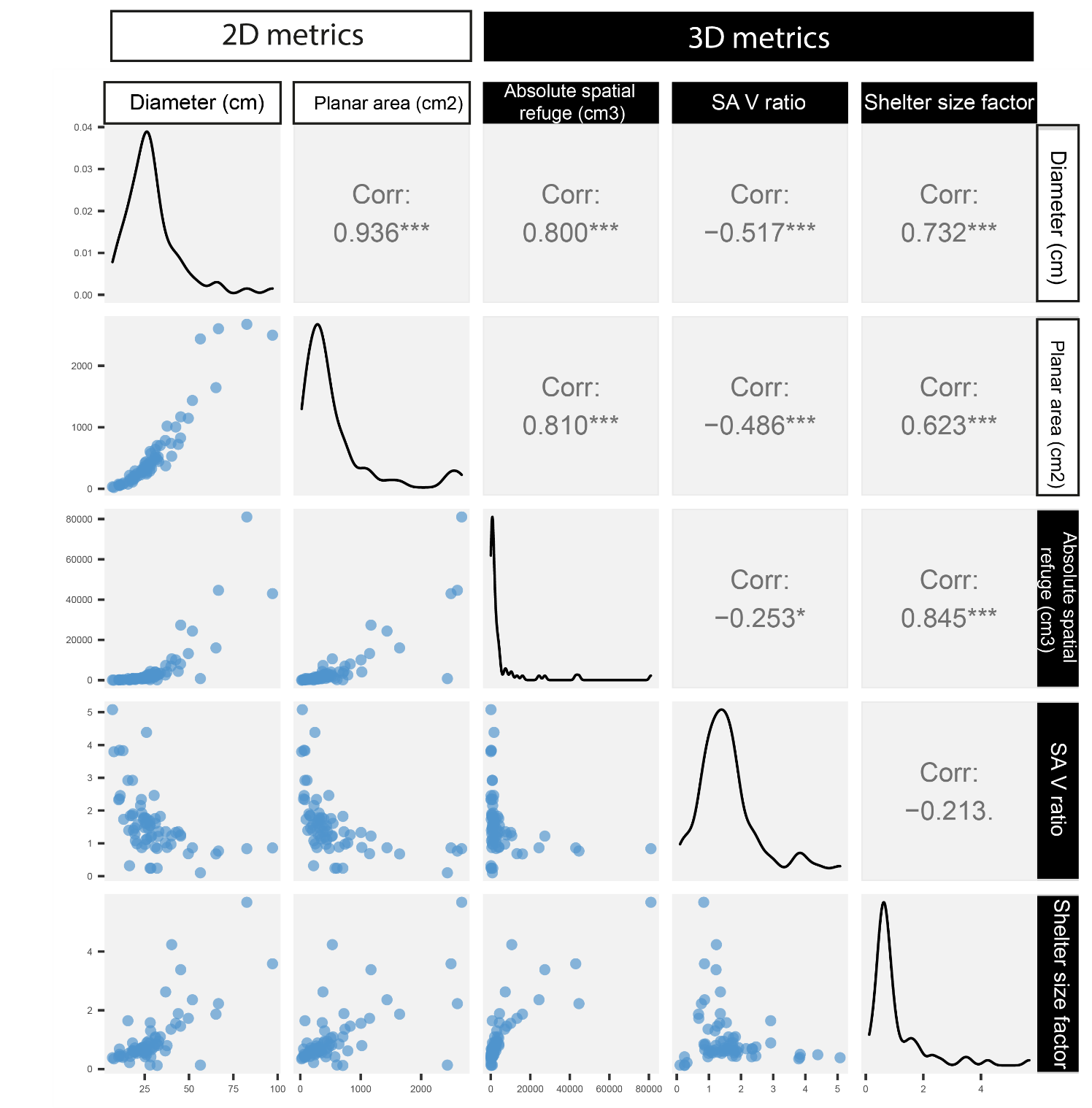
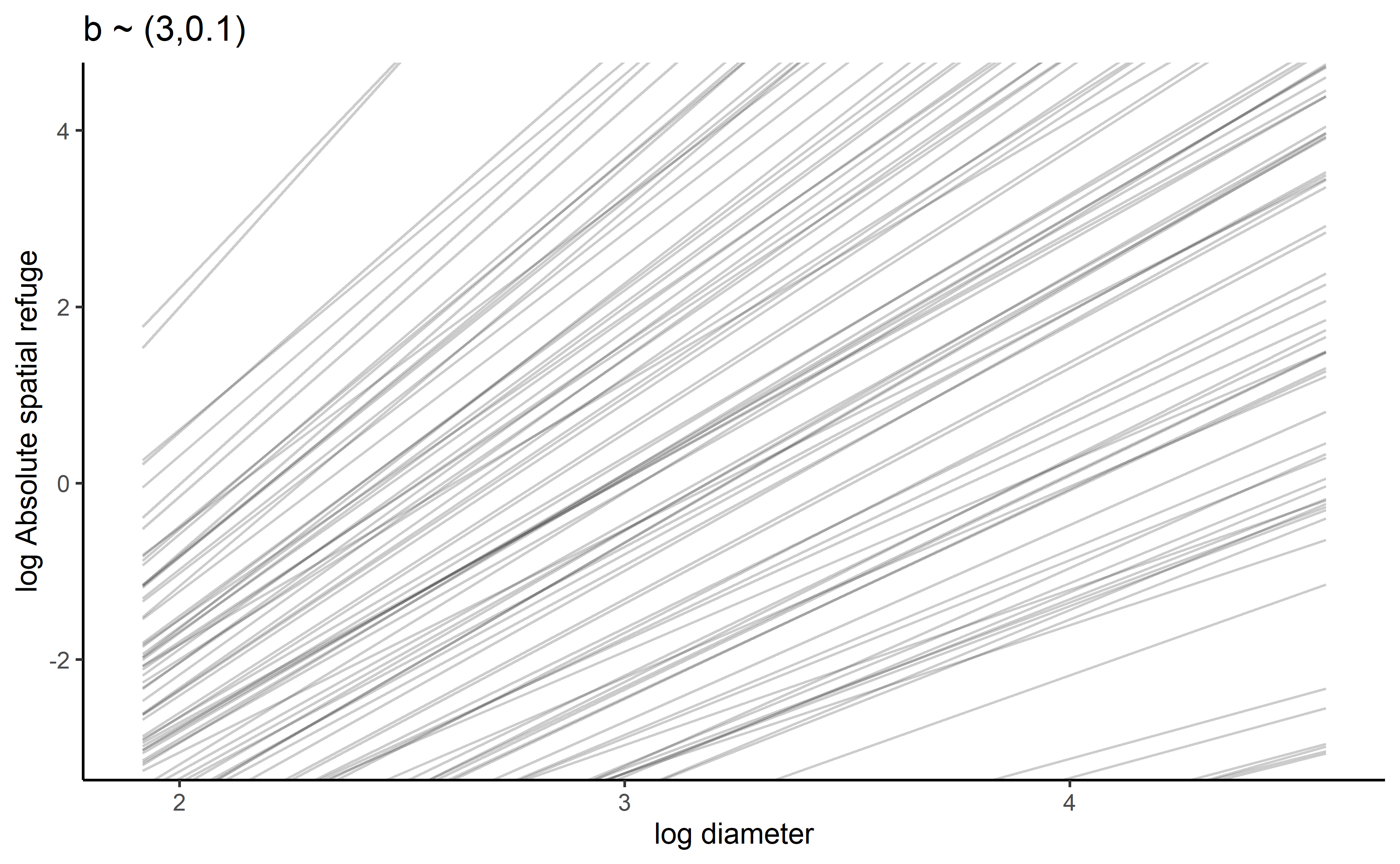
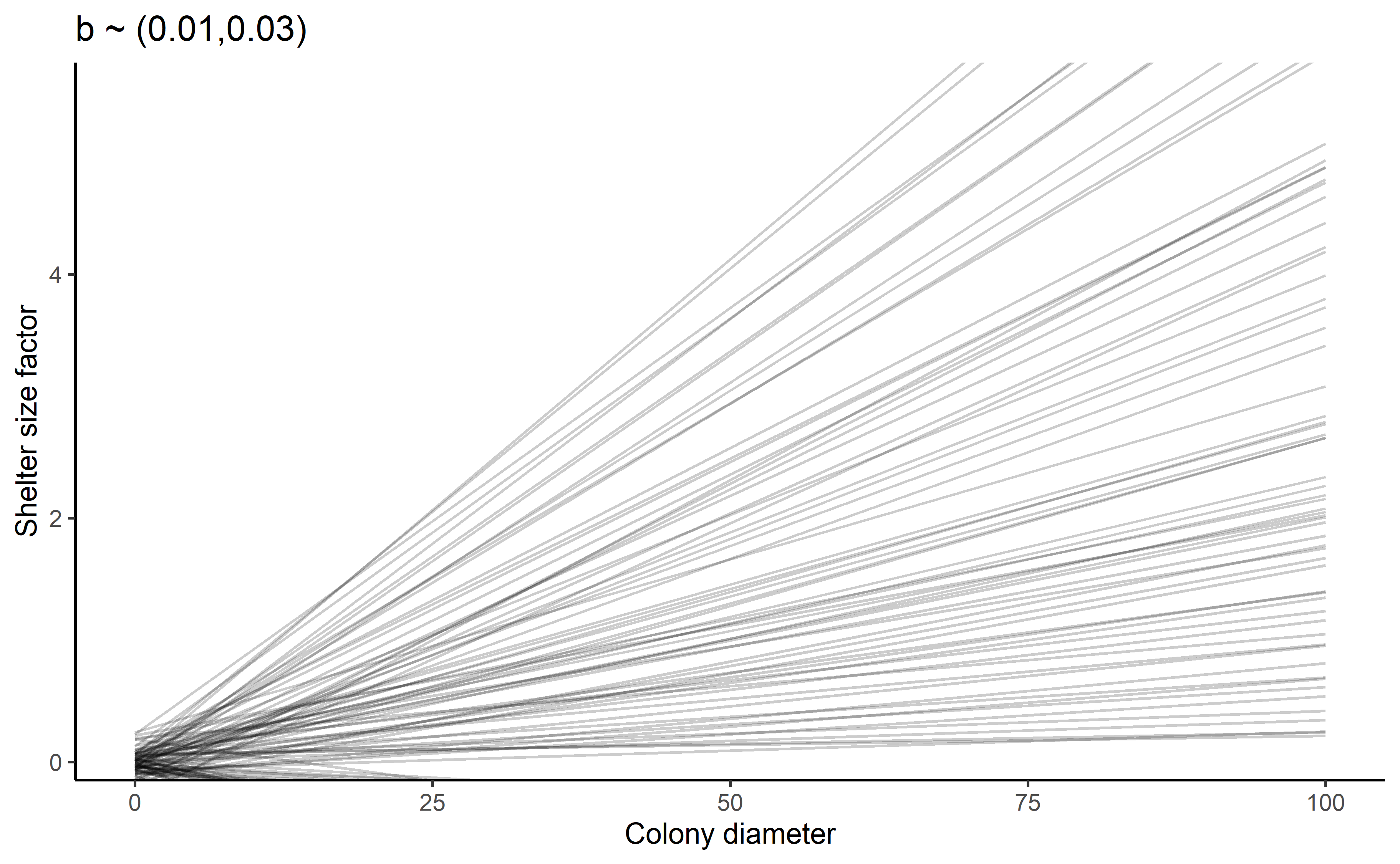


**Figure S1.** Graphical representation of several structural complexity metrics calculated. (A) Colony diameter measures along the widest axis, using the 3D model. (B) Colony height. (C) 2D surface area, measuring the coral footprint using the ruler tool. (D) 3D model used to calculate volume and 3D surface area of the colony. (E) Minimum bounding convex hull encompassing the colony. (F) Wireframe of convex hull overlaid over a tabular colony to show the area of overhang captured in estimates.



**Figure S2**. Pairwise correlations between 2D and 3D variables (raw data). Blue circles are individual coral colonies. Pearson correlation coefficients are available in the upper right panels for each pair of variables. Significance of relationships is denoted by asterisks (\*\*\* = p<0.01). Variable distribution is available on the diagonal.

**Figure S3**: (left) 100 randomly sampled lines from prior predictive distribution simulations for linear regression model predicting log absolute refuge using log colony diameter. Prior values for the intercept and slope were chosen based on Urbina-Barreto et al. (2020), since we performed identical data transformations. The intercept prior was not very informative here since we wanted to allow the data to drive differences in the data. (right) 100 randomly sampled lines from prior predictive distribution simulations for linear regression model predicting shelter size factor from planar area from colony diameter (cm). Prior values were more relaxed for this model since we had no knowledge of how shelter size factor was likely to change with increasing colony size. However, a colony of zero diameter would necessarily have a shelter size factor of 0, we set the intercept close to 0 and allowed only slight variation.



**Table S1**: Posterior estimates for mean parameters for interaction model predicting absolute spatial refuge with increasing colony diameter. 95% estimates are for compatibility intervals. Rhat values of 1 indicate that the within and between-chain estimates are in agreement, and that the chains have mixed well. Bulk and Tail ESS (effective sample size) values measure sampling efficiency in the bulk and the tails of the of the distribution. These should be at least 100 per markov chain (in this case, 4 chains were used).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Estimate | Est. error | Lower 95% CI | Upper 95% CI | Rhat | Bulk-ESS | Tail ESS |
| α Acropora Corymbose | -9.13 | 0.43 | -9.98 | -8.28 | 1.00 | 36873 | 28862 |
| α Acropora Digitate | -8.49 | 0.36 | -9.21 | -7.79 | 1.00 | 37353 | 29497 |
| α Acropora Tabular | -8.95 | 0.45 | -9.83 | -8.08 | 1.00 | 36171 | 29330 |
| α Isopora Branching | -8.80 | 0.48 | -9.74 | -7.86 | 1.00 | 38337 | 28892 |
| α Pocillopora Branching | -8.90 | 0.38 | -9.65 | -8.15 | 1.00 | 36811 | 28286 |
| α Porites Branching | -8.97 | -0.46 | -9.87 | -8.07 | 1.00 | 38289 | 27950 |
| α massive | -8.87 | 0.46 | -9.78 | -7.96 | 1.00 | 37375 | 28438 |
| β Acropora Corymbose | 2.94 | 0.14 | 2.65 | 3.21 | 1.00 | 36899 | 28917 |
| β Acropora Digitate | 2.73 | 0.12 | 2.49 | 2.97 | 1.00 | 37242 | 28101 |
| β Acropora Tabular | 2.80 | 0.13 | 2.54 | 3.06 | 1.00 | 36033 | 29009 |
| β Isopora Branching | 2.93 | 0.12 | 2.69 | 3.17 | 1.00 | 38811 | 28345 |
| β Pocillopora Branching | 2.94 | 0.12 | 2.71 | 3.18 | 1.00 | 36414 | 29743 |
| β Porites Branching | 3.14 | 0.14 | 2.86 | 3.42 | 1.00 | 38018 | 29233 |
| β massive | 2.30 | 0.14 | 2.02 | 2.58 | 1.00 | 37537 | 28800 |
| sigma | 0.39 | 0.04 | 0.32 | 0.47 | 1.00 | 37458 | 29653 |

**Table S2**. Results of pareto-smoothed importance sampling leave-one-out (PSIS-LOO) cross validation between the model predicting absolute spatial refuge using diameter with (Model1) and without (Model1.1) the indicator variable for morphotaxa. Note that all of the model weighting is given to the one including the indicator variable.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | elpd\_diff | s.e\_diff | elpd\_loo | s.e\_elpd\_loo | p\_loo | se\_p\_loo | looic | se\_looic | Model weights |
| Model1 | 0.0 | 0.0 | -39.0 | 8.4 | 11.8 | 3.4 | 77.9 | 16.9 | 1 |
| Model1.1 | -123 | 9 | -162 | 5.5 | 68.8 | 2.5 | 323 | 11 | 0 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Estimate | Est. error | Lower 95% CI | Upper 95% CI | Rhat | Bulk-ESS | Tail ESS |
| α Acropora Corymbose | -0.06 | 0.24 | -0.52 | 0.41 | 1.00 | 34163 | 27316 |
| α Acropora Digitate | 0.05 | 0.21 | -0.36 | 0.46 | 1.00 | 35385 | 28942 |
| α Acropora Tabular | -0.10 | 0.27 | -0.63 | 0.42 | 1.00 | 35975 | 26087 |
| α Isopora Branching | 1.14 | 0.40 | 0.34 | 1.91 | 1.00 | 34278 | 28823 |
| α Pocillopora Branching | -0.05 | 0.20 | -0.44 | 0.34 | 1.00 | 34946 | 28344 |
| α Porites Branching | 0.37 | 034 | -0.30 | 1.03 | 1.00 | 35784 | 29284 |
| α massive | -0.08 | 0.31 | -0.69 | 0.53 | 1.00 | 34474 | 30039 |
| β Acropora Corymbose | 3.58 | 0.93 | 1.74 | 5.41 | 1.00 | 35957 | 30551 |
| β Acropora Digitate | 2.66 | 0.79 | 1.10 | 4.22 | 1.00 | 33785 | 28701 |
| β Acropora Tabular | 3.25 | 0.77 | 1.74 | 4.76 | 1.00 | 36834 | 29297 |
| β Isopora Branching | 3.88 | 0.59 | 2.76 | 5.07 | 1.00 | 33321 | 29401 |
| β Pocillopora Branching | 3.27 | 0.65 | 1.99 | 4.55 | 1.00 | 33074 | 30450 |
| β Porites Branching | 5.98 | 1.05 | 3.94 | 8.03 | 1.00 | 35364 | 29430 |
| β massive | 0.93 | 0.93 | -0.90 | 2.77 | 1.00 | 34343 | 29761 |
| sigma | 0.40 | 0.04 | 0.33 | 0.09 | 1.00 | 36336 | 30396 |

**Table S3**. Posterior estimates for mean parameters for interaction model predicting shelter size factor with increasing colony planar area. 95% estimates are for compatibility intervals. Rhat values of 1 indicate that the within and between-chain estimates are in agreement, and that the chains have mixed well. Bulk and Tail ESS (effective sample size) values measure sampling efficiency in the bulk and the tails of the of the distribution. These should be at least 100 per markov chain (in this case, 4 chains were used).

**Table S4**. Results of pareto-smoothed importance sampling leave-one-out (PSIS-LOO) cross validation between the model predicting shelter size factor using diameter with (Model2) and without (Model2.1) the indicator variable for morphotaxa. Note that all of the model weighting is given to the one including the indicator variable.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | elpd\_diff | s.e\_diff | elpd\_loo | s.e\_elpd\_loo | p\_loo | se\_p\_loo | looic | se\_looic | Model weights |
| Model2 | 0.0 | 0.0 | -52.2 | 19.2 | 26.2 | 11.5 | 104.5 | 38.3 | 1 |
| Model2.1 | -22.9 | 13.4 | -75.1 | 13.9 | 7.8 | 3.9 | 150.2 | 27.9 | 0 |

**Instructions for using python scripts to serially process models and use scripts**

NOTE: Version control has been updated and checked for Python 3.10 and pymeshlab 2021.01 (As of Feb 2022 the most up-to-date version of both packages). Installation instructions are reflective of this.

Integration of MeshLab with python to compute several geometric measures of individual coral colonies. This is split in to two parts - processing of photographs in Metashape to turn these in to 3D reconstructions, as well as extracting complexity metrics. These scripts are easy to implement and follow assuming some knowledge of both using metashape and using a python interpreter / installing modules from the command line.

**Part I: Metashape processing**

This script is a downloadable .py file that is run directly from the metashape GUI (graphical user interface). For every chunk present in the project, it performs end-to-end digitisation of a coral colony according to some user-input values. Fortunately, these are pop-up boxes that have attached instructions. The only manual step required of the user is detecting markers and scaling models.

Requirements: Active license for metashape.

All you need to do is download the script "Metashape\_Processing.py" and run it from inside Metashape. The process is interactive at the beginning then applies the chosen settings to all chunks in the project. To set it up correctly, just populate the document with photos of colonies, each one of these added as a separate chunk.

To run the script, simply navigate to Tools > Run Script in the metashape software and locate the downloaded script.

Defaults (and recommended values) for processing are as follows:

Photo quality threshold: <0.35

Alignment quality: High (1)

Dense cloud quality: Medium (4)

Face count for model: High.

**Part II: Automated metrics extraction**

This code automates the extraction of 10 3D complexity metrics from an obj. file. It loops the function over every file in a specified folder, saving hours/days/weeks of clicking through pesky menus. The most efficient way of running this script is to put the .obj file of each coral in to a single folder, which you refer to in the script. (instructions are embedded)

Requirements: Python interpreter (we recommend PyCharm), installation of PyMeshlab using pip.

INSTRUCTIONS:

Install python (ver 3.6 or above will work but we assume a fresh installation of 3.10 at the time of writing). Do so directly from the Python website (https://www.python.org/downloads/)

When installing, ensure to check the box entitled "Add Python 3.XX to PATH" or nothing will work without manual intervention. ‘pip’ (the package responsible for installing other packages) comes installed by default.

We will use this to install the dependencies for the script. Open the command prompt terminal (press windows key, then type cmd to find it).

paste the following to install pymeshlab:

**pip install pymeshlab**

To run the code, a python interpreter is needed. We recommend the use of PyCharm, a freely available integrated development engine (IDE) capable of running the code. This software comes with installation and setup instructions for first time users which may vary between systems so we do not provide instructions on how to install here. Note that any interpreter can be used by those familiar with python.

TO MAKE THE CODE WORK: Create a new project in PyCharm (File > New Project) in your desired location, ensuring the "inherit global site packages" is checked, and that the base interpreter is your installed version of python. On creation, an empty project will automatically open.

Download all of the files in this repository. Then, cut and paste the downloaded files from the zipped folder to the location of the new project file that you just created in PyCharm. Once this is done, open the script "geometric\_measures.py" in the project. You should see also in the project tab that the other files are present. The two used in this are the .mlx files - these must both be present in the project file for the code to work.

There is only one thing for you to do to make this code run properly. the directory needs to be changed to direct python to the folder where all your .obj files are stored. This folder can include other file types, it won't break the script - the loop will just ignore all non-.obj files. It should look something like "C:/Documents/Project/objs/". Importantly you must use forward slashes and not backslashes (which are windows defaults). This is line 1 of the script and is marked "INPUT NEEDED".

The code runs a series of filters using Meshlab and returns the following as a .csv file called "Geometric Measures.csv" in the current working directory. It is populated with the following:

File\_Path : File path to the original input mesh. Identifies each coral in the file

Vol: Volume of first mesh (the coral)

CVH\_Vol: Volume of minimum bounding convex hull enclosing original mesh

ASR: Absolute spatial refuge. Volumetric measure of shelter capacity (interstitial space) of the object. Calculation : CVH\_Vol - Vol = ASR

PrOcc: Proportion Occupied. Proportion of the convex hull occupied by the coral lying inside it. Measures compactness. Calculation: Vol / CVH\_Vol = PrOcc

Surface\_Area: 3D surface area of input colony (not the convex hull)

SSF: Shelter size factor. Ratio of ASR to 3D surface area. Measure of size structure of refuges. Calculation: ASR / Surface\_area = SSF

Diameter: Maximum colony diameter (length along x axis which is by default the longest horizontal axis in meshlab)

Height: Colony height (length along Z-axis of bounding box)

Measurement units depend on the input mesh. Transformations must be carried out by the user to get to square and cubic cm. Note that your models must have been scaled in the software you used to create them for this code to work.