Model evaluation

Introduction to SDMs: theory and practice in R

Sapienza University, Rome

9-11 June, 2021

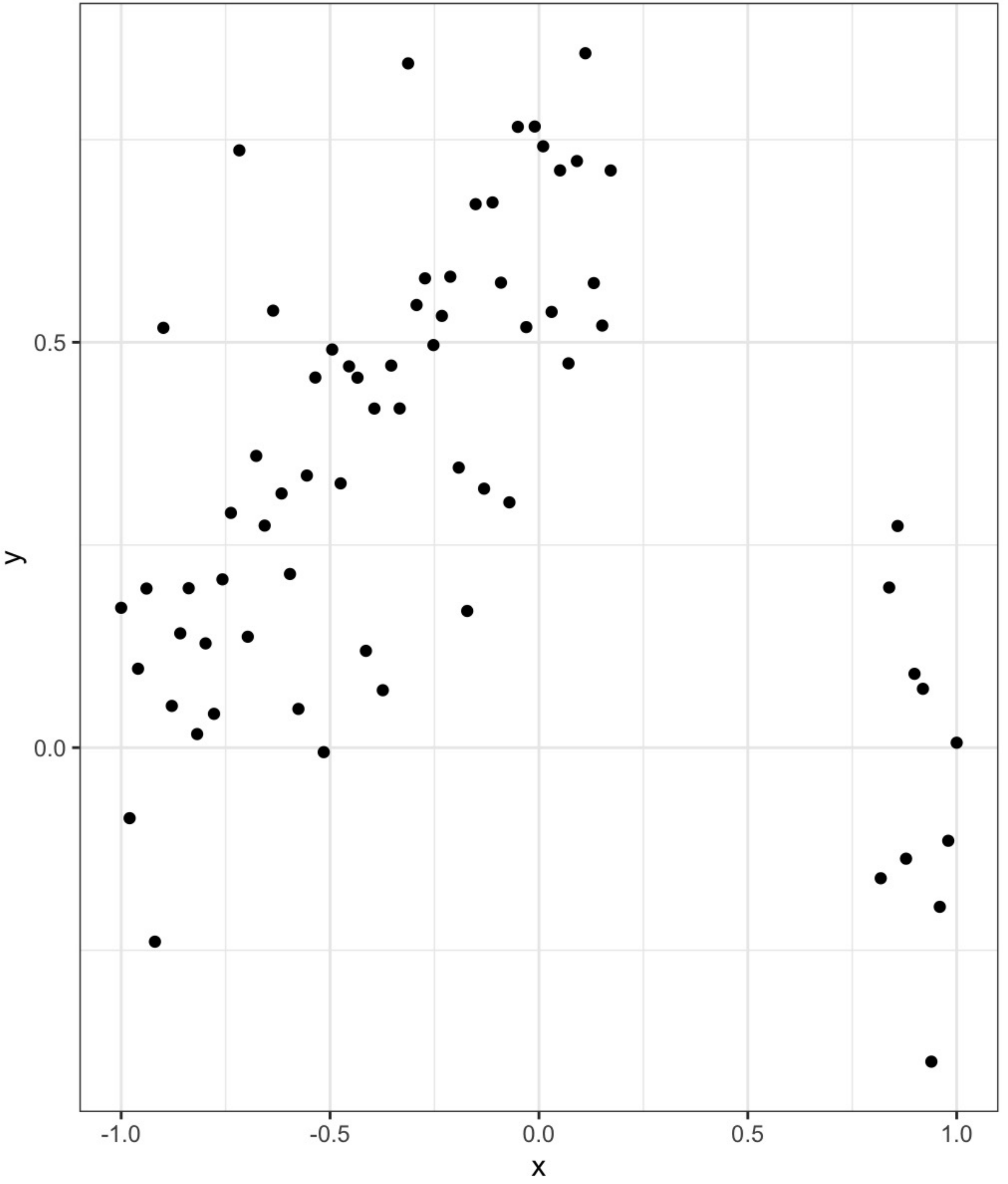
Jamie M. Kass

Postdoctoral Scholar

Okinawa Institute of Science and Technology Graduate University (OIST)

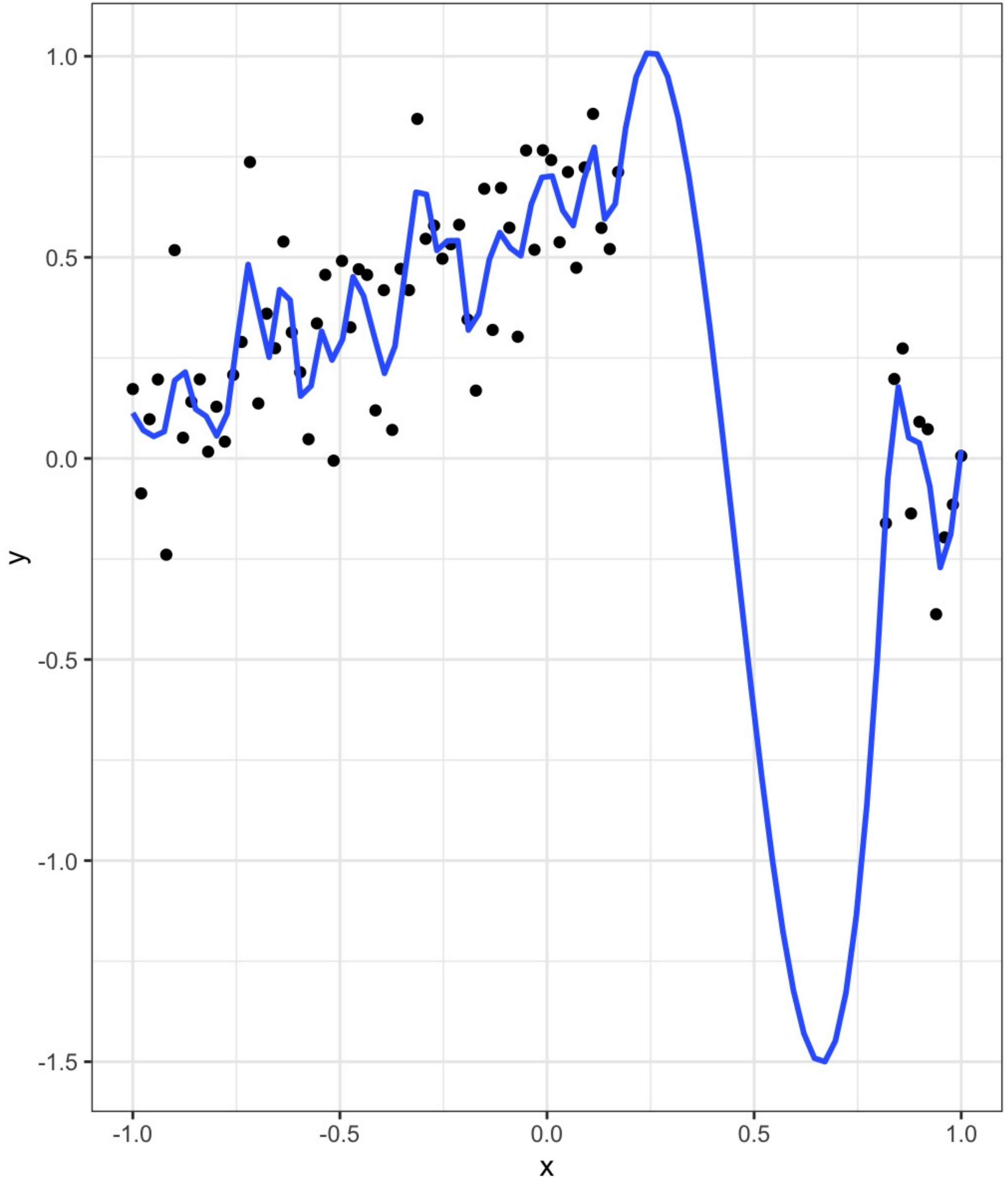
Model complexity

- data is messy in varying degrees



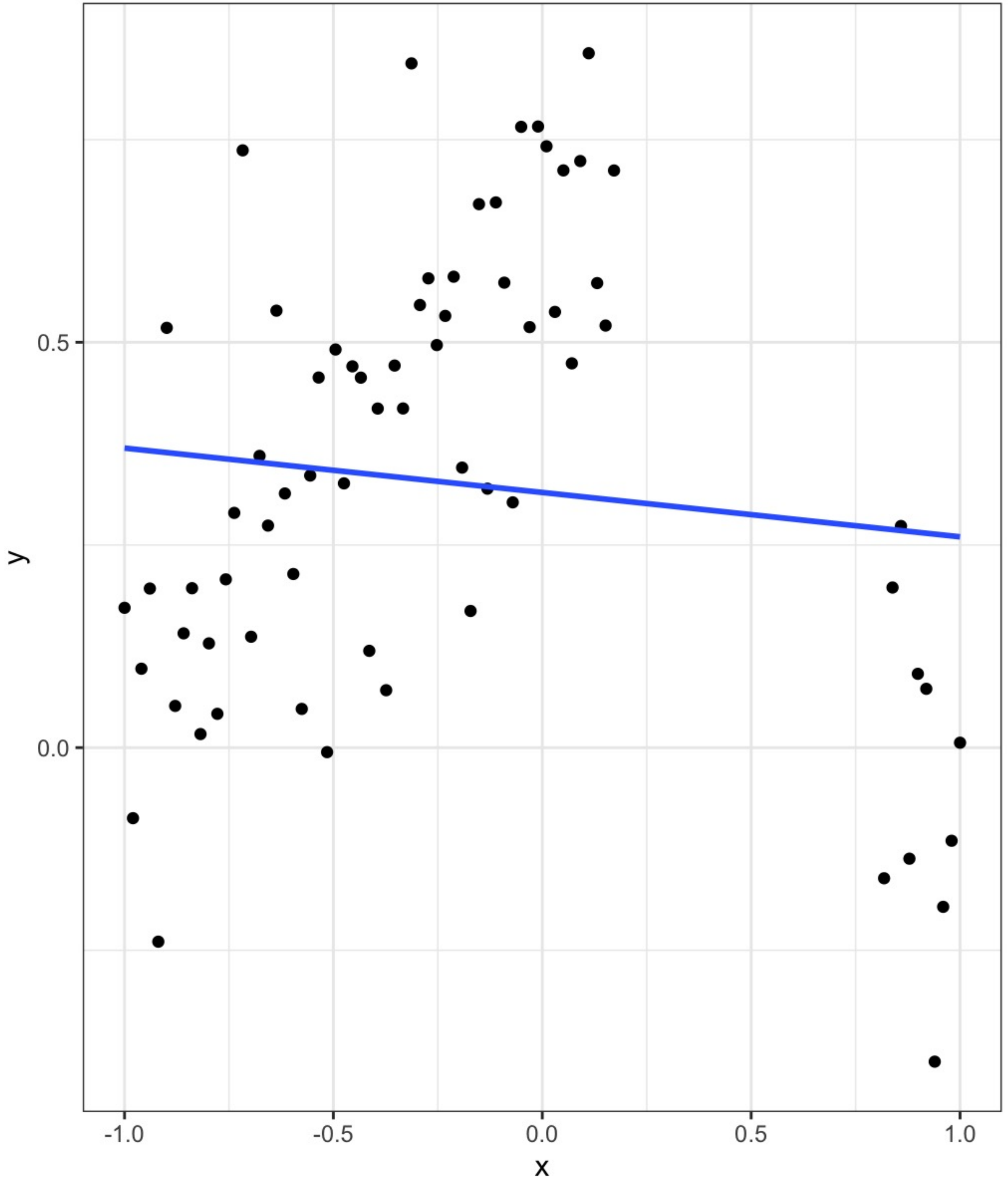
Model complexity

- data is messy in varying degrees
- models that are too complex can overpredict data



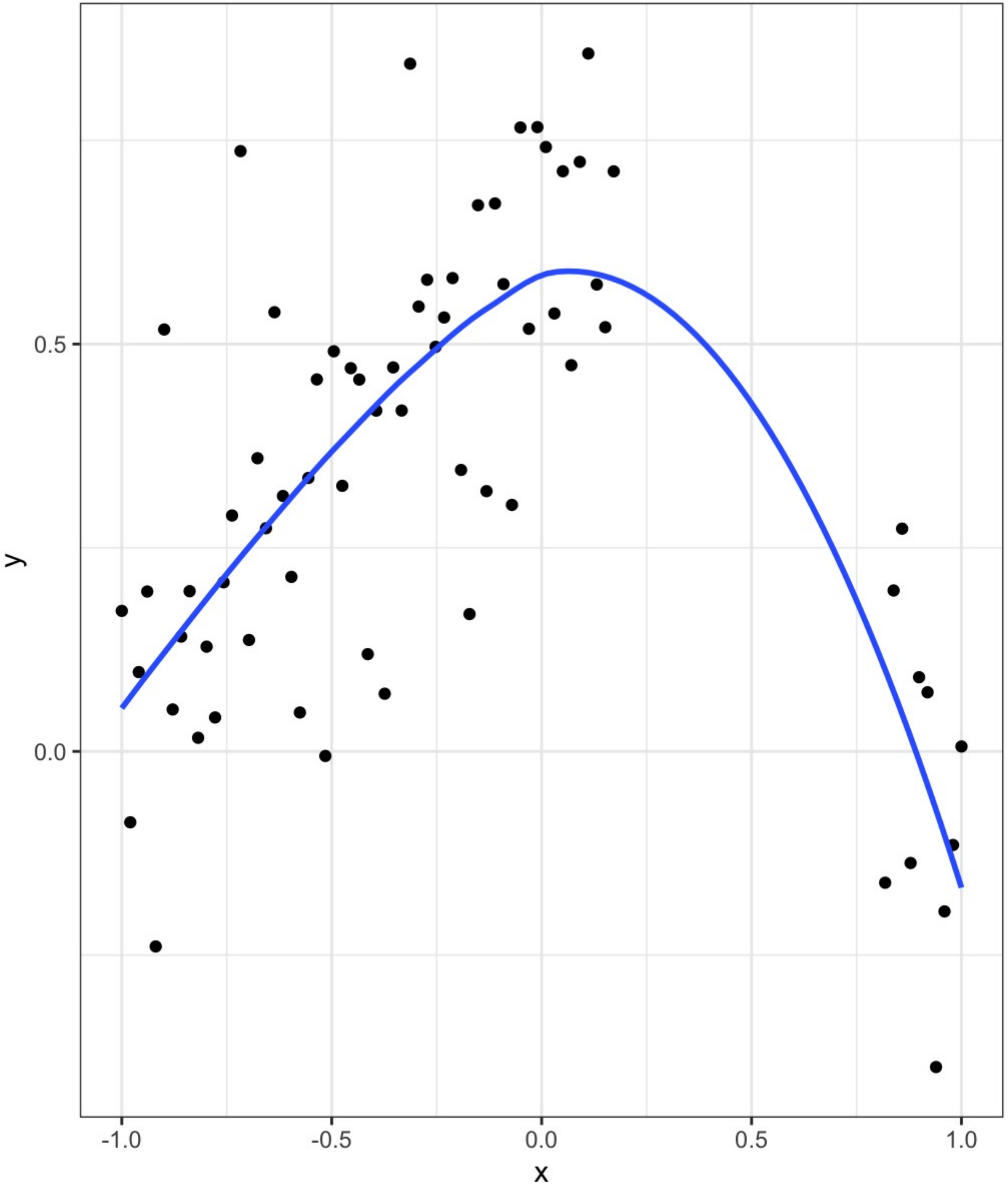
Model complexity

- data is messy in varying degrees
- models that are too complex can overpredict data
- models that are too simple can underpredict data



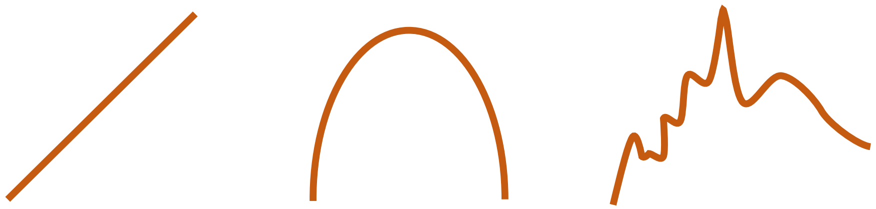
Model complexity

- data is messy in varying degrees
- models that are too complex can overpredict data
- models that are too simple can underpredict data
- we need a model in-between



What creates model complexity?

- the number of predictor variables used to fit the model
- the shapes of the modeled responses



- the presence of variable interactions

r = a + b1 * b2

How do we tell how well the model fits?

- model evaluation: measures of model performance on different data sets
- many metrics exist, and it can get confusing
- interpreting the results of model evaluation is also not straightforward
- key questions: is the model overfit (too complex) or underfit (too simple)?

How do we control complexity?

- exhaustive model selection for standard regression models

- machine learning algorithms have tuning parameters to penalize complexity

- Examples are Maxent, random forest, boosted regression trees, neural networks, lasso regression

What does a model evaluation tell us?

- model performance on the data used to build the model

- model performance on new data

- ecological realism for:
  - relationships with predictor variables
  - spatial predictions

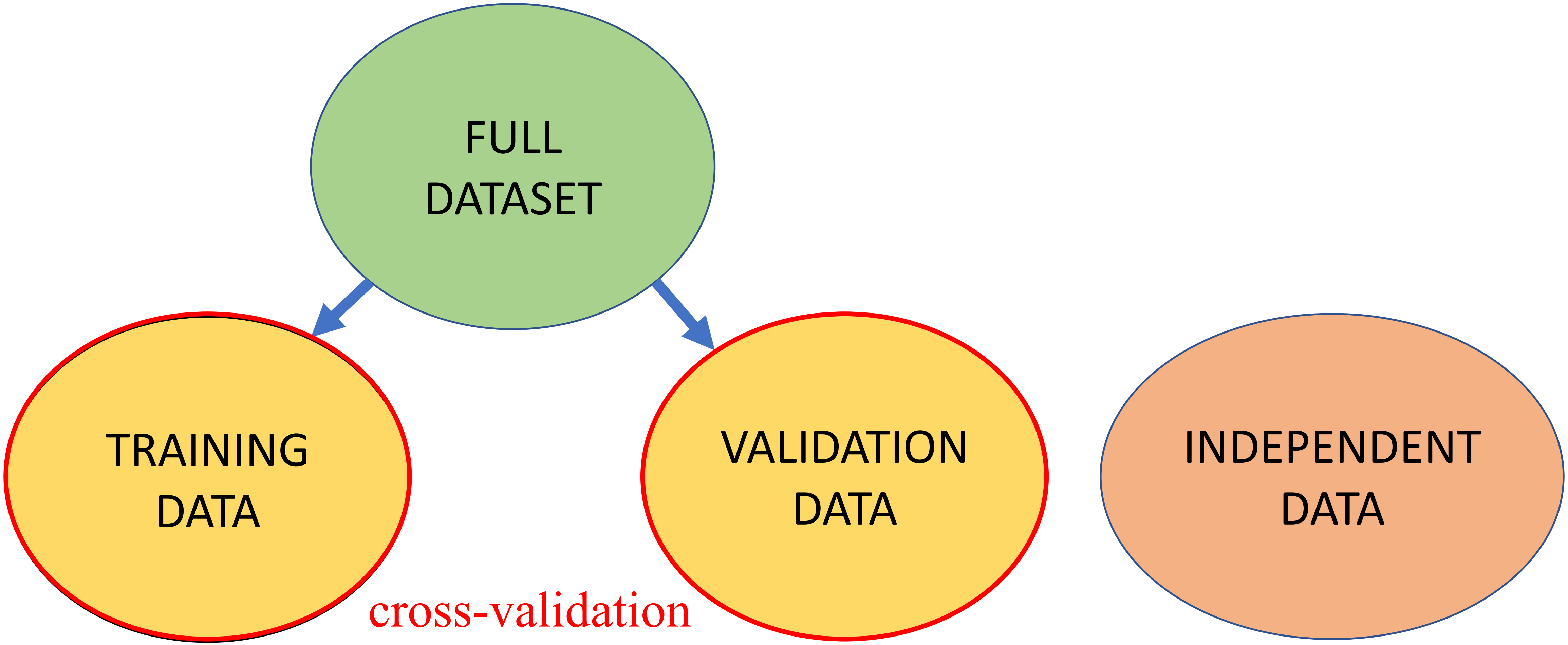
Popular SDM evaluation metrics

metric	threshold	range	high or low?	CV	caveats	R packages
AUC	independent	0 – 1	+	yes	cannot use to compare diff spp or extents	*dismo, ENMeval, SDMtune, ROCR*
pROC	independent	AUC ratio	+	yes	user-set acceptable level of omission error (*e* = 100% for AUC)	*pROC, kuenm, ntbox*
Continuous Boyce Index	independent	-1 – 1	+	yes		*ecospat, ENMeval*
omission rate	dependent	0 – 1	-	yes		*dismo, ENMeval*
TSS	dependent	-1 – 1	+	yes	cannot use to compare diff spp or extents	*SDMtune*
kappa	dependent	-1 – 1	+	yes	cannot use to compare diff spp or extents	*dismo*
AICc	independent	relative	-	no	cannot evaluate transferability	*ENMeval, SDMtune*

Model evaluation strategy

- each model should be able to accurately predict the input data
- but can each model also accurately predict new data?
- if we have independent data, we can evaluate each model on it
- if not, we can evaluate each model on subsets of itself

Data for modeling: terminology

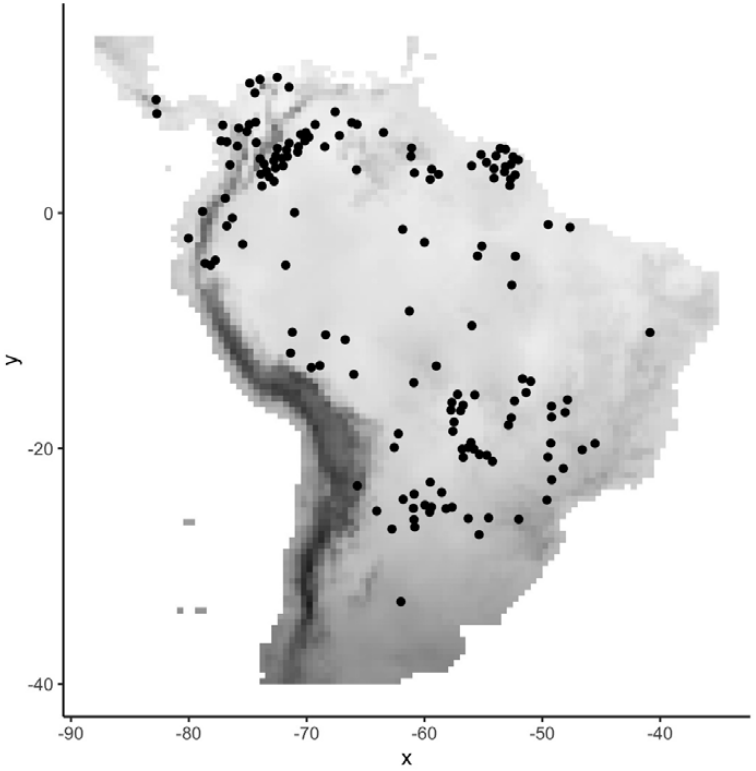


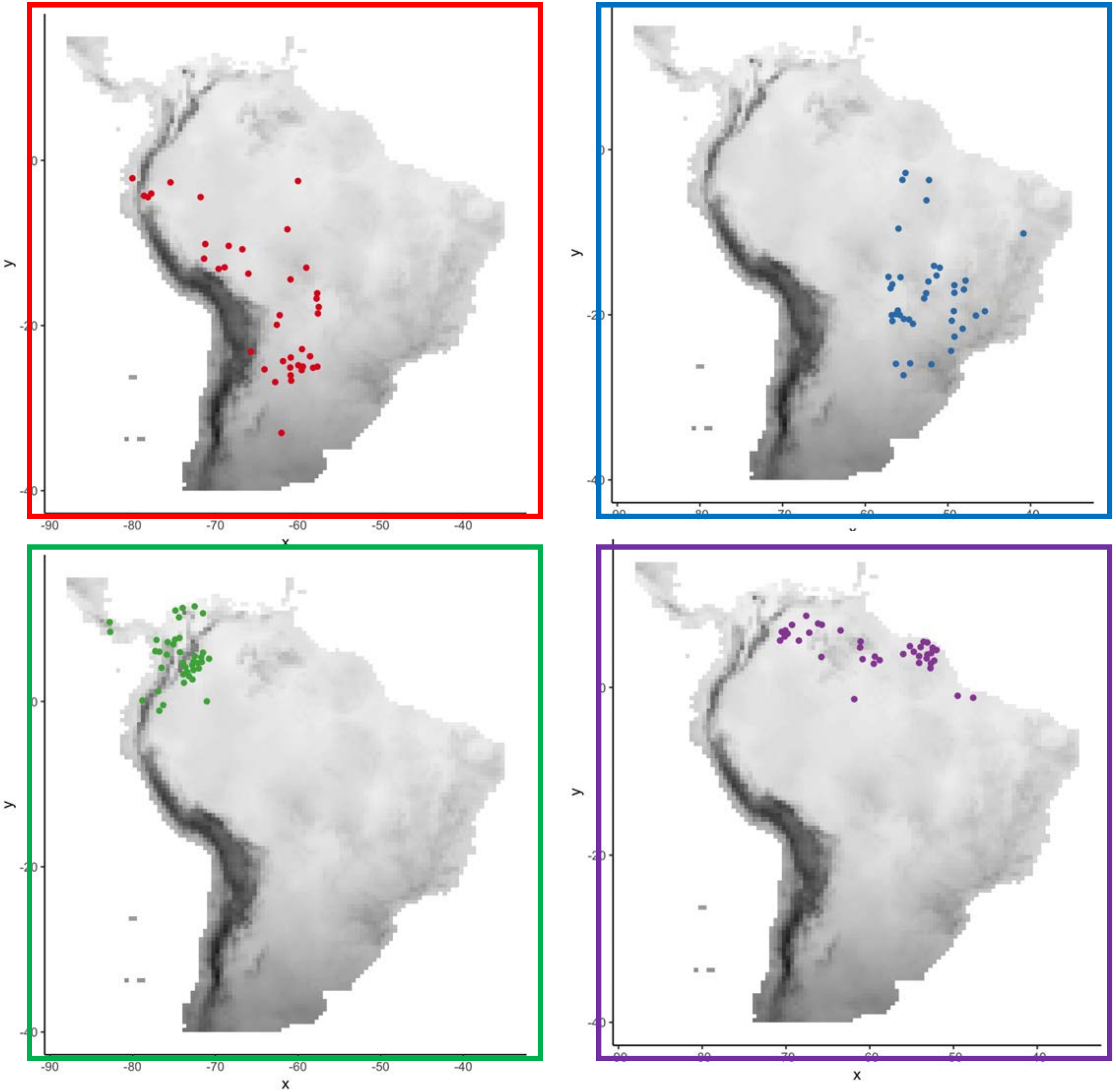
used to fit the model

used to evaluate the model

Cross validation

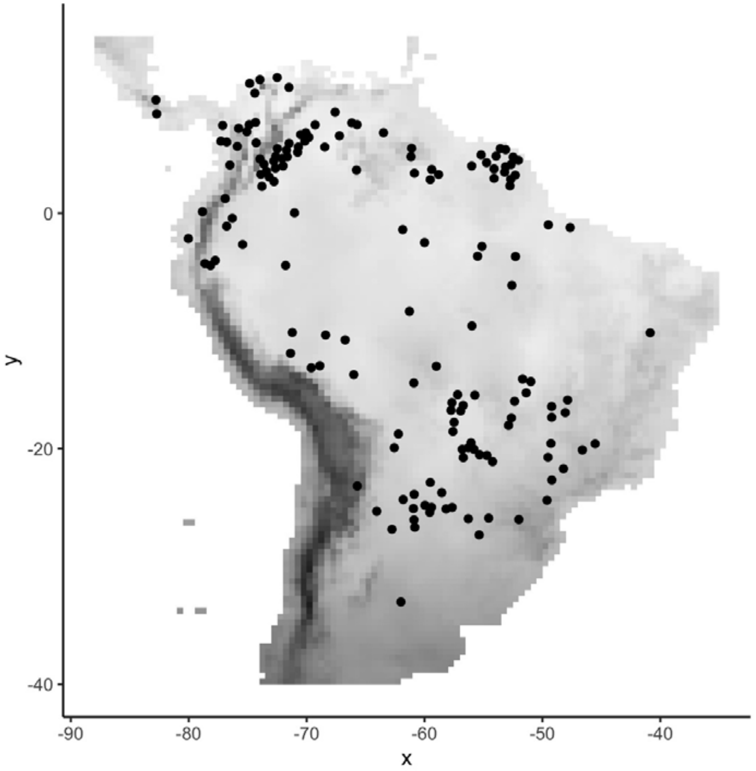
1. Split the data into *k* groups (a.k.a. subsets, partitions)

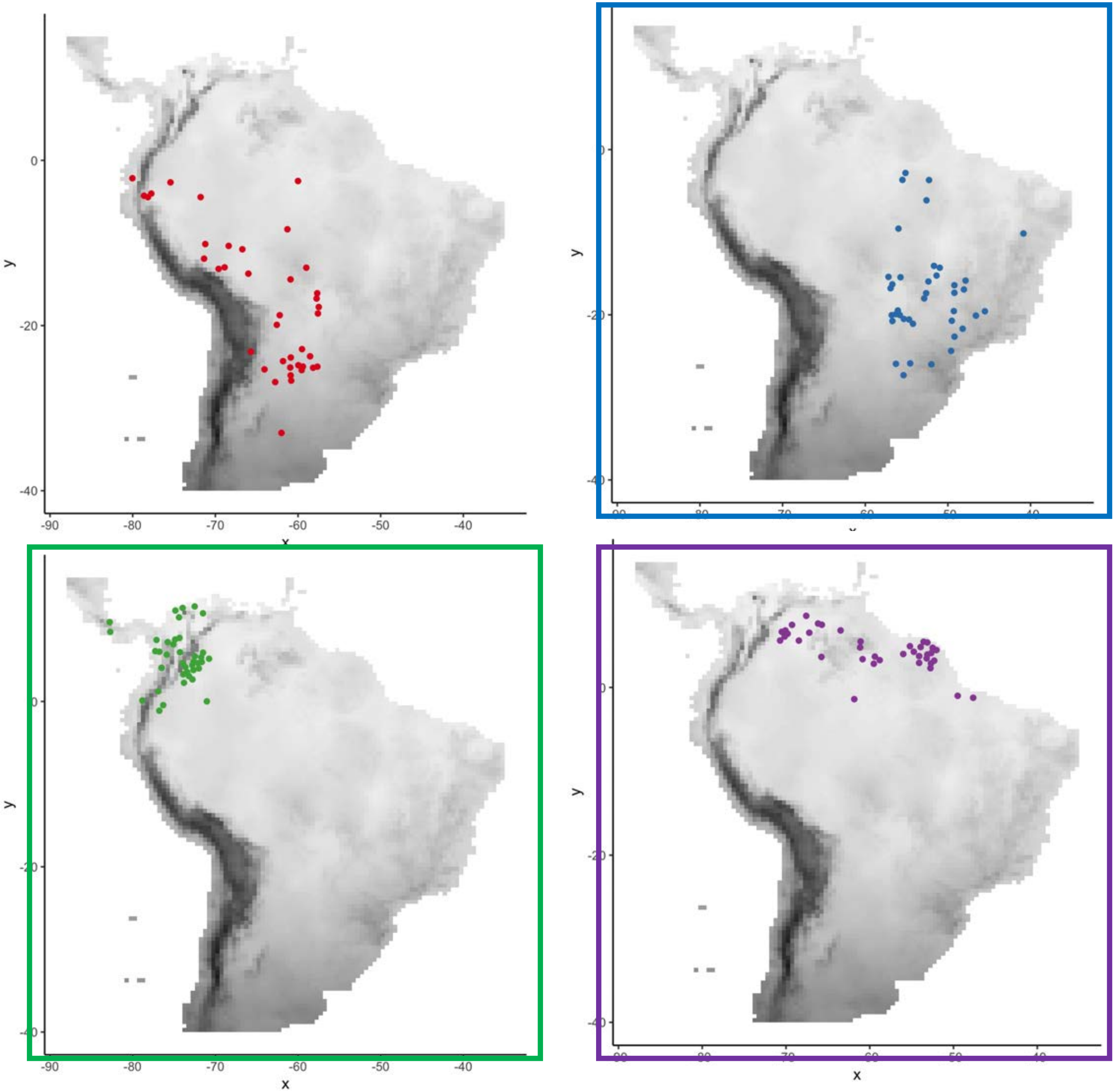




Cross validation

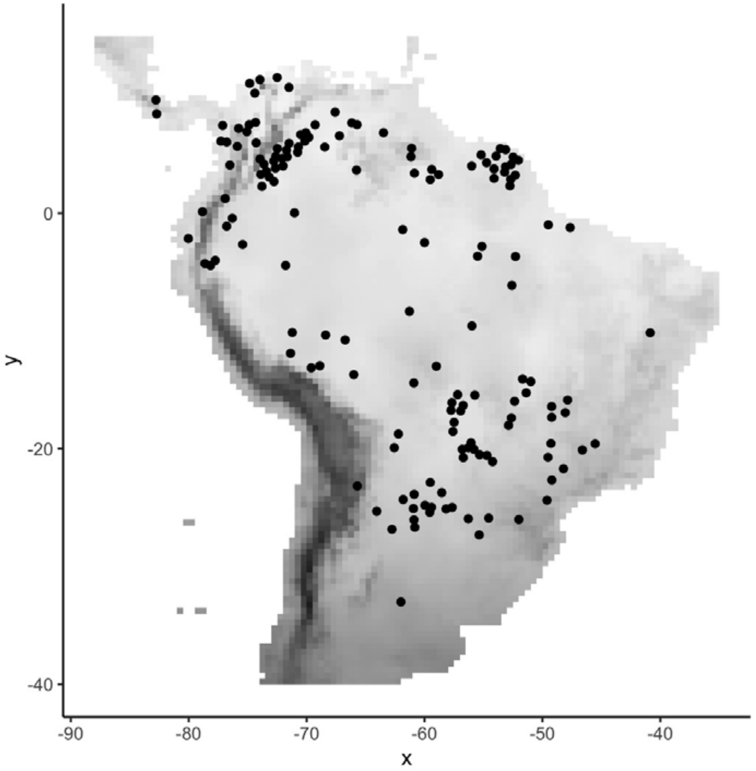
2. Train the model on *k* - 1 subsets

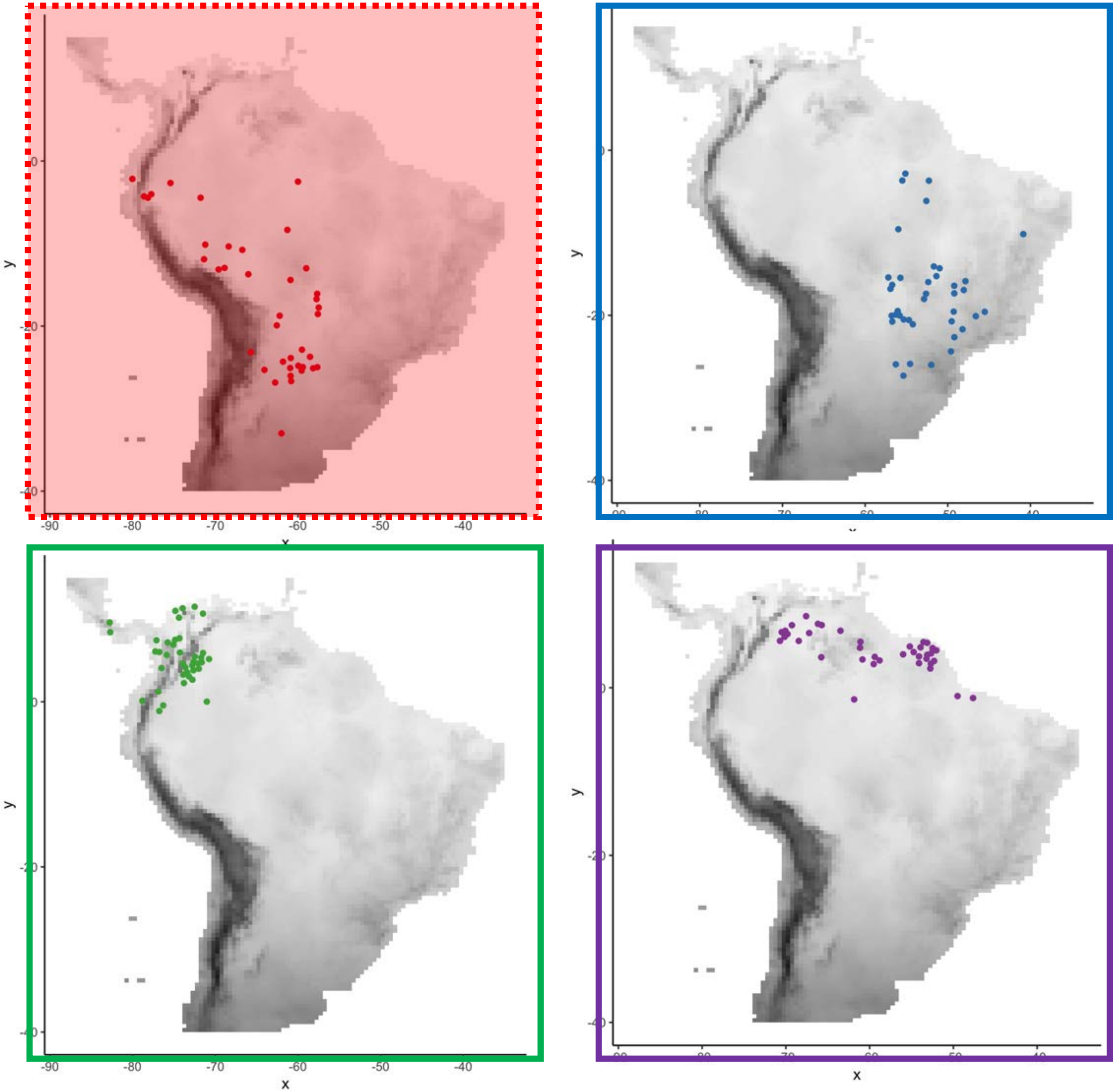




Cross validation

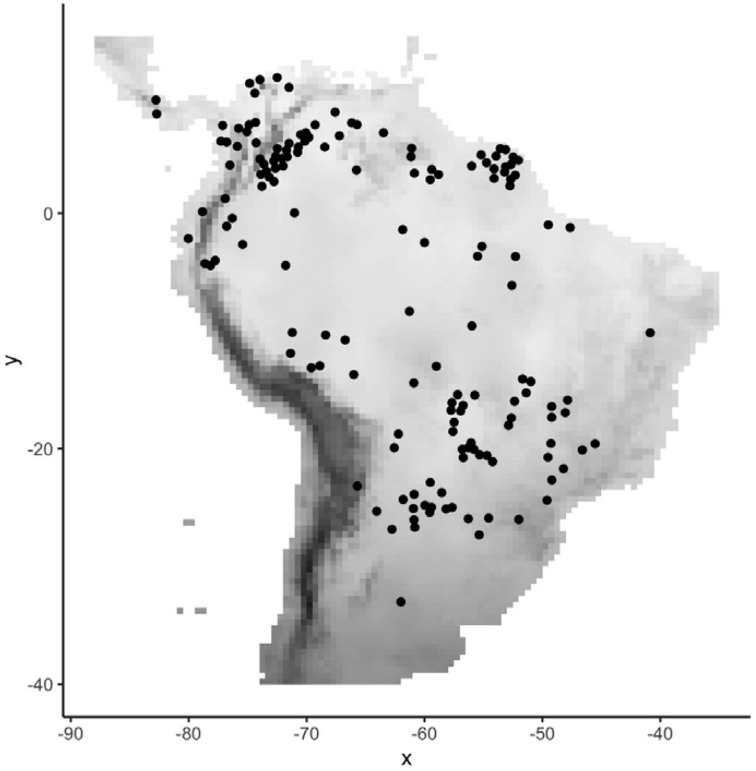
3. Evaluate the model on subset *k* (calculate an evaluation statistic)

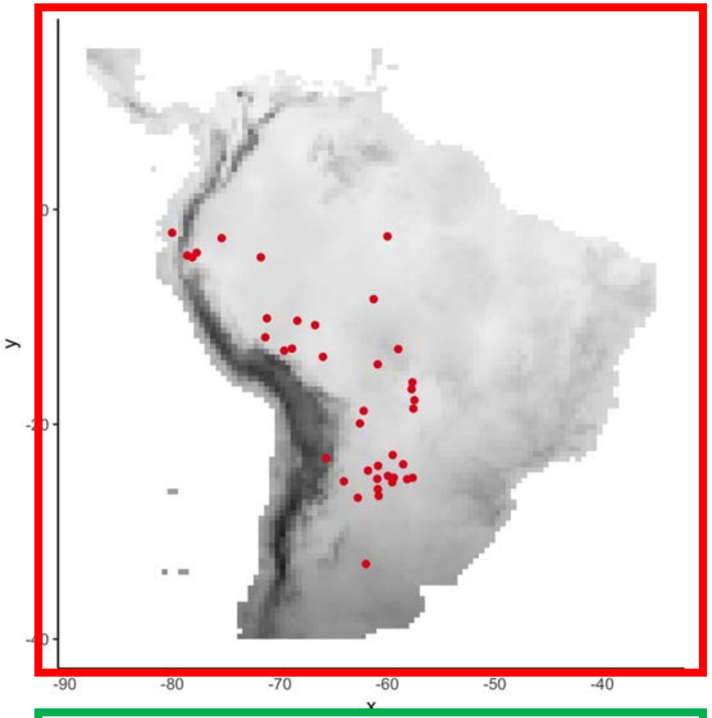


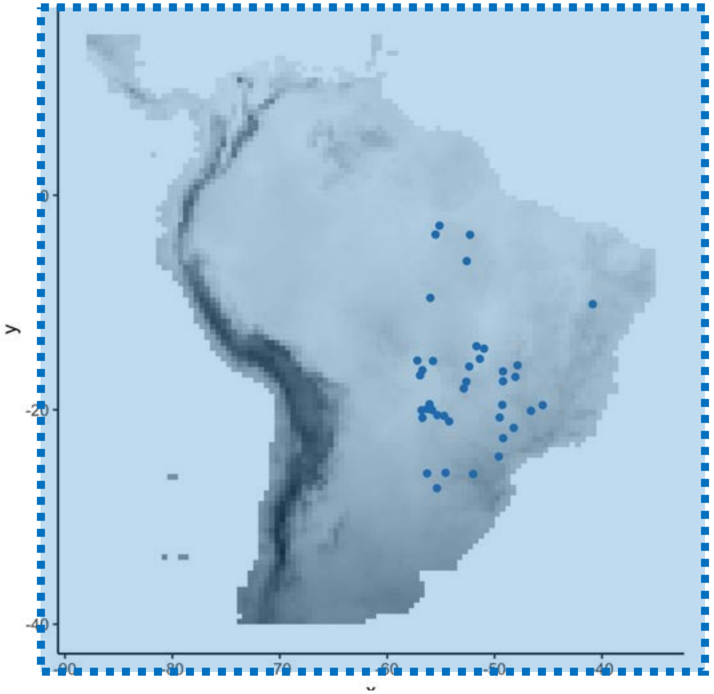


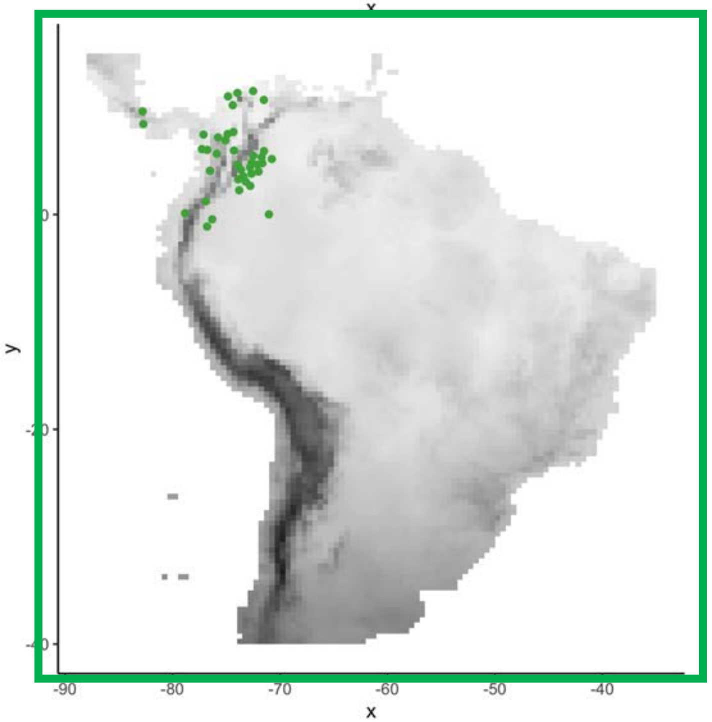
Cross validation

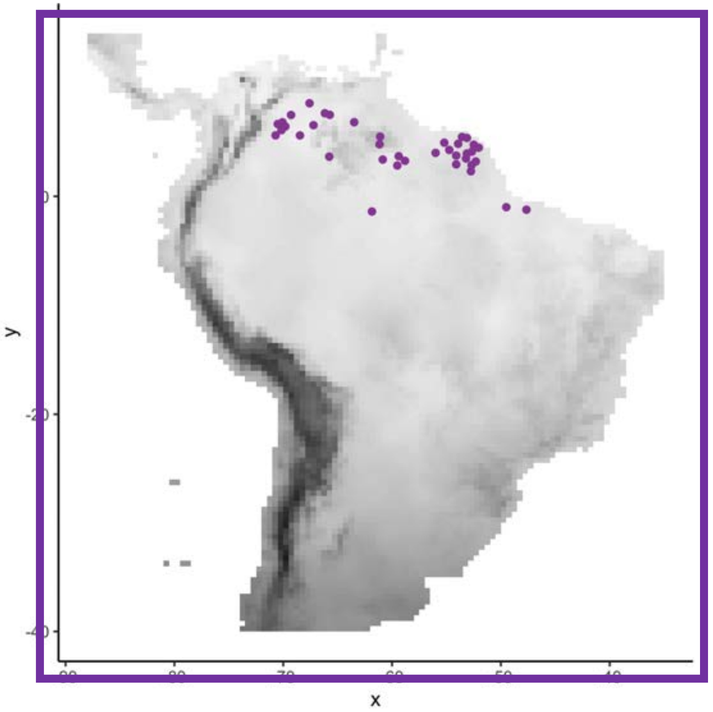
4. Repeat for all *k*





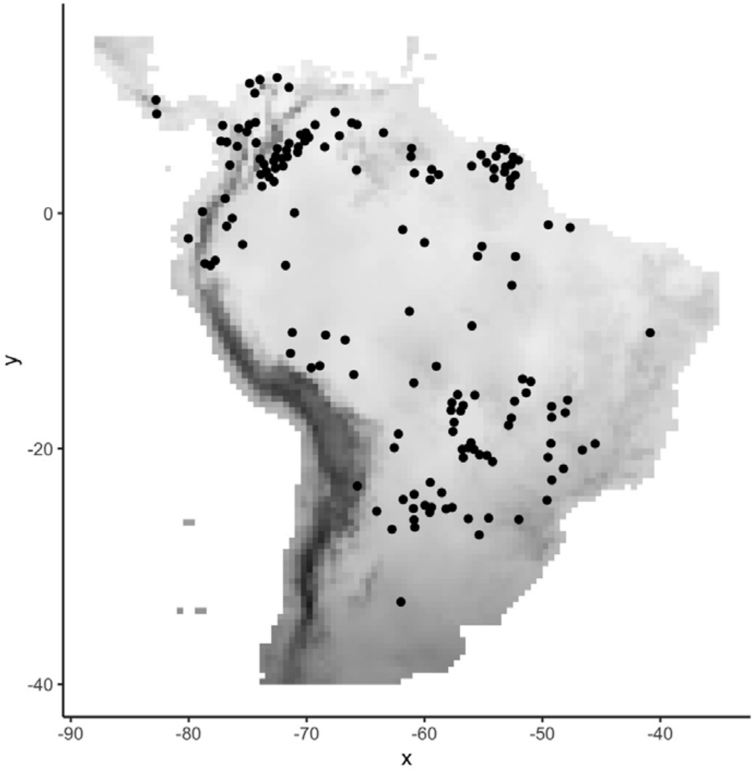


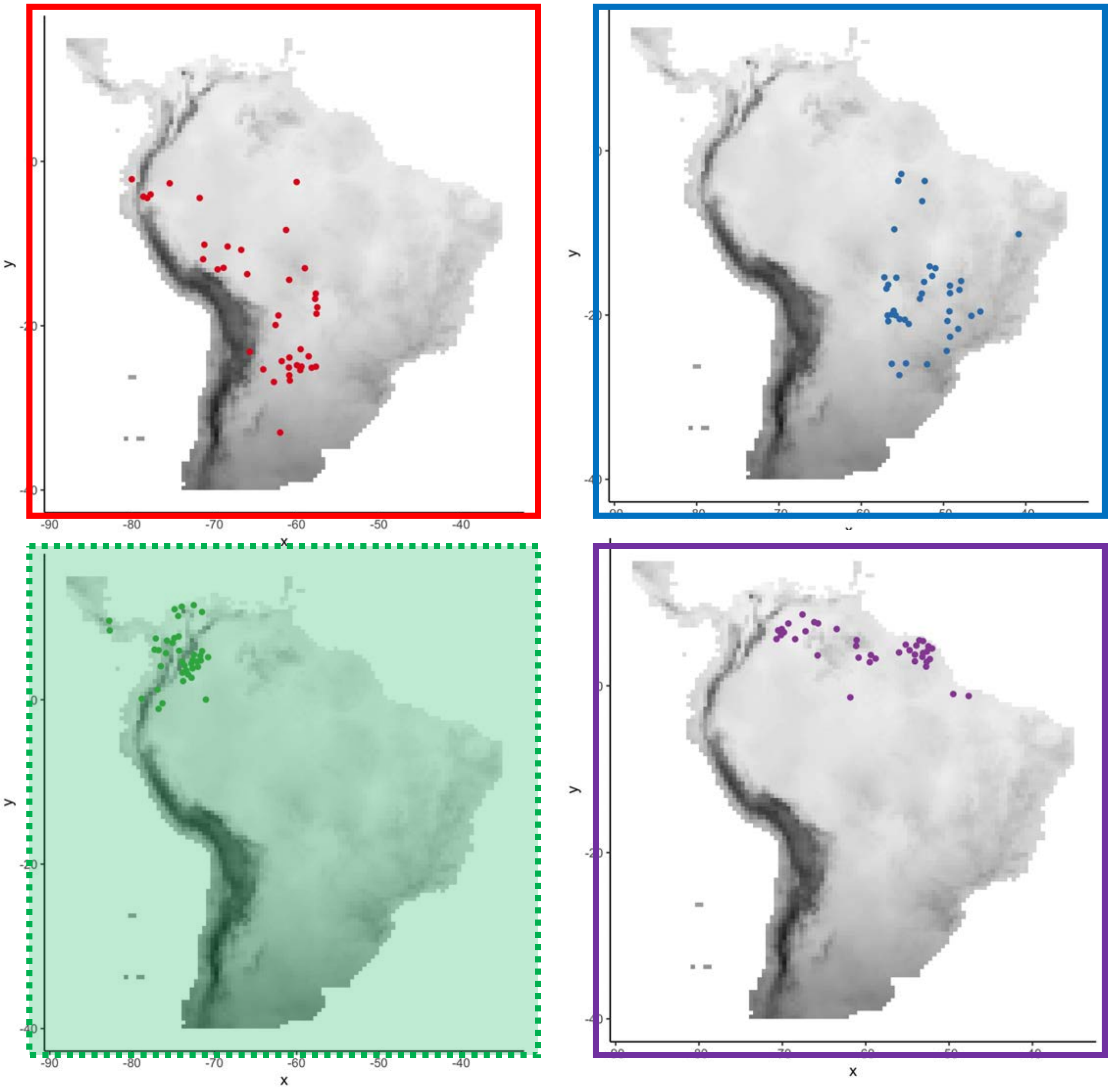




Cross validation

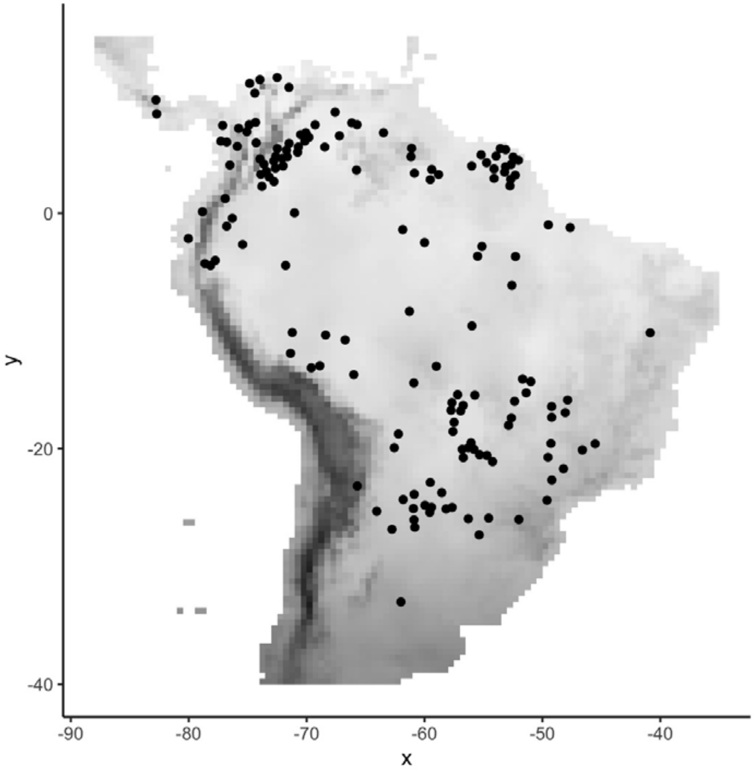
4. Repeat for all *k*

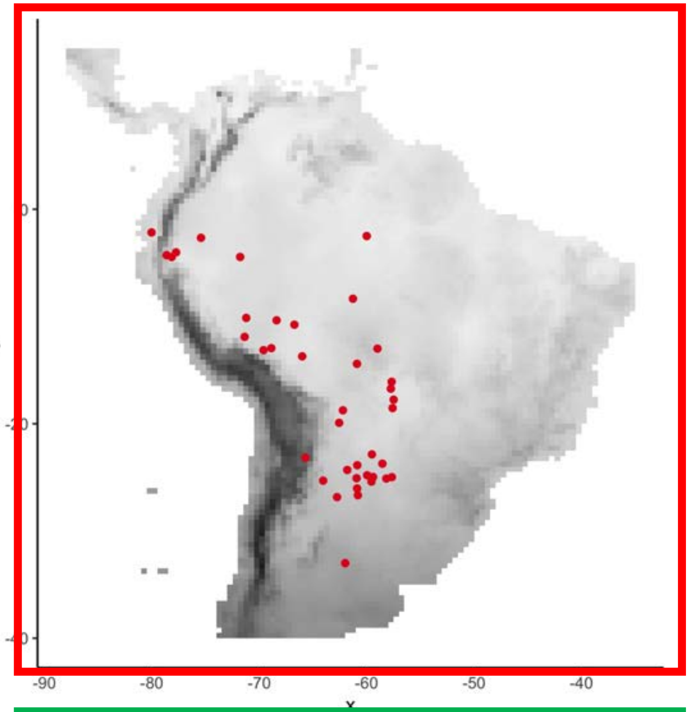


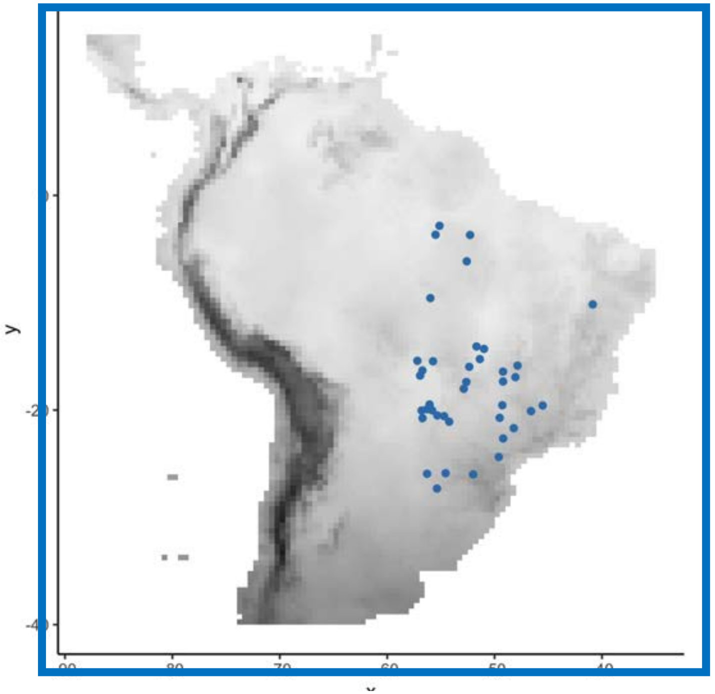


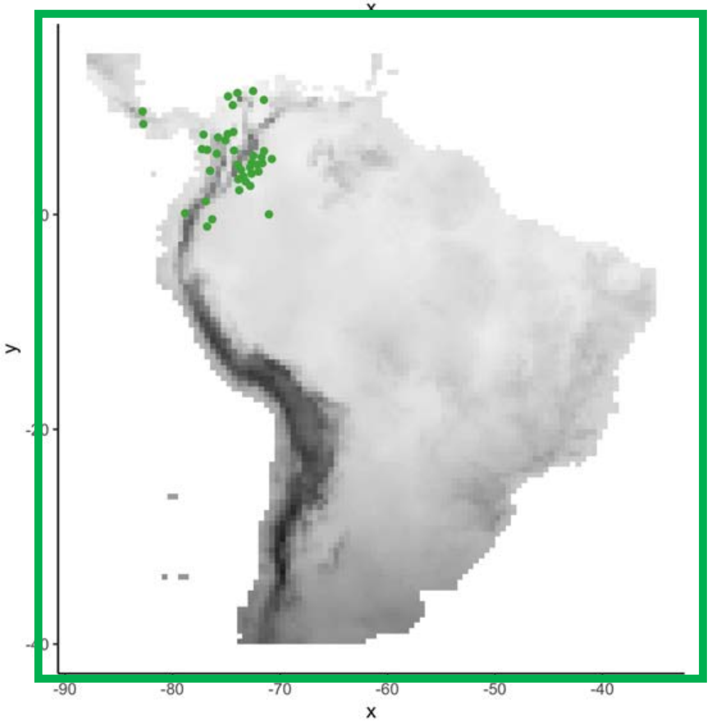
Cross validation

4. Repeat for all k





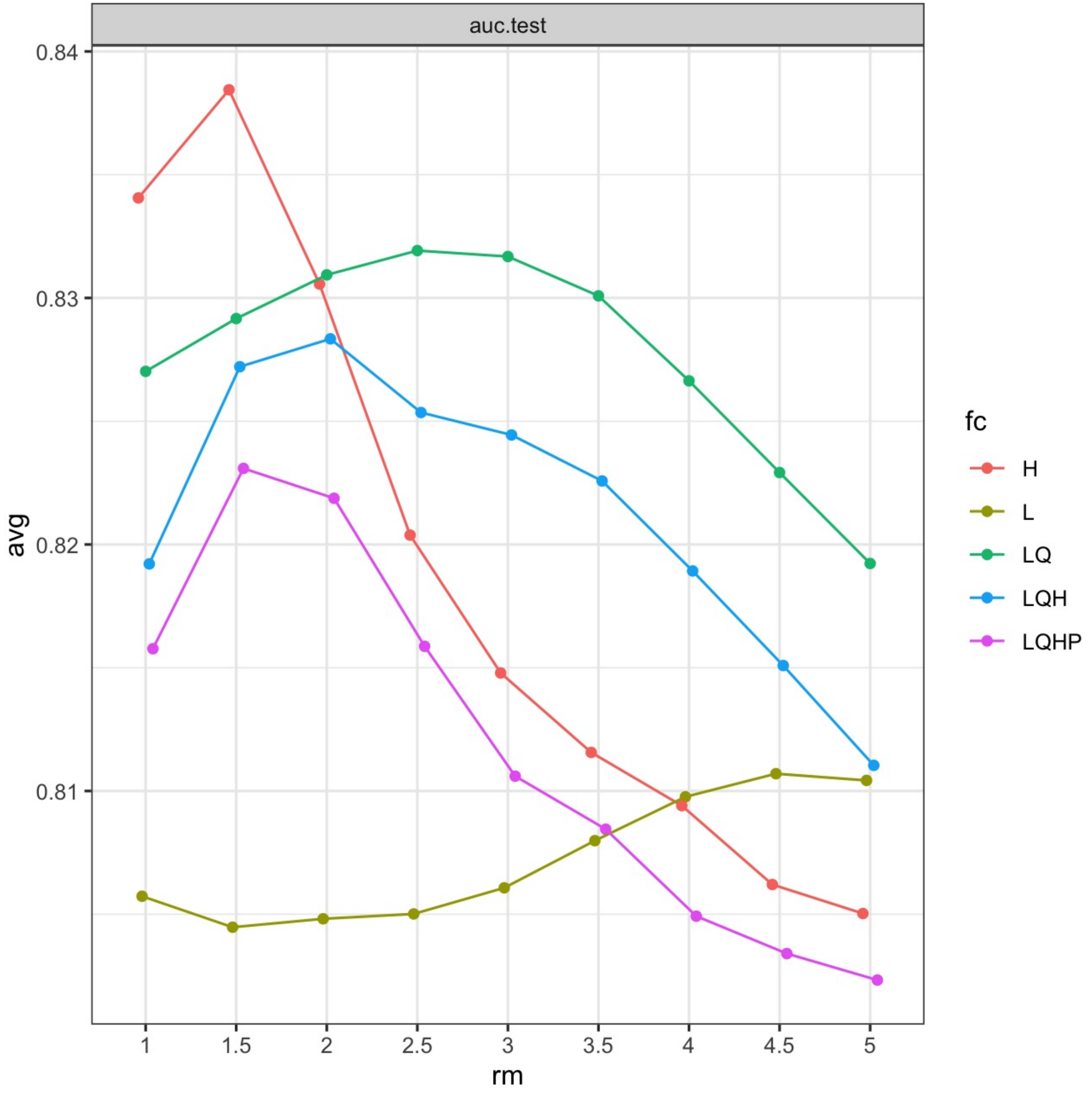




Cross validation

5. Take summary statistics (mean, sd, etc.) on the subset evaluations

Finally, compare the model evaluations to determine the parameter settings for optimal complexity

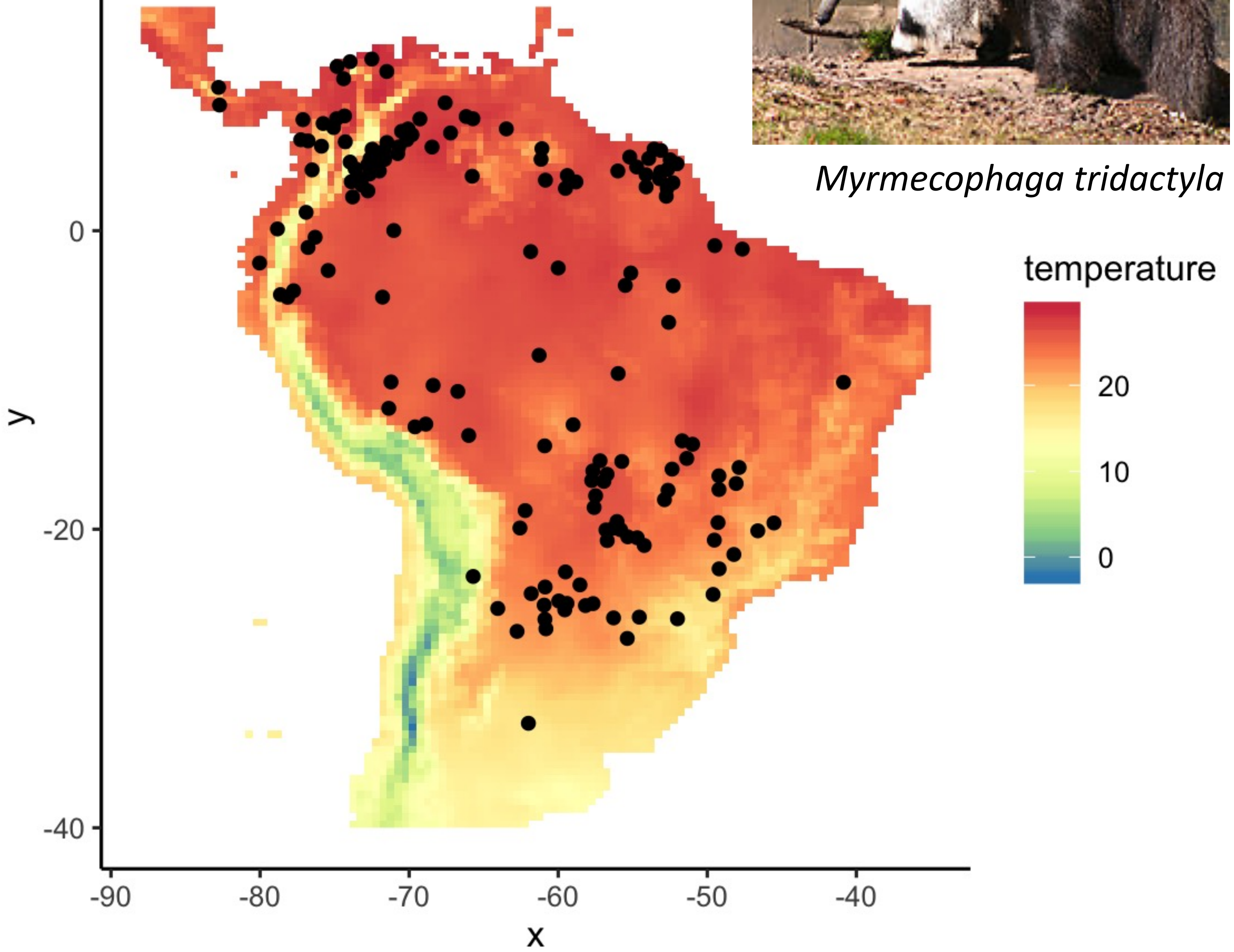


Model tuning

- we can implement cross validation on a suite of models with varying complexity
- each model will have associated performance metrics
- we can then conduct model selection to choose an “optimal” model

Example: Giant Anteater

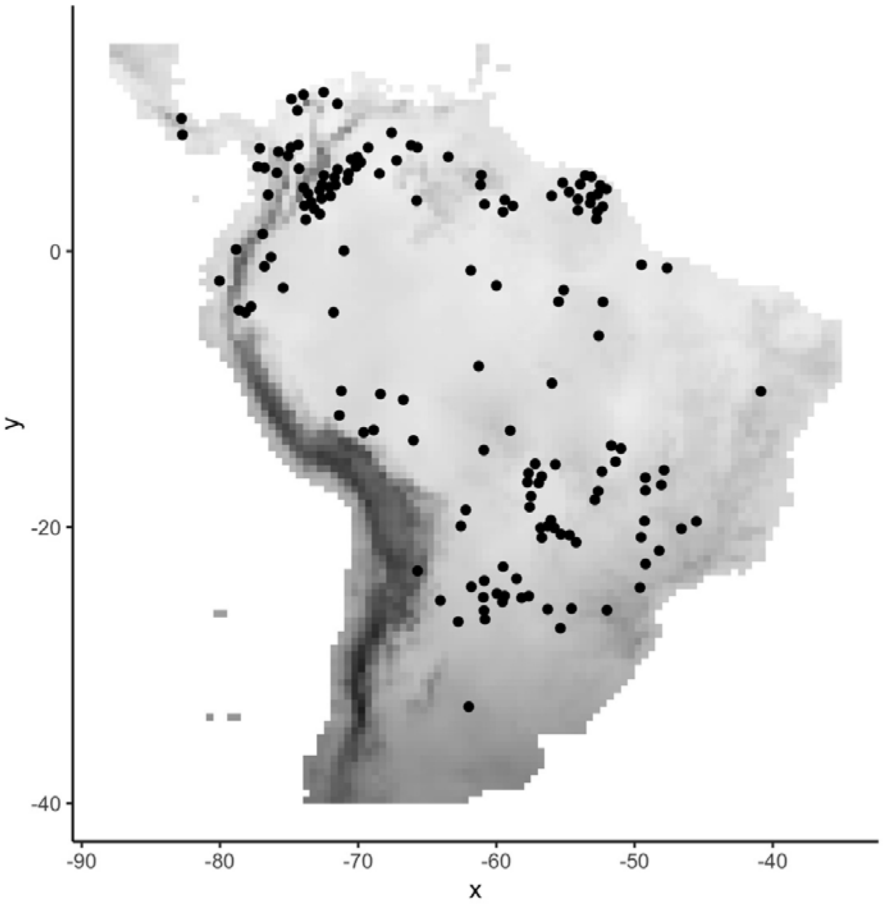
- downloaded from GBIF with R package spocc
- initially, n = 400
- after processing (geographic and spatial filtering), n = 155

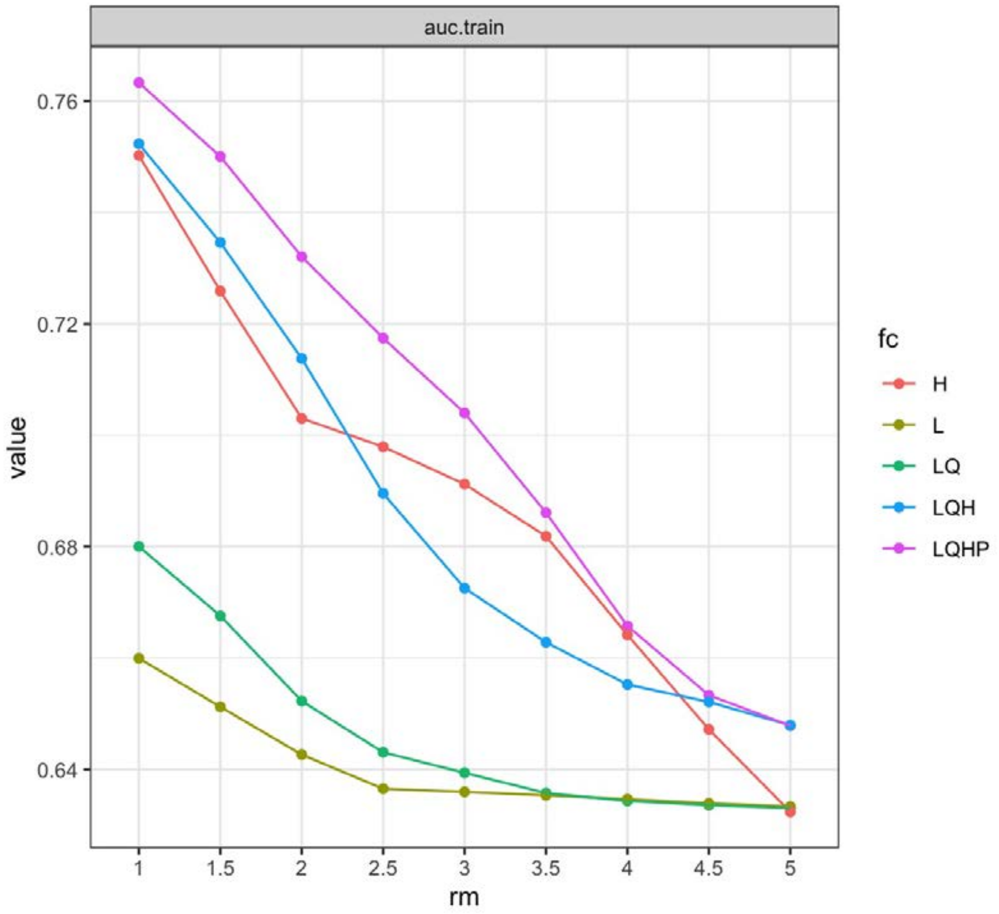


Myrmecophaga tridactyla

How to evaluate models when tuning?

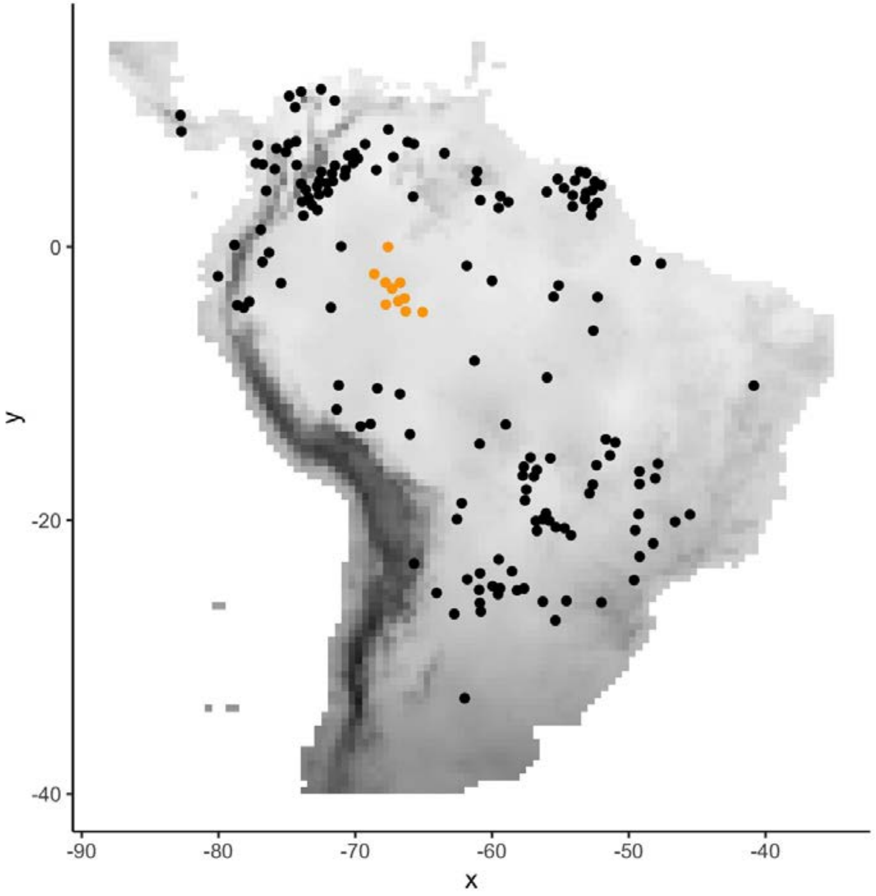
- we could ask how well each model predicts the input data (training data)

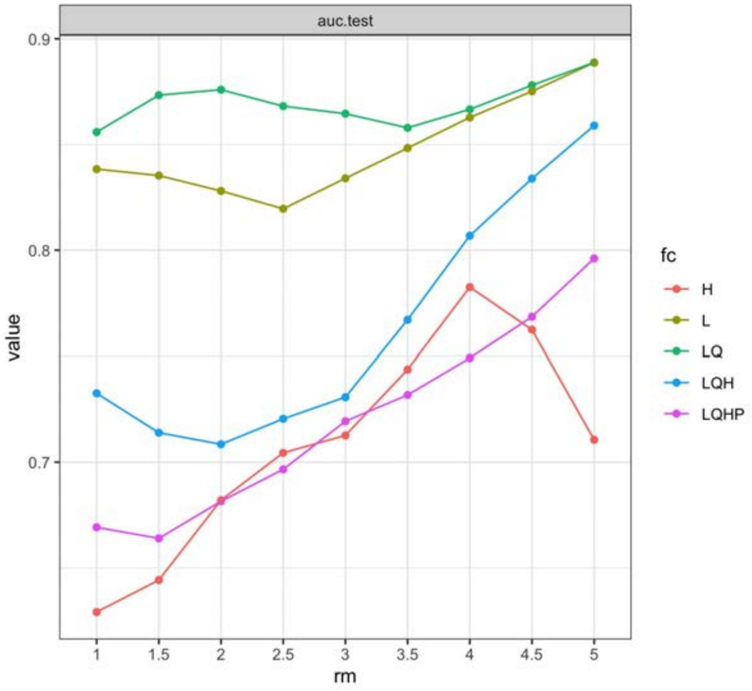


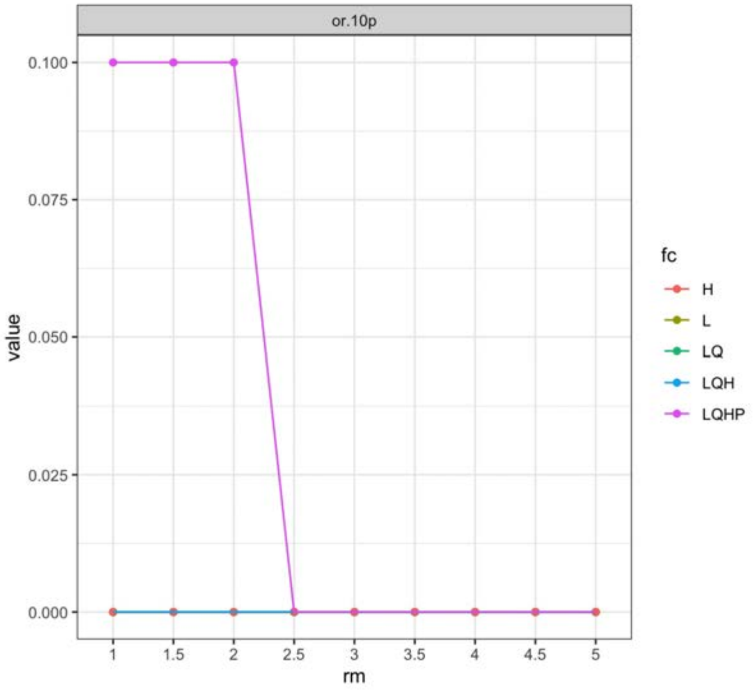


How to evaluate models when tuning?

- we could ask how well each model predicts independent data

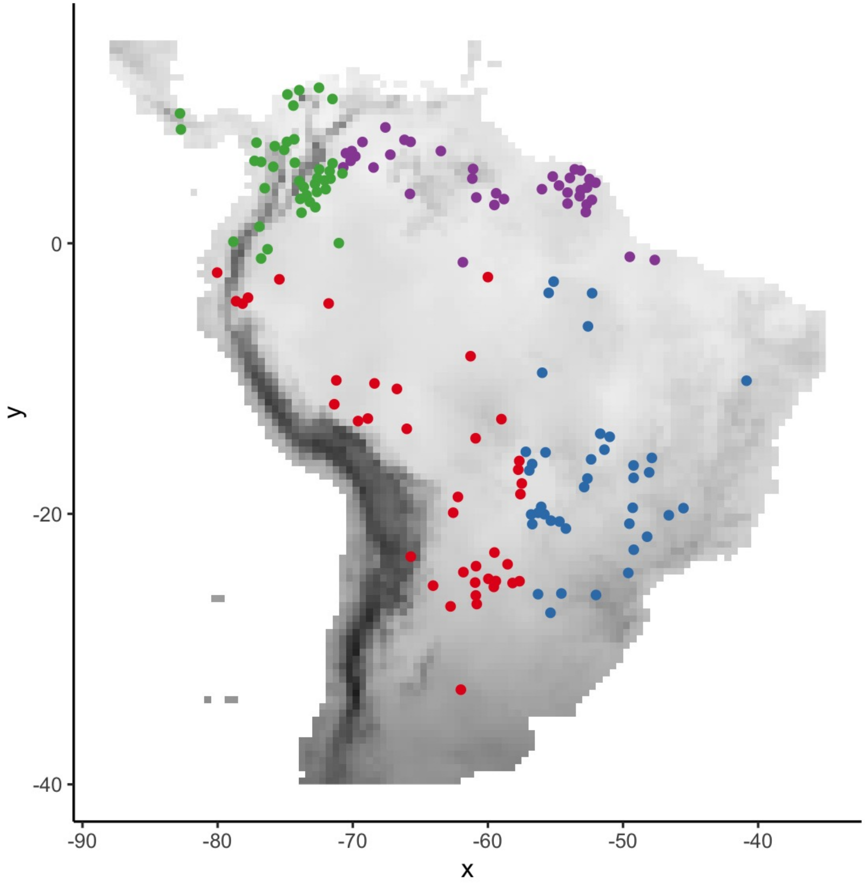


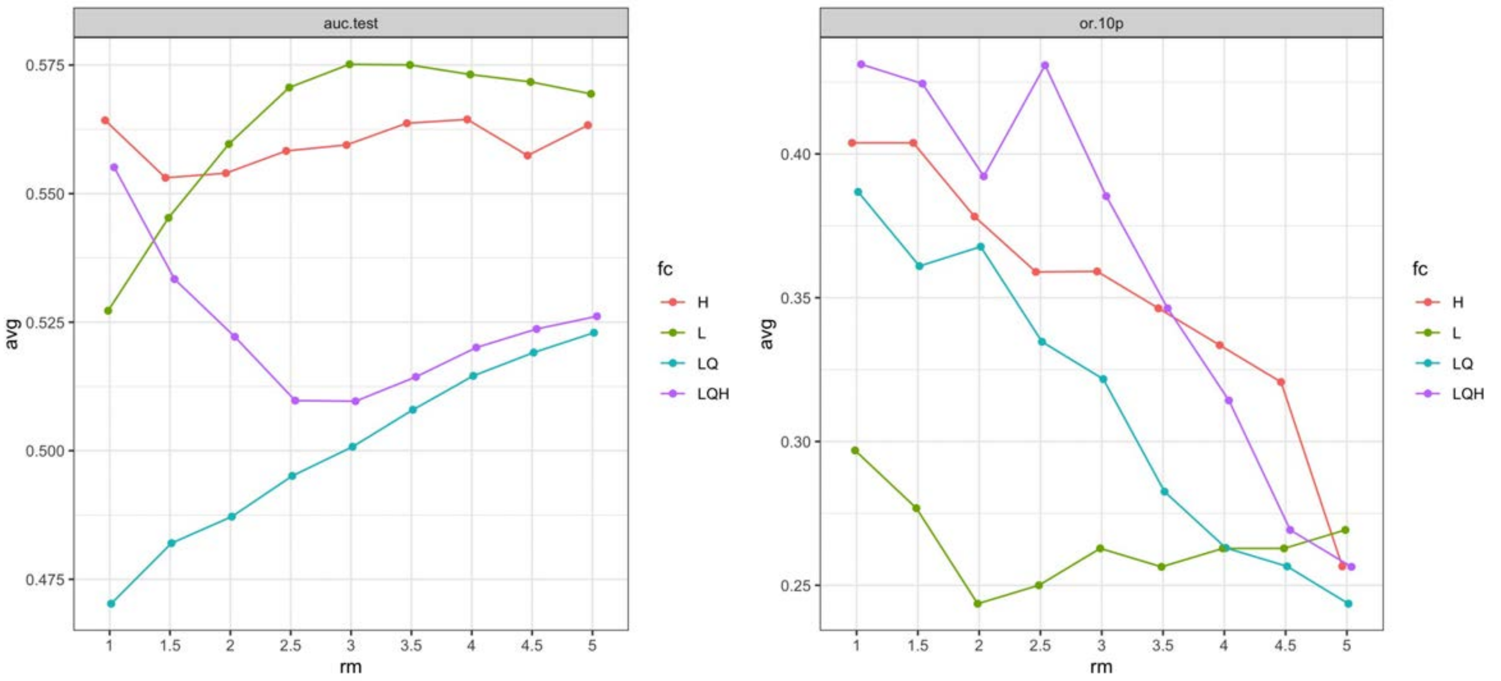




How to evaluate models when tuning?

- we could ask how well each model predicts holdout data on average (testing data)





Ideal data subset for cross validation

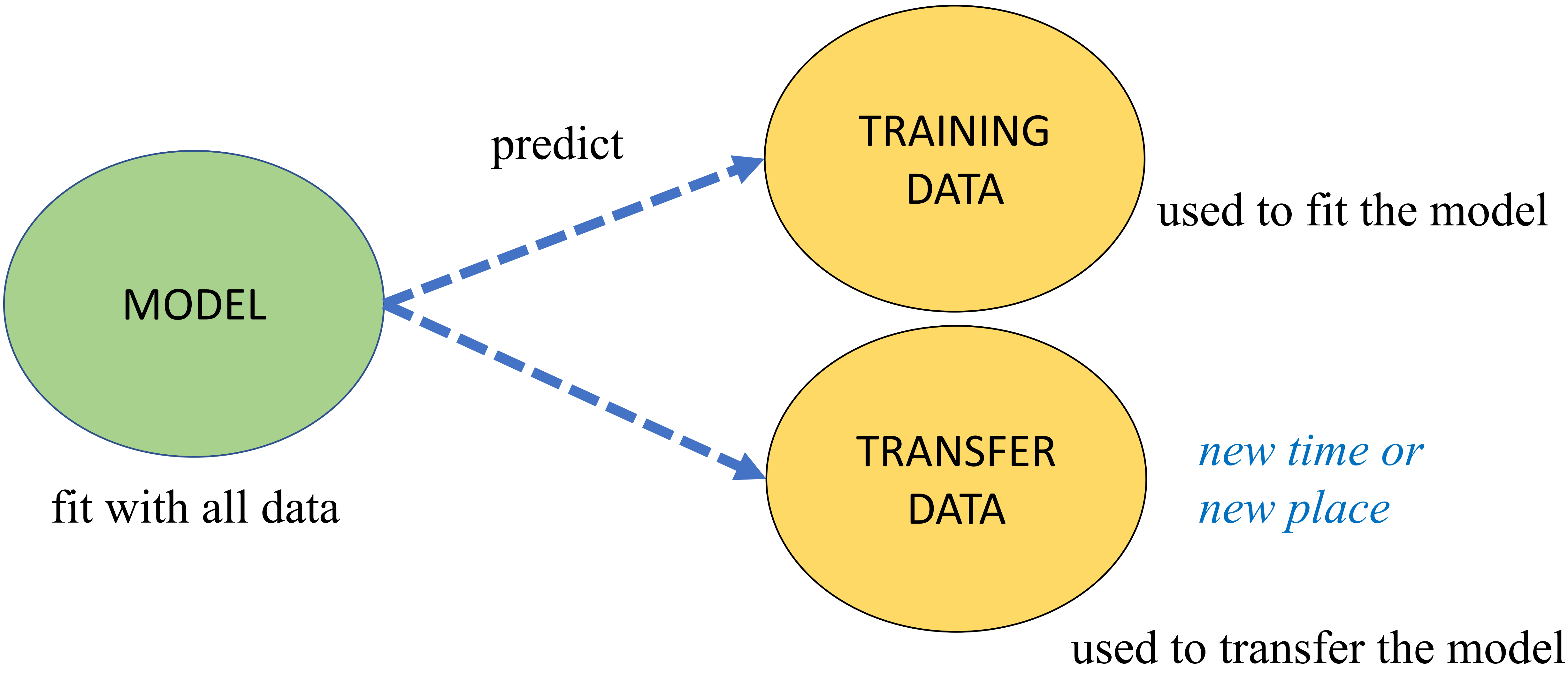
- even number of records across subsets
  - not always feasible when number of records is low

- even sampling across environment
  - not always feasible when records are absent from certain environments
  - not desirable when the goal is extrapolation

Purpose of cross validation evaluation

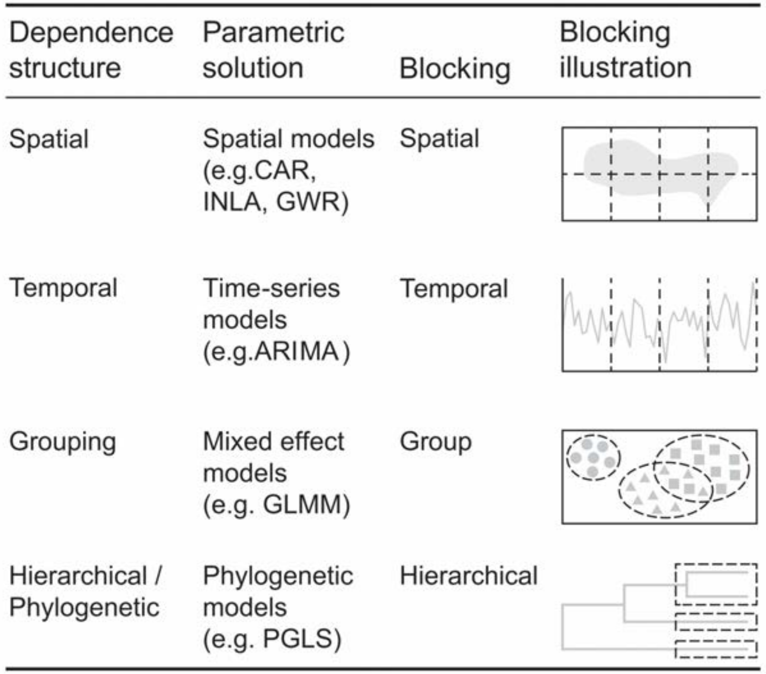
- ability to predict the conditions in your data (interpolation)
- ability to transfer to new conditions (extrapolation)
- need to ask yourself: what do you want your model to do?
- then subset your data to make the model evaluations rate this ability

Data for model transfer: terminology



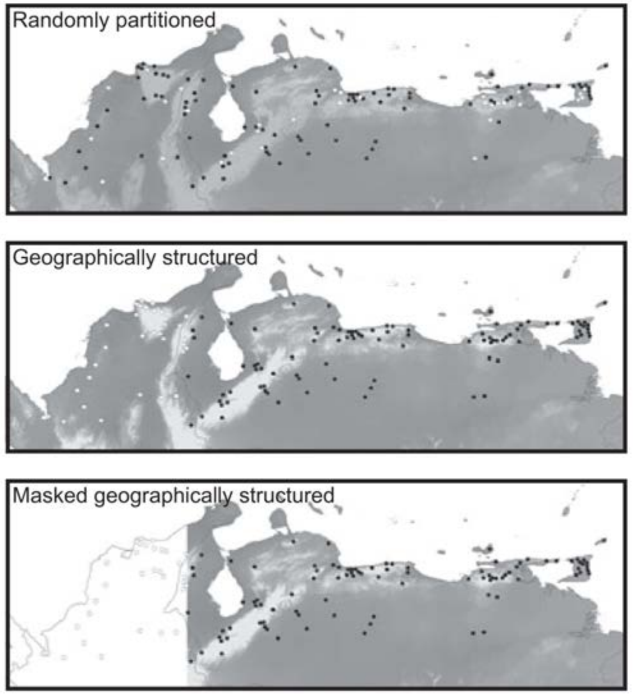
Block vs. non-block subsetting

- block subsetting: partitioning the data with some underlying structure
- usually results in lower performance than random CV
- leads to better evaluations of transferability
- with most block subsets, cross validation should include background data as well



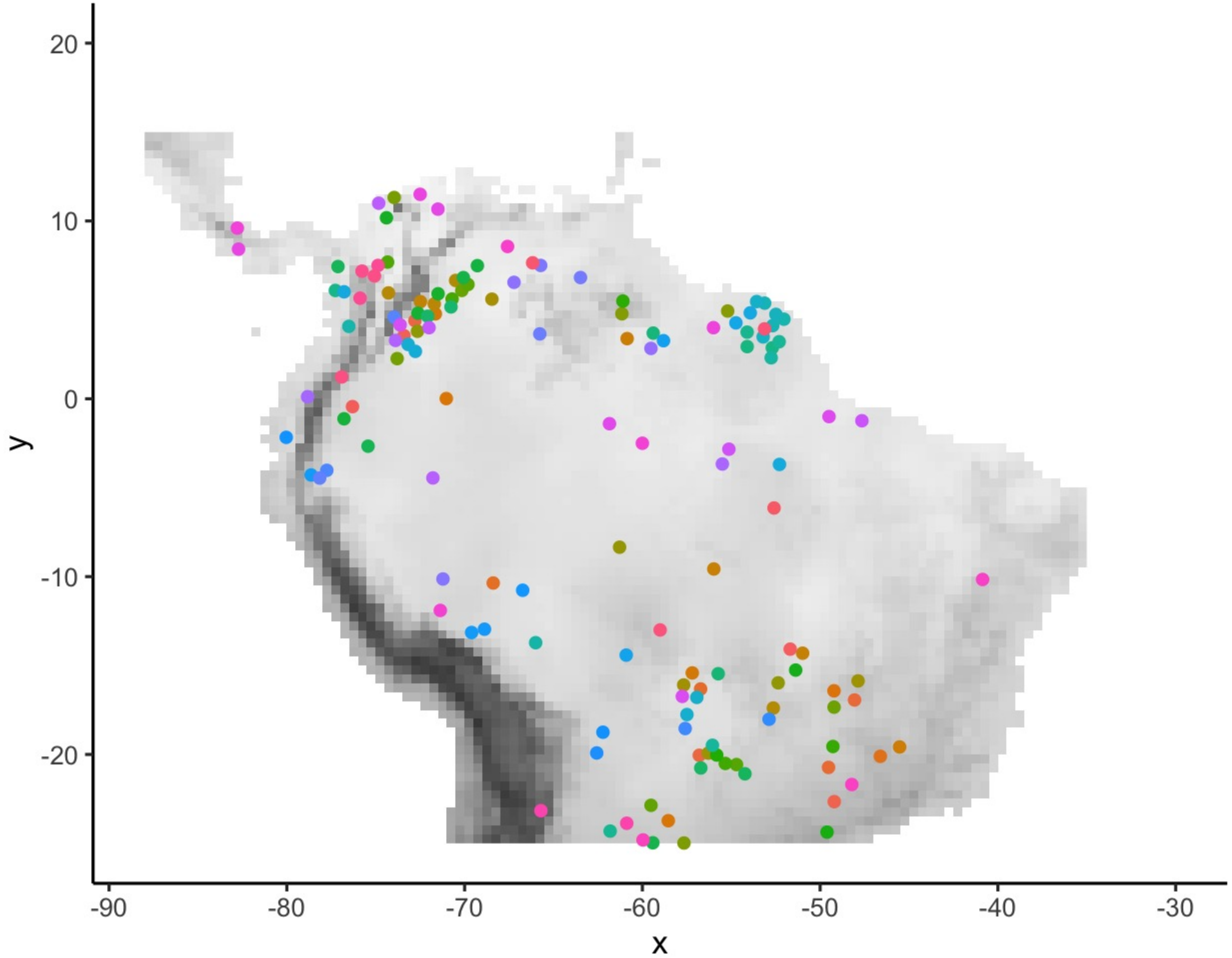
Dependence structure	Parametric solution	Blocking	Blocking illustration
Spatial	Spatial models (e.g.CAR, INLA, GWR)	Spatial	
Temporal	Time-series models (e.g.ARIMA)	Temporal	
Grouping	Mixed effect models (e.g. GLMM)	Group	
Hierarchical / Phylogenetic	Phylogenetic models (e.g. PGLS)	Hierarchical	

Roberts et al. 2017

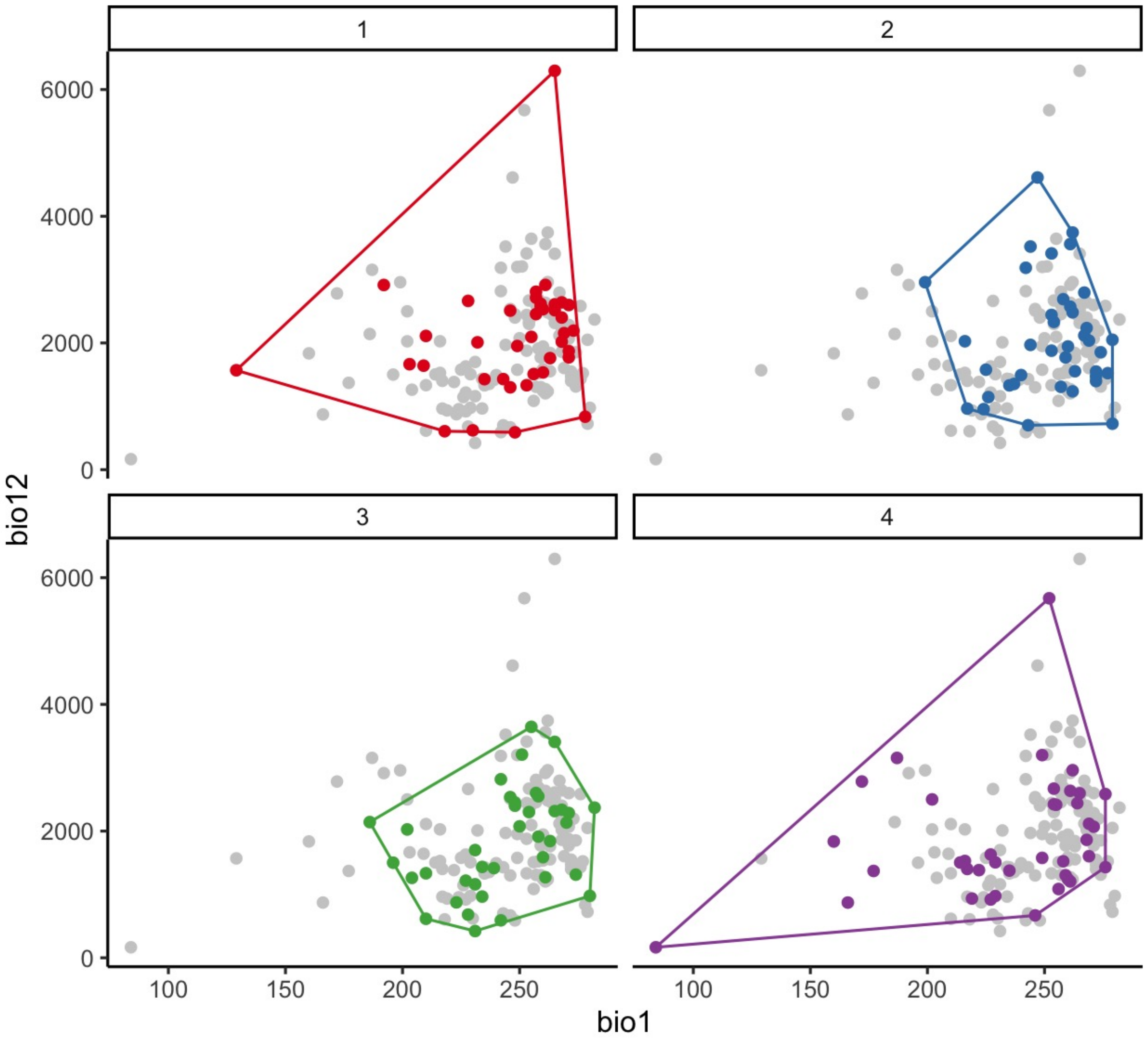


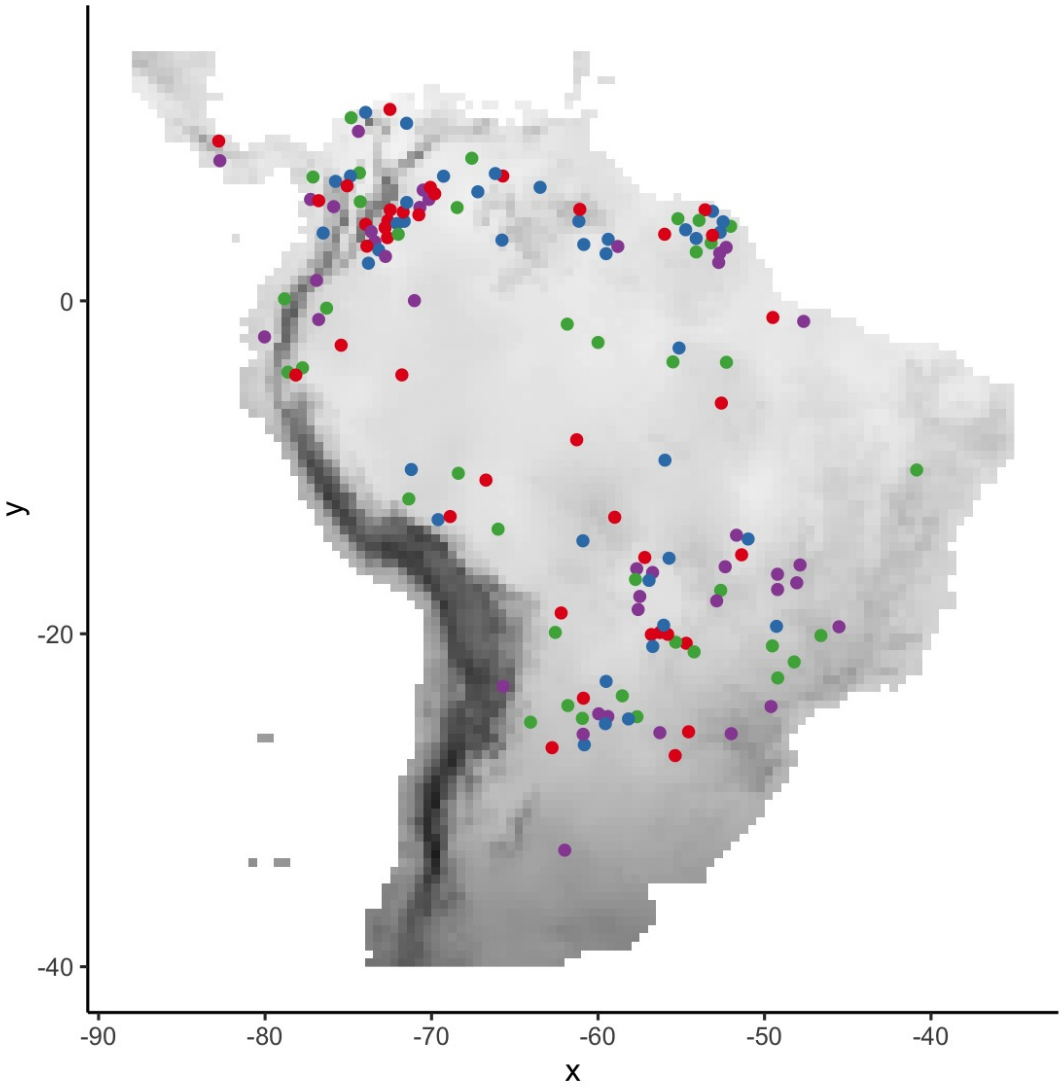
Radosavljevic & Anderson 2014

Ways to subset: leave-one-out (jackknife)



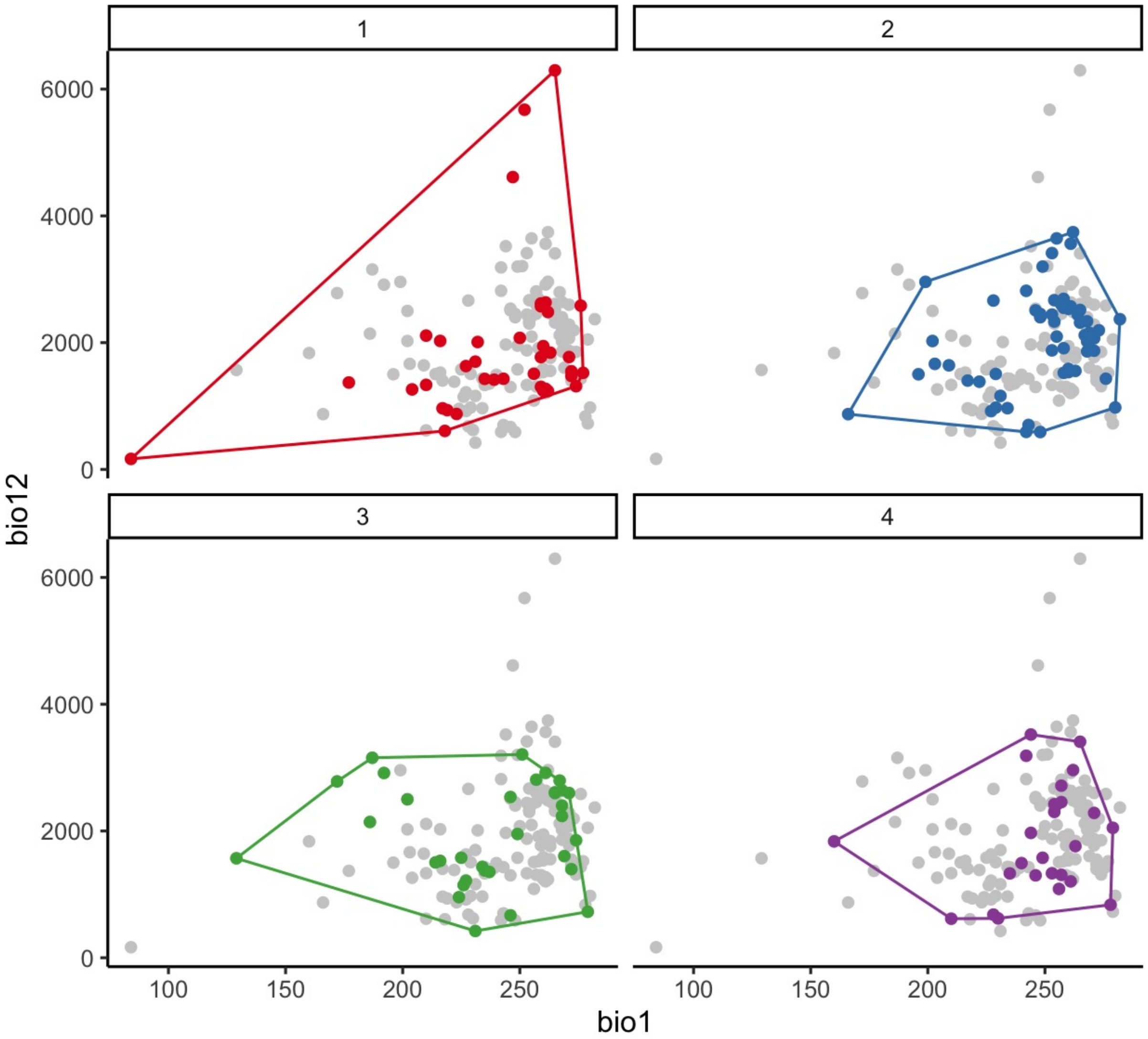
Ways to subset: random

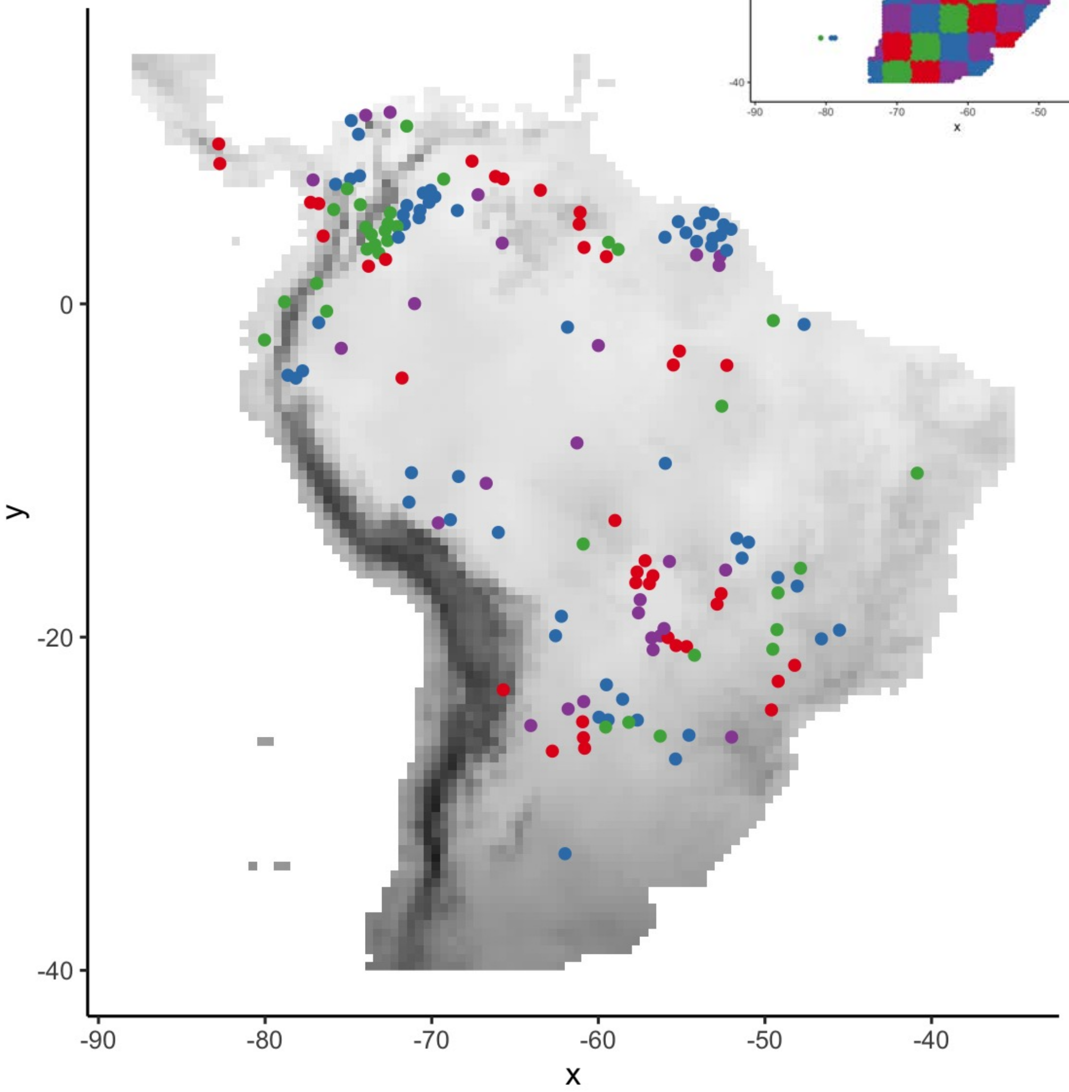


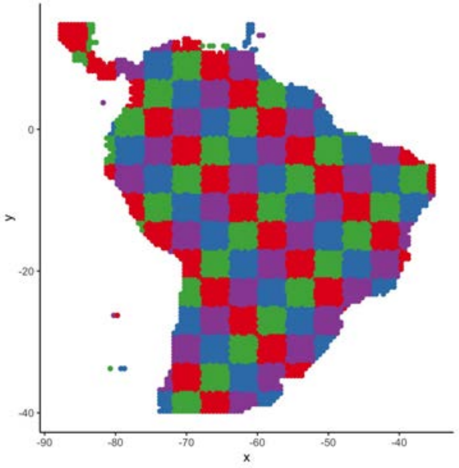


R package ENMeval

Ways to subset: spatial checkerboard

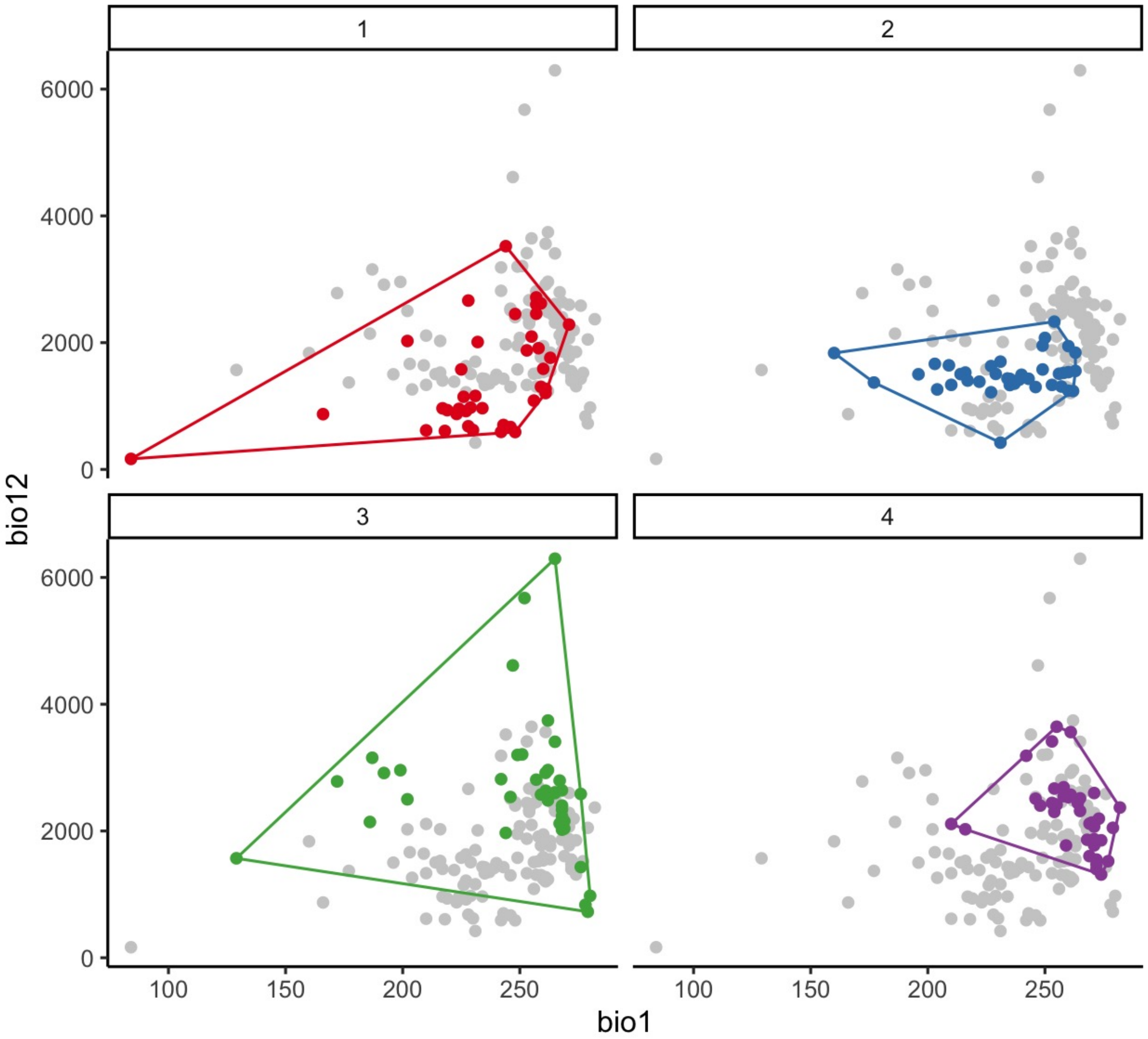


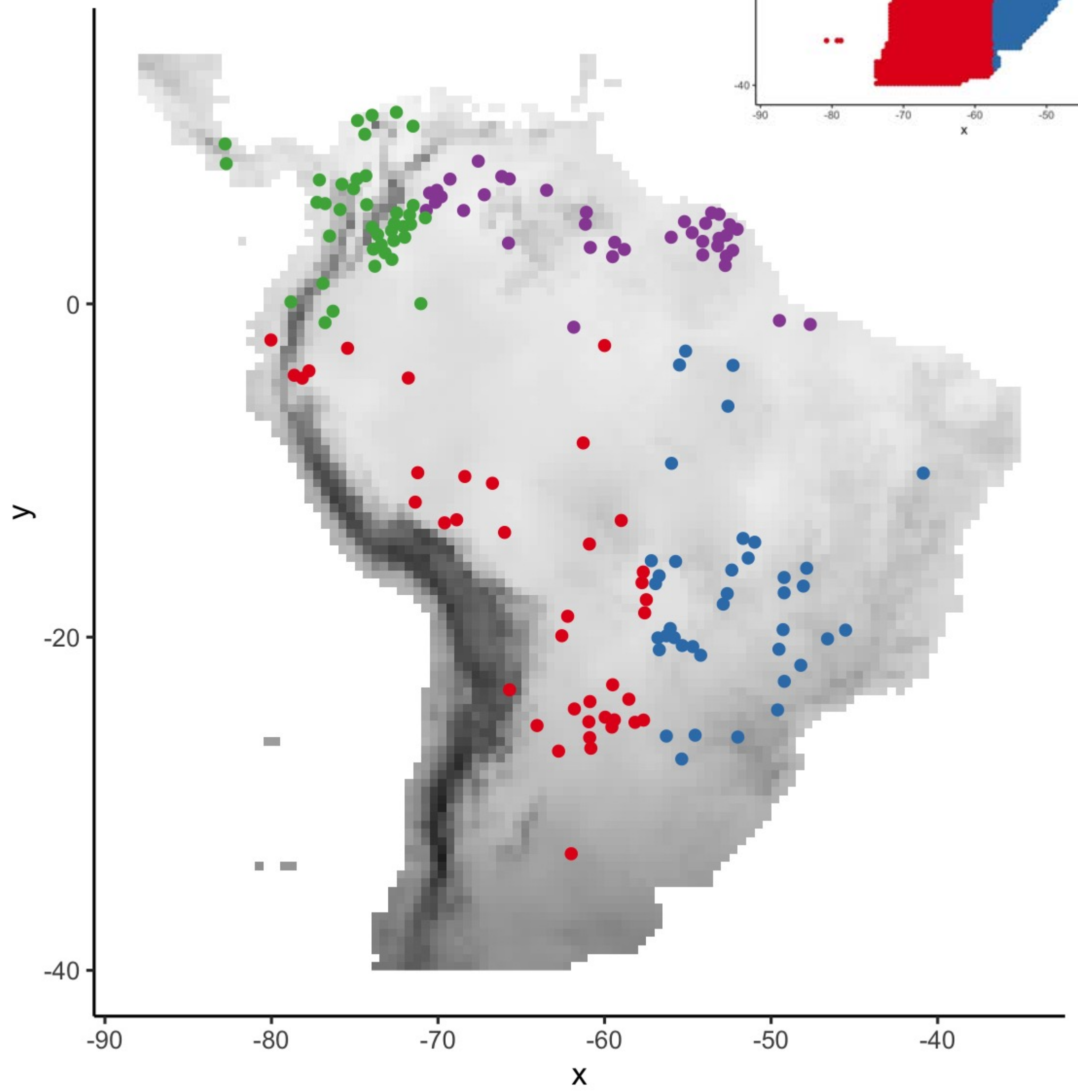


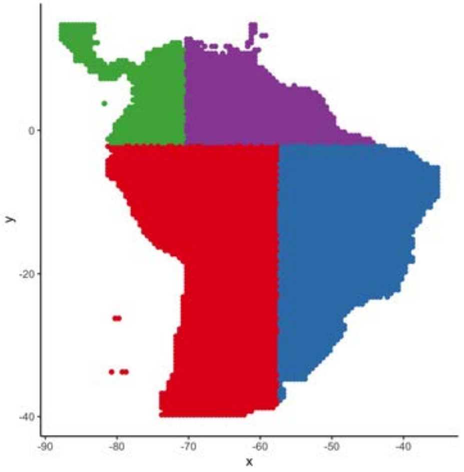


R package ENMeval

Ways to subset: balanced spatial block

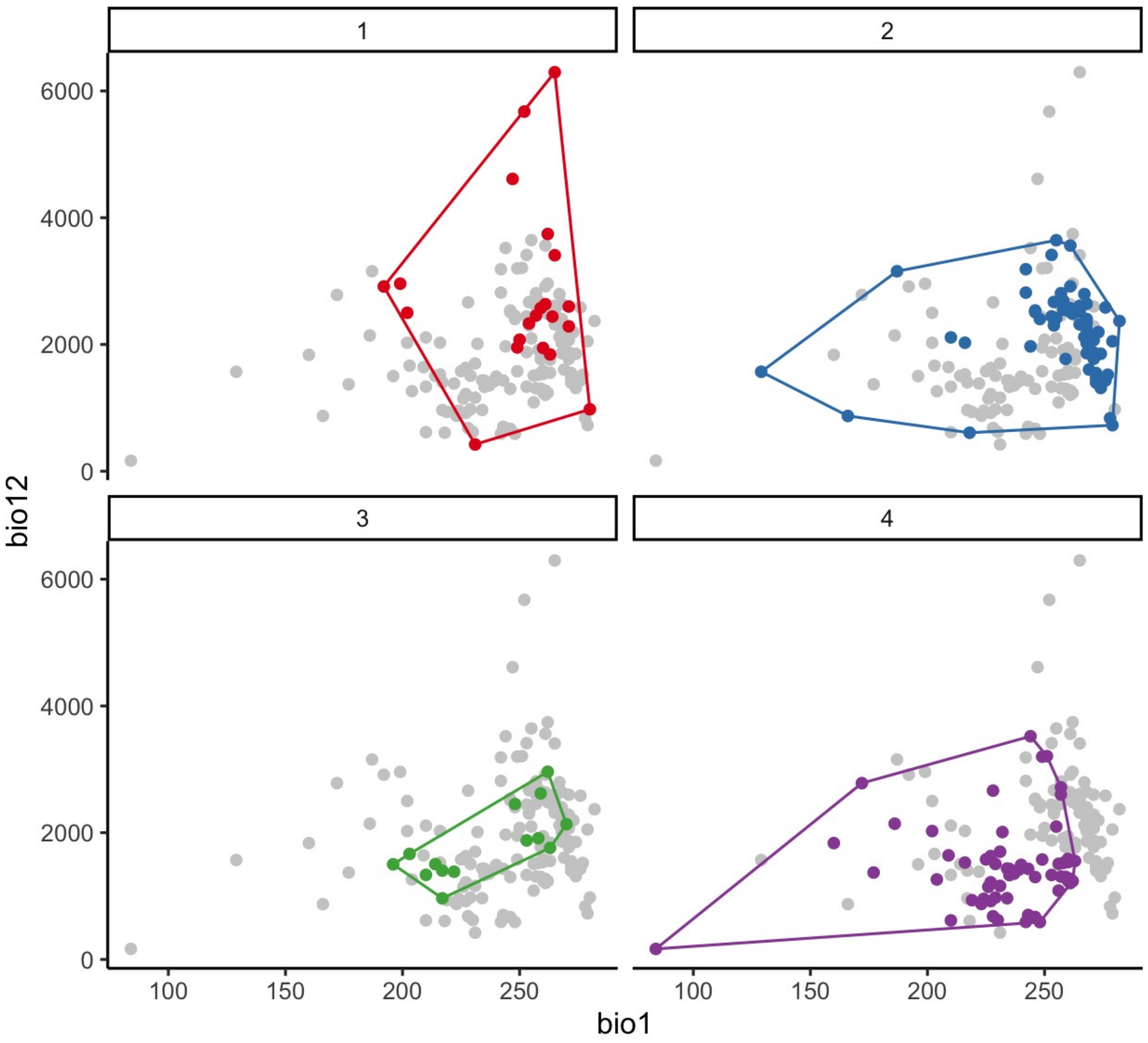


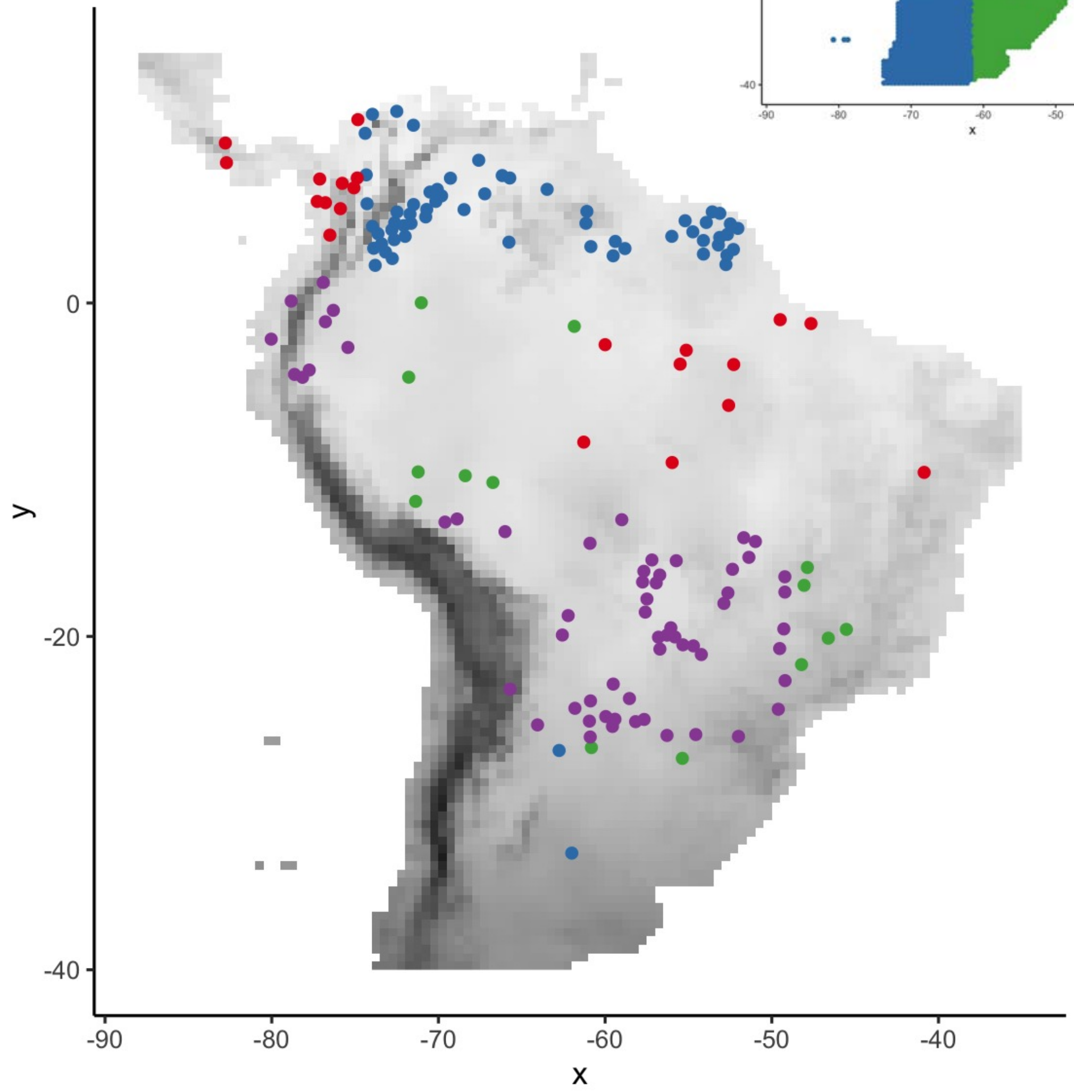


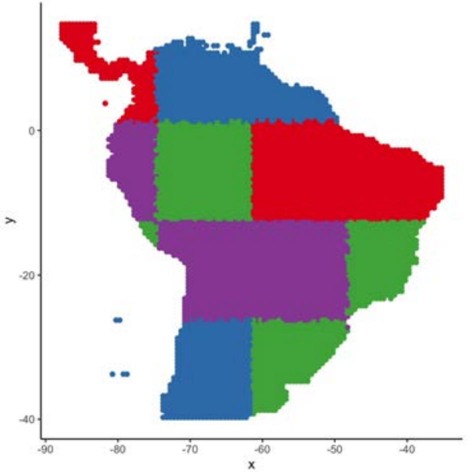


R package blockCV

Ways to subset: random spatial blocks

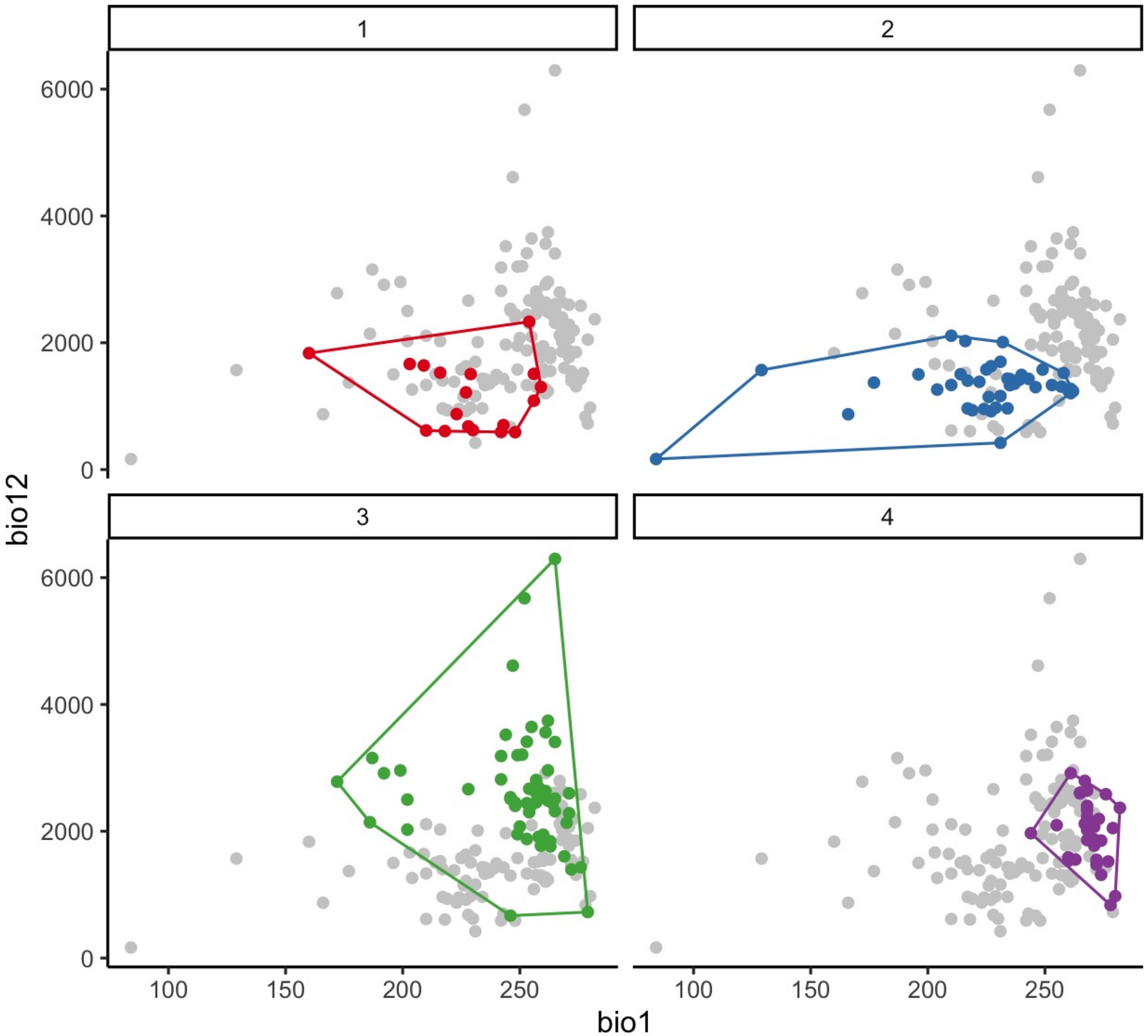


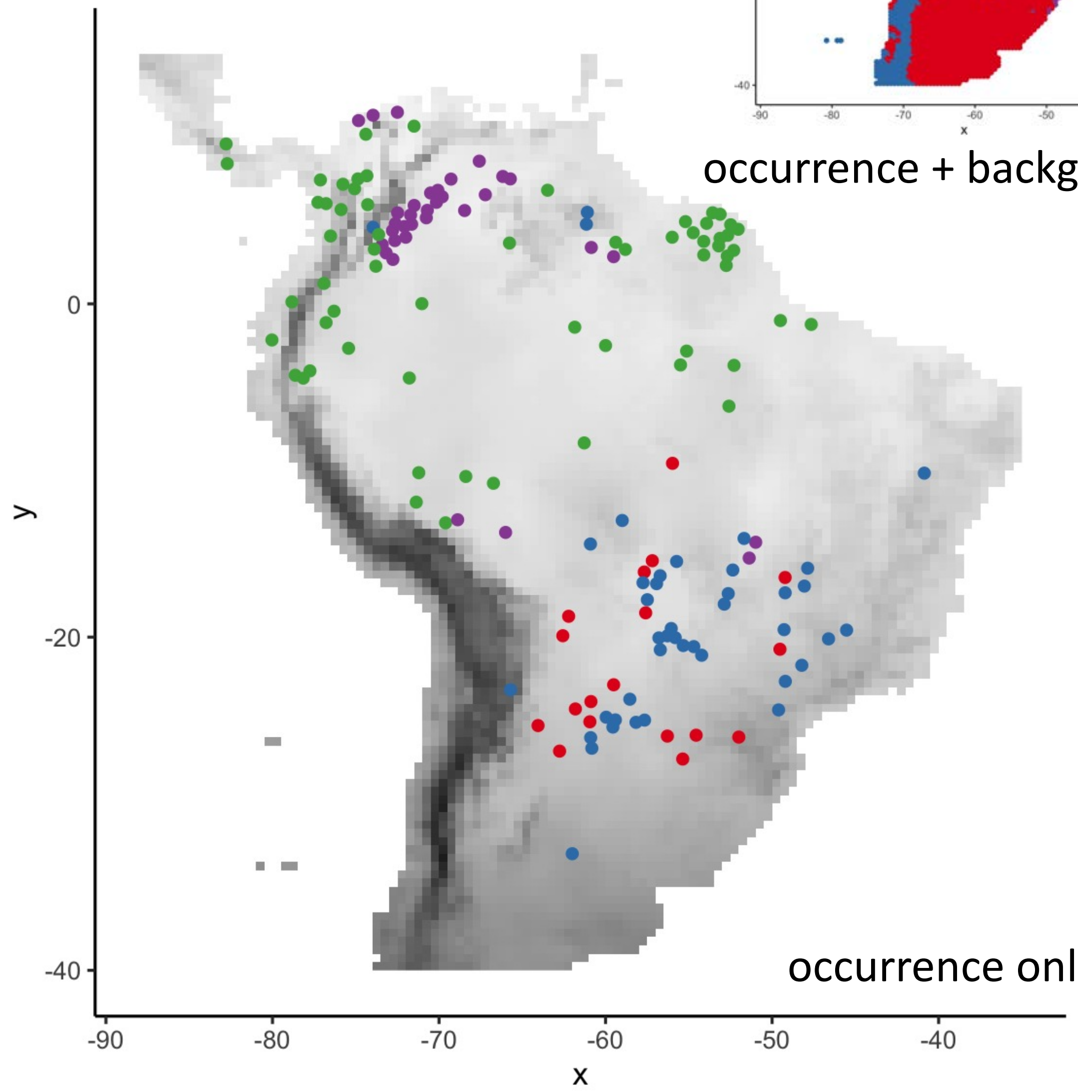




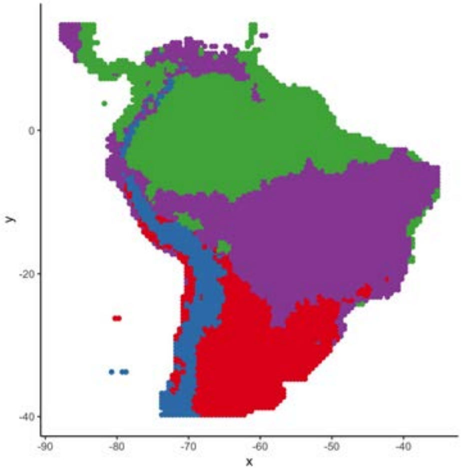
R package blockCV

Ways to subset: environmental blocks

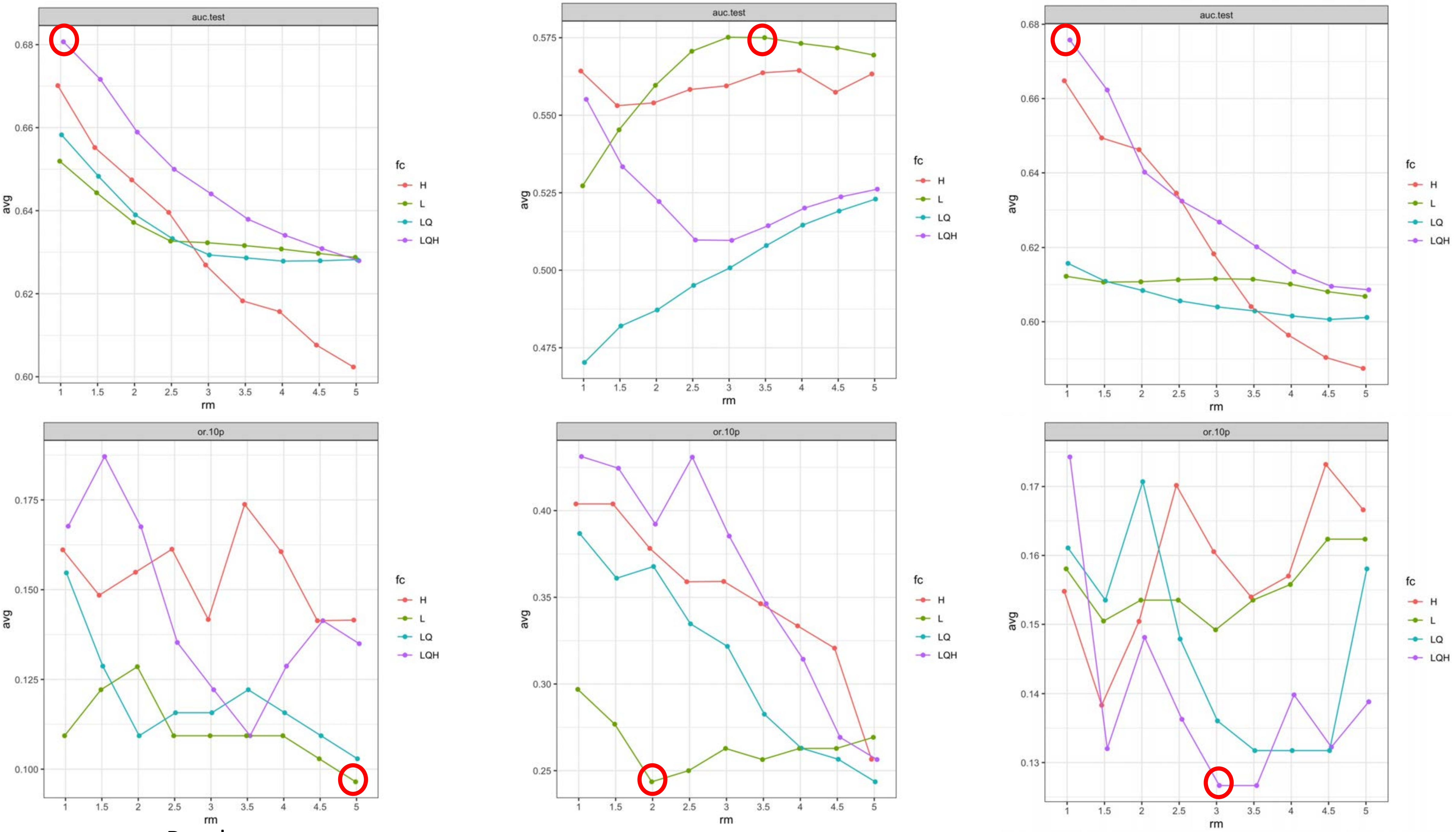




occurrence only



occurrence + background



Random Block Checkerboard

Comments on subsetting techniques

- leave-one-out (jackknife) best for low-data species
- block subsetting should extend to background data
- block subsetting usually results in less optimistic evaluation (**i.e., more realistic**)
- spatial checkerboard is likely to have more even sampling across environments than random
- some techniques do not ensure even sampling of occurrences
- blocking can force model extrapolation

ENMeval 2.0.0

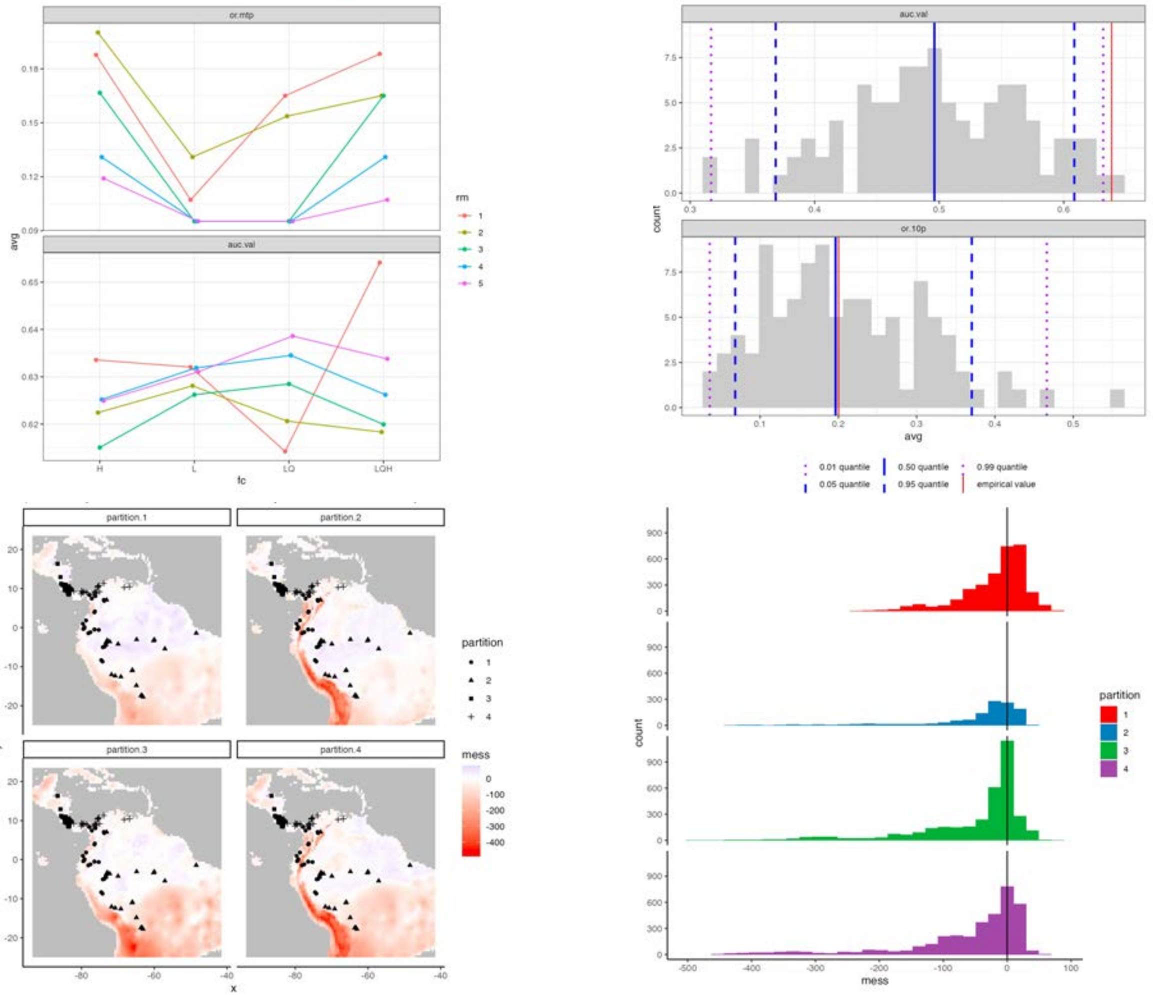
- new structure for adding other algorithms
- customizable model settings and performance metrics
- metadata generation (*rangeModelMetadata*)
- null models to quantify significance and effect sizes
- new visualization tools (*ggplot2*) for mapping partitions and showing environmental differences between them

APPLICATION

Methods in Ecology and Evolution BRITISH ECOLOGICAL SOCIETY

ENMeval 2.0: Redesigned for customizable and reproducible modeling of species' niches and distributions

Jamie M. Kass[1,2,3] | Robert Muscarella[4] | Peter J. Galante[5] | Corentin L. Bohl[6] | Gonzalo E. Pinilla-Buitrago[2,3] | Robert A. Boria[7] | Mariano Soley-Guardia[8] | Robert P. Anderson[2,3,9]



Conclusions

- cross validation can help provide estimates of model evaluation with "independent" data
- many ways to subset data (check out *ENMeval*[1,2] and *blockCV*[3])
- block subsetting has several advantages to random, and becomes very important when models are transferred[4]
- choose subsets based on analysis goals (interpolation or extrapolation)

1. Muscarella et al. 2014
2. Kass et al. 2021
3. Valavi et al. 2018
4. Roberts et al. 2017