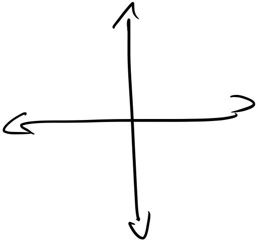
Spatial Data and Super Learners

Big Data y Machine Learning para Economía Aplicada

Ignacio Sarmiento-Barbieri

Universidad de los Andes

Agenda



Agenda

Agenda

Motivation

- In Big Data volume was only a part of the story
- Big Data are data of high complexity: anarchic and spontaneous
- They are the by product of an action: pay with credit card, tweet, move from point A to point B, buy a house, etc.
- Now we are going to center on spatial data

Agenda

1. Spatial Data
   - Motivation
   - Types of Spatial Data
   - Coordinate Systems
   - Spatial Dependence
2. Superlearners

Types of Spatial Data

Spatial data comes in many "shapes" and "sizes", the most common types of spatial data are:

CBD

- Points
- Lines
- Polygons
- Grid (Raster)

Types of Spatial Data: Points

D. Albouy, P. Christensen and I. Sarmiento-Barbieri / Journal of Public Economics 182 (2020) 104110 5

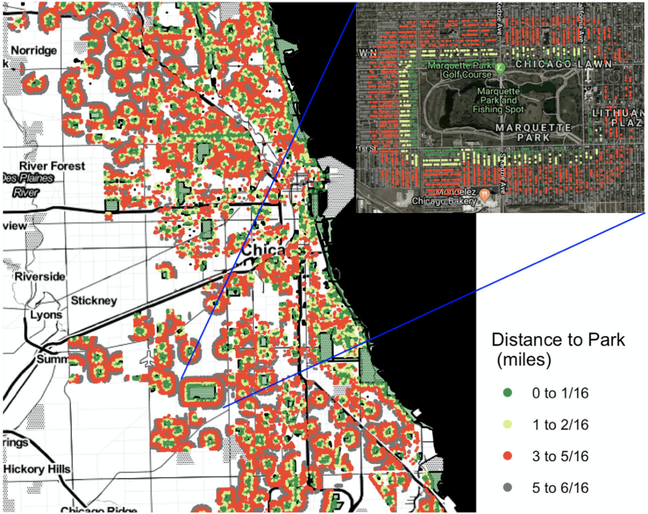


Fig. 1. Housing transactions around parks: neighborhood distance intervals. Notes: The following figure shows transactions within 3/8 miles of the nearest park in Chicago. The

Types of Spatial Data: Lines

D. McMillen, I. Sarmiento-Barbieri and R. Singh   Journal of Urban Economics 110 (2019) 1–25

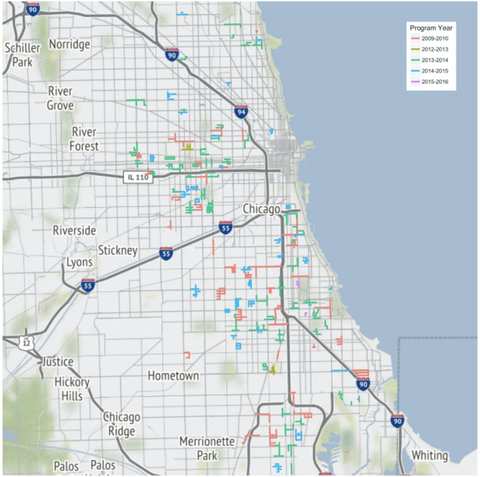
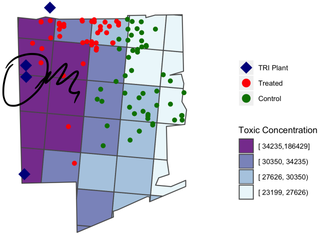
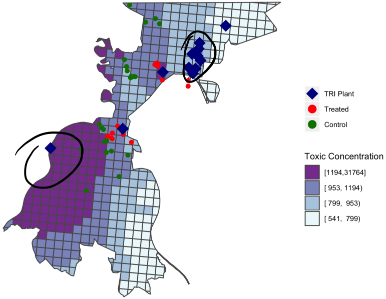


Fig. 1. Safe Passage Routes, by year of program adoption.
Note: Shapefiles with Safe Passage shape and location where obtained from the Chicago Data Portal and year that the program was launched at each location through a FOIA request.

McMillen, Sarmiento-Barbieri & Singh, 2019

Types of Spatial Data: Rasters





Christensen,Sarmiento-Barbieri & Timmins (2022)

Agenda

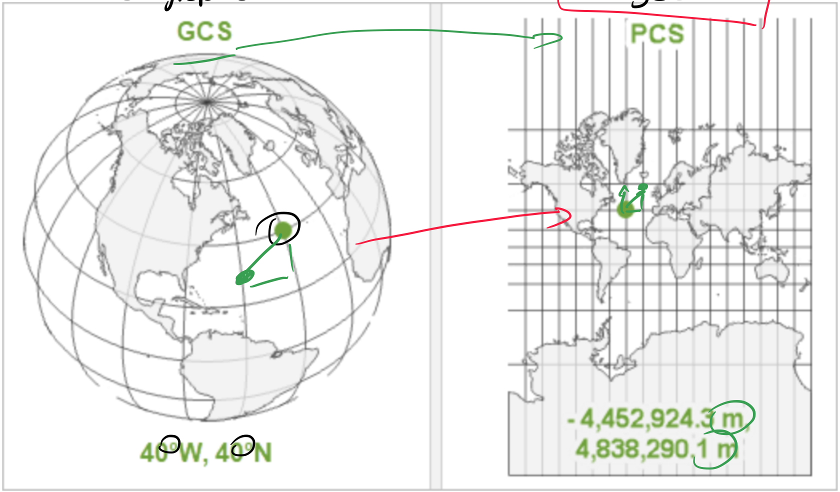
Coordinate Systems

Geographic

WGS84

NAD83

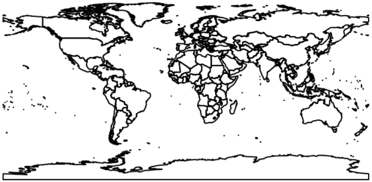
Projected.

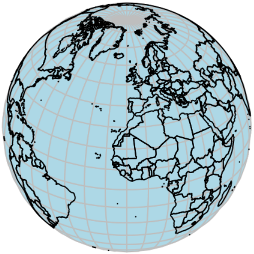


Source: https://www.geoawesomeness.com/all-map-projections-in-compared-and-visualized/

The earth ain't flat

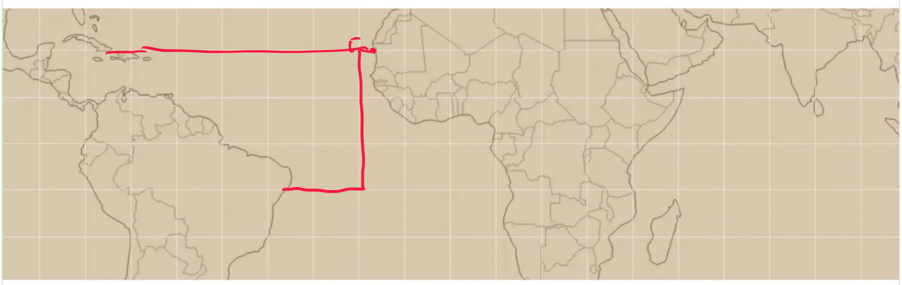
- The world is an irregularly shaped ellipsoid, but plotting devices are flat
- But if you want to show it on a flat map you need a map projection,





The earth ain't flat

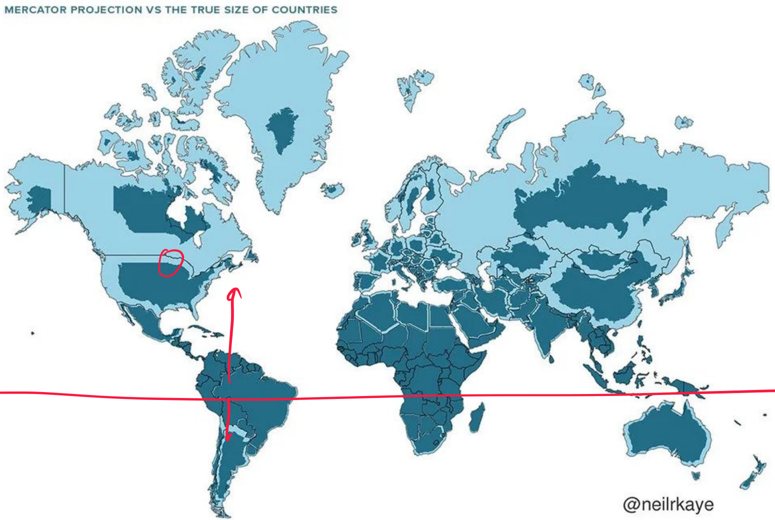
Mercator Projection



Source: https://www.geoawesomeness.com/all-map-projections-in-compared-and-visualized/

The earth ain't flat

Mercator and the true size of countries



Source: https://www.reddit.com/r/Damnthatsinteresting/comments/xziol9/mercator_projection_vs_the_true_size_of_countries

Which Projection should I choose?

- "There exist no all-purpose projections, all involve distortion when far from the center of the specified frame" (Bivand, Pebesma, and Gómez-Rubio 2013)
- The decision as to which map projection and coordinate reference system to use, depends on the regional extent of the area you want to work in, on the analysis you want to do and often on the availability of data.
- In some cases, it is not something that we are free to decide: "often the choice of projection is made by a public mapping agency" (Bivand, Pebesma, and Gómez-Rubio 2013). → IGAC → MAGNA - SIRGAS WGS84
- This means that when working with local data sources, it is likely preferable to work with the CRS in which the data was provided.

Agenda

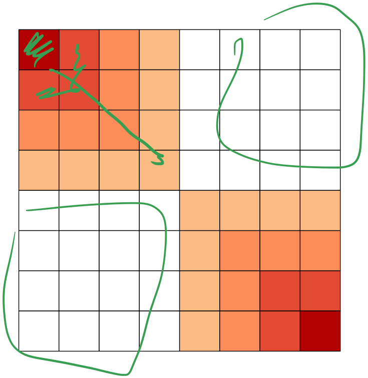
st_transform (, crs = . . .)

st_crs ( )

1. Spatial Data
   - Motivation
   - Types of Spatial Data
   - Coordinate Systems
   - Spatial Dependence
2. Superlearners

Spatial Dependence/ Spatial (Auto)correlation

- Spatial autocorrelation measures the degree to which a phenomenon of interest is correlated to itself in space (Cliff and Ord (1973)).
- For example, positive spatial correlation arises when units that are *close* to one another are more similar than units that are far apart



Spatial Dependence/ Spatial (Auto)correlation

~~iid~~

- We can express the existence of spatial autocorrelation with the following moment condition:

$$Cov(y_i, y_j) \neq 0 \quad for \quad i \neq j \tag{1}$$

were y_i and y_j are observations on a random variable at locations i and j.

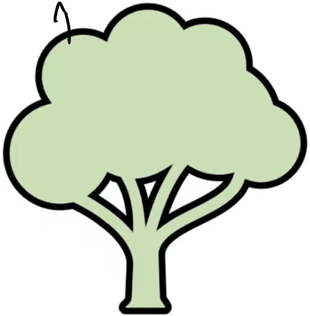
- This autocorrelation can lead to overfitting of the model and poor generalization to new spatial locations.

Spatial Prediction and Cross-Validation

$y =$ crecimiento del árbol

$X = \{$especie, tamaño$\}$

robles

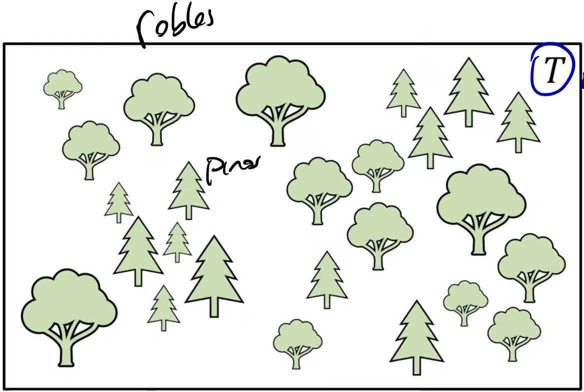


$$y = f(x) + u$$

$$f(x_0) = \hat{y}_0$$

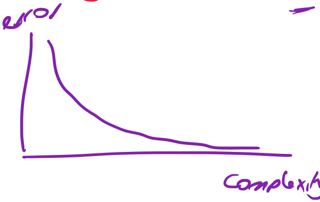
$$\hat{y}_0 \approx y_0$$

Spatial Prediction and Cross-Validation

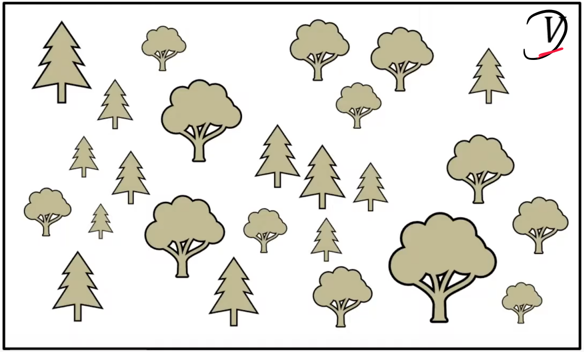


Training set

$$y_T = f(x_T) + \boxed{u_T}$$

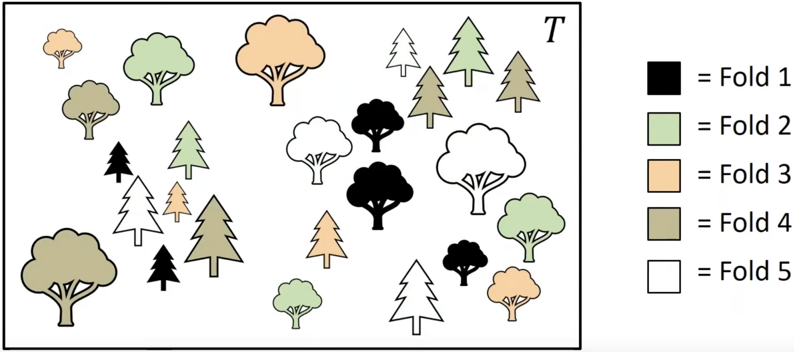


Spatial Prediction and Cross-Validation



$y_v - \hat{f}_T(x_v)$

Spatial Prediction and Cross-Validation



-> Test

Train

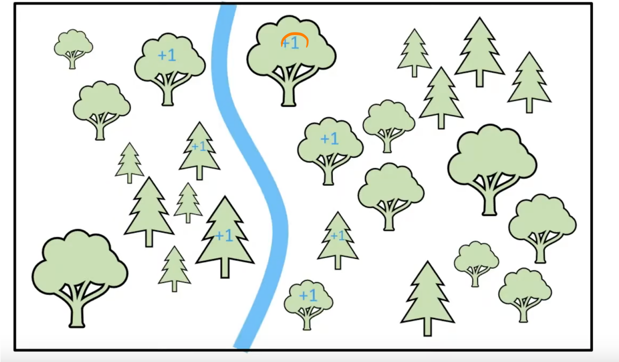
Test

Train

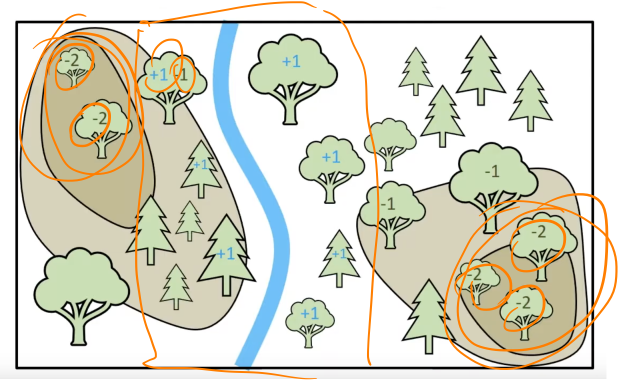
Train

Spatial Prediction and Cross-Validation

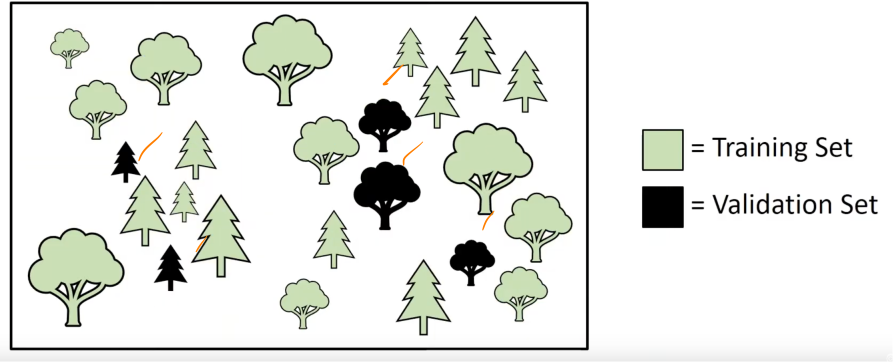
$$y = f(\text{especie}, \text{tamaño}) + u$$



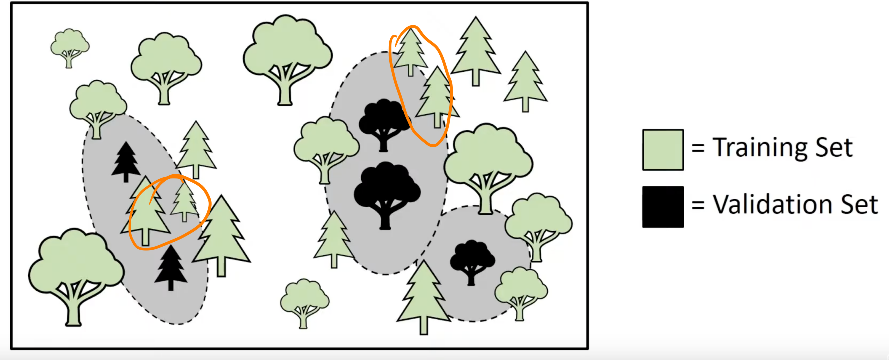
Spatial Prediction and Cross-Validation



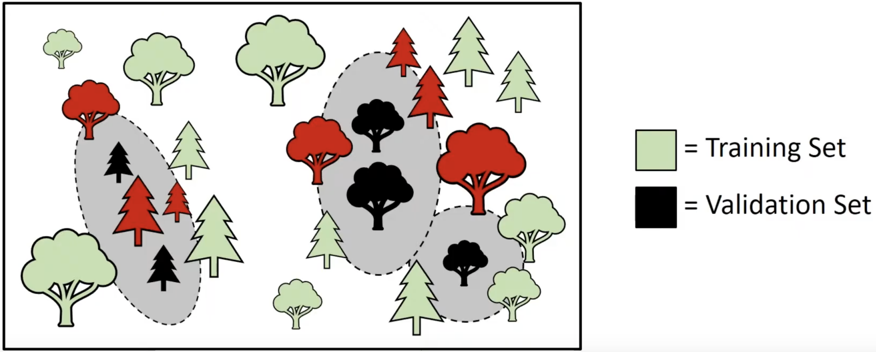
Spatial Prediction and Cross-Validation



Spatial Prediction and Cross-Validation



Spatial Prediction and Cross-Validation



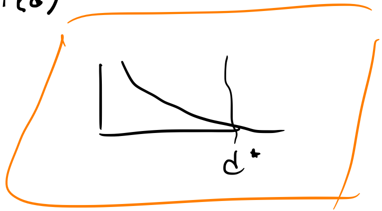
Resumen

$$y(s) = f\big(X(s)\big) + \varepsilon(s),$$

donde:

- s es una ubicación en espacio
- $f(X(s)) \rightarrow$ mi objetivo
- $\varepsilon(s)$ es ruido tal que $\mathbb{E}[\varepsilon(s)] = 0$, $\mathrm{Var}[\varepsilon(s)] = \sigma^2$, ~~y~~ con covarianza espacial:

$$\mathrm{Cov}\big(\varepsilon(s_i), \varepsilon(s_j)\big) = \sigma^2 \rho(d)$$



- $\rho(d)$ la función de correlación decreciente con d.
- $d(s_i, s_j)$ es la distancia entre puntos

Error de generalización verdadero

El error cuadrático medio fuera de muestra en una nueva ubicación s_0 es:

$$\text{Err}_{\text{out}} = \mathbb{E}\left[\Big(Y(s_0) - \hat{f}_{\mathcal{T}}(X(s_0))\Big)^2\right] = \underbrace{\mathbb{E}\left[\Big(f(X(s_0)) - \hat{f}_{\mathcal{T}}(X(s_0))\Big)^2\right]}_{\text{reducible}} + \underbrace{\sigma^2}_{\text{irreducible}},$$

donde $\hat{f}_{\mathcal{T}}$ es el predictor ajustado con el conjunto de entrenamiento $\mathcal{T}$.

Bias^2 + Variance

Ejemplo algebraico sencillo

- Consideremos sólo dos ubicaciones s_i y s_j
- Donde $\text{Corr}\left(\varepsilon(s_i), \varepsilon(s_j)\right) = \rho$

Supongamos que, al estimar $\hat{f}$ con $\mathcal{T} = \{s_j\}$, aprendemos el ruido

$$\hat{f}_{T=\{s_J\}}(X(s_i)) = f(X(s_i)) + \varepsilon(s_J)$$

$$E\left(\left(y(s_i) - \hat{f}_{T=s_J}(x_i)\right)^2\right) = E\left(\left(\varepsilon(s_i) - \varepsilon(s_J)\right)^2\right)$$

$$= Var(\varepsilon(s_i)) + Var(\varepsilon(s_J)) - 2\,Cov(\varepsilon(s_i), \varepsilon(s_J))$$

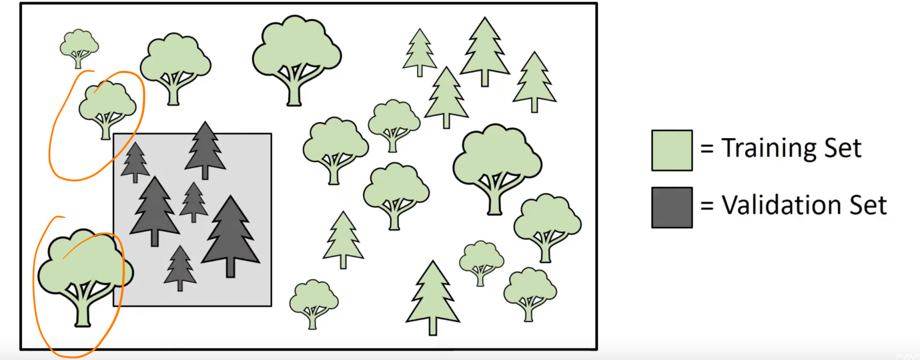
$$= 2\sigma^2 - 2\sigma^2\rho = 2\sigma^2(1-\rho)$$

Resumen

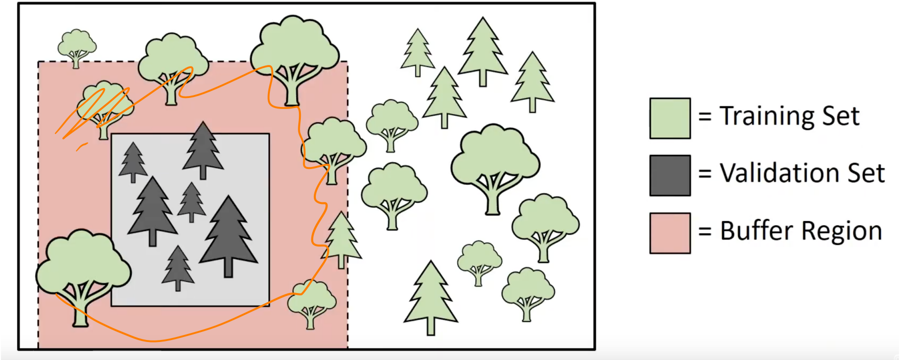
- **CV aleatoria:** $\hat{Err}_{CV} = \boxed{2\sigma^2(1-\rho)} < 2\sigma^2$
- **CV espacial:** correlación es 0

$\hat{Err}_{C\,espacial} = \boxed{2\sigma^2}$

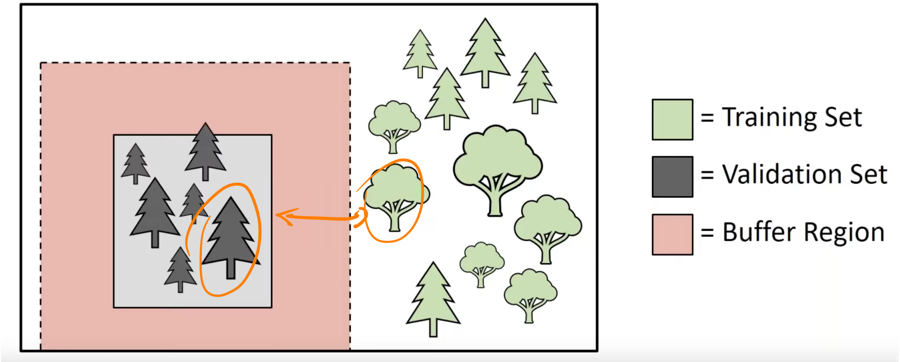
Spatial Prediction and Cross-Validation



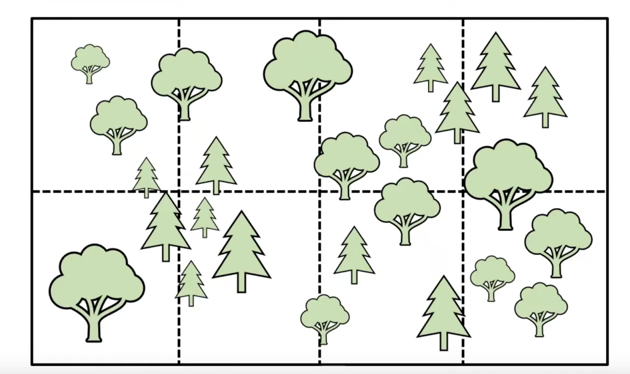
Spatial Prediction and Cross-Validation



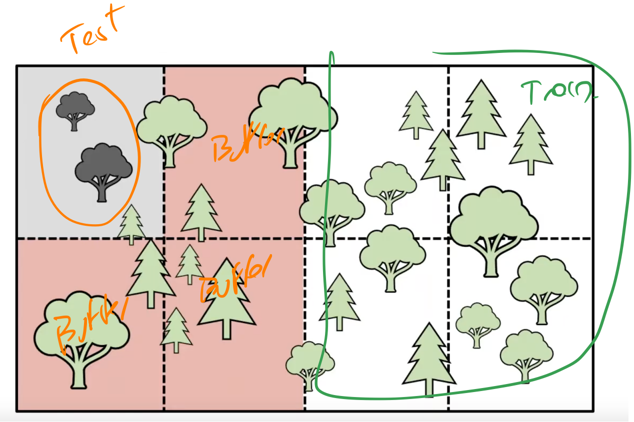
Spatial Prediction and Cross-Validation



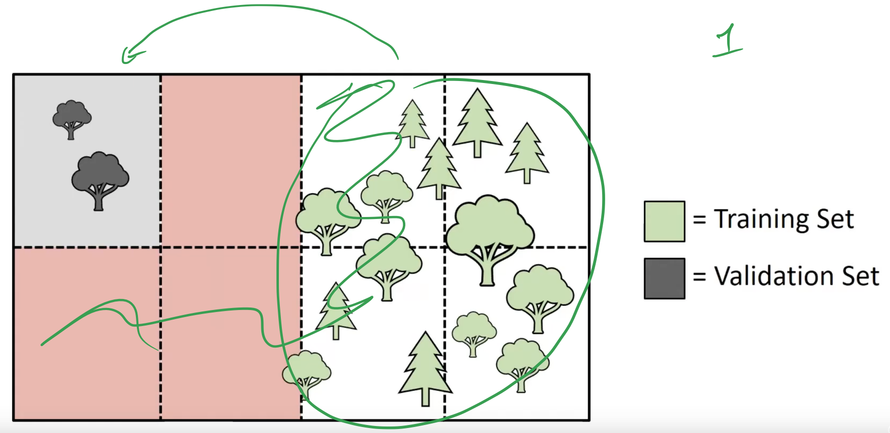
Spatial Prediction and Cross-Validation



Spatial Prediction and Cross-Validation



Spatial Prediction and Cross-Validation



Example: Spatial Cross-Validation

photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/

Agenda

Recap

- Queremos predecir y en funcion de observables ($\mathbf{x}_i$)

$$y = f(\mathbf{x}_i) + u \tag{2}$$

- donde la estimación de f implica la que mejor generalize (prediga mejor fuera de muestra):

$$\hat{f} = \underset{f}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} L(y_i, f(\mathbf{x}_i; \Theta)) \right\} \qquad V(\bar{x}) = \frac{\sigma^2}{n} \tag{3}$$

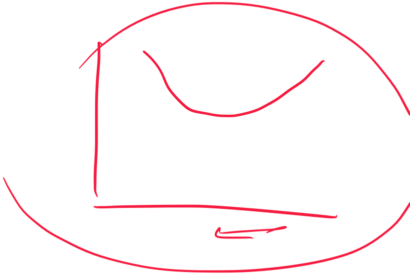
Superlearners: Motivation

- Superlearning is a technique for prediction that involves combining many individual statistical algorithms to create a new, single prediction algorithm that is expected to perform at least as well as any of the individual algorithms.
- The inovation?

SL → decidir como combinar de los algoritmos utilizados

Xgboost NN

mean lm



Superlearners: Algorithm

Denote the library $\mathcal{L}$ and its cardinality as $V(n)$.

1. Fit each algorithm in $\mathcal{L}$ on the entire data set $X = \{X_i : i = 1, ..., n\}$ to estimate $f_v(X)$ with $v = 1, \ldots, V(n)$

1- Lm
2- arbol
3- bosques
⋮
V- NNet

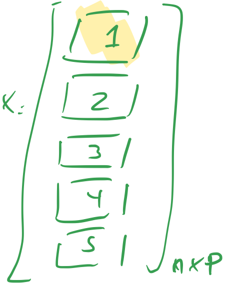
$\hat{y}_{lm}$ $n \times 1$ $\quad$ $\hat{y}_{arbol}$ $n \times 1$ $\quad$ $\hat{y}_{bosques}$ $n \times 1$ $\quad \ldots \quad$ $\hat{y}_{NNet}$ $n \times 1$

$$\hat{y}_{SL} = \sum_{v=1}^{V} \boxed{\alpha_v} \hat{y}_v$$

Superlearners: Algorithm

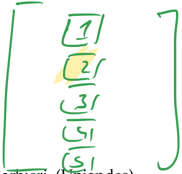
2. Split the data set X into a training and validation sample, according to a K-fold cross-validation scheme:

$K=5$



Test $\rightarrow \hat{y}_v^{(1)}$

Train



$\longrightarrow \hat{y}_v^{(2)}$

Superlearners: Algorithm

3. For the kth fold, fit each algorithm in $\mathcal{L}$ on $T(k)$ and save the predictions on the corresponding test , $\hat{f}_{k,T}(X_i)$ with $X \in Te(k)$

20 1 → $\hat{y}^{(1)}_{lm} \} 20$ $\hat{y}^{(1)}_{orb}$ $\hat{y}^{(1)}_{bosq}$ $\hat{y}^{(1)}_{NNET}$

20 2

20 3

20 4

20 5 $\hat{y}^{(5)}_{lm} \} 20$ $\hat{y}^{(\,)}_{orb}$ $\hat{y}^{(5)}_{bosq}$ $\hat{y}^{(5)}_{NNET}$

Superlearners: Algorithm

4. Bind the predictions from each algorithm together to create a n by V matrix

$$Z = \begin{bmatrix} \hat{y}^{(1)}_{lm\,f20} & \hat{y}^{(1)}_{orb} & \hat{y}^{(1)}_{bosq} & \hat{y}^{(1)}_{NNET} \\ & & & \\ \hat{y}^{(s)}_{lm\,f20} & \hat{y}^{(\,)}_{orb} & \hat{y}^{(s)}_{bosq} & \hat{y}^{(s)}_{NNET} \end{bmatrix}$$

$n \times V$

y

Superlearners: Algorithm

5 Propose a family of weighted combinations of the candidate estimators indexed by weight-vector α:

$$m(z|\alpha) = \sum_{v=1}^{V} \alpha_v \hat{y}_v$$

$$\alpha_v \geq 0 \quad \forall v$$

$$\sum_{v=1}^{V} \alpha_v = 1$$

Superlearners: Algorithm

6. Determine the α that minimizes the cross-validated risk of the candidate estimator $\sum_{v=1}^{V} \alpha_v \hat{f}(X_i)$ over all allowed α-combinations:

$$\hat{\alpha} = \arg\min \sum_{i=1}^{N} \left[\left(y_i - m(z|\alpha) \right)^2 \right]$$

$$\hat{\alpha} = \arg\min \sum_{i=1}^{N} \left[\left(y_i - \sum_{v=1}^{V} \alpha_v \hat{y}_v \right)^2 \right]$$

$y_i \xrightarrow{\text{regres.}} z \rightarrow \hat{\alpha}$

+ restringido $\quad \hat{\alpha} \geq 0$ + Norm $\quad \sum \alpha = 1$

$\rightarrow$ NNLS

Superlearners: Algorithm

7 Combine $\hat{\alpha}_v$ with $\hat{f}_v(X_i)$ according to the weights found, and create the final super learner fit

$$\hat{y}_{SL} = \sum_{v=1}^{V} \hat{\alpha}_v \underbrace{\hat{y}_v}_{\text{predicción del paso 1}}$$

Superlearners: Algorithm

Some considerations:

- The super learner theory does not place any restrictions on the family of weighted combinations used for ensembling the algorithms in the library.
- The restriction of the parameter space for α to be the convex combination of the algorithms in the library provides greater stability of the final super learner prediction.

Example: Superlearners

photo from https://www.dailydot.com/parsec/batman-1966-labels-tumblr-twitter-vine/