Quantitative Research Methods IV - 17.806

Recitation, Week 4.

Topic: Supervised Learning II.

Benjamín Muñoz

March 3, 2023

MIT

Massachusetts Institute of Technology

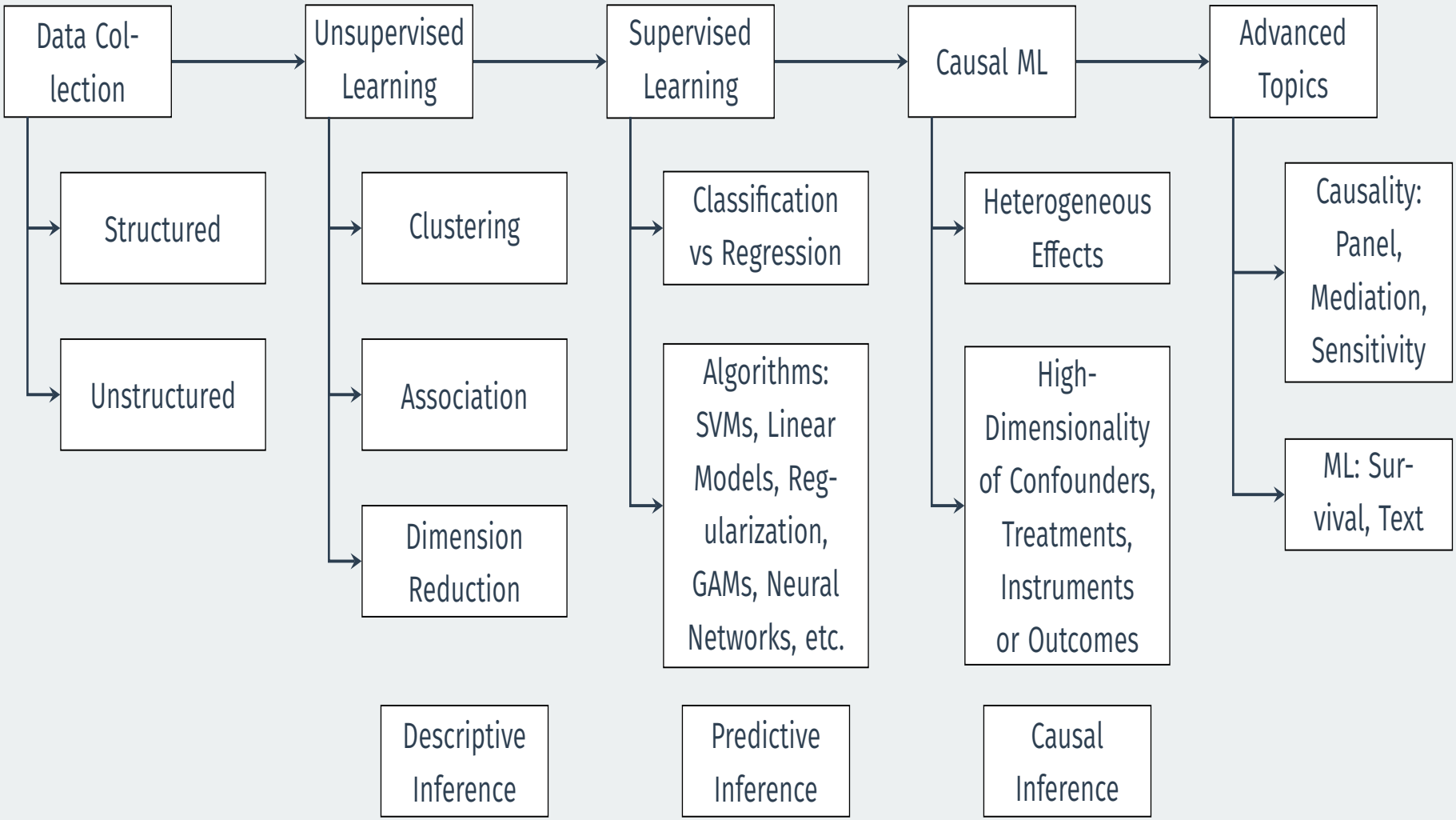
Table of contents

1. Theoretical Review

2. Algorithms

1/ Theoretical Review

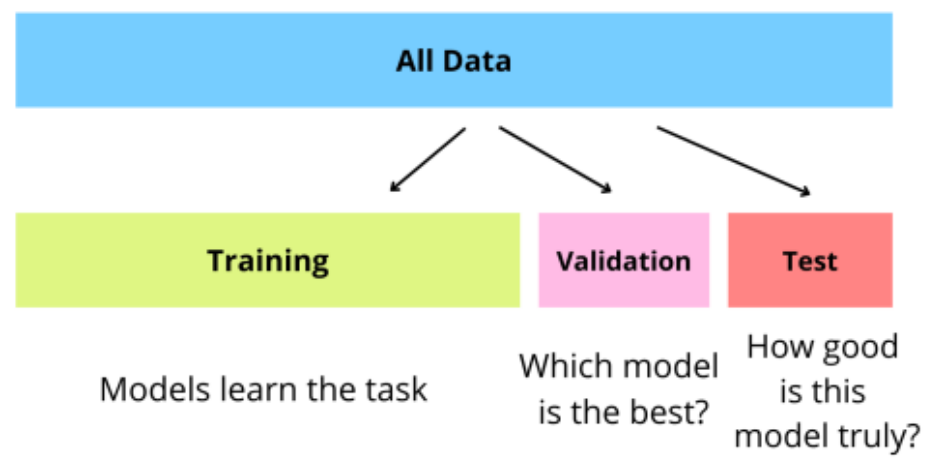
Quantitative Research Methods IV



Supervised Learning

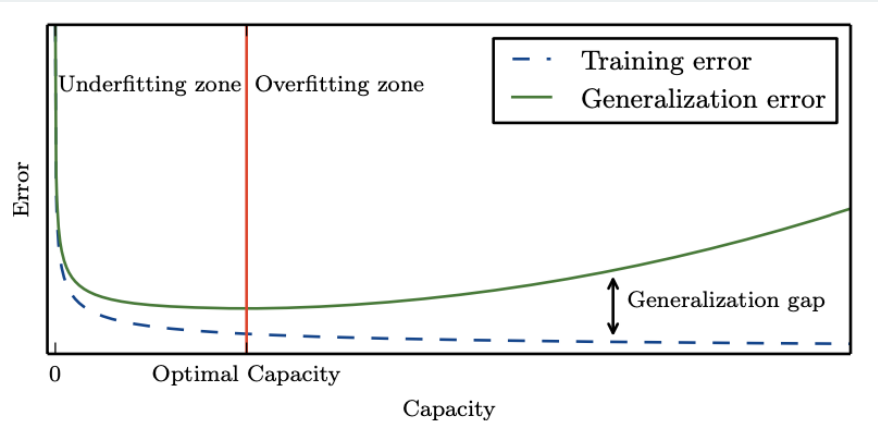
- Algorithms that experience "features" and a supervision signal (target or label). **Labeled Set** of input-output pairs ($D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$).
  - Using this data we build a **prediction** model or learner ($p(Y|\boldsymbol{X})$).
  - Good Learner: accurately predicts the target (*Performance Measure*).
  - Learner: Function $f(X)$ for predicting Y given the input vector $\boldsymbol{X}$. This requires a **Loss Function** ($L(Y, f(X))$) for penalizing errors in prediction.
- **Type of Outcome:** Regression vs Classification.

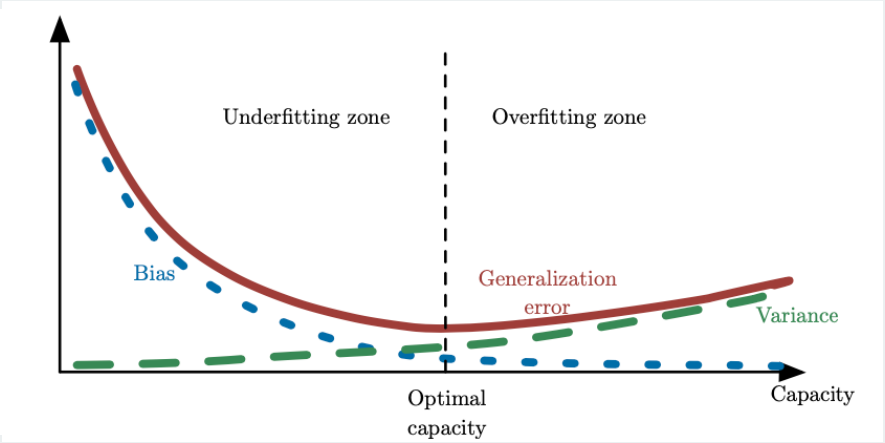
Workflow



- **Generalization:** Algorihm must perform well on *new, previously unseen* inputs.
- Minimization of Training Error $\longrightarrow$ Optimization Problem.
- Minimization of Generalization (Test) Error is the goal.

Model's Capacity, Bias-Variance Decomposition and Overfitting

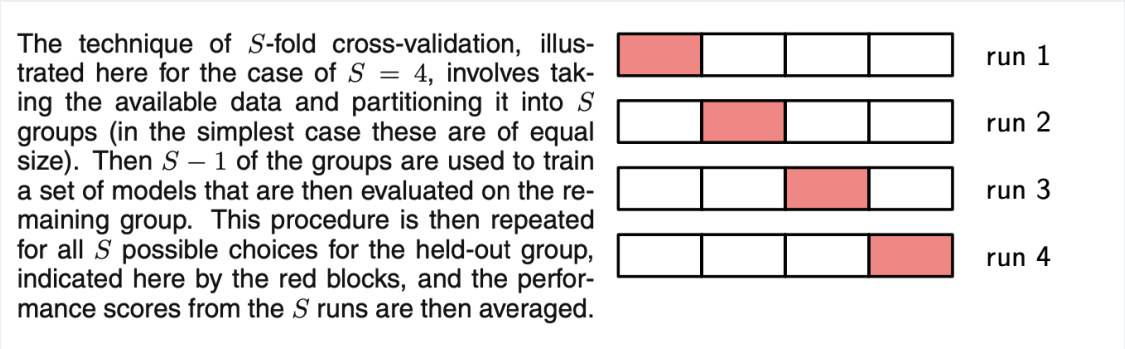




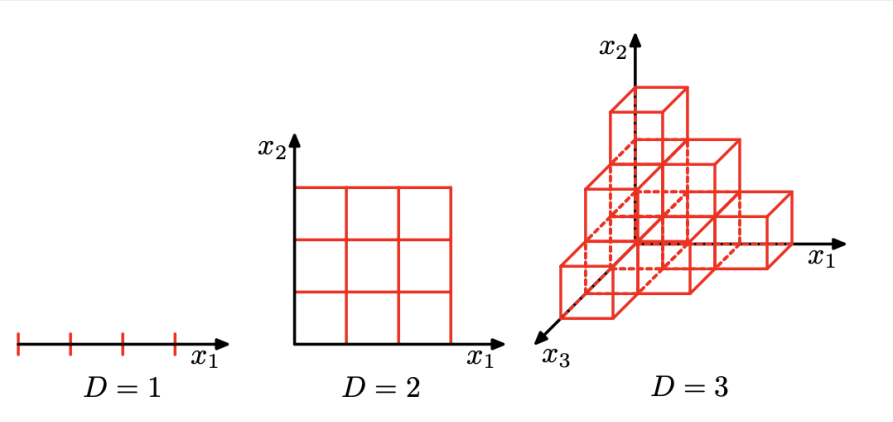
- Ability to fit a wide variety of functions:
  - Underfitting.
  - Overfitting.
- Another Tradeoff: Flexibility (Prediction Accuracy) and Interpretability.

Other Relevant Concepts

- **Cross-Validation:** resampling method that uses different portions of the data to test and train a model on different iterations.



- **Curse of Dimensionality:** increasing data dimensions and its explosive tendencies. We would need an exponentially large quantity of training data (and computational power) to explore the high-dimensional feature space.

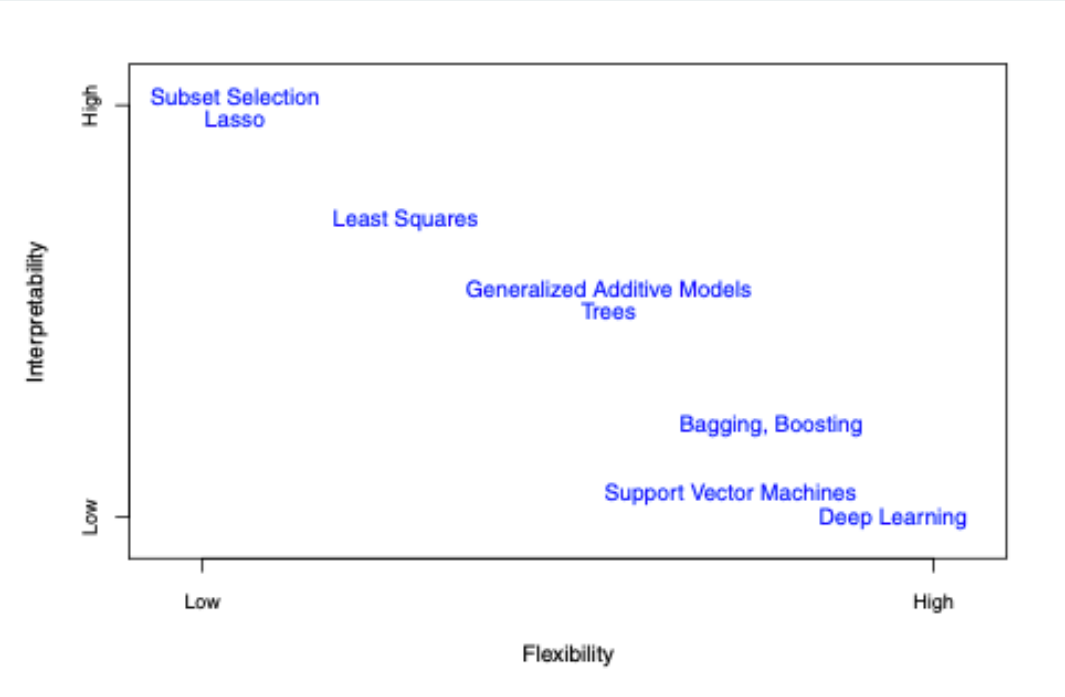


2/ Algorithms

Algorithms

- **No Free Lunch Theorem:** averaged over all possible DGPs, every algorithm has the same error rate when classifying previously unobserved points.
- No ML algorith is universally any better than any other.
- Best performance: capacity is appropriate for the true complexity of the task and the amount of training data.

Algorithms


Subset Selection
Lasso
Least Squares
Generalized Additive Models
Trees
Bagging, Boosting
Support Vector Machines
Deep Learning
High
Low
Interpretability
Low
High
Flexibility

Regularization

- Any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.
- We can regularize a model that learns a function f(x) by adding a penalty (regularizer) to the loss function (controls model complexity $\longrightarrow$ overfitting).

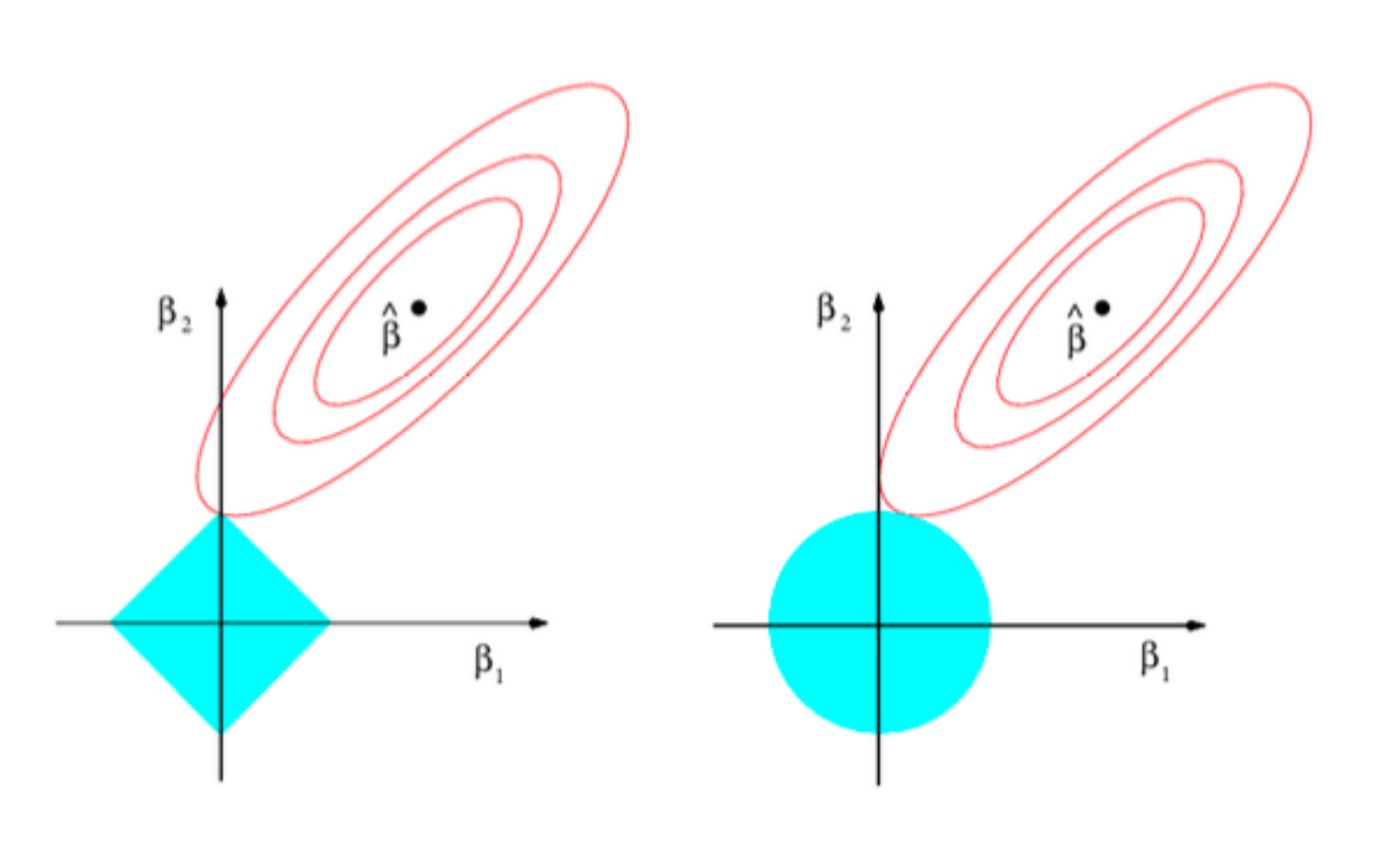
$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \leq s \tag{6.8}$$

and

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \leq s, \tag{6.9}$$

Regularization

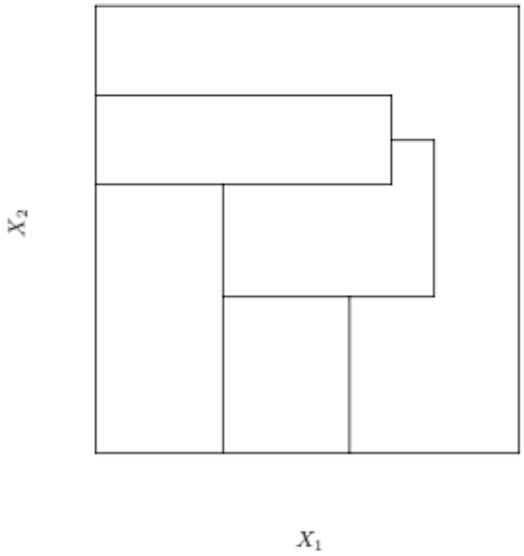
- Ridge, LASSO, Elastic Net.
- LASSO for variable selection ($\beta_j^2 \longrightarrow |\beta_j|$).

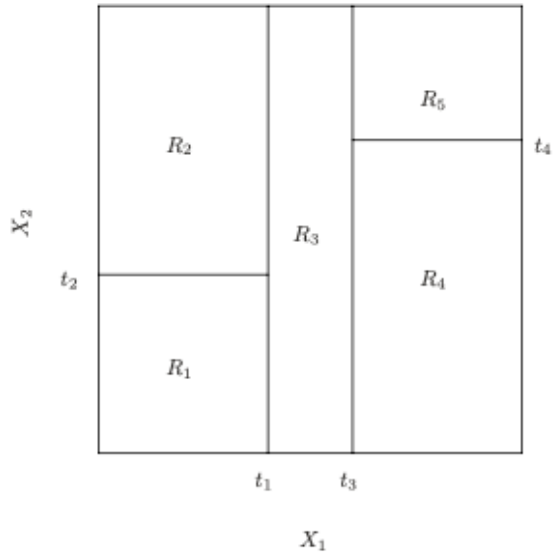


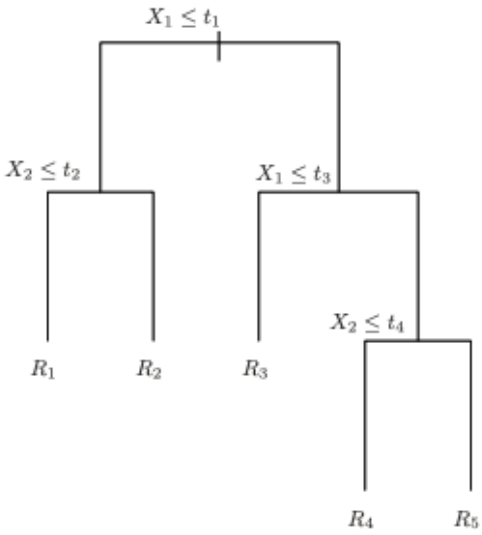
Non-Linear (more Flexible) Algorithms

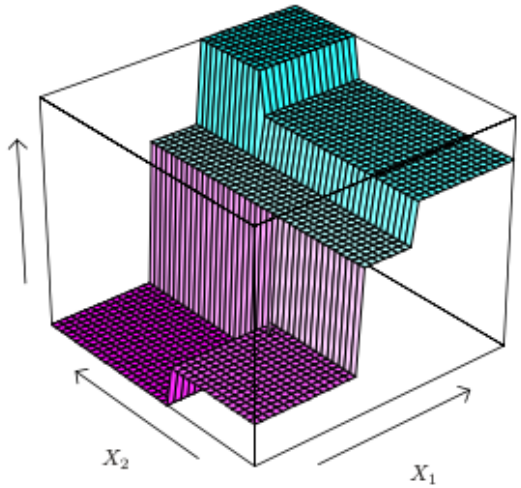
- GAMs.
- Trees
    - Bagging (Bootstrap Aggregation).
    - Random Forests: random sample of predictors (decorrelates the trees).
    - Boosting: sequential learning

Non-Linear (more Flexible) Algorithms









Boosting Algorithms

Algorithm 8.2 *Boosting for Regression Trees*

1. Set $\hat{f}(x) = 0$ and $r_i = y_i$ for all i in the training set.

2. For $b = 1, 2, \ldots, B$, repeat:

   (a) Fit a tree $\hat{f}^b$ with d splits ($d+1$ terminal nodes) to the training data (X, r).

   (b) Update $\hat{f}$ by adding in a shrunken version of the new tree:

   $$\hat{f}(x) \leftarrow \hat{f}(x) + \lambda \hat{f}^b(x). \tag{8.10}$$

   (c) Update the residuals,

   $$r_i \leftarrow r_i - \lambda \hat{f}^b(x_i). \tag{8.11}$$

3. Output the boosted model,

$$\hat{f}(x) = \sum_{b=1}^{B} \lambda \hat{f}^b(x). \tag{8.12}$$