Machine Learning for Education Policies

Flemish Ministry of Education, February 20th

Falco J. Bargagli-Stoffi, Kristof De Witte

KU Leuven

SONO

STEUNPUNT
ONDERWIJS ONDERZOEK

Table of contents

1. General introduction to machine learning

2. Supervised Machine Learning in a Nutshell

3. Classification and Regression Trees

   CART for prediction

   CART for causal inference

4. Random forests

Intro

General introduction to machine learning (ML)

ML is the sub-field of computer science that gives computers the ability to learn without being explicitly programmed, ***Arthur Samuel, 1959***

General introduction to machine learning (ML)

> *There are two cultures in the use of statistical modeling to reach conclusions from data. One assumes that the data are generated by a given stochastic data model. The other uses algorithmic models and treats the data mechanism as unknown. [...] If our goal as a field is to use data to solve problems, then we need to move away from exclusive dependence on data models and adopt a diverse set of tools,* ***Leo Breiman, 2001***

Key features of ML

- ML explores the study and construction of algorithms that can **learn** from and **make predictions** on data
- ML algorithms **overcome following strictly static program instructions**
- ML models do not make any assumptions about the data generating model (**model free**)
- The algorithms build a mathematical model from a set of data **(data-driven models)**

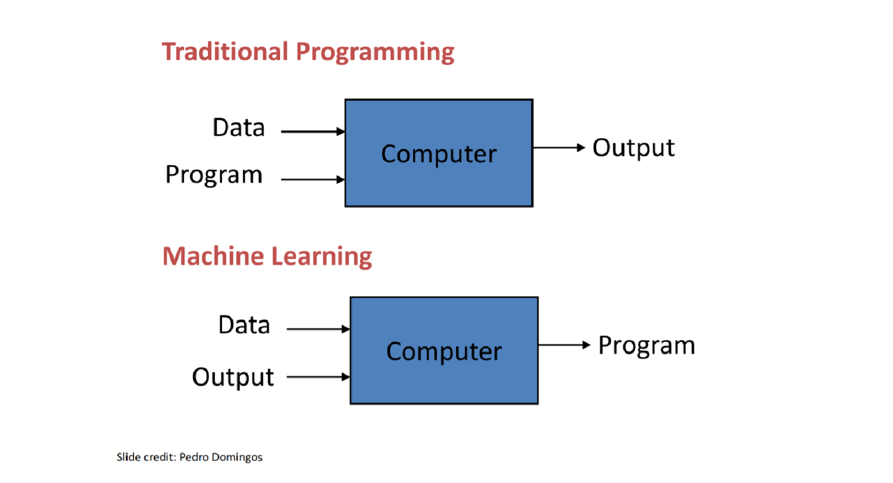
What is learning?

> *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E,* ***Tom M. Mitchell****, 1997*

- *E*: experience is a set of inputs, colloquially a data set
- *T*: the class of tasks can be predictive, clustering, dimensionality reduction, anomaly detection, etc.
- *P*: performance measure that varies based on the task being tackled

> *E.g. Use data on students' SES, background and abilities to predict partially unobserved financial literacy scores*

Inductive algorithms



Traditional Programming

Data → Computer

Program → Computer

Computer → Output

Machine Learning

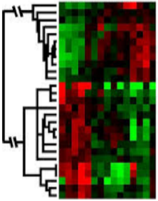
Data → Computer

Output → Computer

Computer → Program

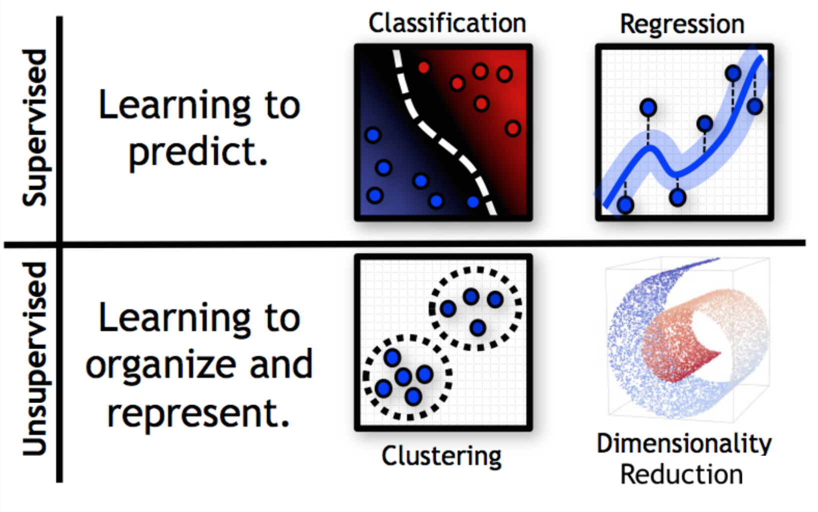
Slide credit: Pedro Domingos

When is ML used?



- Humans expertise does not exist (**navigating on Mars**)
- Humans can’t explain their expertise (**speech recognition**)
- Models are based on huge amount of data (**genomics**)
- Models must be customized (**targeted policies**)

Supervised vs Unsupervised ML



Learning to predict.

Learning to organize and represent.

ML in social sciences

- Following Mullainathan and Speiss (2017 JEP) four main branches of applications:
  1. ML for **causal inference** (SL)
  2. ML for **policy prediction** (SL)
  3. ML to **test theory** (SL)
  4. ML for **creation of new data sources** (mostly UL)

- The focus will be on (1), (2)

- A brief overview on the some packages and functionalities for ML in **R** will be provided

Supervised Machine Learning in a Nutshell

Supervised Machine Learning (SML)

- The common denominator of SML algorithms is that they take an information set $\mathsf{X}_{N \times P}$ and map it to a vector of outputs y
- The functional form of this relationship is very **flexible** and gets updated by evaluating a loss function in two steps:
  1. pick the best **loss-minimizing function** $f(\cdot)$:

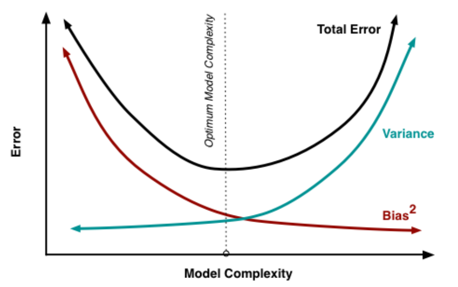
$$argmin \sum_{i=1}^{N} L(f(x_i), y_i) \quad over \quad f(\cdot) \in F \qquad s.t. \qquad R(f(\cdot)) \leq c$$

  2. estimate the optimal level of complexity using empirical tuning through **cross-validation**

Why is ML tailored for prediction?

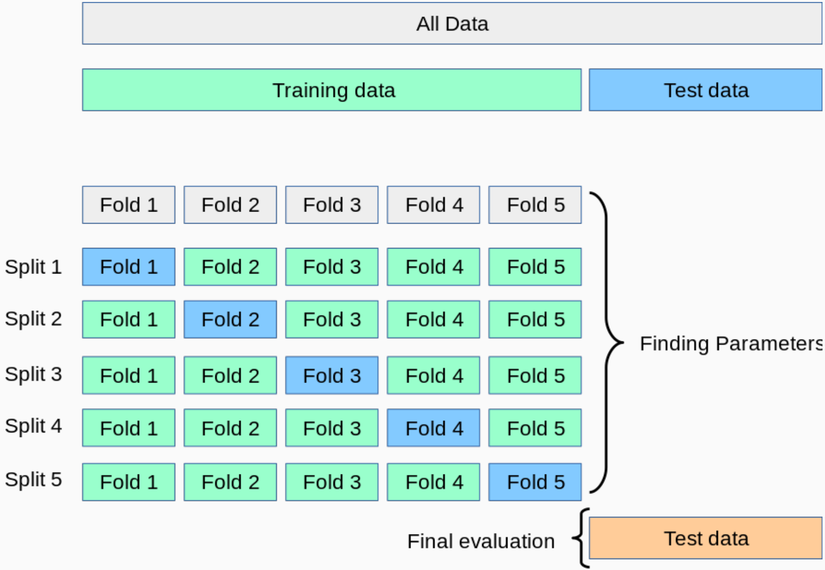
- Take a generic loss function such as the MSE of prediction:

$$E_{\mathcal{D}}[(y-\hat{f}(x))^2] = \underbrace{E_{\mathcal{D}}\left[\left(E_{\mathcal{D}}[\hat{y_0}] - \hat{f}(x)\right)^2\right]}_{\text{Variance}} + \underbrace{(E_{\mathcal{D}}[\hat{y_0}] - y)^2}_{\text{Bias}^2}$$



- By fixing the bias to be zero, the OLS regression rules out the possibility of this trade-off
- Impossibly to tune in a data-driven way the model with unbiased methods

How is model complexity chosen?



All Data

Training data

Test data

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

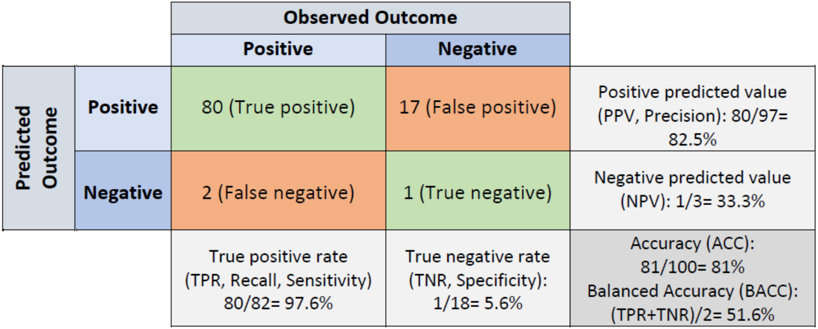
Finding Parameters

Final evaluation

Test data

Accuracy measures for discrete outcomes

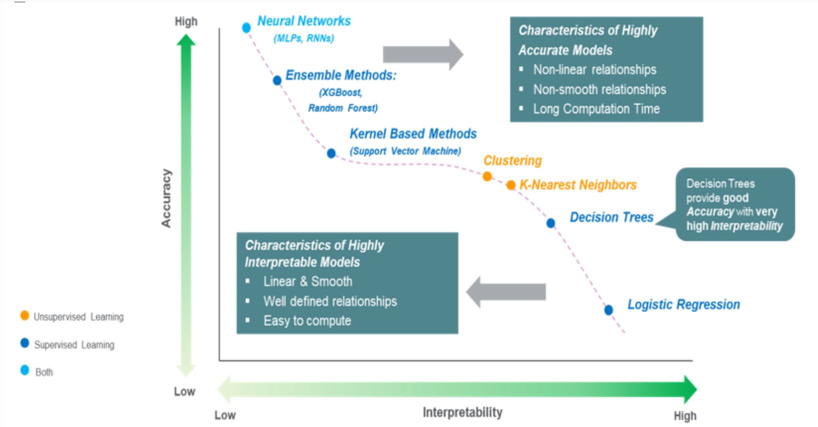
- Imagine the following scenario: 82 positive outcomes (e.g. high financial literacy score) and 18 negative outcomes (e.g. low financial literacy)



		Observed Outcome		
		Positive	Negative	
Predicted Outcome	Positive	80 (True positive)	17 (False positive)	Positive predicted value (PPV, Precision): 80/97= 82.5%
	Negative	2 (False negative)	1 (True negative)	Negative predicted value (NPV): 1/3= 33.3%
		True positive rate (TPR, Recall, Sensitivity) 80/82= 97.6%	True negative rate (TNR, Specificity): 1/18= 5.6%	Accuracy (ACC): 81/100= 81% Balanced Accuracy (BACC): (TPR+TNR)/2= 51.6%

$$F1-score = \frac{2 \cdot precision \cdot recall}{precision + recall}$$

Interpretability vs Accuracy



Characteristics of Highly Accurate Models

- Non-linear relationships
- Non-smooth relationships
- Long Computation Time

Characteristics of Highly Interpretable Models

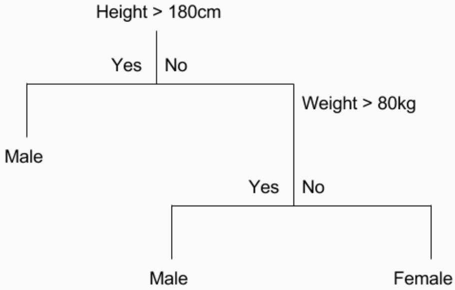
- Linear & Smooth
- Well defined relationships
- Easy to compute

Decision Trees provide **good *Accuracy*** with **very** high ***Interpretability***

Classification and Regression Trees

Decision Trees

Definition 1 (Decision Tree)
Decision tree learning uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves)



Classification and Regression Trees

Definition 2 (CART)
The CART methodology - introduced by Breiman, Friedman, Olshen and Stone in 1984 - is an algorithm for construction of binary trees, or trees where each node is splitted in only two branches

- **Classification tree** analysis is when the predicted outcome is the class to which the data belongs (e.g. *qualitative* outcomes)
- **Regression tree** analysis is when the predicted outcome can be considered a real number (e.g. *quantitative* outcomes)

CART is the basis for other algorithms that generate more complex trees. It is divided into two phases:

1. **Generation of the tree**
2. **Pruning of the tree**

1. Generation of a Regression tree

Generation of a tree:

1. Splitting of the predictor space (set of possible values for $X_1, X_2, ..., X_p$) into J distinct and non-overlapping regions, $R_1, R_2, ..., R_J$
2. Predict Y conditional on realization of X_j in each region R_j using the sample mean in that region

The construction of the regions $R_1, R_2, ..., R_J$ (high-dimensional rectangles) proceeds by finding boxes $R_1, R_2, ..., R_J$ that minimize the MSE given by:

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2$$

where $-\hat{y}_{R_j}$ is the mean response for the training obs withing the j-th box

Binary splitting

1. Computationally infeasible to consider every possible partition of feature space

2. *Top-down* approach for the *recursive binary splitting*

   a. Select a predictor X_j and a cut point s s.t.:

   $$R_1(j,s) = \{X|X_j \leq s\} \textit{ and } R_2(j,s) = \{X|X_j > s\}$$

   minimizes:

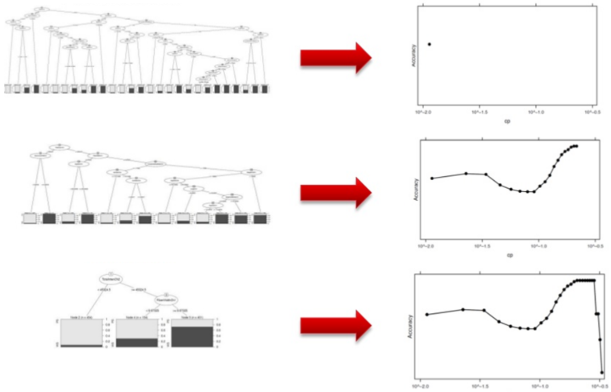
   $$\sum_{i:x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{i:x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2$$

   b. Repeat the process onto the two previously identified regions, so to minimize the MSE more

   c. Do it for all predictors and then choose the predictor and cut-point such that the resulting tree has the lowest MSE

2. Pruning of the tree (1)

- Too complex trees lead to data overfitting



2. Pruning of the tree (2)

- Two ways out:
    1. Split until the decrease in the MSE exceeds some threshold
    2. Grow a very large tree $\mathbb{T}$ and then prune it back to obtain a sub-tree

- This second strategy is implemented by minimizing:

$$\sum_{m=1}^{|\mathbb{T}|} \sum_{i:x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha|\mathbb{T}|$$

where $|\mathbb{T}|$ indicates the number of nodes of the tree $\mathbb{T}$, R_m is the rectangle corresponding to the m-th terminal node and α is a non-negative tuning parameter chosen by Cross-Validation

Classification Trees

- Classification Trees are similar to Regression Trees except they are used to predict a qualitative response
- The main difference is that instead of minimizing the MSE it is used the Classification Error Rate

$$MSE \rightarrow CER$$

- CER is the fraction of training obs. in a region that do not belong to the most common class

$$CER = 1 - \max_{k}(\hat{p}_{m,k})$$

where $\hat{p}_{m,k}$ represents the proportion of training obs. in the m-th region that are from the k-th class

Impurity measure: Entropy and Information Gain

- Entropy and Information Gain
  - Definition: degree of disorder of our dataset Ω: if we define by F_1 and F_2 the fraction of observations Ω classified with “1” and “2”, the entropy of the entire system S is defined as the following function $H(S)$:

    $$H(S) = -F_1 log F_1 - F_2 log F_2$$

  - Respect to the J subclasses entropy is defined as:

    $$H(S) = -\sum_{j=1}^{J} F_j log F_j$$

- The concept of information gain is a formalization of the entropic gain obtained through a partition of the data:

  $$G = H(S) - H(S, s) \text{ where } H(S, s) = H'(S)$$

Pros and Cons of CART

- Strengths and weaknesses of the CART methodology

- **Pros**:
  1. CART results are invariant under monotone transformations of the independent variables;
  2. CART can use the data set with a complex structure have been developed to be able to detect the dominant structures of the data;
  3. CART are extremely robust to outliers;
  4. CART can use linear combinations of variables to make the split: no need to *discretize* continuous variables

- **Cons**:
  1. We don't use all the data (cross-validation);
  2. Every time our algorithm chooses a split, it chooses the best split in that exact moment (no bigger picture) → *greedy algorithm*

Application in R

CART to predict students with low financial literacy scores using PISA data

Machine Learning and Causality

Using CART to estimate heterogeneous causal effect

Machine Learning and Causality

- *Econometrics/ Statistics/ Social Science*
    - Formal theory of causality
        - Potential outcomes methods (Rubin) maps onto economic approaches
    - Well-developed and widely used tools for estimation and inference of causal effect in experimental and observational studies
        - Used by social science, policy-makers, development organizations, medicine, business, experimentation
    - Weaknesses
        - Non-parametric approaches fail with many covariates
        - Model selection unprincipled

Motivations

- Experiments and Data-Mining
    - Concerns about ex-post “data-mining”
        - In medicine, scholars are required to pre-specify analysis plan (similar in economic field experiments)
- How is it possible to deal with sets of treatment effects among subsets of the entire population?
- Estimate of treatment effect heterogeneity needed for optimal decision-making

Definition 3 (Athey and Imbens, 2015; 2016)

1. Estimating heterogeneity by features in causal effects in experimental or observational studies
2. Conduct inference about the magnitude of the differences in the treatment effects across subsets of the population

Causal Inference Framework

- Causal inference in observational studies:
  - As we saw previously, assuming unconfoundedness to hold, we can treat observations as having come from a randomized experiment
  - Therefore we can define the **conditional average treatment effect (CATE)** as follows:

$$\tau(x) = E[Y_i(1) - Y_i(0)|X_i = x]$$

  - The population average treatment effect then is:

$$\tau^p = E[Y_i(1) - Y_i(0)] = E[\tau(X_i)]$$

Why is CATE important?

- There are a variety of reasons that researchers wish to conduct estimation and inference on $\tau(x)$:

1. It my be used to assign future units to their optimal treatment (in presence of different levels of the treatment):

$$W_i^{opt} = max\, \tau(X_i)$$

2. If we don't pre-specify the sub-populations it can be the case that the overall effect is negative, but it can be positive on subpopulations, then:

$$W_i^{PTE} = \mathbf{1}_{\tau(X_i)\geq 0}$$

e.g.: treatment is a drug → prescribe it just to those who benefit from it

Using Trees to Estimate Causal Effects

Athey and Imbens (2015; 2016) propose 3 different approaches:

- Approach I: Analyze two groups separately:
  - Estimate $\hat{\mu}(1, x)$ using dataset where W_i=1
  - Estimate $\hat{\mu}(0, x)$ using dataset where W_i=0
  - Preform within group cross-validation to choose tuning parameters
  - Predict $\hat{\tau} = \hat{\mu}(1, x) - \hat{\mu}(0, x)$

- Approach II: Estimate $\mu(w, x)$ using just one tree:
  - Estimate $\hat{\mu}(1, x)$ and $\hat{\mu}(0, x)$ using just one tree
  - Preform within tree cross-validation to choose tuning parameters
  - Predict $\hat{\tau} = \hat{\mu}(1, x) - \hat{\mu}(0, x)$
  - Estimate is zero for x where tree does not split on w

The CATE Transformation of the Outcome

- The authors’ goal is to develop an algorithm that generally leads to an accurate approximation of $\hat{\tau}$ the Conditional Average Treatment Effect.

1. Ideally we would measure the quality of the approximation in terms of goodness of fit using the MSE:

$$Q^{infeas} = \frac{1}{N}\sum_{i=1}^{N}(Y_i(1) - Y_i(0) - \hat{\tau}(X_i))^2$$

2. We can address this problem of infeasibiliy by transforming the outcome using the treatment indicator W_i and $e(X)$:

$$Y_i^* = Y_i^{obs} \cdot \frac{W_i - e(X_i)}{(1 - e(X_i)) \cdot e(X_i)}$$

3. Then:

$$E[Y_i^*|X_I = x] = \tau(x)$$

How to estimate the In-Sample Goodness of fit?

- The ideal goodness of fit measure would be:

$$Q^{infeas}(\hat{\tau}) = \mathbb{E}[(\tau_i - \hat{\tau}(X_i))^2].$$

- A useful proxy that can be used for the goodness of fit measure is:

$$\mathbb{E}[\tau_i^2 | X_i \in S_j] = \frac{1}{N} \sum_i \hat{\tau}(x_i)^2.$$

  This leads to our In-sample goodness of fit function:

$$Q^{is} = -\frac{1}{N} \sum_i \hat{\tau}(x_i)^2.$$

Transformed Outcome Tree Model

- Approach 3:
  1. Model and Estimation
     - Model Type: Tree structure
     - Estimator $\hat{\tau}_i^{TOT}$: sample average treatment effect within leaf
  2. Criterion function (for fixed tuning parameter λ)
     - In-sample Goodness-of-fit function:

$$Q^{is} = -MSE = -\frac{1}{N}\sum_{i=1}^{N}(\hat{\tau}_i^{TOT})^2$$

     - Structure and use of criterion:

$$Q^{crit} = Q^{is} - \lambda \times leaves$$

     - Select member of set of candidate estimators that maximizes Q^{crit}, given λ
  3. Cross-validation approach
     - Out-of-Sample Goodness-of-fit function:

$$Q^{oos} = -MSE = -\frac{1}{N}\sum_{i=1}^{N}(\hat{\tau}_i^{TOT} - Y_i^*)^2$$

     - Approach: select tuning parameter λ with highest Q^{os}

Critique to the TOT approach

- Transformation of the Outcome in a randomized set-up:

$$Y_i^* = Y_i^{obs} \cdot \frac{W_i - p}{(1-p) \cdot p} = \begin{cases} \frac{1}{p} \cdot Y_i^{obs} & if \; W_i = 1 \\ -\frac{1}{1-p} \cdot Y_i^{obs} & if \; W_i = 0 \end{cases}$$

- Within a leaf the sample average of Y_i^* is not the most efficient estimator of treatment effect

- The proportion of treated units within the leaf is not the same as the overall sample proportion

- We use a weighted estimator similar to the Hirano, Imbens and Ridder (2003) estimator

Causal Tree Approach

- In details the Treatment Effect in a generic leaf $\mathbb{X}_j$ is:

$$\tau^{CT}(X_i) = \frac{\sum_{j:X_j \in \mathbb{X}_j} Y_i^{obs} \cdot \frac{W_i}{\hat{e}(X_i)}}{\sum_{j:X_j \in \mathbb{X}_j} \frac{W_i}{\hat{e}(X_i)}} - \frac{\sum_{j:X_j \in \mathbb{X}_j} Y_i^{obs} \cdot \frac{(1-W_i)}{(1-\hat{e}(X_i))}}{\sum_{j:X_j \in \mathbb{X}_j} \frac{(1-W_i)}{(1-\hat{e}(X_i))}}$$

- This estimator is a consistent estimator of:

$$\tau_{\mathbb{X}_j} = \mathbb{E}[Y_i(1) - Y_i(0) | X_i \in \mathbb{X}_j]$$

- The variance can be estimated the Neyman estimator:

$$\hat{\mathbb{V}}_{Neyman} = \frac{s_t^2}{N_t} + \frac{s_c^2}{N_c}$$

These two quantities can be estimated as:

$$s_{t,j}^{te,2} = \frac{1}{N_t - 1} \sum_{i:W_i=1} (Y_i(1) - \overline{Y}_t^{obs})^2 = \frac{1}{N_t - 1} \sum_{i:W_i=1} (Y_i - \overline{Y}_t^{obs})^2$$

$$s_{c,j}^{te,2} = \frac{1}{N_c - 1} \sum_{i:W_i=0} (Y_i(0) - \overline{Y}_c^{obs})^2 = \frac{1}{N_c - 1} \sum_{i:W_i=0} (Y_i - \overline{Y}_c^{obs})^2$$

Attractive features of Causal trees

1. Can easily separate tree construction from treatment effect estimation

2. Tree constructed on training sample is independent of sampling variation in the test sample

3. Holding tree from training sample fixed, can use standard methods to conduct inference within each leaf of the tree on test sample

4. Can use any valid method for treatment effect estimation, not just the methods used in training

5. Simulations run by the authors show that the Causal Tree Algorithm over-performs the ST, TT and TOT approaches

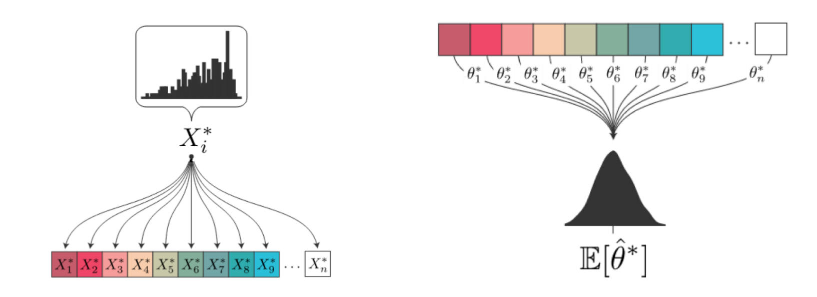
Extra on ML for causal inference

- Causal Inference with random forests in randomized experiments (Wager and Athey, 2018) and in observational studies (Athey et al., 2019)
- Heterogeneous effects in IV settings (Guber and Farbmacher, 2018; Bargagli-Stoffi et al. 2019; Johnson et al., 2019)
- Heterogeneous effects with network interference (Bargagli-Stoffi et al., 2020)
- Interpretable inference (Lee et al., 2018; Bargagli-Stoffi et al., 2020)
- Personalized treatment (Kallus, 2017; 2018)

Random forests

Random Forest

- RF: A Random Forest (Breiman, 2001) is a collection of fully grown CART. A Random Forest is a substantial transformation of the bagging method by introducing a collection of trees uncorrelated with each other.
  1. Bagging;
  2. Independence.



1. Bagging

- The techniques called bagging take shape **bootstrap** by the method. The term itself comes from *bootstrap aggregation*.
    1. Bradley Efron (1979)
    2. Sample **X** of dimension n
    3. Estimate the parameter θ by simulating **B** samples of the same abundance, obtained sampling by assuming **X** as if for the overpopulation of reference
- $\hat{f}(X) \mapsto B$ samples $X_1^*, \ldots, X_B^* \mapsto t(X_1^*), \ldots, t(X_B^*)$
- **Bagging estimator**

$$\hat{t}_{bag}(X) = \frac{1}{B} \sum_{b=1}^{B} t(X_b^*)$$

2. Independence

- If we develop the *bagged* variance estimator $\hat{t}^B_{bag}(X)$:

$$
\begin{aligned}
Var(\hat{t}^B_{bag}(X)) &= ar\left(\frac{1}{B}\sum_{i=1}^{B} t(X_i)\right) \\
&= \sigma^2 \cdot \rho + \sigma^2 \frac{1-\rho}{B}
\end{aligned}
$$

.

- The idea behind the Random Forest is that we can significantly increase the benefits of *bagging* through a reduction in the correlation of trees
- Random selection mechanism to select m variables between the p total splitting variables

Random Forest Algorithm for Regression and Classification

1. For b that goes from 1 to B:
   - Draw a sample $\mathbf{Z}^*$ of N units through the bootstrap method from our starting datasets $\boldsymbol{\Omega}$;
   - Grow a tree of the random forest T_b repeating, recursively, the following steps for each terminal node of the tree until you reach the minimum number of nodes n_{min}
     - 1.i Select m randomly variables between the p available variables;
     - 1.ii Choose the best combination of variables used for the split between the m variables;
     - 1.iii Splitting the node into 2 children nodes.
2. Through the output of all the trees $\{T_b\}^B$ we can proceed as follows:
   - *Regression*: $\hat{t}_{rf}^{B}(X)=\frac{1}{B}\sum_{b=1}^{B} T_b(x)$;
   - *Classification*: we can think about prediction of the k-th class and the b-th tree of the random forest as $\hat{C}_{rf}^{B,K}(x)$. Where: $\hat{C}_{rf}^{B,K}(x)=\textit{majority vote }\{\hat{C}_b(x)^{B,K}\}$

Pros and Cons of RF

- Pros:
  1. There is no need to rework or transform the data before building the model. Data must not be normalized and this approach is particularly robust to outliers;
  2. If we have a lot of input variables, we must not do any variable selection in a prior stage to construction of the model because it will be the same Random Forest to identify what are the most useful variables.;
  3. Many trees are built through random mechanisms and therefore every tree is actually an independent model that does not bring the model to an overfitting.
- Cons:
  1. Strong data dependency
  2. Lower interpretability
  3. Higher computational costs

Extra on RF

- Causal forests (Wager and Athey, 2019)
- Bayesian Forests (Chipman et al., 2010)
- Sensitivity of predictions analysis (Bargagli-Stoffi et al., 2020)
- Overlap and predictions (Bargagli-Stoffi & De Witte, 2020)

Main References

- Athey, S., Imbens, G. (2016). *Recursive partitioning for heterogeneous causal effects*. Proceedings of the National Academy of Sciences, 113(27), 7353-7360.
- Breiman, L. (2001). *Random forests*. Machine learning, 45(1), 5-32.
- Breiman, L., Friedman, J., Stone, C. J., Olshen, R. A. (1984). *Classification and Regression Trees*. CRC Press.
- Friedman, J., Hastie, T., Tibshirani, R. (2001). *The elements of statistical learning* (Vol. 1, No. 10). New York: Springer series in statistics.
- Efron, B., Hastie, T. (2016). *Computer age statistical inference* (Vol. 5). Cambridge University Press.

Application in R

Using RF to predict students with low financial literacy scores using PISA data

Application in R

Using CT to detect the heterogeneous effect of additional funding to schools