CS 446/ECE 449: Machine Learning

A. G. Schwing

University of Illinois at Urbana-Champaign, 2020

L11: Boosting, Classification and Regression Trees, Ensembles and Regularization/Cross-validation

Goals of this lecture

- Getting to know Boosting
- Getting to know Classification and Regression Trees
- Getting to know Random Forests
- Getting to know Ensembles
- Getting to know Cross-Validation

Reading material:

- Shai Shalev-Shwartz & Shai Ben-David, Understanding Machine Learning: From Theory to Algorithms, Chapter 4

Recap:

Focus on regression:

- Linear regression
- Logistic regression
- Non-linear logistic regression

Other machine learning frameworks exist

Boosting:

Kearns and Valiant (1988, 1989):

> "Can a set of weak learners create a single strong learner?"

Rob Schapire (1990):

"Yes!"

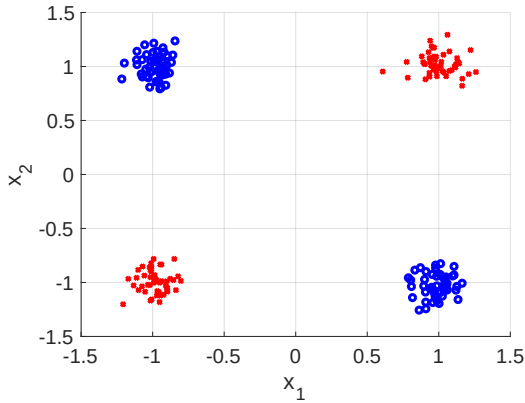
Adaboost

Given:

- Dataset $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}$ with $x^{(i)} \in \mathcal{X}$, $y^{(i)} \in \mathcal{Y} = \{-1, 1\}$
- Set of 'weak' classifiers $\mathcal{F} = \{f_t\}$ with $f_t : \mathcal{X} \to \mathcal{Y}$

Example of a 'weak' classifier: (decision stump)

$$f_t(x) = \begin{cases} -1 & \text{if } x_{\text{Index}(t)} < \text{Threshold}(t) \\ 1 & \text{otherwise} \end{cases}$$



Output: classifier

$$F_T(x) = \sum_{t=1}^{T} \alpha_t f_t(x)$$

no need to optimize threshold

How to build F_T?

Build F_T iteratively:

$$F_t(x) = F_{t-1}(x) + \alpha_t f_t(x)$$

Question:

Which $f_t \in \mathcal{F}$ and which α_t to use given F_{t-1}?

Let's define the error of F_T to be

$$E(F_T) = \sum_i \exp\left(-y^{(i)} F_T(x^{(i)})\right)$$

Let's also define factors $\gamma_t^{(i)}$:

$$\gamma_1^{(i)} = 1 \; \forall \; i; \qquad \gamma_t^{(i)} = \exp\left(-y^{(i)} F_{t-1}(x^{(i)})\right)$$

Consequently

$$\begin{aligned} E(F_t) &= \sum_i \gamma_t^{(i)} \exp\left(-y^{(i)} \alpha_t f_t(x^{(i)})\right) = \sum_{i: y^{(i)} = f_t(x^{(i)})} \gamma_t^{(i)} e^{-\alpha_t} + \sum_{i: y^{(i)} \neq f_t(x^{(i)})} \gamma_t^{(i)} e^{\alpha_t} \\ &= \sum_i \gamma_t^{(i)} e^{-\alpha_t} + \sum_{i: y^{(i)} \neq f_t(x^{(i)})} \gamma_t^{(i)} \left(e^{\alpha_t} - e^{-\alpha_t}\right) \end{aligned}$$

Therefore:

Pick f_t with lowest weighted error $\sum_{i: y^{(i)} \neq f_t(x^{(i)})} \gamma_t^{(i)}$

How to pick α_t?

$$\frac{dE(F_t)}{d\alpha_t} = 0$$

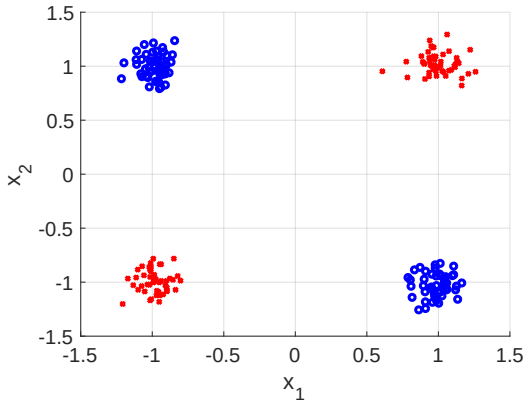
Result:

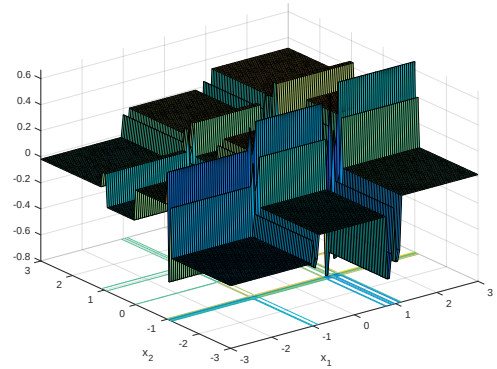
$$\alpha_t = \frac{1}{2} \ln \left(\frac{1 - \epsilon_t}{\epsilon_t} \right) \qquad \text{where} \quad \epsilon_t = \left(\sum_{i: y^{(i)} \neq f_t(x^{(i)})} \gamma_t^{(i)} \right) / \left(\sum_i \gamma_t^{(i)} \right)$$

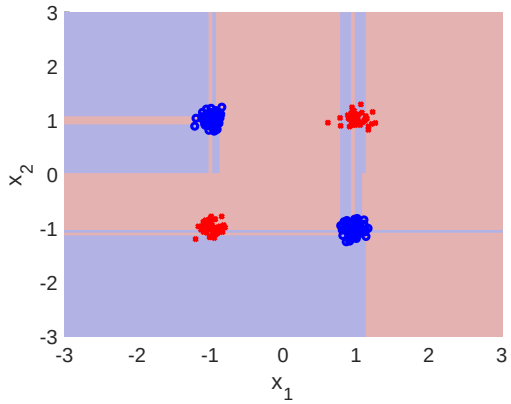
Algorithm: AdaBoost for binary classification:

- Initialize $\gamma_1^{(i)} = 1 \forall i \in \mathcal{D}$
- Iterate for $t = 1, \ldots, T$
  - Pick $f_t \in \mathcal{F}$ which minimizes weighted error $\sum_{i: y^{(i)} \neq f_t(x^{(i)})} \gamma_t^{(i)}$ (exhaustive search)
  - Calculate error rate ϵ_t and weight α_t
  - Improve F_{t-1} to $F_t = F_{t-1} + \alpha_t f_t$
  - Update $\gamma_{t+1}^{(i)} = \exp\left(-y^{(i)} F_t(x^{(i)})\right)$

Demo:



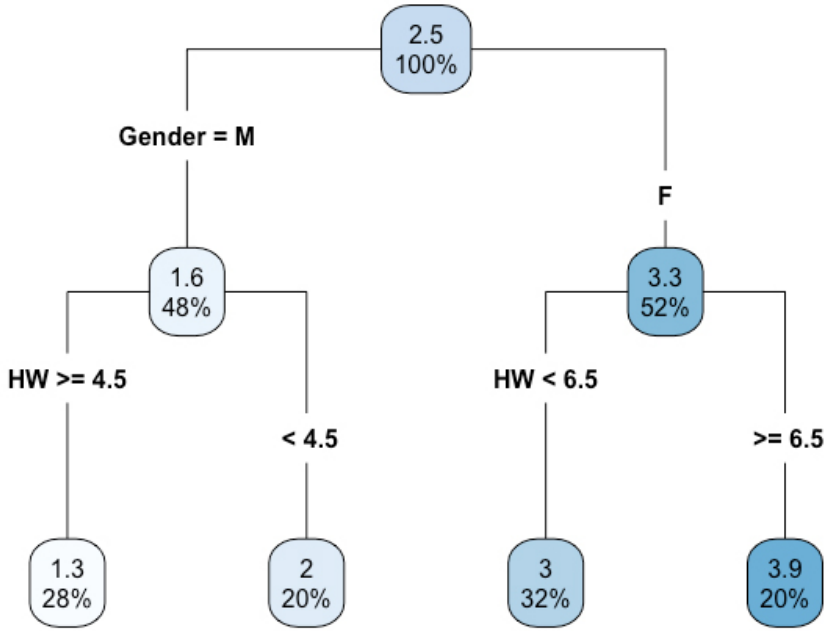




Training set classification accuracy: 72%

Code

Decision Tree Learning



Inference:

- Start at the root and follow the decisions
- The leaf node reveals the result

Question:

How to learn the decisions of such a tree?

Many specific decision tree learning algorithms:

- Iterative Dichotomiser 3 (ID3)
- C4.5 (successor of ID3)
- CART (classification and regression tree)
- ...

General algorithm flavor (top down):

- Choose a variable that “best” splits the set of data items
- Split the data according to the chosen rule, append two nodes and let them process their data
- Stop once the number of datapoints in a node is reasonably small and compute the leaf node statistics

The leaf node statistics are the classification result.

Metric for measuring "best" split: Information Gain (N child nodes, $f \in \mathcal{F}$ split function, $\mathcal{D}$ data at parent node, $\mathcal{D}_j$ data at j-th child)

$$IG(\mathcal{D}, f) = I(\mathcal{D}) - \sum_{j=1}^{N} \frac{|\mathcal{D}_j|}{|\mathcal{D}|} I(\mathcal{D}_j)$$

- Gini impurity

$$I(\mathcal{D}) = 1 - \sum_{c=1}^{C} p(c|\mathcal{D})^2$$

- Entropy

$$I(\mathcal{D}) = -\sum_{c=1}^{C} p(c|\mathcal{D}) \log p(c|\mathcal{D})$$

- Classification error

$$I(\mathcal{D}) = 1 - \max_{c \in \{1,\ldots,C\}} p(c|\mathcal{D})$$

Information Gain:

$$IG(\mathcal{D}, f) = I(\mathcal{D}) - \sum_{j=1}^{N} \frac{|\mathcal{D}_j|}{|\mathcal{D}|} I(\mathcal{D}_j)$$

Example:

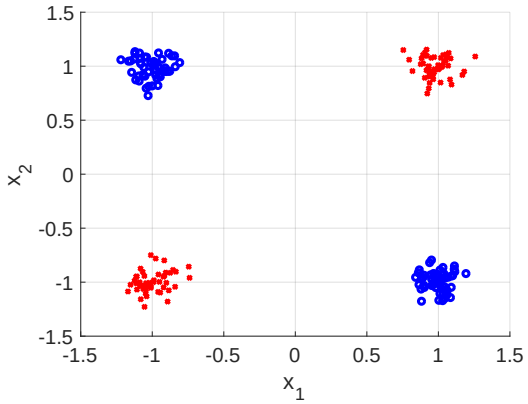
- Gini impurity:

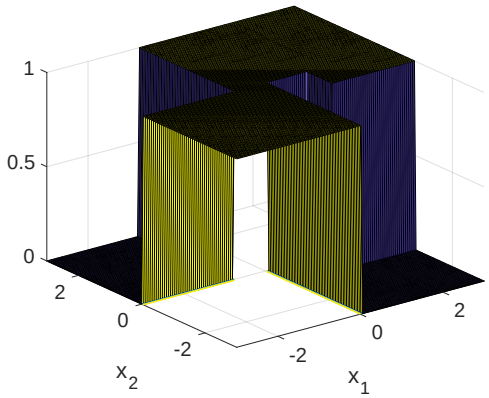
$$I(\mathcal{D}) = 1 - \sum_{c=1}^{C} p(c|\mathcal{D})^2$$

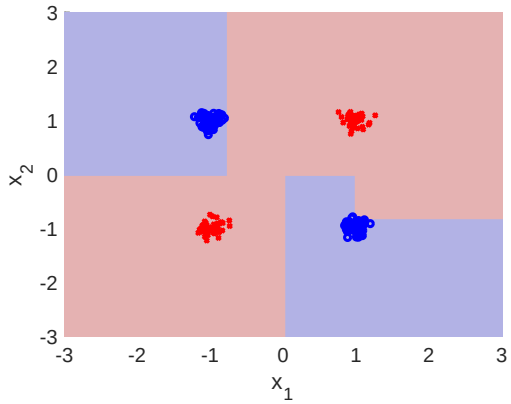
- $\mathcal{D}$: 10 examples of class 0 and 10 examples of class 1
- $\mathcal{D}_1$: 10 examples of class 0 and 0 examples of class 1
- $\mathcal{D}_2$: 0 examples of class 0 and 10 examples of class 1

$$1 - 0.5^2 - 0.5^2 - \frac{1}{2}(1 - 0^2 - 1^2) - \frac{1}{2}(1 - 1^2 - 0^2)$$

Demo:







Training set classification accuracy: 99.5%

Code

Classification Ensembles

- Train a variety of classification algorithms (same or different type)
- Average their result

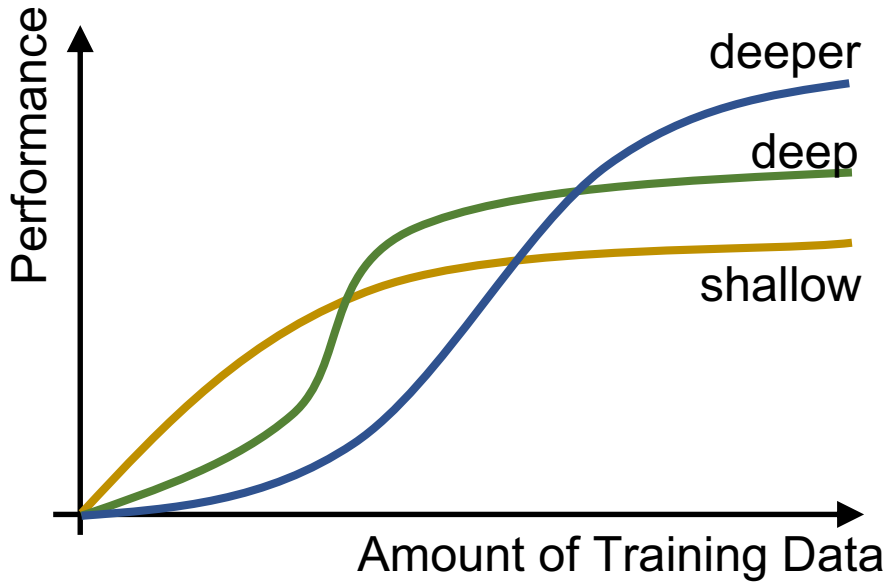
Example:

- Random Forest (parallel ensemble)
- Boosting (sequential ensemble)

Bagging (bootstrap aggregation):

$$F(x) = \frac{1}{M} \sum_{m=1}^{M} F_T^{(m)}(x)$$

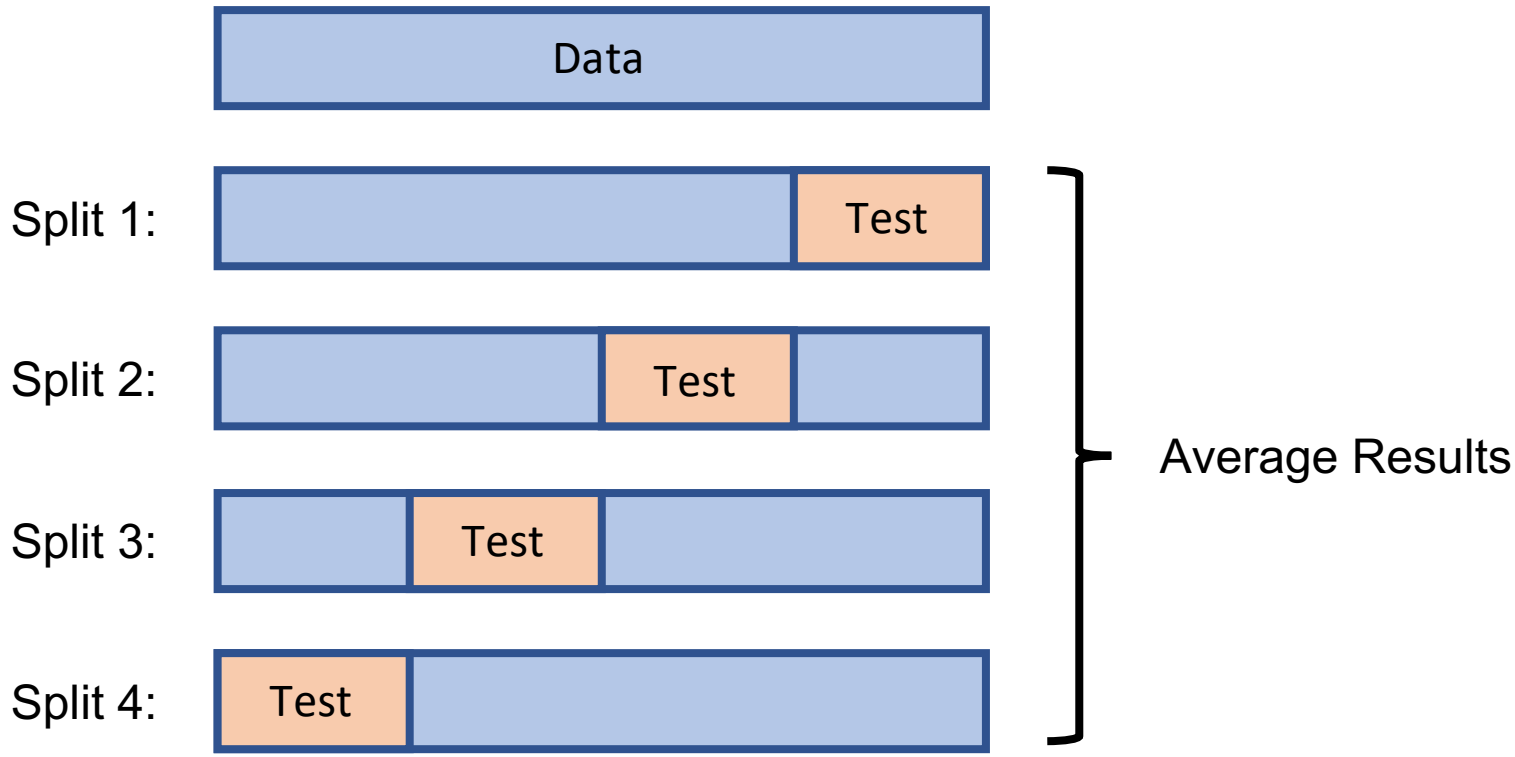
When to choose which method?



How to select hyper-parameters of machine learning models?

- Split your data into train/val/test set
- Choose parameters based on val set
- Report results on test set

4-fold Cross validation:



Quiz:

- What is Boosting?
- How do we construct decision trees?
- What are ensembles?

Important topics of this lecture

- Boosting
- Classification and regression trees
- Ensembles

Up next:

- Structured Models