Introduction to Machine Learning
Lecture 10

Mehryar Mohri
Courant Institute and Google Research
mohri@cims.nyu.edu
Decision Trees
Supervised Learning Problem

- **Training data**: sample drawn i.i.d. from set $X$ according to some distribution $D$, 
  \[ S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \in X \times Y, \]
- classification: $\text{Card}(Y) = k$.
- regression: $Y \subseteq \mathbb{R}$.

- **Problem**: find classifier $h : X \rightarrow Y$ in $H$ with small generalization error.
Advantages

- Interpretation: explain complex data, result easy to analyze and understand.
- Adaptation: easy to update to new data.
- Different types of variables: categorical, numerical.
- Monotone transformation invariance: measuring unit is not a concern.
- Dealing with missing labels.
- But: beware of interpretation!
Example - Playing Golf

Misclassification rates are indicated at each node.
Decision Trees

![Decision Tree Diagram]

- **Decision Trees**

- **X1 < a1**
  - YES
  - NO

- **X1 < a2**
  - YES
  - NO

- **X2 < a3**
  - YES
  - NO

- **X2 < a4**
  - YES
  - NO

- **R1**
  - YES
  - NO

- **R2**
  - YES
  - NO

- **R3**
  - YES
  - NO

- **R4**
  - YES
  - NO

- **R5**
  - YES
  - NO

- **X2**
  - R2
  - a4

- **a2**
  - R3
  - R1

- **a3**
  - R4
  - R5

- **X1**
  - a1
Different Types of Questions

- Decision trees
  - $X \in \{\text{blue, white, red}\}$: categorical questions.
  - $X \leq a$: continuous variables.

- Binary space partition (BSP) trees:
  - $\sum_{i=1}^{n} \alpha_i X_i \leq a$: partitioning with convex polyhedral regions.

- Sphere trees:
  - $\|X - a_0\| \leq a$: partitioning with pieces of spheres.
Prediction

- In each region $R_t$ (tree leaf):
  - **classification**: majority vote - ties broken arbitrarily.
    \[
    \hat{y}_t = \arg\max_{y \in Y} \left| \left\{ x_i \in R_t : i \in [1, m], y_i = y \right\} \right|.
    \]
  - **regression**: average value.
    \[
    \hat{y}_t = \frac{1}{|R_t|} \sum_{x_i \in R_t} y_i.
    \]

- for confident predictions, need enough points in each region.
Learning

How to build a decision tree from data:

• choose question, e.g., $x \leq 3$, yielding best purity.
• partition data into corresponding subsets.
• reiterate with resulting subsets.
• stop when regions are approximately pure.
Impurity Criteria - Classification

- Binary case: $p$ fraction of positive instances.
  - misclassification: $F(p) = \min(p, 1 - p)$.
  - entropy: $F(p) = -p \log_2(p) - (1 - p) \log_2(1 - p)$.
  - Gini index: $F(p) = 2p(1 - p)$.
Impurity Criteria - Regression

- Mean squared error:

\[ F(R) = \frac{1}{|R|} \sum_{x_i \in R} (y_i - \langle y \rangle)^2. \]

- Other similar \( L_p \) norm criteria.
Training

- **Problem:** general problem of determining partition with minimum empirical error is NP-hard.

- **Heuristics:** greedy algorithm.

  - for all $j \in [1, N], \theta \in \mathbb{R}$, $R^+(j, \theta) = \{x_i \in R: x_i[j] \geq \theta, i \in [1, m]\}$
  
  $R^-(j, \theta) = \{x_i \in R: x_i[j] < \theta, i \in [1, m]\}$.

### Decision-Trees ($S = ((x_1, y_1), \ldots, (x_m, y_m))$)

1. $P \leftarrow \{S\}$ \triangleright initial partition
2. for each region $R \in P$ such that $\text{Pred}(R)$ do
3.  $(j, \theta) \leftarrow \arg\min_{(j, \theta)} \text{error}(R^-(j, \theta)) + \text{error}(R^+(j, \theta))$
4.  $P \leftarrow P - R \cup \{R^-(j, \theta), R^+(j, \theta)\}$
5. return $P$
Overfitting

Problem: size of tree?
- tree must be large enough to fit the data.
- tree must be small enough not to overfit.
- minimizing training error or impurity does not help.

Theory: generalization bound.

\[ R(h) \leq \hat{R}(h) + O \left( \sqrt{\frac{\text{complexity measure}}{m}} \right) . \]
- minimize \( \text{impurity} + \alpha |\text{tree}| \).
Controlling Size of Tree

- Grow-then-prune strategy (CART):
  - create very large tree.
  - prune back according to some criterion.

- Pruning criteria:
  - $(\text{impurity} + \alpha |\text{tree}|)$.
  - $\alpha$ determined by cross-validation.
Categorical Variables

Problem: with $N$ possible unordered variables, e.g., color (blue, white, red), there are $2^{N-1} - 1$ possible partitions.

Solution (when only two possible outcomes): sort variables according to the number of 1s in each, e.g., white .9, red .45, blue .3. Split predictor as with ordered variables.
Missing Values

**Problem:** points \( x \) with missing values \( y \), due to:
- the proper measurement not taken,
- a source causing the absence of labels.

**Solution:**
- categorical case: create new category missing;
- use surrogate variables: use only those variables that are available for a split.
Instability

- **Problem**: high variance
  - small changes in the data may lead to very different splits,
  - price to pay for the hierarchical nature of decision trees,
  - more stable criteria could be used.
Decision Tree Tools

Most commonly used tools for learning decision trees:

- **CART** (classification and regression tree) (Breiman et al., 1984).

- **C4.5** (Quinlan, 1986, 1993) and **C5.0** (RuleQuest Research) a commercial system.

Differences: minor between latest versions.
Summary

- Straightforward to train.
- Easily interpretable (modulo instability).
- Often not best results in practice.

Boosting decision trees (next lecture).
References


