Escuela Politécnica de Ingeniería de Gijón EPI Universidad de Oviedo
Grado en Ciencia e Ingeniería de Datos

AA1 Aprendizaje Automático I

T8: Ensemble learning

Antonio Bahamonde
Departamento de Informática

Ensemble Methods

Nandita Bhaskhar

content adapted from
(a) Hastie, Tibshirani & Friedmanand
(b) Protopapas, Rader & Pan
(c) Jason Brownlee

May 27th, 2022

Cs229 Stanford University

Contents

Decision Trees Recap

Pros

- Can handle large datasets
- Can handle mixed predictors (continuous, discrete, qualitative)
- Can ignore redundant variables
- Can easily handle missing data
- Easy to interpret if small

Cons

- Prediction performance is poor
- Does not generalize well
- Large trees are hard to interpret

Ensemble Methods: Intro

- Methods to improve the performance of weak learners
- Weak learners (e.g., classification trees) don't perform that well
- What do we do??
- **Wisdom of the crowds!**

Ensemble Methods: Intro

- Wisdom of the crowds!
- Shift responsibility from 1 weak learner to an "ensemble" of such weak learners
- Set of weak learners are combined to form a strong learner with better performance than any of them individually

Ensemble Methods: Intro

- A single decision tree often produces noisy / weak classifiers
- They DON'T generalize well
- But they are super fast, adaptive and robust!
- Solution: Let's learn multiple trees!
- How to ensure they don't all just learn the same thing??
- ~~TRIVIAL Solution~~

Bagging

- Bagging (Breiman, 1996)
- Bootstrap Aggregating: to ensure lower variance
- Bootstrap sampling: get different splits / subsets of the data
- Aggregating: majority voting or averaging

Bagging

- Averages a given procedure over many samples to reduce its variance
- Multiple realizations of the data (via multiple samples) →
  - calculate predictions multiple times →
  - average the predictions →
  - more certain estimations (lesser variance)

Bagging

- Let $f(x)$ be the classifier and let b be a sample set from data

$$\hat{f}_{agg}(x) = \frac{1}{B}\sum_{b=1}^{B} f_b(x)$$

Or

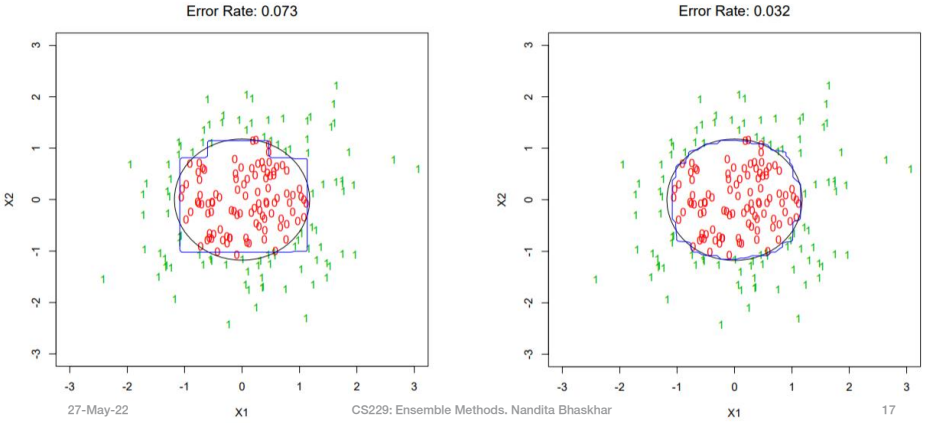
$$\hat{f}_{agg}(x) = \text{Majority Vote } \{f_b(x)\}_{b=1}^{B}$$

- Independent of type of classifier

Bagging

- Bootstrap sampling:
- Collect B $(\cong 100)$ subsets by sampling with replacement from training data
- Construct B trees (one classifier for one subset)
- Aggregate them using aggregator of your choice
- Parallelizable

Bagging



Bagging

- What about cross validation?
- Each bootstrap sample set uses only a subset of the data
- Unused samples: out-of-bag samples (OOB)
- Calculate overall error rate on out-of-bag samples for all bootstraps

Bagging

- Reduces overfitting (i.e., variance)
- Can work with any type of classifier (here focus on trees)
- Easy to parallelize
- But loses on interpretability to single decision tree

Random Forests

Issues with Bagging:

- Expectation of bagged trees is equal to expectation of individual trees

$$\mathbf{E}\left[\hat{f}_{agg}(x)\right] = \mathbf{E}\left[f_b(x)\right]$$

- Bias of bagged trees is the same as that of individual trees
- Each tree is identically distributed (i.d. not i.i.d). Bagged trees are correlated!

Random Forests

- How to decorrelate the trees generated for bagging?
- We want to generate B i.i.d. trees such that their bias is the same, but variance reduces
- Ideas:
  - We can restrict how many times a feature can be used
  - We only allow a certain number of features
  - Etc..

Random Forests

- Ideas:
  - We can restrict how many times a feature can be used
  - We only allow a certain number of features
  - Etc..
- Bias changes for the above ideas ☹
- Instead, choose only subset of features for each bag
- Decorrelated trees when you randomly select the subset

Random Forests

- As in bagging, choose B bootstrapped splits (or bags)
- For each split in the B trees, consider only k features from the full feature set m
- $k = m$ → same as Bagging
- $k < m$ → Random Forests
- OOB error rate can be used to fit RF in one sequence with cross validation done along the way

Random Forests

- Works great in practice. k to be treated as a hyperparameter

Issues:

- When you have large number of features, yet very small number of relevant features
- Prob(selecting the relevant feature in k) is very small

Más información

- https://cs229.stanford.edu/lectures-spring2022/cs229-boosting_slides.pdf