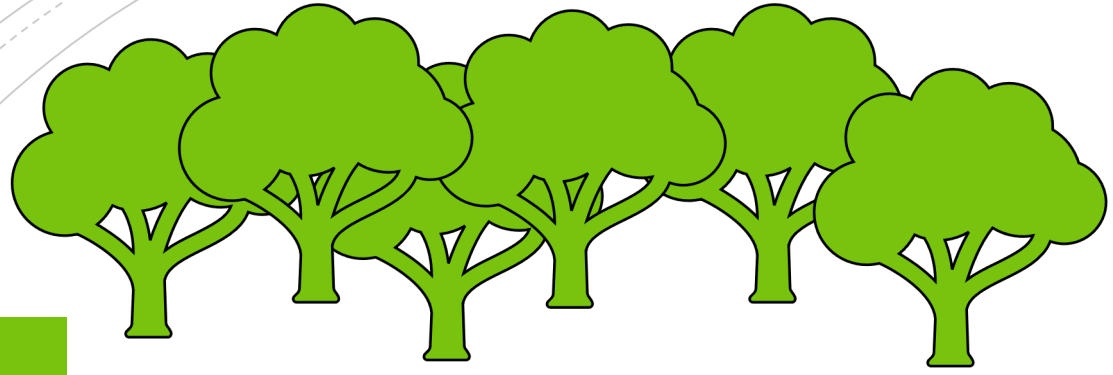
13.3 RANDOM FORESTS

Decision by Committee: Ensemble Learning

Random forest



- The idea is largely that if one tree is good, then many trees (a forest) should be better, if there is enough variety between them.
- creates randomness
  - Bagging (training them on slightly different data)
  - Limit the choices that the decision tree can make

The Basic Random Forest Training Algorithm

- For each of N trees:
  - create a new bootstrap sample of the training set
  - use this bootstrap sample to train a decision tree
  - at each node of the decision tree, randomly select m features, and compute the information gain (or Gini impurity) only on that set of features, selecting the optimal one
  - repeat until the tree is complete

Gini impurity

- Metric that is used when training decision trees.
- `Gini Impurity` is a measurement of the likelihood of an **incorrect classification** of a new instance of a random variable, if that new instance were **randomly classified according to the distribution of class labels** from the data set
- `G(k) = Σ P(i) * (1 - P(i))`

Other metric

- **Information Entropy and Information Gain**
  - Entropy could be used to describe the amount of unpredictability in a random variable.
- **Gini Gain**
  - Similar to entropy, which had the concept of `information gain`, `gini gain` is calculated when building a decision tree to help determine which attribute gives us the most information about which class a new data point belongs to.

https://bambielli.com/til/2017-10-29-gini-impurity/#:~:text=Gini%20Impurity%20is%20a%20measurement%20of%20the%20likelihood,if%20the%20data%20set%20contains%20only%20one%20class.

Out-of-bootstrap examples

- The bootstrap sample will miss out about 35% of the data on average,
- Keep track of these datapoints for Test set
- Avoid cross-validation

Boosting Bagging

- Boosting has to run sequentially (boosting has to run sequentially), Random forest is in parallel
- Bagging puts most of its effort into ensuring that the different classifiers see different data and the importance of each datapoint changes for the different classifiers
- Both boosting and bagging take a vote from amongst the classifiers
  - boosting takes a weighted vote
  - bagging takes the majority vote

DIFFERENT WAYS TO COMBINE CLASSIFIERS

- Voting is not necessarily simple
  - Binary? outputis max common outputs.
  - Regression? output is the mean value or median of outputs.

Mixture of experts

- Another way is combining classifiers
- Each individual classifier's assessments are weighted by the relevant gate, which produces a weight w using the current inputs, and this is propagated further up the hierarchy.

The Mixture of Experts Algorithm

- For each expert:
  - calculate the probability of the input belonging to each possible class by computing (where the $\mathbf{w}_i$ are the weights for that classifier):

$$o_i(\mathbf{x}, \mathbf{w}_i) = \frac{1}{1 + \exp(-\mathbf{w}_i \cdot \mathbf{x})}. \tag{13.6}$$

- For each gating network up the tree:
  - compute:

$$g_i(\mathbf{x}, \mathbf{v}_i) = \frac{\exp(\mathbf{v}_i \mathbf{x})}{\sum_l \exp(\mathbf{v}_l \mathbf{x})}. \tag{13.7}$$

- Pass as input to the next level gates (where the sum is over the relevant inputs to that gate):

$$\sum_k o_j g_j. \tag{13.8}$$