

Random Forest

AbdElMoniem Bayoumi, PhD

Fall 2021

Uniform Sampling with Replacement

- Give a set of training examples \mathcal{S}
- Sample a subset \mathcal{S}_i of examples
- **Uniform sampling:** selection probability of any example in \mathcal{S} is $\frac{1}{|\mathcal{S}|}$
- **Replacement:** a selected item can be reselected multiple times for the same subset

Random Forest

- Given M training examples
- Uniformly sample T subsets with replacement
 - each of size M
- Build T decision trees with zero-training error
 - Tree for each training subset S_i
 - No pruning
- Take the average/votes of T trees
 - Subsets are different so no overfitting!
 - Can get certainty of your prediction, i.e., how many classifiers?

Random Forest

- Given D features, i.e., dimensions
- Building T decision trees:
 - Sample K features randomly ($K < D$)
 - Only split on these K random features
 - New K features are sampled for every single split
- Sampling features leads to different trees
 - Different mistakes by each tree
 - Mistakes averaged out → no overfitting

Random Forest

- Out of the box approach
 - No need to tune hyperparameters
 - No need to pre-process or scale inputs
- $K = \lceil \sqrt{D} \rceil \rightarrow \text{round up!}$
- Increase T as much as you can afford
 - Parallel processing

Random Forest: Out-of-Bag Error

- Aka. out-of-bag estimate
- No need for training/validation split
- Can estimate test error directly from training set
- Compute error for each training example
 - consider only trees that do not include that example in their training subset
 - approx. 60% of trees are considered

Random Forest: Out-of-Bag Error

- For each data point, average the loss of classifiers that do not have that data point,
 - we obtain an estimate of the true test error
 - without reducing the training set

$$E_{OFB} = \frac{1}{M} \sum_{i=1}^M \left[\frac{1}{z_i} \sum_{\substack{j \\ (x_i, y_i) \notin S_j}}^T \text{loss}(h_j(x_i), y_i) \right]$$

$$z_i = \sum_{\substack{j \\ (x_i, y_i) \notin S_j}}^T 1$$

num of trees **NOT** trained on (x_i, y_i)

Random Forest

- Easy to use!
- Second best approach! → Rule of thumb!
- No need for training/validation split
- For regression, use regression trees

Random Forest

- Wisdom of the crowd!
- Improvement: prune the last split of trees (i.e., from bottom) if that improves E_{OFB}
 - Decrease size of trees
 - Decrease noise
- Not suitable for raw images

Acknowledgement

- These slides have been designed relying on materials of Victor Lavrenko and Kilian Weinberger