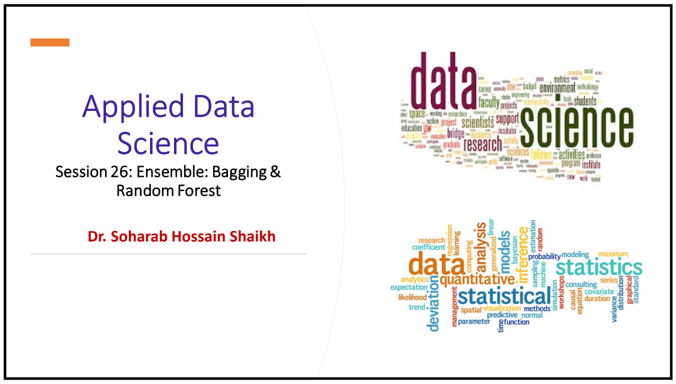
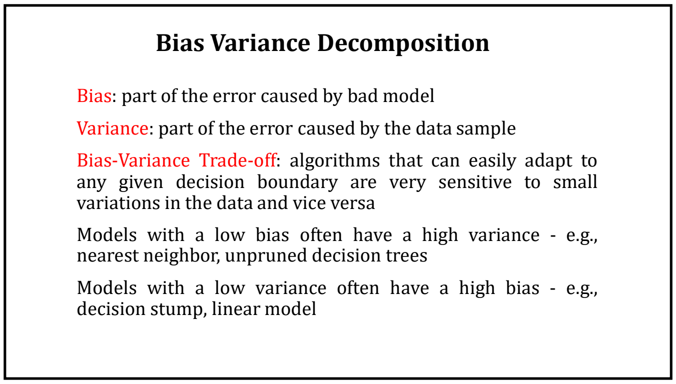


Applied Data Science

Session 26: Ensemble: Bagging & Random Forest

Dr. Soharab Hossain Shaikh

1



Bias Variance Decomposition

Bias: part of the error caused by bad model

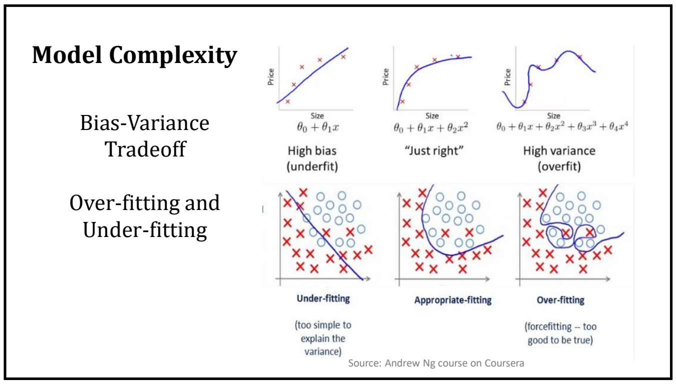
Variance: part of the error caused by the data sample

Bias-Variance Trade-off: algorithms that can easily adapt to any given decision boundary are very sensitive to small variations in the data and vice versa

Models with a low bias often have a high variance - e.g., nearest neighbor, unpruned decision trees

Models with a low variance often have a high bias - e.g., decision stump, linear model

2



Model Complexity

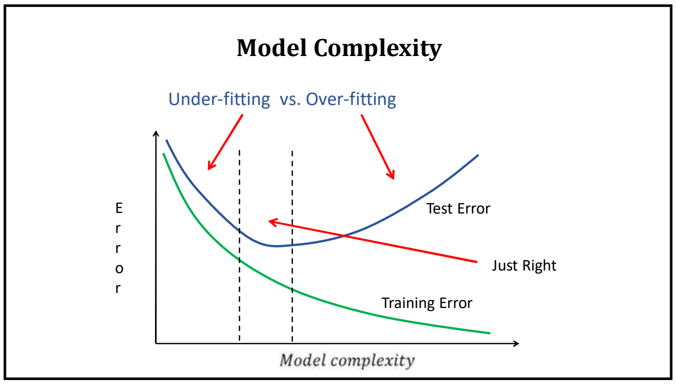
Bias-Variance Tradeoff

Over-fitting and Under-fitting

Price
Size
$\theta_0 + \theta_1 x$
High bias (underfit)
Price
Size
$\theta_0 + \theta_1 x + \theta_2 x^2$
"Just right"
Price
Size
$\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$
High variance (overfit)
Under-fitting
(too simple to explain the variance)
Appropriate-fitting
Over-fitting
(forcefitting -- too good to be true)

Source: Andrew Ng course on Coursera

3



Model Complexity

4

Splitting Data: Importance of Validation Set

- Training Set – data at out disposal to make use of. (80/70%)
- Validation Set – parameter tuning
- Test Set – make prediction – check for model performance (20/30%)

5

Ensemble Learning

IDEA:

- do not learn a *single* classifier but learn a *set of classifiers*
- *combine the predictions* of multiple classifiers

MOTIVATION:

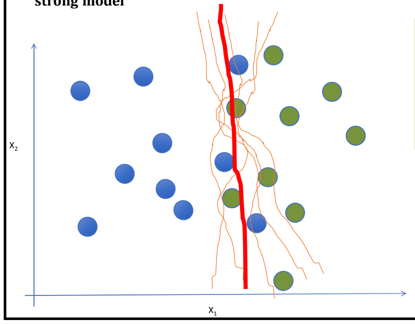
- reduce variance: results are less dependent on peculiarities of a single training set
- reduce bias: a combination of multiple classifiers may learn a more expressive concept class than a single classifier

KEY STEP:

- formation of an ensemble of *diverse* classifiers from a single training set (?)

6

Learning Ensemble: Combining several weak models in order to create a strong model



Wisdom of the crowd – decisions taken by averaging the decisions of a large number of non-experts (weak learners) is usually better/correct than the one based on only a single decision made by an expert.

- ➢ Train a number of models (usually weak learners).
- ➢ Take decision based on aggregating the results of all the learners (majority vote or average score)

7

Why do Ensembles Work?

Suppose there are 25 base classifiers

- Each classifier has error rate, ε = 0.35
- Assume classifiers are independent i.e., probability that a classifier makes a mistake does not depend on whether other classifiers made a mistake

Note: in practice they are not independent!

Probability that the ensemble classifier makes a wrong prediction

- The ensemble makes a wrong prediction if the majority of the classifiers makes a wrong prediction
- The probability that 13 or more classifiers err is

$$\sum_{i=13}^{25}\binom{25}{i}\varepsilon^{i}(1-\varepsilon)^{25-i}\approx 0.06\ll\varepsilon$$

8

Ensemble Methods

Like Crowd-sourced Machine Learning Algorithms:

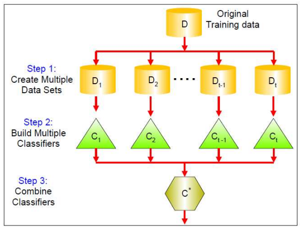
- Take a collection of simple or *weak* learners
- Combine their results to make a single, better/strong learner
- Diversity among weak learners is required

Types:

- **Bagging:** train learners in parallel on different samples of the data, then combine by voting (discrete output) or by averaging (continuous output).
- **Random Forest** – Grow many Decision Trees
- **Stacking:** combine model outputs using a second-stage learner like linear regression (different types of learners).
- **Boosting:** train learners on the filtered output of other learners (sequential).

9

Bagging


D
Original Training data
Step 1: Create Multiple Data Sets
D_1
D_2
D_{t-1}
D_t
Step 2: Build Multiple Classifiers
C_1
C_2
C_{t-1}
C_t
Step 3: Combine Classifiers
C^*

Bagging: Bootstrap **agg**regat**ing** is a method that results in low variance.

If we had multiple realizations of the data (or multiple samples) we could calculate the predictions multiple times and take the average of all the predictions

Based on the idea that averaging multiple onerous estimations produce less uncertain results

10

Bootstrapping: Random Sub-sampling of Data with Replacement

```
1. Choose n, the number of the samples to be selected in a bootstrap.
2. i=1
3. While (i<=n)
      {
         3.i. Randomly select an observation/sample from the training dataset
         3.ii. Add it to the sample
          i=i+1
      }
```

N=number of samples in the training dataset
n=number of samples per bootstrap

Generally, in Ensemble learning like bagging/random-forest, n=N
Note: step 3.ii is randomly selecting a sample from the dataset; the same sample may be selected in the bootstrap for more than one allowing duplication (replacement)

11

Bagging

Say for each sample *b*, we calculate $f^b(x)$, then:

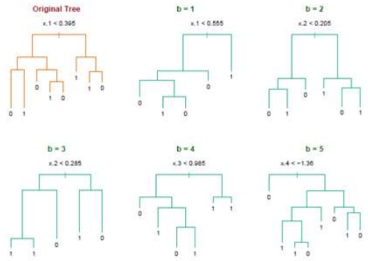
$$\hat{f}_{avg}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^b(x)$$

> Construct B (hundreds) of trees (no pruning) - Learn a classifier for each bootstrap sample based on all the predictors/features

> Average the results of all of them : Very effective

12

Bagging Decision Trees



Hastie et al.,"The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer (2009)

13

Out-of-Bag Error Estimation

- In bootstrapping we sample with replacement, and therefore **not all observations are used for each bootstrap sample**.
- On average 1/3 of them are not used!
- We call them out-of-bag samples (OOB)
- We can predict the response for the *i-th* observation using each of the trees in which that observation was OOB and do this for *n* observations
- Calculate overall OOB MSE or classification error

14

Bagging

- Reduces overfitting (variance) – random sub-samples do not contain all the training samples; therefore each learner (decision tree) do not see all the training samples and therefore do not adapt to the samples (memorizes) too much – reducing variance.
- Normally uses one type of classifier - Decision Trees are popular
- Easy to parallelize

15

Variable Importance Measures

- Bagging results in improved accuracy over prediction using a single tree
- Unfortunately, difficult to interpret the resulting model.
- Bagging improves prediction accuracy at the expense of interpretability.

Calculate the total amount that the RSS or Gini index is decreased due to splits over a given predictor, averaged over all B trees.

16

Bagging - Issues

Each tree is identically distributed, the expectation of the average of *B* such trees is the same as the expectation of any one of them

The bias of bagged trees is the same as that of the individual trees

Suppose that there is one very strong predictor in the data set, along with a number of other moderately strong predictors.

Then all bagged trees will select the strong predictor at the top of the tree and therefore all trees will look similar.

We can penalize the splitting (like in pruning) with a penalty term that depends on the number of times a predictor is selected at a given length

We can restrict how many times a predictor can be used

We only allow a certain number of predictors

17

Decision Trees

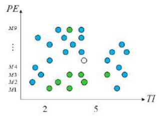
- Decision trees are individual learners that are combined. They are one of the most popular learning methods commonly used for data exploration.
- One type of decision tree is called CART - Classification and Regression Tree.
- CART - greedy, top-down, binary, recursive partitioning, that divides feature space into sets of disjoint rectangular sub-regions.
  - Regions should be pure w.r.t response variable
  - Simple model is fit in each region – majority vote for classification, constant value for regression.

18

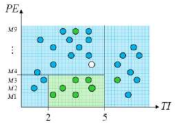
Decision Trees involve Greedy -Recursive Partitioning

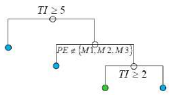
Simple dataset with two predictors

TI	PE	Response
1.0	M2	good
2.0	M1	bad
...	...	...
4.5	M5	?



- Greedy, recursive partitioning along TI and PE





19

Random Forest

- **Random forest** (or **random forests**) is an Ensemble classifier that consists of **many decision trees** and outputs the class that is the mode of the class's output by individual trees.
- The term came from **random decision forests** that was first proposed by Tin Kam Ho of Bell Labs in 1995.
- The method combines Breiman's "bagging" idea and the random selection of features.

20

Random Forest - justification for the name

- Random- Randomly select a subset of features/predictors from the set of all features/predictors.
- Forest – construct a number of Decision Trees forming a Forest of trees

21

Algorithm: Construct a Random Forest

Each Tree is constructed using the following algorithm:

1. Let the number of training cases be *N*, and the number of variables/predictors/features in the classifier be *M*.
2. Subset of Features: The number *m* of input variables/features/predictors to be used to determine the decision at a node of the tree; $m << M$.
3. Bootstrapping - Random Sub-sampling with Replacement: Choose a training set for this tree by choosing *n* times with replacement from all *N* available training cases (i.e. take a bootstrap sample). Each subsample contains approximately 2/3 samples from the training set and approximately 1/3 samples are duplicates.
4. Error Estimation: Use the rest of the cases (approximately 1/3 out-of-bag (OOB) samples from the training set) to estimate the error of the tree, by predicting their classes.
5. For each node of the tree, randomly choose *m* variables on which to base the decision at that node. Calculate the **best split** based on these ***m*** variables in the training set. When a node has less than some predetermined $\boldsymbol{n_{min}}$ number of nodes, then stop further splitting.

Bootstrapping and random selection of features make trees diverse reducing variance.

6. Each tree is fully grown and not pruned (as may be done in constructing a normal tree classifier).

No pruning makes trees fit data more tightly reducing bias.

22

Prediction with Random Forest

For prediction of a new sample

> The new sample is pushed down the tree.

> It is assigned the label of the training samples in the terminal/leaf node it ends up.

> This procedure is iterated over all trees in the ensemble.

> The average score of all trees is reported as random forest prediction.

Score:

Classification – a majority voting scheme

Regression – mean/median of all the predictions

23

Random Forests Tuning

The inventors make the following recommendations:

How minimum number of features and nodes?

- For classification, the default value for ***m*** is $\sqrt{M}$ and the minimum node size is **one**.
- For regression, the default value for m is *M/3* and the minimum node size is **five**.

In practice the best values for these parameters will depend on the problem, and they should be treated as tuning parameters.

When to terminate train?

Like with Bagging, we can use OOB and therefore RF can be fit in one sequence, with cross-validation being performed along the way.

Once the OOB error stabilizes, the training can be terminated.

24

Random Forest – Advantages

- It is one of the most accurate learning algorithms available. For many data sets, it produces a highly accurate classifier.
- It runs efficiently on large databases.
- It can handle thousands of input variables without variable deletion.
- It gives estimates of what variables are important in the classification.
- It generates an internal unbiased estimate of the generalization error as the forest building progresses.
- It works reasonably well when a large proportion of the data are missing.
- It computes proximities between pairs of cases that can be used in clustering, locating outliers, or (by scaling) give interesting views of the data. This capabilities can be extended to unlabeled data, leading to unsupervised clustering, data views and outlier detection.
- It offers an experimental method for detecting variable interactions.

25

Estimating Test Error

- While growing forest, estimate test error from training samples
- For each tree grown, approximately, 33% of samples are not selected in bootstrap, called out of bootstrap/bag (OOB) samples
- Using OOB samples as input to the corresponding tree, predictions are made as if they were novel test samples
- Through book-keeping, majority vote (classification), average (regression) is computed for all OOB samples from all trees.
- Such estimated test error is very accurate in practice, with reasonable N, size of training data.

26

Feature Importance

By computing OOB Error:

- Denote by $\hat{e}$ the OOB estimate of the loss when using original training set, D.
- For each predictor x_p where $p \in \{1,..,k\}$
  - Randomly permute p^th^ predictor to generate a new set of samples

    $D' = \{(y1,x'1),\dots,(yN,x'N)\}$
  - Compute OOB estimate $\hat{e}_k$ of prediction error with the new samples
- A measure of importance of predictor x_p is $\hat{e}_k - \hat{e}$, the increase in error due to random perturbation of p^{th} predictor.

By Computing OOB Accuracy:

- Randomly permute the data for column *j* in the OOB samples the record the accuracy again.
- The decrease in accuracy as a result of this permuting is averaged over all trees and is used as a measure of the importance of variable j in the random forest.

27

Random Forests Issues

When the number of variables is large, but the fraction of relevant variables is small, random forests are likely to perform poorly when *m* is small - at each split the chance can be small that the relevant variables will be selected.

Tends to overfit in some datasets

28

29

Case-study

30

Classification of Credit Risk - Predicting Possible Credit Defaulter based on German Credit Dataset

Context
The original dataset contains 1000 entries with 20 categorial/symbolic attributes prepared by Prof. Hofmann.

In this dataset, each entry represents a person who takes a credit by a bank. ach person is classified as good or bad credit risks according to the set of attributes.

Description of the Attributes

> Age (numeric)

> Sex (text: male, female)

> Job (numeric: 0 - unskilled and non-resident, 1 - unskilled and resident, 2 - skilled, 3 - highly skilled)

> Housing (text: own, rent, or free)

> Saving accounts (text - little, moderate, quite rich, rich)

> Checking account (numeric, in DM - Deutsch Mark)

> Credit amount (numeric, in DM)

> Duration (numeric, in month)

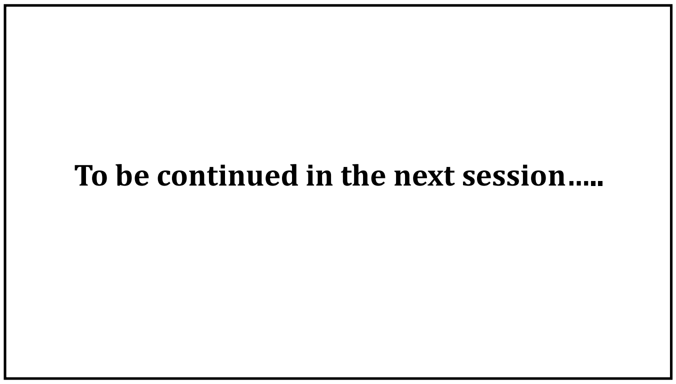
> Purpose (text: car, furniture/equipment, radio/TV, domestic appliances, repairs, education, business, vacation/others) etc.....

Total of 17 features (16 input features and 1 target variable (*'default'* column))

31

Let’s go to the Coding Demo…

32



To be continued in the next session…..

33