Simple guide for ensemble learning methods

What, why, how and Bagging–Boosting demystified, explained rather unconventionally, read on:)

Juhi Ramzai
Follow

Feb 25, 2019 · 8 min read

Before this post, I have published a "[Holy grail for Bias variance trade-off, Overfitting and Underfitting](#)". This comprehensive article serves as an important prequel to this post if you are a newbie or would just like to brush up the concepts of *bias and variance* before diving in with full force in the sea of Ensemble modelling. All the others in the audience can readily move on to know more about Ensemble modelling from my pen. I will resort to quoting some real life examples to simplify the concepts of ***what, why and how of the ensemble models with focus on bagging and boosting techniques***.

Scenario 1: You require a new pair of headphones. Now, is it likely that you'll just walk into a store and buy the headphones that the sales people show you? At the juncture of this day and age, I am sure that the answer is NO because we rely heavily on our "*research*" before buying anything nowadays. You would browse a few web technology portals and check the user reviews and would then compare different models that interest you while checking for their features and prices. You will also probably ask your friends and colleagues for their opinion. In short, you wouldn't directly reach a conclusion, but will instead make an informed decision after thoroughly researching your way through.

Now, can take a look at the technical definition of Ensemble learning methods.

What is an ensemble method?

Ensemble models in machine learning combine the decisions from multiple models to improve the overall performance. They operate on the similar idea as employed while buying headphones.

The main causes of error in learning models are due to **noise, bias and variance**.

Ensemble methods help to minimize these factors. These methods are designed to improve the stability and the accuracy of Machine Learning algorithms.

Scenario 2: Let's suppose that you have developed a health and fitness app. Before making it public, you wish to receive critical feedback to close down the potential loopholes, if any. You can resort to one of the following methods, read and decide which method is the best:

1. **You can take the opinion of your spouse or your closest friends.**

2. **You can ask a bunch of your friends and office colleagues.**

3. **You can launch a beta version of the app and receive feedback from the web development community and non-biased users.**

No brownie points for guessing the answer :D Yes, of course we will roll with the third option.

Now, pause and think what you just did. You took multiple opinions from a large enough bunch of people and then made an informed decision based on them. This is what Ensemble methods also do.

Ensemble models in machine learning combine the decisions from multiple models to improve the overall performance.

Scenario 3: Take a look at the following picture; we can see a group of blindfolded children playing the game of "Touch and tell" while examining an elephant which none of them had ever seen before. Each of them will have a different version as to how does an elephant looks like because each of them is exposed to a different part of the elephant. Now, if we give them a task of submitting a report on elephant description, their individual reports will be able to describe only one part accurately as per their experience but collectively they can combine their observations to give a very accurate report on the description of an elephant.

Similarly, ensemble learning methods employ a group of models where the combined result out of them is almost always better in terms of prediction accuracy as compared to using a single model.

Ensembles are a divide and conquer approach used to improve performance.
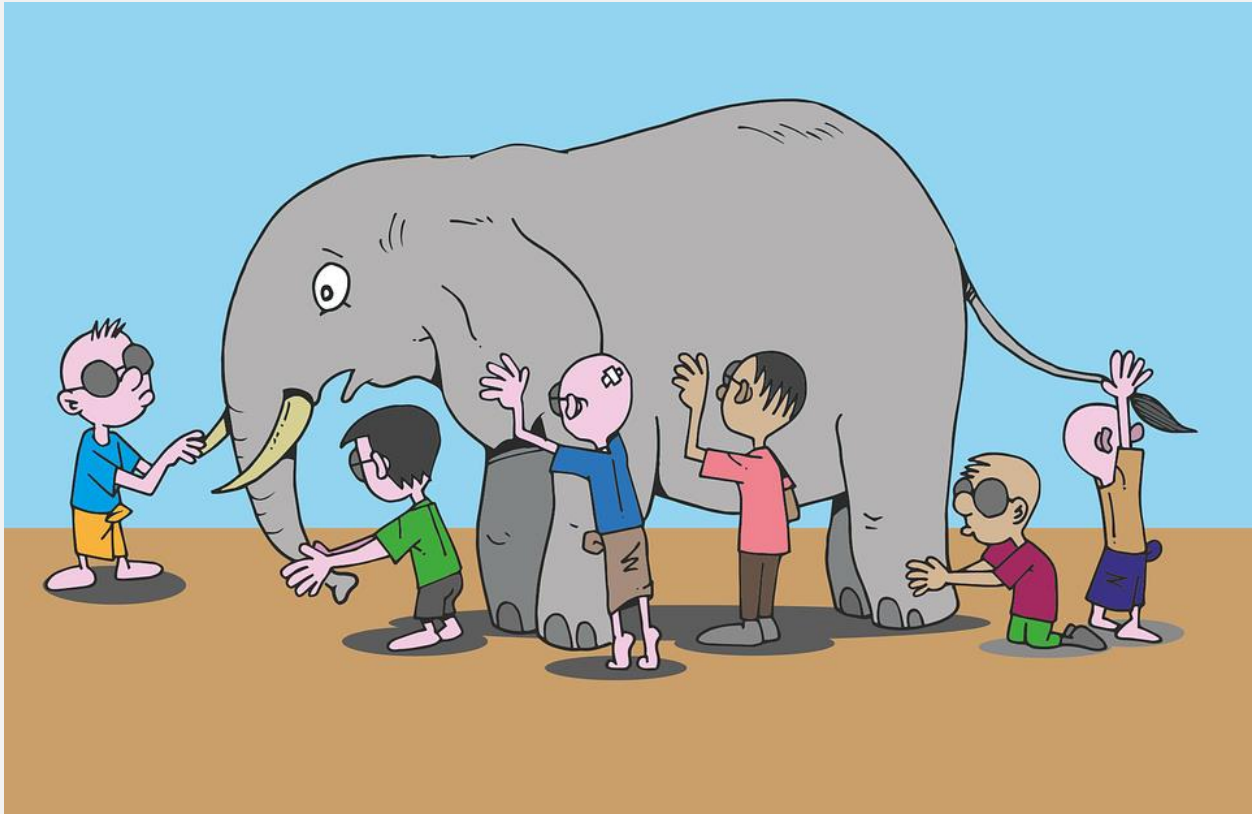
Now, let's dive into some of the important Ensemble techniques.

Crack up a little :D You are doing great :)

Simple Ensemble techniques

1. **Taking the mode of the results**

MODE: The mode is a statistical term that refers to the most frequently occurring number found in a set of numbers.

In this technique, multiple models are used to make predictions for each data point. The predictions by each model are considered as a separate vote. The prediction which we get from the majority of the models is used as the final prediction.

For instance: We can understand this by referring back to Scenario 2 above. I have inserted a chart below to demonstrate the ratings that the beta version of our health and fitness app got from the user community. (*Consider each person as a different model*)

Output= MODE=3, as majority people voted this
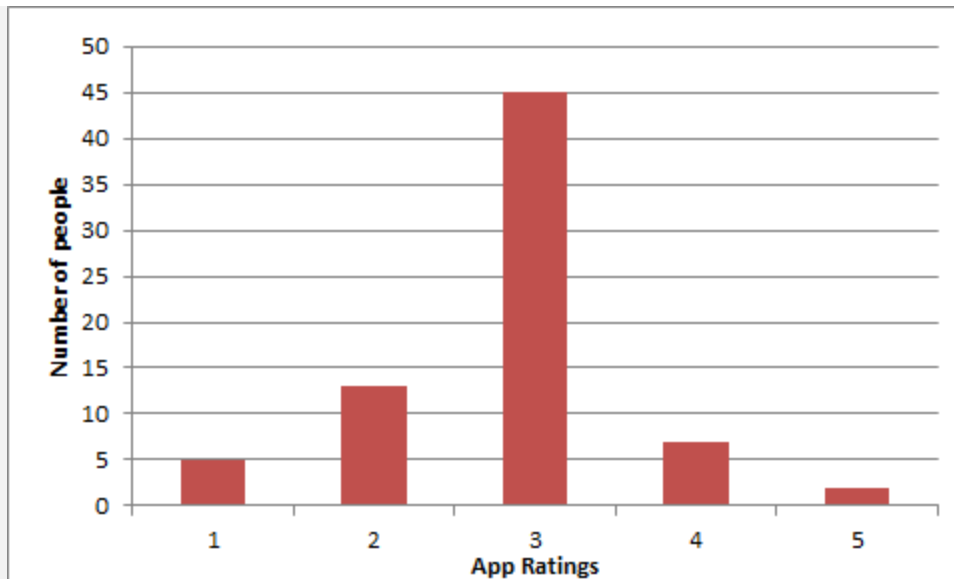
2. **Taking the average of the results**

In this technique, we take an average of predictions from all the models and use it to make the final prediction.

AVERAGE= sum(Rating*Number of people)/Total number of people= (1*5)+(2*13)+(3*45)+(4*7)+(5*2)/72 = 2.833 =Rounded to nearest integer would be 3

3. **Taking weighted average of the results**

This is an extension of the averaging method. All models are assigned different weights defining the importance of each model for prediction. For instance, if about 25 of your responders are professional app developers, while others have no prior experience in this field, then the answers by these 25 people are given more importance as compared to the other people.

For example: For posterity, I am trimming down the scale of the example to 5 people

WEIGHTED AVERAGE= (0.3*3)+(0.3*2)+(0.3*2)+(0.15*4)+(0.15*3) =3.15 = rounded to nearest integer would give us 3

Person	Professional	Weight	Rating
A	Y	0.3	3
B	Y	0.3	2
C	Y	0.3	2
D	N	0.15	4
E	N	0.15	3

Advanced Ensemble techniques

We'll learn about Bagging and Boosting techniques now. But, to use them you must select a base learner algorithm. For example, if we choose a classification tree, Bagging and Boosting would consist of a pool of trees as big as we want.

1. Bagging (Bootstrap AGGregatING)

Bootstrap Aggregating is an ensemble method. First, we create random samples of the training data set with replacment (sub sets of training data set). Then, we build a model (classifier or Decision tree) for each sample. Finally, results of these multiple models are combined using average or majority voting.

As each model is exposed to a different subset of data and we use their collective output at the end, so we are making sure that problem of overfitting is taken care of by not clinging too closely to our training data set. Thus, Bagging helps us to reduce the variance error.

Combinations of multiple models decreases variance, especially in the case of unstable models, and may produce a more reliable prediction than a single model.

Random forest technique actually uses this concept but it goes a step ahead to further reduce the variance by randomly choosing a subset of features as well for each bootstrapped sample to make the splits while training *(My next post will detail all about Random forest technique)*
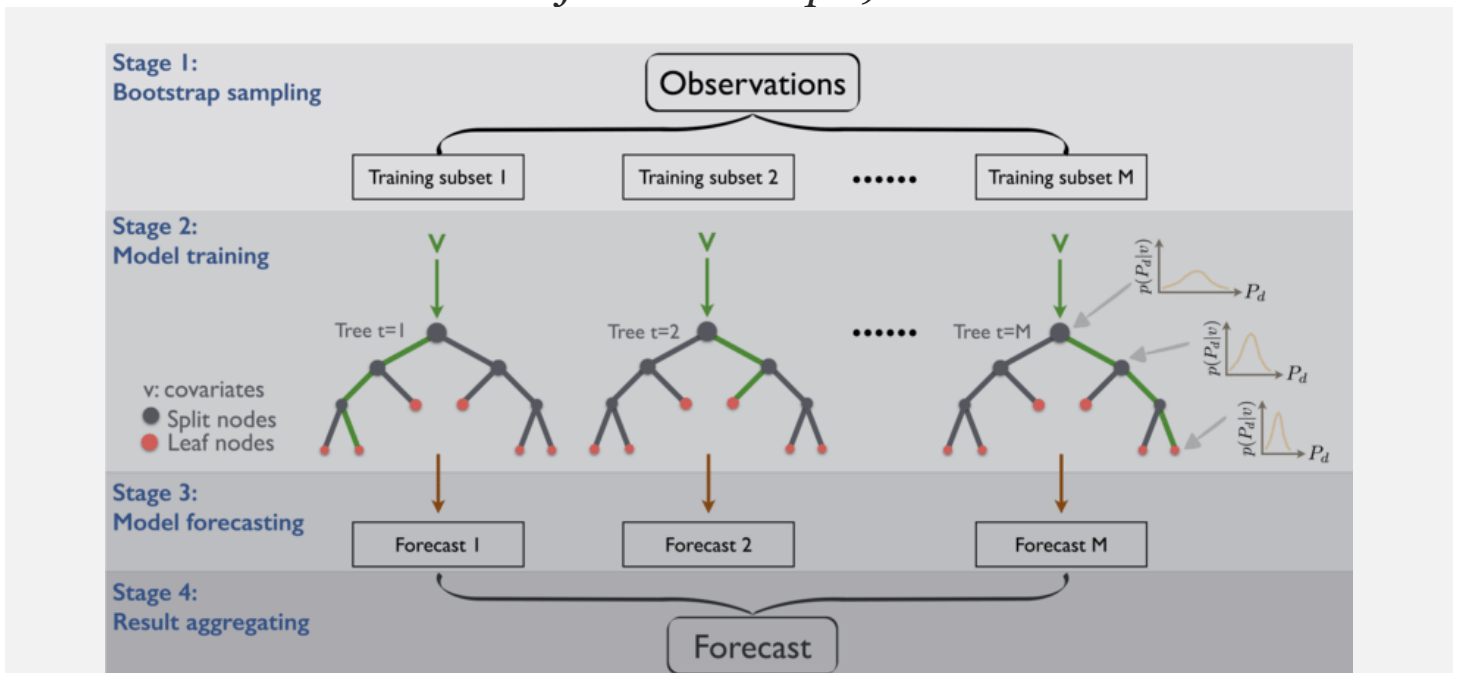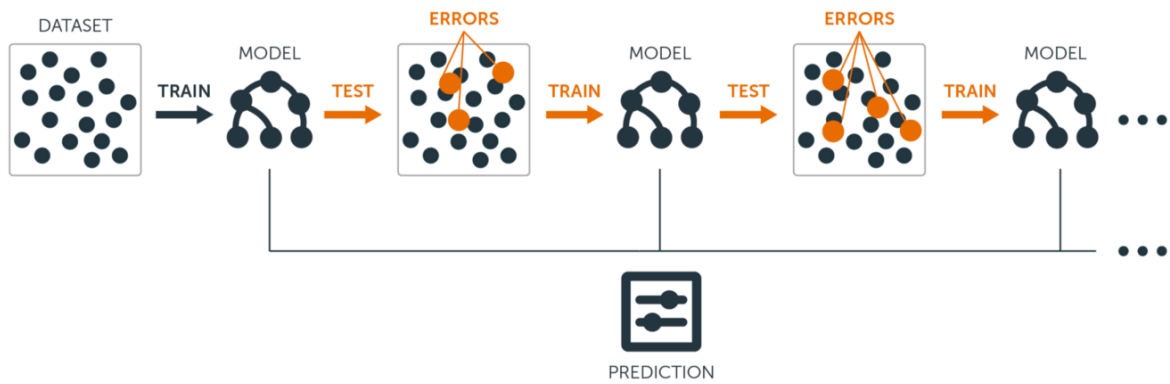

2. Boosting

Boosting is an iterative technique which adjusts the weight of an observation based on the last classification. If an observation was classified incorrectly, it tries to increase the weight of this observation and vice versa.

Boosting in general decreases the bias error and builds strong predictive models. Boosting has shown better predictive accuracy than bagging, but it also tends to over-fit the training data as well.*Thus, parameter tuning becomes a crucial part of boosting algorithms to make them avoid overfitting.*

Boosting is a sequential technique in which, the first algorithm is trained on the entire data set and the subsequent algorithms are built by fitting the residuals of the first algorithm, thus giving higher weight to those observations that were poorly predicted by the previous model.
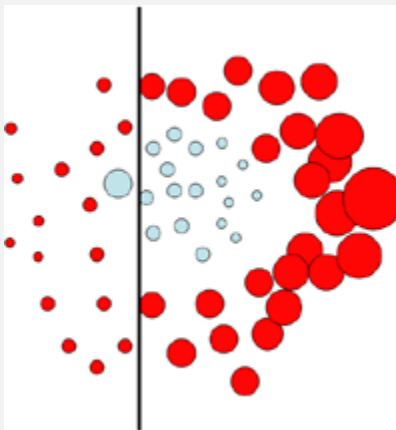
It relies on creating a series of weak learners each of which might not be good for the entire data set but is good for some part of the data set. Thus, each model actually boosts the performance of the ensemble.

Summary of differences between Bagging and Boosting

	Bagging	Boosting
Similarities	• Uses voting • Combines models of the same type	
Differences	Individual models are built separately	Each new model is influenced by the performance of those built previously
	Equal weight is given to all models	Weights a model's contribution by its performance

V. EXPERIMENTAL DESIGN

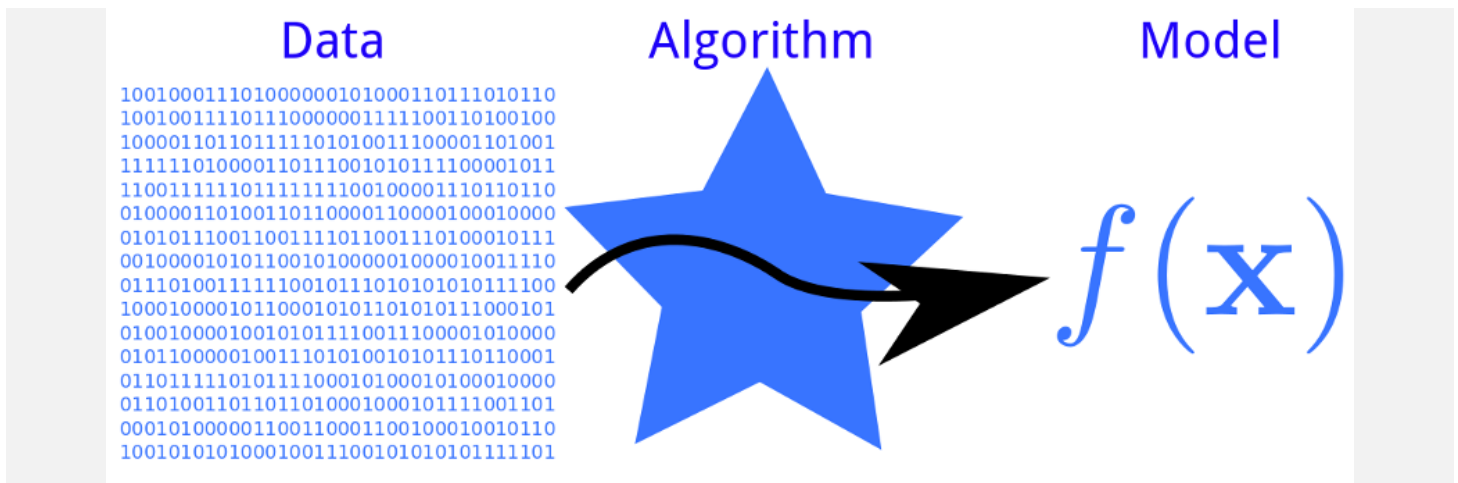# Advantages/Benefits of ensemble methods

Ensemble methods are used in almost all the ML hackathons to enhance the prediction abilities of the models.

Let's take a look at the advantages of using ensemble methods:

1. **More accurate prediction results**- We can compare the working of the ensemble methods to the *Diversification* of our financial portfolios. It is advised to keep a mixed portfolio across debt and equity to reduce the variability and hence, to minimize the risk. Similarly, the ensemble of models will give better performance on the test case scenarios (unseen data) as compared to the individual models in most of the cases.

2. **Stable and more robust model**- The aggregate result of multiple models is always less noisy than the individual models. This leads to model stability and robustness.

Data   Algorithm   Model

$$f(\mathbf{x})$$

3. Ensemble models can be used to capture the linear as well as the non-linear relationships in the data.This can be accomplished by using 2 different models and forming an ensemble of the two.

Disadvantages of ensemble methods

1. **Reduction in model interpret-ability-** Using ensemble methods reduces the model interpret-ability due to increased complexity and makes it very difficult to draw any crucial business insights at the end.

2. **Computation and design time is high-** It is not good for real time applications.

3. The selection of models for creating an ensemble is an art which is really hard to master.

___

So, now that we have covered all the fundamentals of ensemble methods, you can proceed to trying them hands on to enhance your understanding even further.

Thank you for reading!:)