Bagging, boosting and stacking in machine learning

What's the similarities and differences beetween this 3 methods: bagging, boosting, stacking?

Which is the best one? And why?

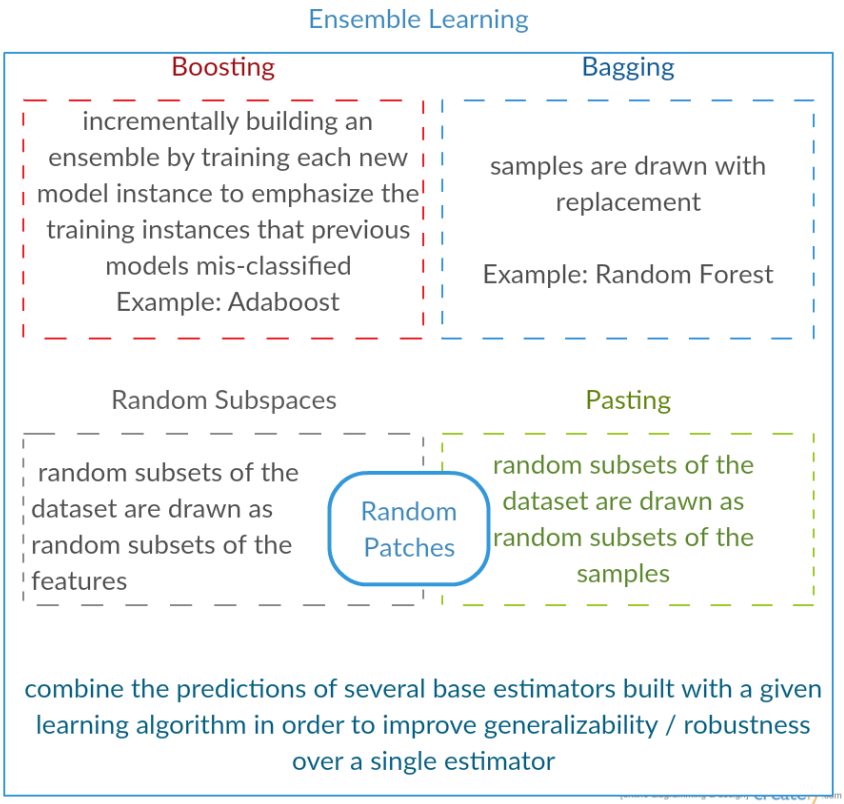
Can you give me an example for each?

machine-learning ensemble model-averaging

edited Dec 20 '15 at 21:45
Tim
11.2k 2 26 58

asked Nov 24 '11 at 16:51
Bucsa Lucian
436 1 5 3

4 Answers



Sources for this image:

- Wikipedia
- sklearn

answered Dec 22 '15 at 22:43
moose
74 10

Just to elaborate on Yuqian's answer a bit. The idea behind bagging is that when you OVERFIT with a nonparametric regression method (usually regression or classification trees, but can be just about any nonparametric method), you tend to go to the high variance, no (or low) bias part of the bias/variance tradeoff. This is because an overfitting model is very flexible (so low bias over many resamples from the same population, if those were available) but has high variability (if I

collect a sample and overfit it, and you collect a sample and overfit it, our results will differ because the non-parametric regression tracks noise in the data). What can we do? We can take many resamples (from bootstrapping), each overfitting, and average them together. This should lead to the same bias (low) but cancel out some of the variance, at least in theory.

Gradient boosting at its heart works with UNDERFIT nonparametric regressions, that are too simple and thus aren't flexible enough to describe the real relationship in the data (i.e. biased) but, because they are under fitting, have low variance (you'd tend to get the same result if you collect new data sets). How do you correct for this? Basically, if you under fit, the RESIDUALS of your model still contain useful structure (information about the population), so you augment the tree you have (or whatever nonparametric predictor) with a tree built on the residuals. This should be more flexible than the original tree. You repeatedly generate more and more trees, each at step k augmented by a weighted tree based on a tree fitted to the residuals from step k-1. One of these trees should be optimal, so you either end up by weighting all these trees together or selecting one that appears to be the best fit. Thus gradient boosting is a way to build a bunch of more flexible candidate trees.

Like all nonparametric regression or classification approaches, sometimes bagging or boosting works great, sometimes one or the other approach is mediocre, and sometimes one or the other approach (or both) will crash and burn.

Also, both of these techniques can be applied to regression approaches other than trees, but they are most commonly associated with trees, perhaps because it is difficult to set parameters so as to avoid under fitting or overfitting.

answered Dec 21 '15 at 2:25

AlaskaRon
941 6

Bagging:

1. **parallel** ensemble: each model is built independently
2. aim to **decrease variance**, not bias
3. suitable for high variance low bias models (complex models)
4. an example of a tree based method is **random forest**, which develop fully grown trees (note that RF modifies the grown procedure to reduce the correlation between trees)

Boosting:

1. **sequential** ensemble: try to add new models that do well where previous models lack
2. aim to **decrease bias**, not variance
3. suitable for low variance high bias models
4. an example of a tree based method is **gradient boosting**

edited Dec 16 '15 at 16:31

answered Dec 16 '15 at 3:23

yuqian
111 1 3

1 Commenting each of the points to answer *why* is it so and *how* it is achieved would be a great improvement in your answer. – Tim Dec 16 '15 at 8:12

1 Can you share any document/link which explains that boosting reduce variance and how it does it? Just want to understand in more depth – ML_Pro Dec 16 '15 at 13:55

Thanks Tim, I'll add some comments later. @ML_Pro, from the procedure of boosting (e.g. page 23 of cs.cornell.edu/courses/cs578/2005fa/…), it's understandable that boosting can reduce bias. – yuqian Dec 17 '15 at 1:01

These are different approaches to improve the performance of your model (so-called meta-algorithms):

1. **Bagging** (stands for **B**ootstrap **Agg**regation) is the way decrease the variance of your prediction by generating additional data for training from your original dataset using combinations with repetitions to produce multisets of the same cardinality/size as your original data. By increasing the size of your training set you can't improve the model predictive force, but just decrease the variance, narrowly tuning the prediction to expected outcome.
2. **Boosting** is a an approach to calculate the output using several different models and then average the result using a weighted average approach. By combining the advantages and

pitfalls of these approaches by varying your weighting formula you can come up with a good predictive force for a wider range of input data, using different narrowly tuned models.

3. **Stacking** is a similar to boosting: you also apply several models to you original data. The difference here is, however, that you don't have just an empirical formula for your weight function, rather you introduce a meta-level and use another model/approach to estimate the input together with outputs of every model to estimate the weights or, in other words, to determine what models perform well and what badly given these input data.

As you see, these all are different approaches to combine several models into a better one, and there is no single winner here: everything depends upon your domain and what you're going to do. You can still treat *stacking* as a sort of more advances *boosting*, however, the difficulty of finding a good approach for your meta-level makes it difficult to apply this approach in practice.

Short examples of each:

1. *Bagging*: Ozone data.
2. *Boosting*: is used to improve optical character recognition (OCR) accuracy.
3. *Stacking*: is used in K-fold cross validation algorithms.

edited Jun 4 '15 at 10:06
Community ♦
1

answered Nov 28 '11 at 12:32

Alexander Galkin
1,671 8 6

1 Thank you for the helpful overview. Can you elaborate on the relationship between stacking and k-fold cross validation? I don't see the connection, and the link doesn't clarify (the word "stacking" doesn't even appear there). – D.W. Dec 10 '15 at 21:15