Ensemble Models

Stacking, Bagging, and Boosting

Greg Damico

// FLATIRON SCHOOL

Agenda

Justification

- An old mantra

Stacking

- Weighted Averaging

Bagging

- Layers of randomization
- Synchronic Aggregation

Boosting

- Diachronic Learning

Justification

An Old Mantra

Data Collection:
- One data point is good, but more data points are better!

Bootstrapping:
- One sample is good, but more samples are better!

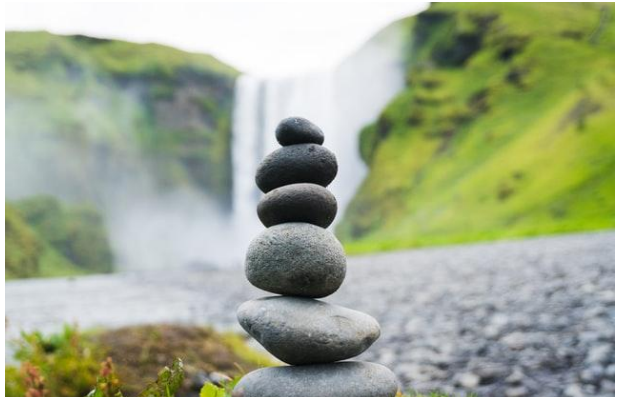
Modeling:
- One model is good, but more models are better!

Justification

Model *Composed of* Other Models

- Building models beyond the first is good for comparison's sake.
- But we can also combine models together to form new models.

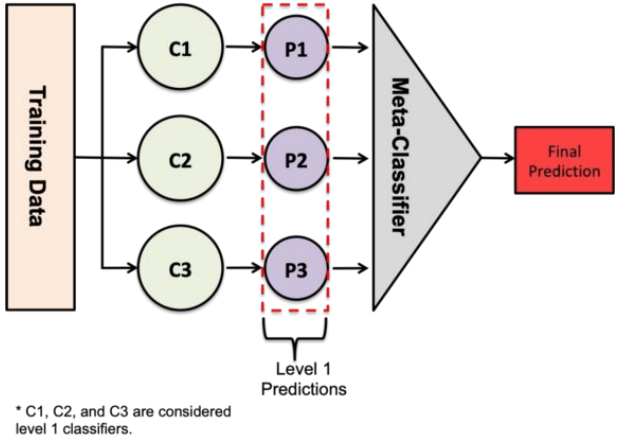
Stacking



Different Models, Same Data

- Similar to a bagging approach, stacking is a form of **averaging**
- Unlike a bagging approach, stacking typically uses the **same training data** for every model
- The innovation of stacking is the use of **different kinds of models**

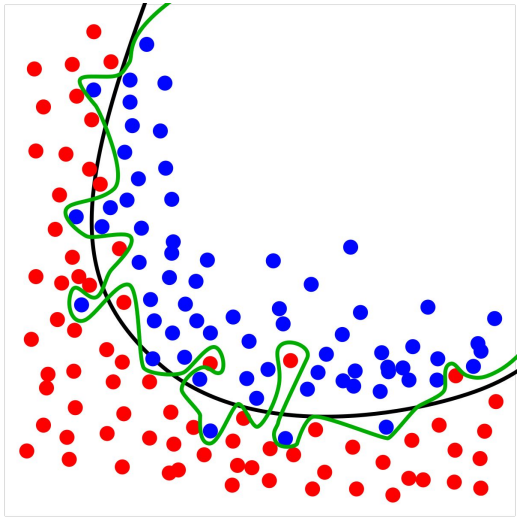
Stacking



Meta-Classifier/Meta-Regressor

- First, we ask several different models to make predictions about the target
- Rather than taking a simple average or vote to determine the outcome, feed these results into a **final model that makes the prediction based on the other models' predictions**
- If it seems to you like we are approaching a neural network...you are correct!

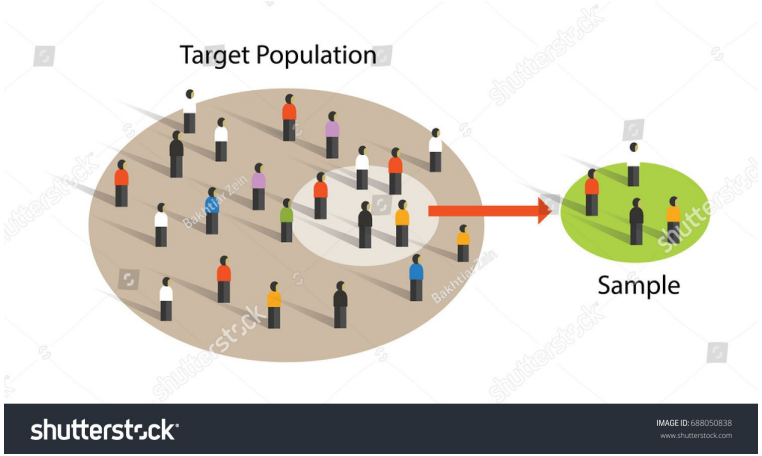
Bagging



Strategy

- Many models naturally **overfit**
- Randomization → New models
- New models overfit in different ways
- Aggregation → Smooth over different ways of overfitting to reduce variance

Bagging



Synchronic Aggregation

- ***B***ootstrap ***A***ggregating.
- Algorithm to repeat many times:
  a. Create a sample from your data.
  b. Train a model (e.g. a decision tree) on that sample.
- Final model comes by averaging over those many models.

Bagging

Random Forests

- We already have: Level 1 of Randomization
  - Train each model on random sample
- **Add: Level 2 of Randomization**
  - **Choose a random set of features at each decision point**

Bagging

Extremely Randomized Trees

- We already have: Level 1 and Level 2 of Randomization
  - Train each model on random sample
  - Choose a random set of features at each decision point

- **Add: Level 3 of Randomization**
  - Choose a path at random!

Boosting

Strategy

- Prevent overfitting from start
- Train an **underfit** model *m*
- Update *m* by training a new model on **the residuals of *m***

Boosting

Diachronic Learning

- Prevent overfitting from start
- Train a bad model *m*
- Improve *m* by making quantitative use of the **residuals of *m***