Supervised, Unsupervised and Reinforcement Learning in Finance

Week 1: Supervised Learning

Tree methods: Boosting

Igor Halperin

NYU Tandon School of Engineering, 2017

What is boosting

Boosting is any Ensemble method that produces a strong learner out of week learners

Two elements:

- how to pick weak learners (typically shallow CART trees)
- how to construct a strong learners

Two most popular boosting approaches:

- **AdaBoost** (Adaptive Boosting)
- **Gradient Boosting**

Hyper-parameters of Boosting algorithms:

- number of base classifiers
- hyper-parameters of base classifiers
- learning rate

Boosting as gradient descent

Boosting = gradient descent in function space (Breiman 1998)
Boosting solves the following optimization problem:

$$\min_f \sum_{i=1}^{N} L\big(y_i, f(x_i)\big) \leftarrow f(x) = w_0 + \sum_{m=1}^{M} w_m \phi_m(x,\gamma)$$

Examples of Loss functions:

Regression: the squared loss $L\big(y_i, f(x_i)\big) = \frac{1}{2}\big(y_i - f(x_i)\big)^2$ ==> L2Boosting

Regression: L1-loss $L\big(y_i, f(x_i)\big) = \left|y_i - f(x_i)\right|$ ==> Gradient Boosting

Classification: Exponential loss $L\big(y_i, f(x_i)\big) = \exp\big(-y_i f(x_i)\big)$ ==> AdaBoost

Boosting as gradient descent

Boosting = gradient descent in function space (Breiman 1998)
Boosting solves the following optimization problem:

$$\min_f \sum_{i=1}^{N} L\big(y_i, f(x_i)\big) \qquad \longleftarrow \qquad f(x) = w_0 + \sum_{m=1}^{M} w_m \phi_m(x,\gamma)$$

Solve iteratively:

- initialize $f_0(x) = \arg\min_w \sum_{i=1}^{N} L\big(y_i, f(x_i,\gamma)\big)$

(e.g. for a squared error $L\big(y_i, f(x_i)\big) = \frac{1}{2}\big(y_i - f(x_i)\big)^2$, we set $f_0(x) = \bar{y}$)

Boosting as gradient descent

Boosting = gradient descent in function space (Breiman 1998)
Boosting solves the following optimization problem:

$$\min_f \sum_{i=1}^{N} L\big(y_i, f(x_i)\big) \qquad \longleftarrow \qquad f(x) = w_0 + \sum_{m=1}^{M} w_m \phi_m(x,\gamma)$$

Solve iteratively:

- initialize $f_0(x) = \arg\min_w \sum_{i=1}^{N} L\big(y_i, f(x_i,\gamma)\big)$

(e.g. for a squared error $L\big(y_i, f(x_i)\big) = \frac{1}{2}\big(y_i - f(x_i)\big)^2$, we set $f_0(x) = \bar{y}$)

- At iteration m, compute:

$$\big(\beta_m, \gamma_m\big) = \arg\min_{\beta,\gamma} \sum_{i=1}^{N} L\big(y_i, f_{m-1}(x_i) + \beta\phi(x_i,\gamma)\big)$$

$$f_m(x) = f_{m-1}(x) + \beta_m \phi(x, \gamma_m)$$

This is called forward stage-wise additive modeling (no going back and updating earlier parameters). For more on boosting methods, see. Chap. 16.4 in Murphy)

Example: L2Boosting as gradient descent

Boosting = gradient descent in function space (Breiman 1998)
Boosting solves the following optimization problem:

$$\min_f \sum_{i=1}^{N} L\big(y_i, f(x_i)\big) \quad \leftarrow \quad f(x) = w_0 + \sum_{m=1}^{M} w_m \phi_m(x, \gamma)$$

Solve iteratively for the squared error $L\big(y_i, f(x_i)\big) = \frac{1}{2}\big(y_i - f(x_i)\big)^2$:

- initialize $f_0(x) = \bar{y}$

- At iteration m, compute:

$$\big(\beta_m, \gamma_m\big) = \arg\min_{\beta,\gamma} \sum_{i=1}^{N} \big(y_i - f_{m-1}(x_i) - \beta\phi(x_i, \gamma)\big)^2$$

$$f_m(x) = f_{m-1}(x) + \beta_m \phi(x, \gamma_m)$$

Each new basis function is optimized to fit the current residual $r_{im} = y_i - f_{m-1}(x_i)$
This is called L2Boosting, or Least Squares Boosting (Buhlmann and Yu, 2003)

Control question

Select all correct answers

1. Boosting amounts to inflating the weight of a best weak learner among all weak learners.
2. If your boosted tree overfit, you should increase the number of weak learners, which adds more noise to the problem, and hence reduces the generalization error.
3. Boosting methods typically use shallow CART trees as weak learners.
4. Boosting can be understood as optimization in a functional space. Depending on the specification of loss function, such procedure gives rise to algorithms such as L2Boosting, AdaBoost, or Gradient Boosting.

Correct answers: 3,4.