STAT562 Lecture 13 Gradient Boosting

Beidi Qiang

SIUE

Gradient boosting is a method that fits complicated models by refitting sub-models typically decision trees to either residuals or pseudo residuals. It involves three elements:

- A loss function to be optimized.
- A weak learner to make predictions.
- An additive model to add weak learners to minimize the loss function.

General Gradient Boosting Algorithm

Initialize: $F_0(x)$
At each iteration m, $m = 1, \cdots M$:

- Optimize:

$$(\beta_m, \boldsymbol{\alpha}_m) = argmin \sum_{i-1}^{n} L(y_i, F_{m-1}(x_i) + \beta h(x_i, \boldsymbol{\alpha}))$$

- Update model:

$$F_m(x) = F_{m-1}(x) + \epsilon \beta_m h(x, \boldsymbol{\alpha}_m)$$

Here,
$L(y, F)$ is a loss function.
$h(x, a)$ is a weak (base) learner with parameters $\boldsymbol{\alpha}$. ϵ is a shrinkage factor that slows the stagewise learning.

The Loss function

- Regression: (squared-error L2 loss)

$$L(y, F) = \frac{1}{n} \sum (y_i - F(x_i))^2$$

- Classification: (logistic loss)

$$L(y, p) = -(y \log(p) + (1 - y) \log(1 - p)),$$

where $p = Pr(y = 1)$

The algorithm is

- Start with $\hat{f}(x) = 0$ and residuals $r_i = y_i$.
- b=1,2,...,B, Fit a new tree $\hat{f}^b$ with d splits to the data (X, r)
- update $\hat{f}$ by adding a shrunken of the new tree: $\hat{f}(x) + \epsilon \hat{f}^b(x)$
- update the residual $r_i \to r_i - \epsilon \hat{f}^b(x_i)$.
- Repeat, and finally output

$$\hat{f}(x) = \sum_{b=1}^{B} \epsilon \hat{f}^b(x)$$

.

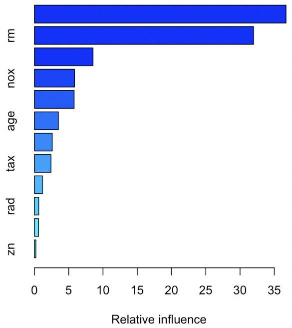
Comments

- In boosting we fit a tree using the current residual, rather than the outcome Y.
- Each of these trees can be rather small with d split.
- The shrinkage parameter ϵ, a small positive number usually 0.01 or 0.001, slows the updating process down more.
- Boosting is an approach that learns slowly, reducing the potentially overfitting issue and usually performs better.

Example: Boston data

```
> library(gbm)
Loaded gbm 2.1.8.1
> train = sample(1:506,350)
> boost.boston=gbm(medv~.,data=Boston[train,],
+                  distribution="gaussian",n.trees=5000, interaction.depth=4)
> summary(boost.boston)
            var    rel.inf
lstat     lstat 36.7083465
rm           rm 31.9682625
dis         dis  8.5481945
nox         nox  5.8495002
crim       crim  5.7897636
age         age  3.4983779
ptratio ptratio  2.5997693
tax         tax  2.4168475
chas       chas  1.1764697
rad         rad  0.6287636
indus     indus  0.6061389
zn           zn  0.2095657
> yhat.boost = predict(boost.boston,newdata=Boston[-train,])
Using 5000 trees...

> |
```



Stochastic Gradient Boosting

stochastic gradient boosting is a variation of grading boosting that reduces the correlation between the trees in the sequence in gradient boosting models. At each iteration a subsample of the training data is drawn at random (without replacement) from the full training dataset. The randomly selected subsample is then used, instead of the full sample, to fit the base learner.

A few variants of stochastic boosting that can be used:

- Subsample rows before creating each tree.
- Subsample columns before creating each tree
- Subsample columns before considering each split.