Introduction to boosting

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Erin LeDell
Instructor

Boosting Algorithms

- Adaboost
- Gradient Boosting Machine ("GBM")

Adaboost Algorithm

- Train decision tree with equally weighted observations
- Increase/Lower the weights of the observations
- Second tree is grown on weighted data
- **New model: Tree 1 + Tree 2**
- Classification error from this new 2-tree ensemble model
- Grow 3rd tree to predict the revised residuals
- Repeat this process for a specified number of iterations

Gradient Boosting Machine (GBM)

Gradient Boosting = Gradient Descent + Boosting

- Fit an additive model (ensemble) in a forward, stage-wise manner.
- In each stage, introduce a "weak learner" (e.g. decision tree) to compensate the shortcomings of existing weak learners.
- In Adaboost, "shortcomings" are identified by high-weight data points.
- In Gradient Boosting, the "shortcomings" are identified by gradients.

Advantages & Disadvantages

- Often performs better than any other algorithm
- Directly optimizes cost function

- Overfits (need to find a proper stopping point)
- Sensitive to extreme values and noises

Train a GBM Model

```r
# Train a 5000-tree GBM model
model <- gbm(formula = response ~ .,
               distribution = "bernoulli",
               data = train,
               n.trees = 5000)
```

Let's practice!

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Understanding GBM model output

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Erin LeDell
Instructor

Examine model output

```
print(credit_model)
```

```
gbm(formula = default ~ ., distribution = "bernoulli",
    data = credit_train,
    n.trees = 20000)
A gradient boosted model with bernoulli loss function.
20000 iterations were performed.
There were 16 predictors of which 16 had non-zero influence
```

Variable Importance

```
summary(credit_model)
```

```
                                      var    rel.inf
checking_balance         checking_balance 25.4977193
amount                             amount 15.5225137
credit_history             credit_history 10.6469955
...                                   ...        ...
housing                           housing  1.7772694
job                                   job  1.0878588
existing_loans_count existing_loans_count  0.4069210
phone                               phone  0.2527371
dependents                     dependents  0.1100395
```

Prediction using GBM

```
?predict.gbm
predict(model, type = "response", n.trees = 10000)
```

Let's practice!

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Tuning a GBM model

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Erin LeDell

Instructor

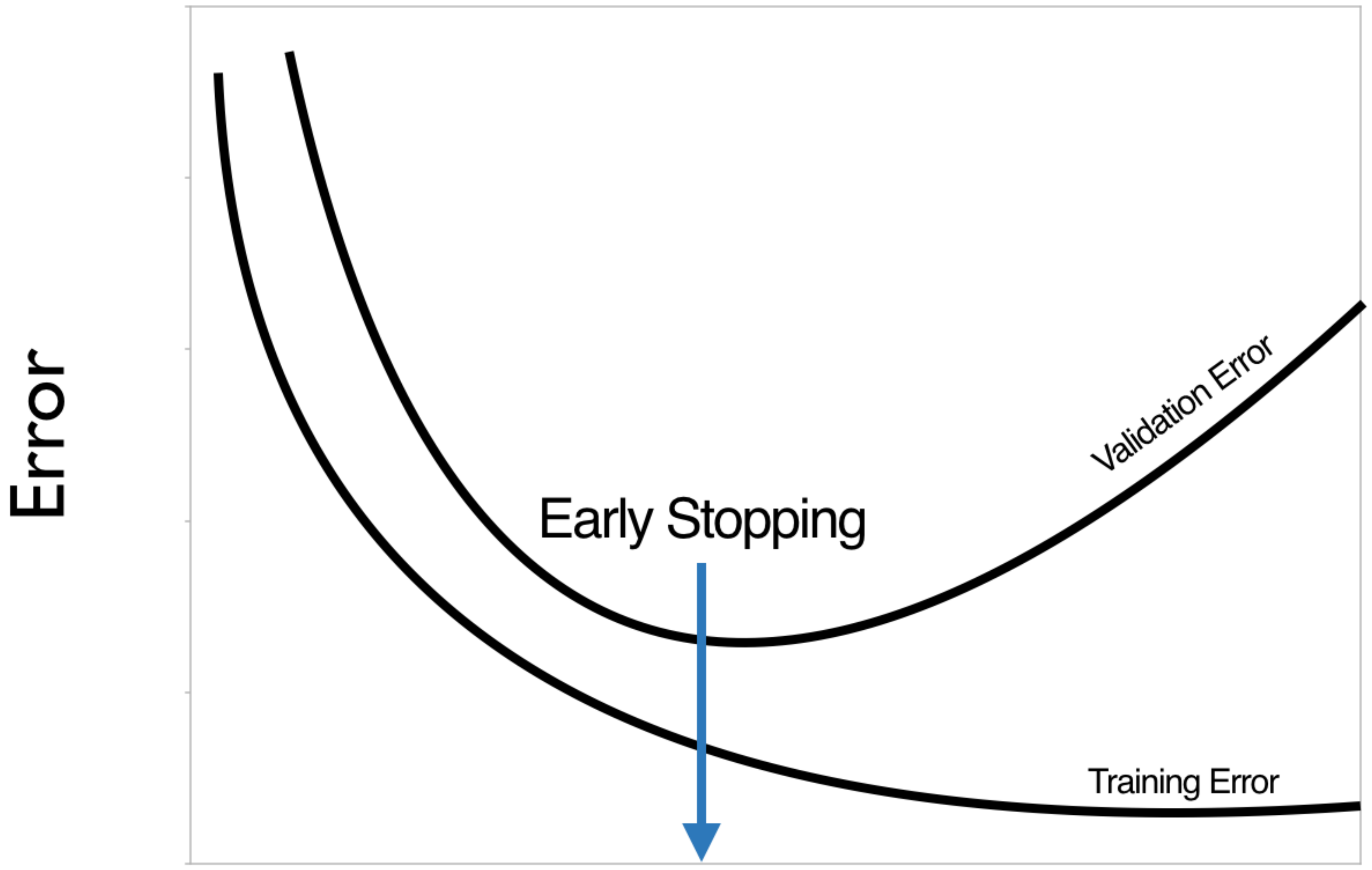
GBM Hyperparameters

- n.trees: number of trees
- bag.fraction: proportion of observations to be sampled in each tree
- n.minobsinnode: minimum number of observations in the trees

terminal nodes

- interaction.depth: maximum nodes per tree
- shrinkage: learning rate

Early Stopping



Early Stopping in GBMs

```r
# train a GBM model
model <- gbm(formula = response ~ .,
             distribution = "bernoulli",
             data = train,
             n.trees = 5000,
             cv.folds = 3)
```

```r
# get optimal ntree based on OOB error
ntree_opt_oob <- gbm.perf(model, method = "OOB")
```

```r
# get optimal ntree based on CV error
ntree_opt_cv <- gbm.perf(model, method = "cv")
```

Let's practice!

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Model comparison via ROC Curve & AUC

MACHINE LEARNING WITH TREE-BASED MODELS IN R

Erin LeDell
Instructor

Let's practice!

MACHINE LEARNING WITH TREE-BASED MODELS IN R