

Post-processing boosted regression models: model and variable selection

Greg Ridgeway

RAND Statistics Group

Santa Monica, CA

Gradient-based approaches

- Goal: find a model, $f(\mathbf{x})$, that minimizes $J(f)$
 - $J(f) = \text{E}_{y,\mathbf{x}} (y - f(\mathbf{x}))^2$
 - $J(f) = -2\text{E}_{y,\mathbf{x}} yf(\mathbf{x}) - \log(1 + \exp(f(\mathbf{x})))$
- General strategy:
 - Initialize $f(\mathbf{x}) = c$
 - Iteratively set $f(\mathbf{x}) \leftarrow f(\mathbf{x}) + g(\mathbf{x})$, where $J(f + g) < J(f)$
 - Use the gradient $\frac{J(f)}{f(\mathbf{x}_i)}$ to suggest $g(\mathbf{x})$

Examples

- IRLS (Nelder and Wedderburn, 1972)
 - $f(\mathbf{x}) \leftarrow f(\mathbf{x}) + \beta \mathbf{x}$ where $\beta \mathbf{x}$ is a particular linear approximation to $\frac{J(f)}{f(\mathbf{x})}$
- LARS (Efron, Hastie, Johnstone, Tibshirani 2004)
 - $f(\mathbf{x}) \leftarrow f(\mathbf{x}) + \lambda x_j$ where x_j is the predictor most correlated with $\frac{J(f)}{f(\mathbf{x}_i)}$. $\lambda \approx 0.0001$
- Boosting (Freund & Schapire, 1997; Friedman, 2001)
 - $f(\mathbf{x}) \leftarrow f(\mathbf{x}) + \lambda \times \text{tree}(\mathbf{x})$ where $\text{tree}(\mathbf{x})$ is a regression tree fit to $\frac{J(f)}{f(\mathbf{x}_i)}$. $\lambda \approx 0.0001$

Open issues

- Model selection (number of iterations)
 - IRLS: If $d < N$, iterate until convergence
 - LARS: Use cross-validation
 - Boosting: Use a held out test dataset
- Variable selection
 - IRLS does none, LARS essentially uses the LASSO penalty, $\sum |\beta_j|$, for selection
 - Boosting uses the LASSO for selecting a set of trees, but is not useful in eliminating redundant predictors

Generalized boosted models

- This presentation will focus on boosting as implemented in the `gbm` library

$$f(\mathbf{x}) = -0.452 + \begin{cases} -0.571 & \text{TMI} < 1.5 \\ -0.094 & \text{Use days} < 2.5 \\ 0.597 & \end{cases} + \begin{cases} -0.048 & \text{Complex beh} < 1.5 \\ -0.002 & \text{Somatic} < 0.5 \\ 0.159 & \end{cases} + \dots$$

- To predict for a new observation, predict with each tree and sum the results

Generalized boosted models

GBM's advantages include:

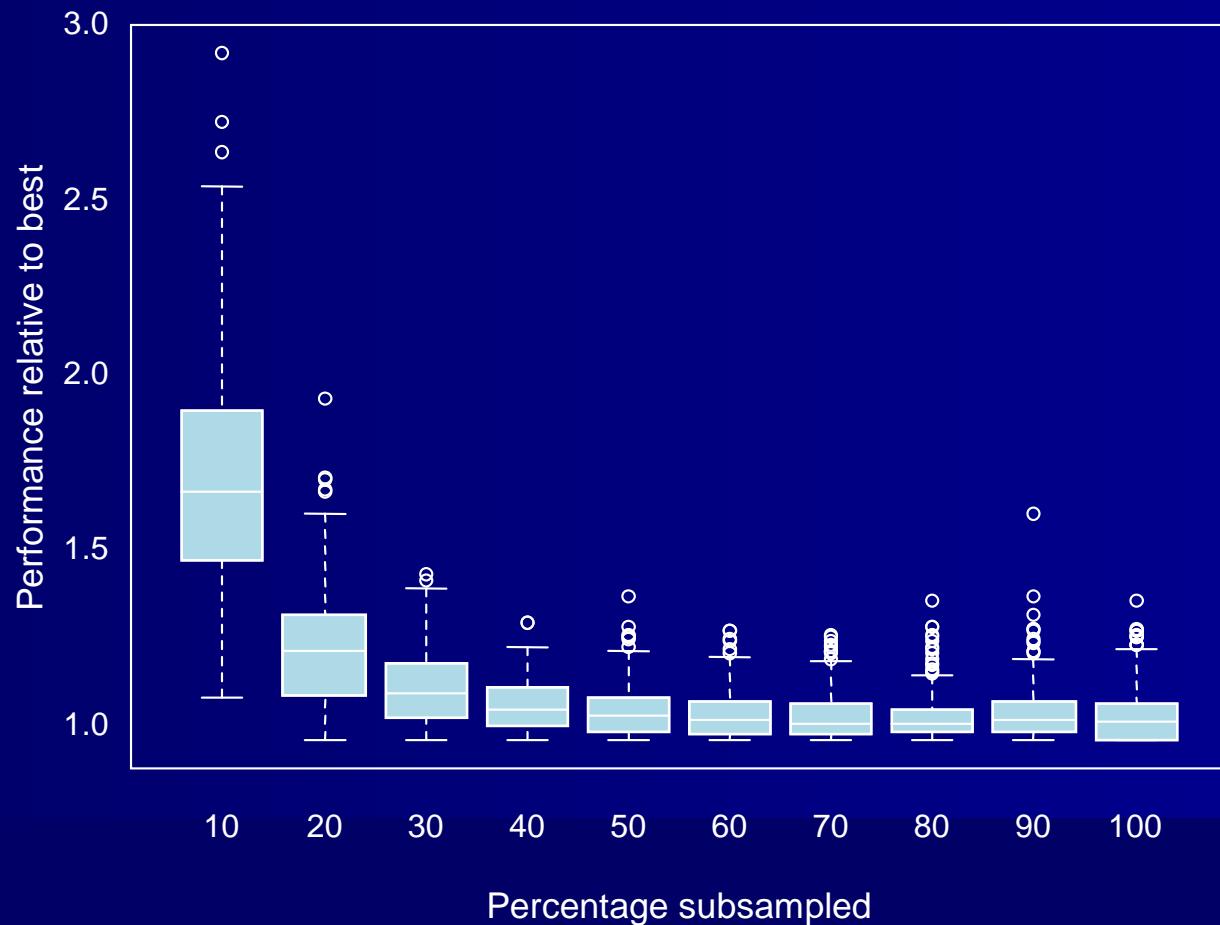
1. Excellent estimation of $f(\mathbf{x})$
2. The resulting model handles continuous, nominal, ordinal, and missing x s
3. Invariant to 1-to-1 transformations of the x s
4. Model higher interaction terms with more complex regression trees
5. Implemented in R in the `gbm` library

Estimating number of iterations

- Current practice is to set aside some fraction of observations as a test set
 - Those left out observations may have useful information on the model structure
 - Seems excessive to use 80% to estimate model structure and 20% to estimate regularization
 - In high dimensions, each left out variable is likely to be informative about a region with little data in the training set

Stochastic gradient boosting

- Friedman (2002), performance improves using a random subsample each iteration



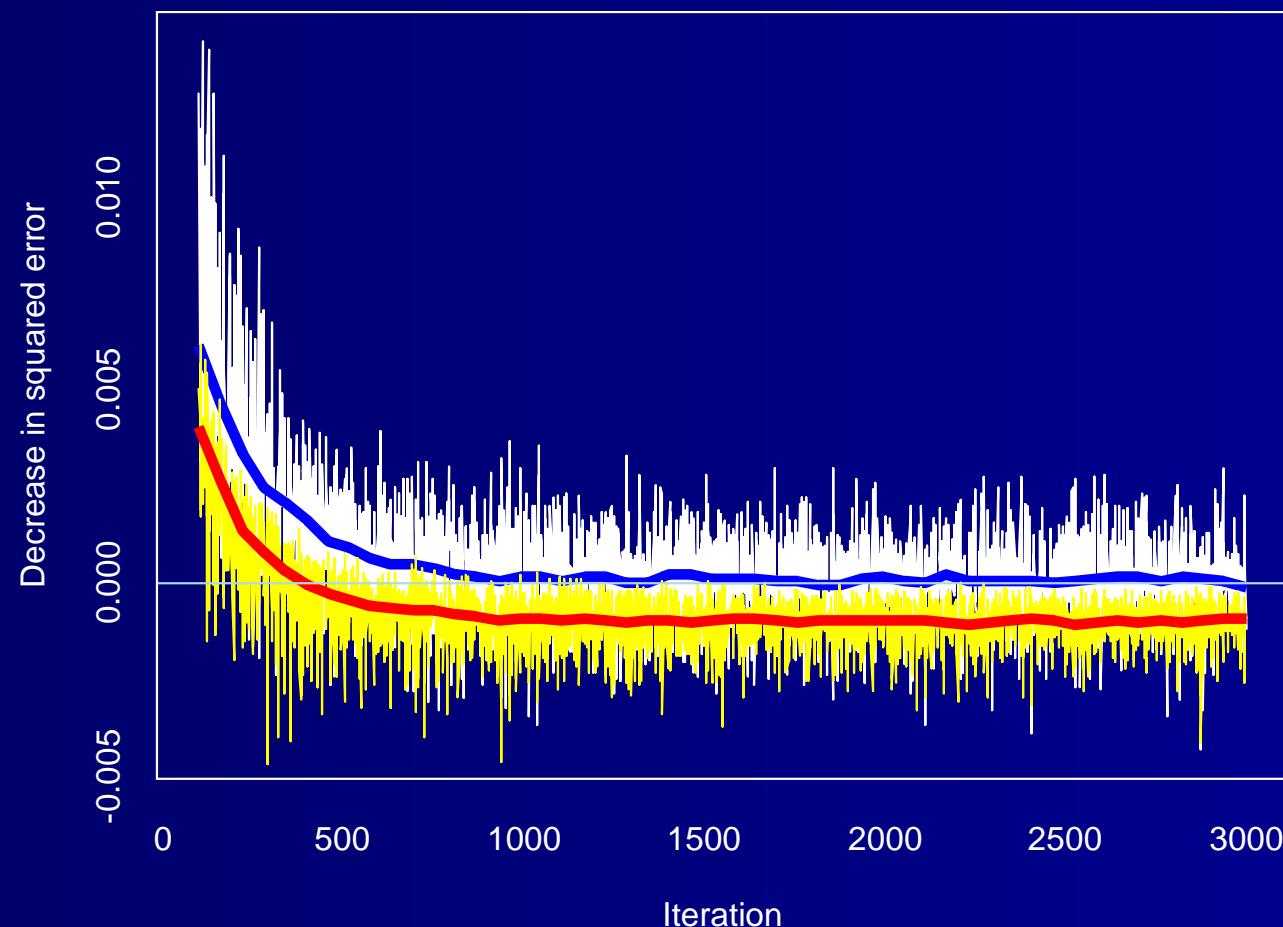
Out-of-bag estimation

- When bootstrapping, Efron (1983) & Breiman (1996) utilized the 27% of the observations not in the bootstrap sample as an independent test set
- Idea: Use those “out-of-bag” observations to estimate the improvement in predictive performance

$$\begin{aligned}\Delta J = J(f_t) - J(f_{t+1}) \approx \\ \sum_{i \in \text{OOB}} L(y_i, f_t(\mathbf{x}_i)) - L(y_i, f_t(\mathbf{x}_i) + \lambda g(\mathbf{x}_i))\end{aligned}$$

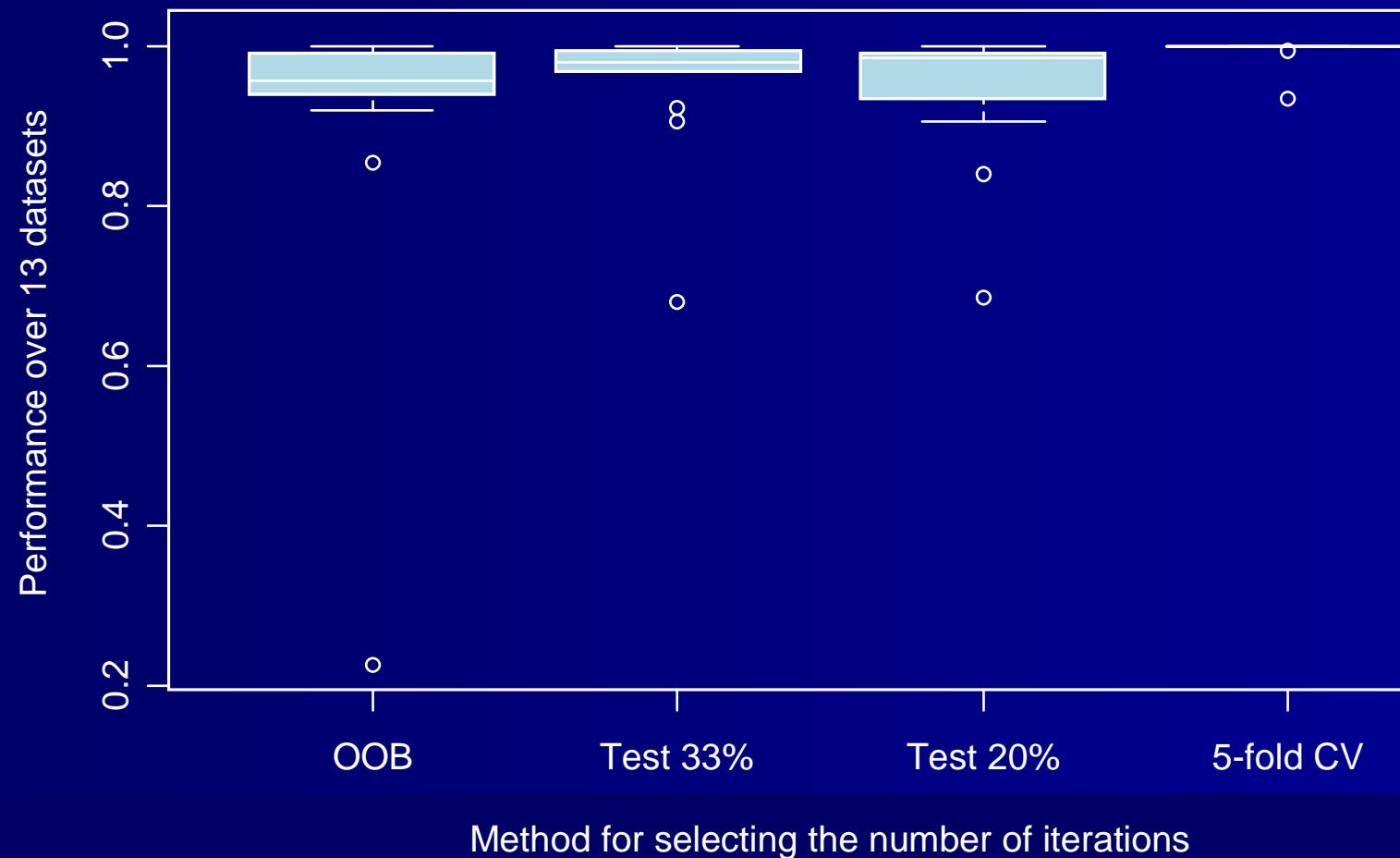
Bias in the OOB estimator

Out-of-bag underestimates performance



OOB underperforms

- Reduction in error relative to the best
- Best performer is the most expensive

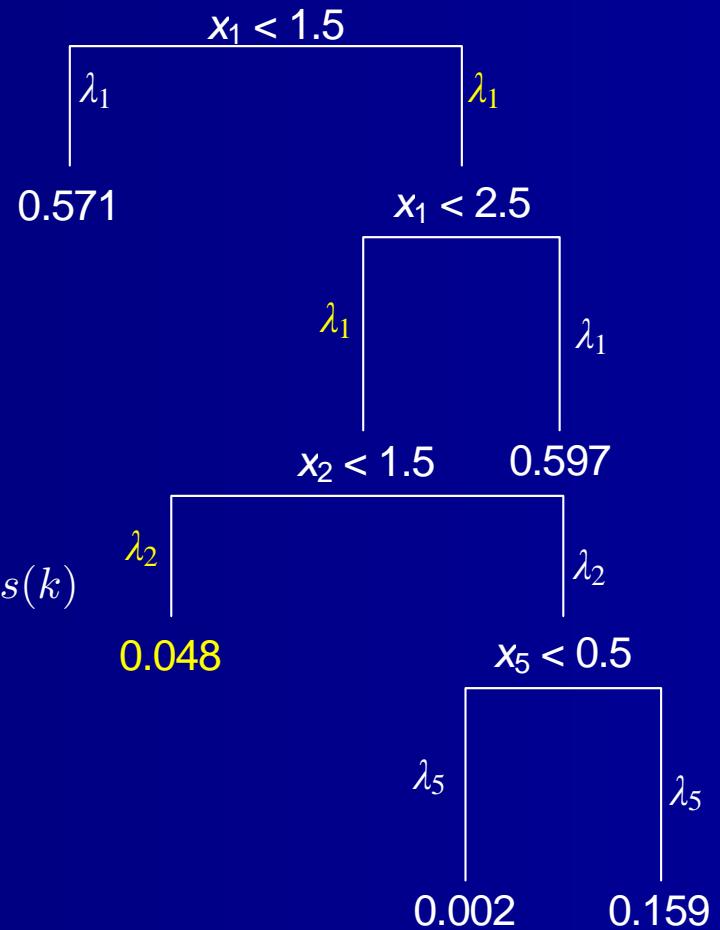


Variable selection

- Hastie and Pregibon (1990), shrinking trees
- Extending, $\lambda_j \in [0, 1]$

$$f(\mathbf{x}_i, \lambda) = \sum_{j \in \text{path}(i)} \theta_j (1 - \lambda_{s(j)}) \prod_{k < j} \lambda_{s(k)}$$

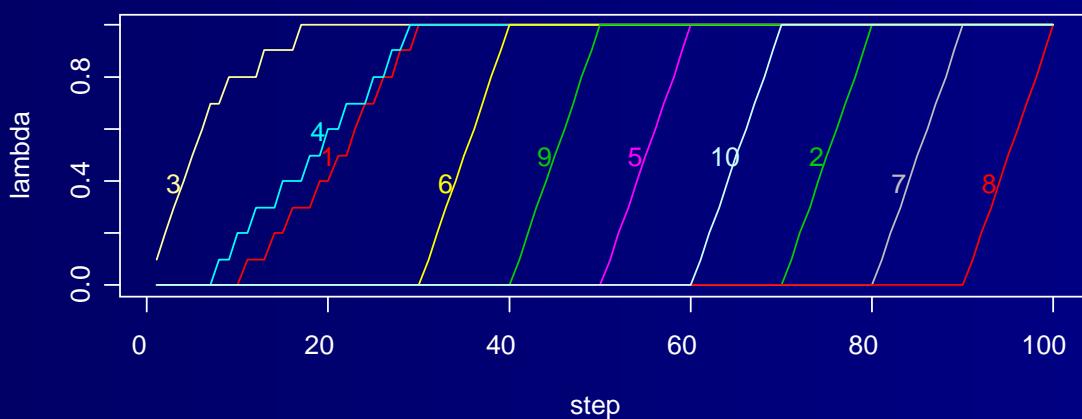
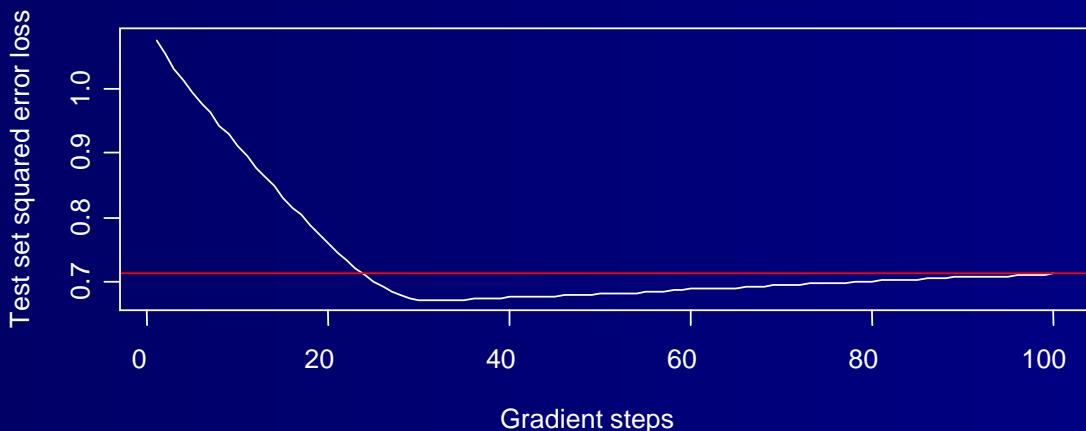
- $\frac{\partial f(\mathbf{x}_i, \lambda)}{\partial \lambda_j}$ is also computable



Variable selection

1. Set $\lambda_j = 0$ for all j
2. Let $j^* = \arg \min_j \frac{\partial J(f, \lambda)}{\partial \lambda_j}$
3. Update $\lambda_{j^*} \leftarrow \lambda_{j^*} + \epsilon$
4. Go to step 2.

Variable selection



- Simulated data where first 4 predictors affect y
- Optimal number of iterations implies use all variables
- Can do better by eliminating 7 of the predictors

R with gbm screenshot

R Gui

R Console

```
> # fit the propensity score model
> # don't use the response variable in fitting model
> l.y <- which(names(mydata)=="y")
> gbm1 <- gbm(z~.,
+               data=mydata,-l.y),
+               distribution="bernoulli",
+               n.trees=20000,
+               shrinkage=0.0005,
+               interaction.depth=4,
+               bag.fraction=0.5,
+               train.fraction=1.0,
+               n.minobsinnode=10)
Iter TrainLL
1 -0.6685
2 -0.6684
3 -0.6682
4 -0.6681
5 -0.6679
6 -0.6678
7 -0.6677
8 -0.6675
9 -0.6674
10 -0.6672
100 -0.6550
200 -0.6423
300 -0.6305
400 -0.6197
500 -0.6093
600 -0.5997
700 -0.5906
800 -0.5821
900 -0.5738
1000 -0.5662
1100 -0.5591
1200 -0.5523
1300 -0.5457
1400 -0.5395
```

R Graphics: Device 2 (inactive)

R Graphics: Device 3 (ACTIVE)

HTML Help

Hide Back Print Options

Contents Index Search

gbm(gbm): Contents
Package gbm: Objects
calibrate.plot
gbm
gbm.fit
gbm.loss
gbm.mse
gbm.object
gbm.perf
gbm.rms
permutation.test.gbm
plot.gbm
predict.gbm
pretty.gbm.tree
quantile.rug
relative.influence
summary.gbm
Package gbm: Titles

gbm(gbm)

R Documentation

Generalized Boosted Regression Modeling

Description

Fits generalized boosted regression models.

Usage

```
gbm(formula = formula(data),
     distribution = "bernoulli",
     data = list(),
     weights,
     offset = NULL,
     var.monotone = NULL,
     n.trees = 100,
     interaction.depth = 1,
     n.minobsinnode = 10,
     shrinkage = 0.001,
     bag.fraction = 0.5,
     train.fraction = 1.0,
     keep.data = TRUE,
     verbose = TRUE)

gbm.fit(x,y,
        offset = NULL,
        misc = NULL,
```

GBM Summary

- An effective nonparametric modeling tool
- Need efficient regularization of boosted models
 - Out-of-bag estimate is conservative
- Variable selection can improve predictive performance
 - On some real datasets we find post hoc selection removes no variables
 - Indicates a need to simultaneously fit model and select variables

Related talks at JSM

Dan McCaffrey

Propensity Score Estimation
with Boosted Regression

Tuesday 10:35AM, TCC-714A

Saharon Rosset

1-norm Regularization: Efficient and Effective
Wednesday 2:05PM, TCC-709