Modern Prediction Methods: Bagging and Boosting

Greg Ridgeway

Department of Statistics
University of Washington,
http://www.stat.washington.edu/greg

Outline

- Reducing modeling uncertainty through *Bayesian Model Averaging*
- Stabilizing predictors through *bagging*
- Improving performance through *boosting*
- Emerging theory illuminates empirical success
- Latest algorithms

Reasons to combine predictions

- Decreases variability in the predictions.
- Accounts for uncertainty in the model class.
- Improved accuracy on new data.

What is model uncertainty?

- Suppose we wish to predict y from predictors x.
- Given a dataset of observations, D, for a new observation with predictors $\boldsymbol{x}^*$ we want to derive the predictive distribution of y^* given $\boldsymbol{x}^*$ and D.

$$p(y^* \mid \boldsymbol{x}^*, D)$$

In practice…

- Although we want to use all the information in D to make the best estimate of y^* for an individual with covariates $\boldsymbol{x}^*$…

$$P(y_* \mid \boldsymbol{x}_*, D)$$

- In practice, however, we always use

$$P(y_* \mid \boldsymbol{x}_*, M)$$

where M is a model constructed from D.

Selecting *M*

- The process of selecting a model usually involves
  - Model class selection
    - Linear regression, tree regression, neural network
  - Variable selection
    - variable exclusion, transformation, smoothing
  - Parameter estimation
- We tend to choose the one model that fits the data or performs best as *the* model.

What's wrong with that?

- Two models may equally fit a dataset (with repect to some loss) but have different predictions.
- Competing interpretable models with equivalent performance offer ambiguious conclusions.
- Model search dilutes the evidence. "Part of the evidence is spent specifying the model."

Bayesian Model Averaging

Goal: Account for model uncertainty

Method: Use Bayes' Theorem and average the models by their posterior probabilities

Properties:

- Improves predictive performance
- Theoretically elegant
- Computationally costly

Averaging the models

Consider a set containing the K candidate models — $M_1, \ldots, M_K$.

With a few probability manipulations we can make predictions using all of them.

$$\mathrm{P}(y^* \mid \boldsymbol{x}^*, D) = \sum_k \mathrm{P}(y^* \mid \boldsymbol{x}^*, M_k)\,\mathrm{P}(M_k \mid D)$$

The probability mass for a particular prediction value of y is a weighted average of the probability mass that each model places on that value of y. The weight is based on the posterior probability of that model given the data.

Bayes' Theorem

$$P(M_k \mid D) = \frac{P(D \mid M_k)P(M_k)}{\sum_{l=1}^{K} P(D \mid M_l)P(M_l)}$$

- M_k - model
- D - data
- P(D|M_k) - integrated likelihood of M_k
- P(M_k) - prior model probability

Challenges

- The size of the model set may cause exhaustive summation to be impossible.
- The integrated likelihood of each model is usually complex.
- Specifying a prior distribution (even a non-informative one) across the space of models is non-trivial.
- Proposed solutions to these challenges often involve MCMC, BIC approximation, MLE approximation, Occam's window, Occam's razor.

Performance

- Survival model: Primary biliary cirrhosis
  - BMA vs. Stepwise regression — 2% improvement
  - BMA vs. expert selected model — 10% improvement
- Linear regression: Body fat prediction
  - BMA provides best 90% predictive coverage.
- Graphical models
  - BMA yields an improvement

BMA References

- Chris Volinsky’s BMA homepage *www.research.att.com/~volinsky/bma.html*
- J. Hoeting, D. Madigan, A. Raftery, C. Volinsky (1999). “Bayesian Model Averaging: A Practical Tutorial” (to appear in *Statistical Science*), *www.stat.colostate.edu/~jah/documents/bma2.ps*

Unstable predictors

We can always assume

$$y = f(\boldsymbol{x}) + \varepsilon, \text{ where } \mathrm{E}(\varepsilon \mid \boldsymbol{x}) = 0$$

Assume that we have a way of constructing a predictor, $\hat{\boldsymbol{f}}_D(\boldsymbol{x})$, from a dataset D.

We want to choose the estimator of f that minimizes J, squared loss for example.

$$J(\hat{f}, D) = \mathrm{E}_{y,\boldsymbol{x}}(y - \hat{f}_D(\boldsymbol{x}))^2$$

Bias-variance decomposition

If we could average over all possible datasets, let the average prediction be

$$\underline{\lambda}(\boldsymbol{x}) = \mathrm{E}^{D}\, \check{\lambda}^{D}(\boldsymbol{x})$$

The average prediction error over all datasets that we might see is decomposable

Bias-variance decomposition

The squared-error averaged over all datasets…

$$
\begin{aligned}
E_D\, \ell(\hat{F}, D) = \quad & E\,\varepsilon^2 + && \text{noise} \\
& E_x\left(F(\mathbf{x}) - \bar{F}(\mathbf{x})\right)^2 + && \text{bias} \\
& E_{x,D}\left(\hat{F}_D(\mathbf{x}) - \bar{F}(\mathbf{x})\right)^2 && \text{variance}
\end{aligned}
$$

where $\bar{F}(\mathbf{x}) = E_D\, \hat{F}_D(\mathbf{x})$

Combining Multiple Models

- Boosting
- Bagging
- Adaptive Bagging
- Bumping
- Bundling
- Stacking
- Leveraging
- Ensemble learning
- Pasting
- Crumpling
- Arcing
- Bayesian Model Averaging
- Group Method of Data Handling

Words of Wisdom

- "[With proportional representation]… there would be a fair comparison of intellectual strength."

  John Stuart Mill - *Representative Government* (1861)

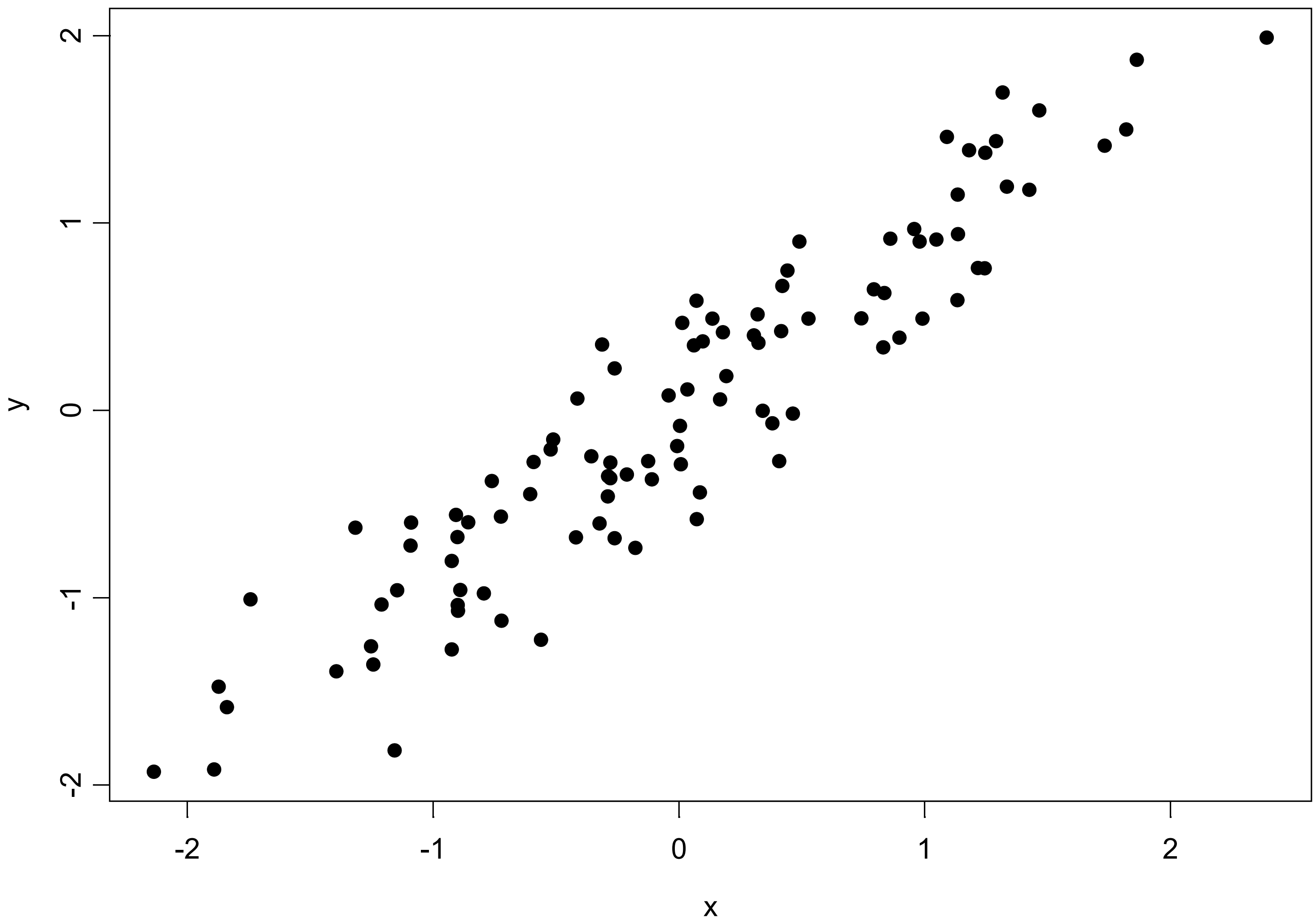
- "Don't put all your eggs in one basket."

  Miguel de Cervantes - *Don Quixote* (1605)

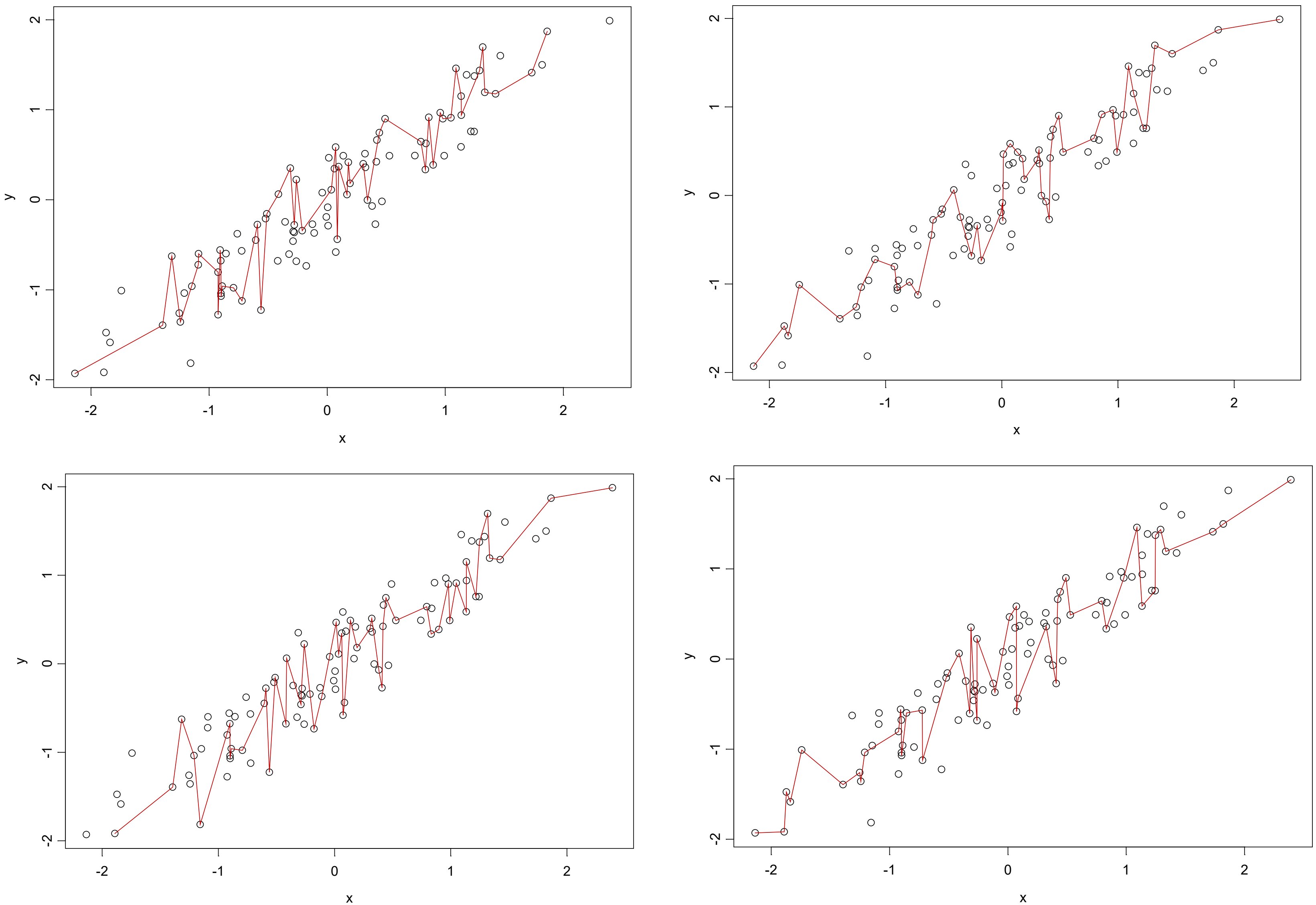
- "For by wise guidance you will wage war, And in abundance of counselors there is victory."

  King Solomon - Proverbs 24:6 (930 BC?)

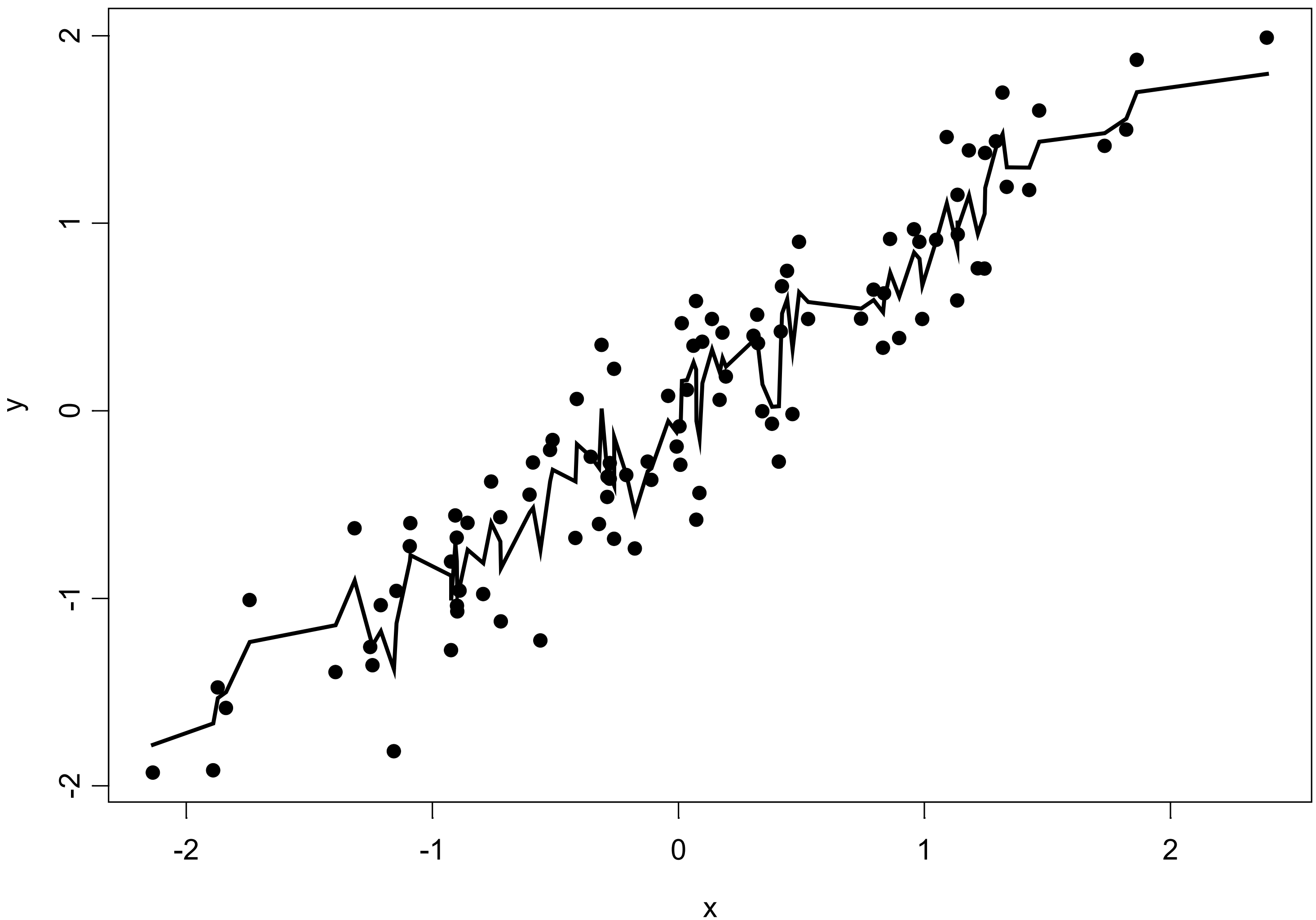
A very important model



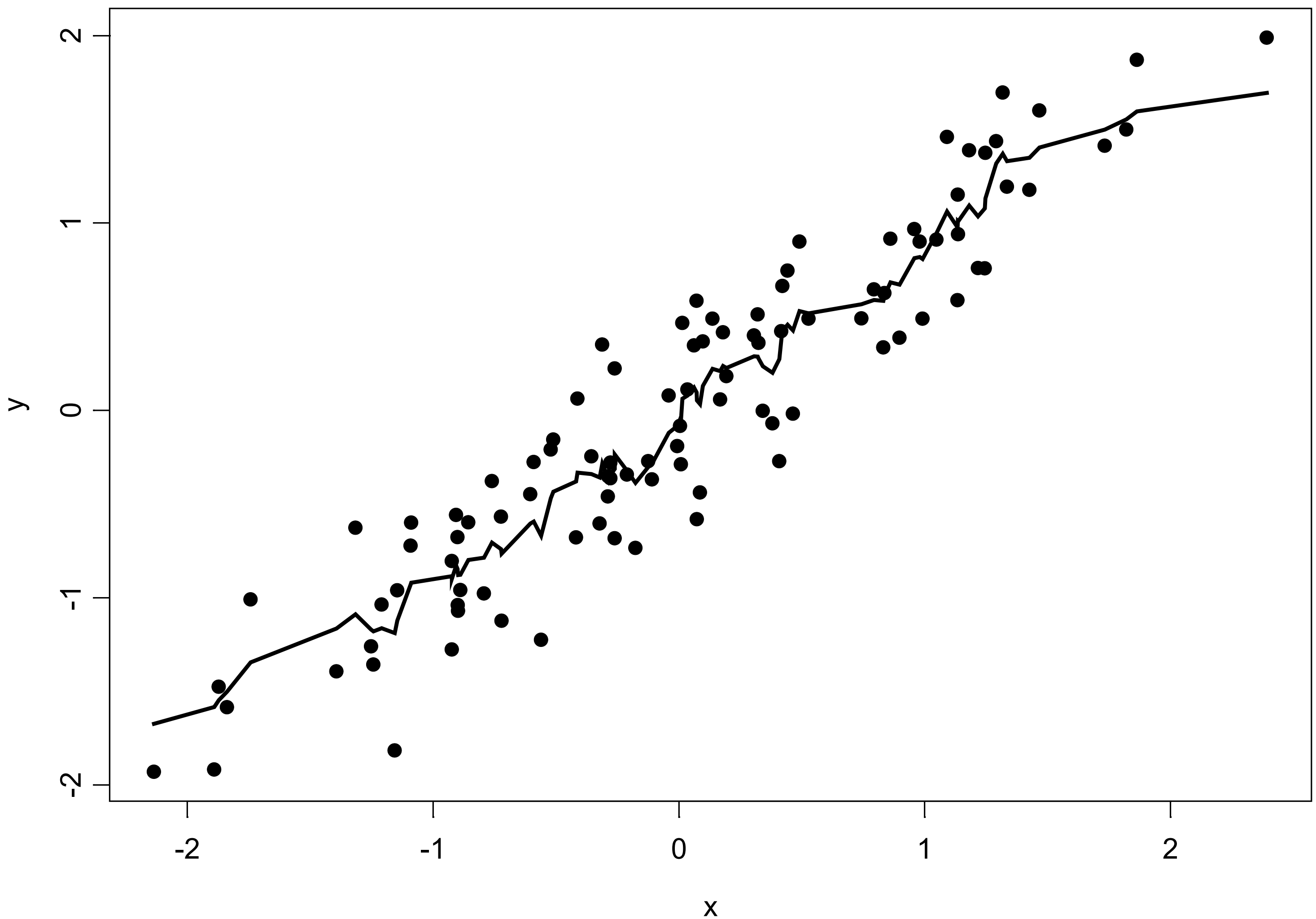
On half-samples…



Average over half-samples



Average over quarter-samples



Bias vs. Variance

Variance reduction

- Limit $\hat{F}(\mathbf{x})$ to depend on a small number of parameters
- Enforce smoothness constraints
- Additive models

Bias reduction

- Allow $\hat{F}(\mathbf{x})$ to be more flexible (more parameters)
- Model complex interactions

Bagging (*B*ootstrap *Agg*regating)

Goal: Variance reduction

Method: Create bootstrap replicates of the dataset and fit a model to each. Average the predictions of each model.

Properties:

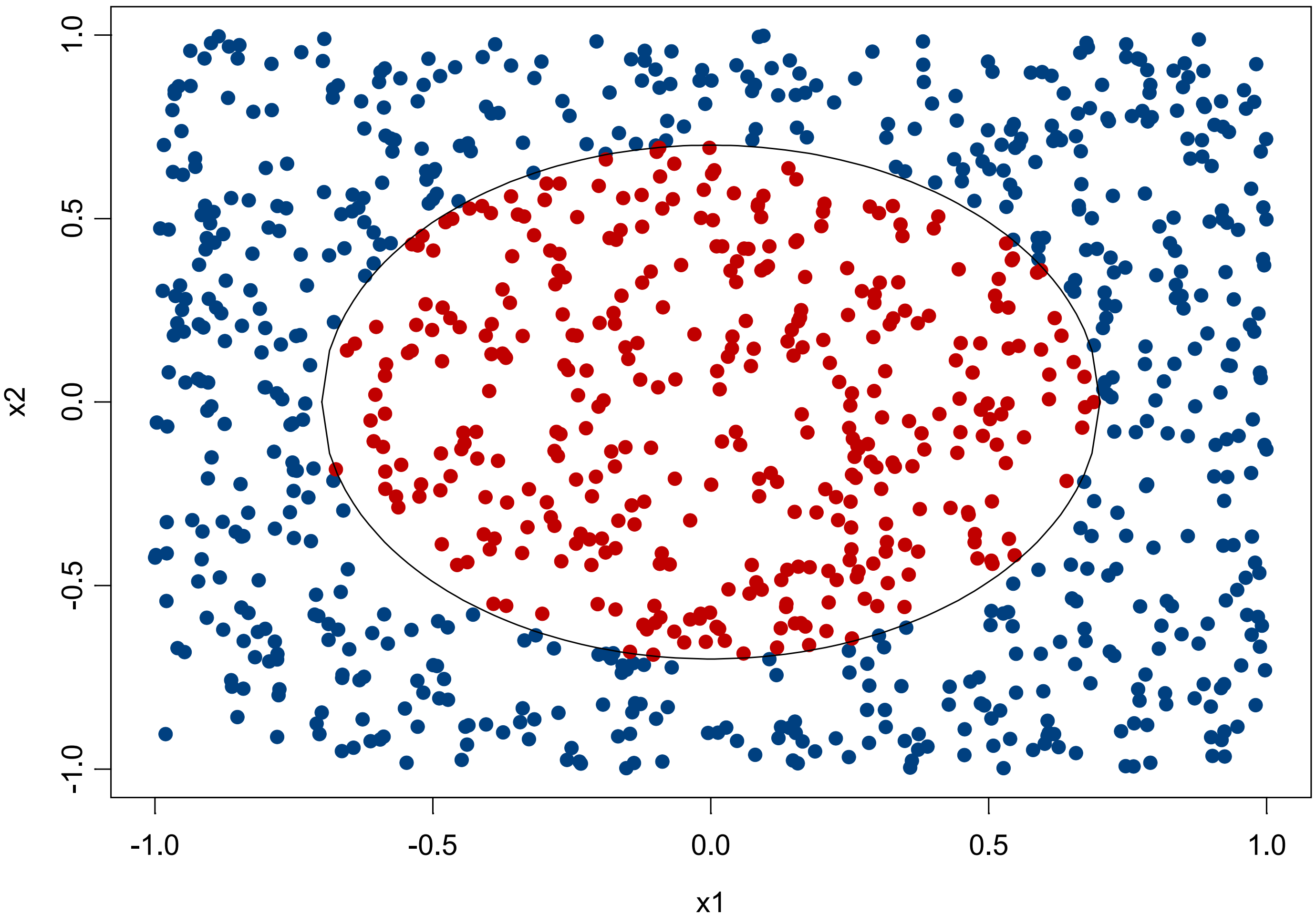
- Stabilizes “unstable” methods
- Easy to implement, parallelizable
- Theory is not fully explained

Bagging algorithm

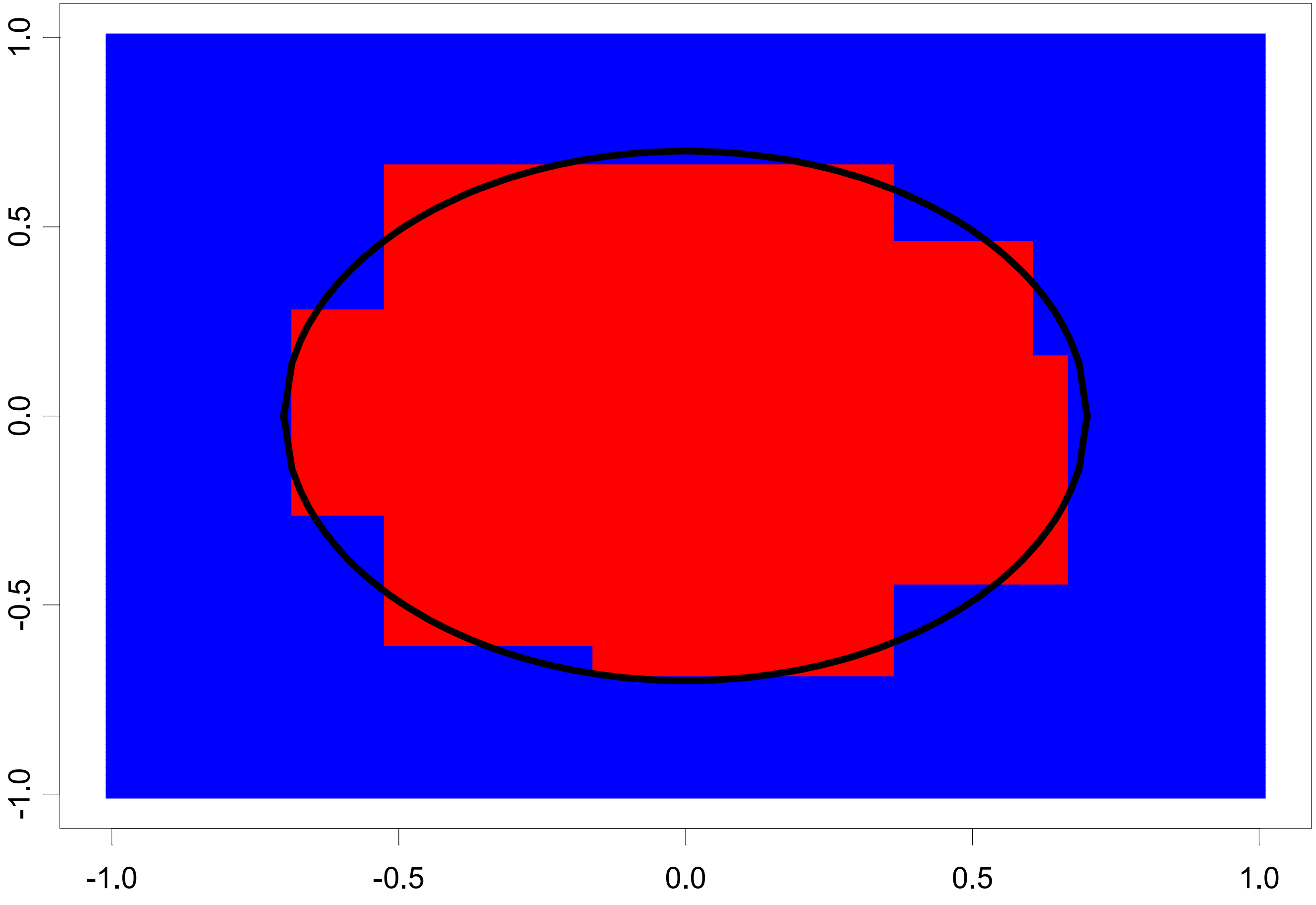
1. Create K bootstrap replicates of the dataset.
2. Fit a model to each of the replicates.
3. Average (or vote) the predictions of the K models.

Bootstrapping simulates the stream of infinite datasets in the bias-variance decomposition.

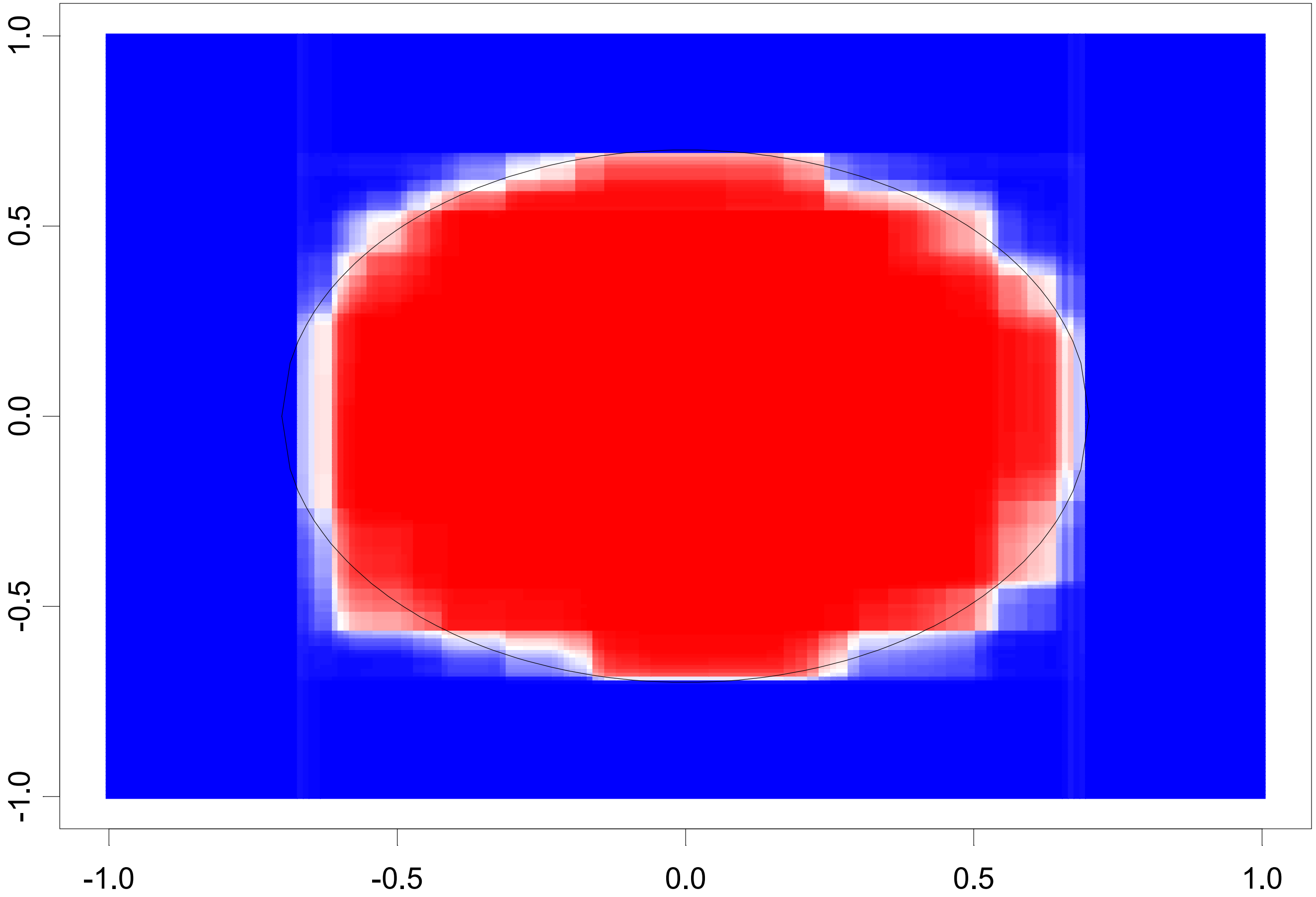
Bagging Example



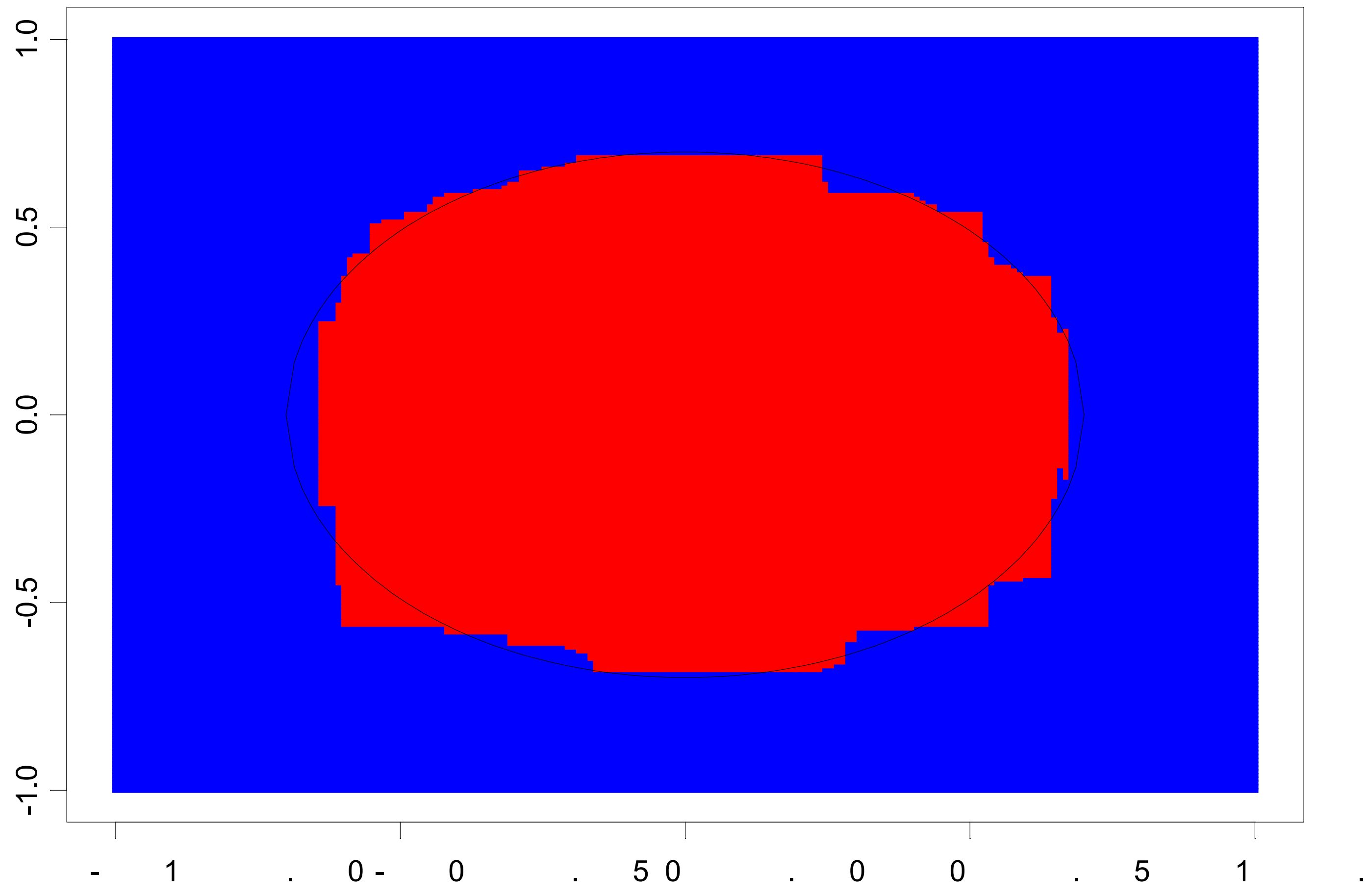
CART decision boundary



100 bagged trees

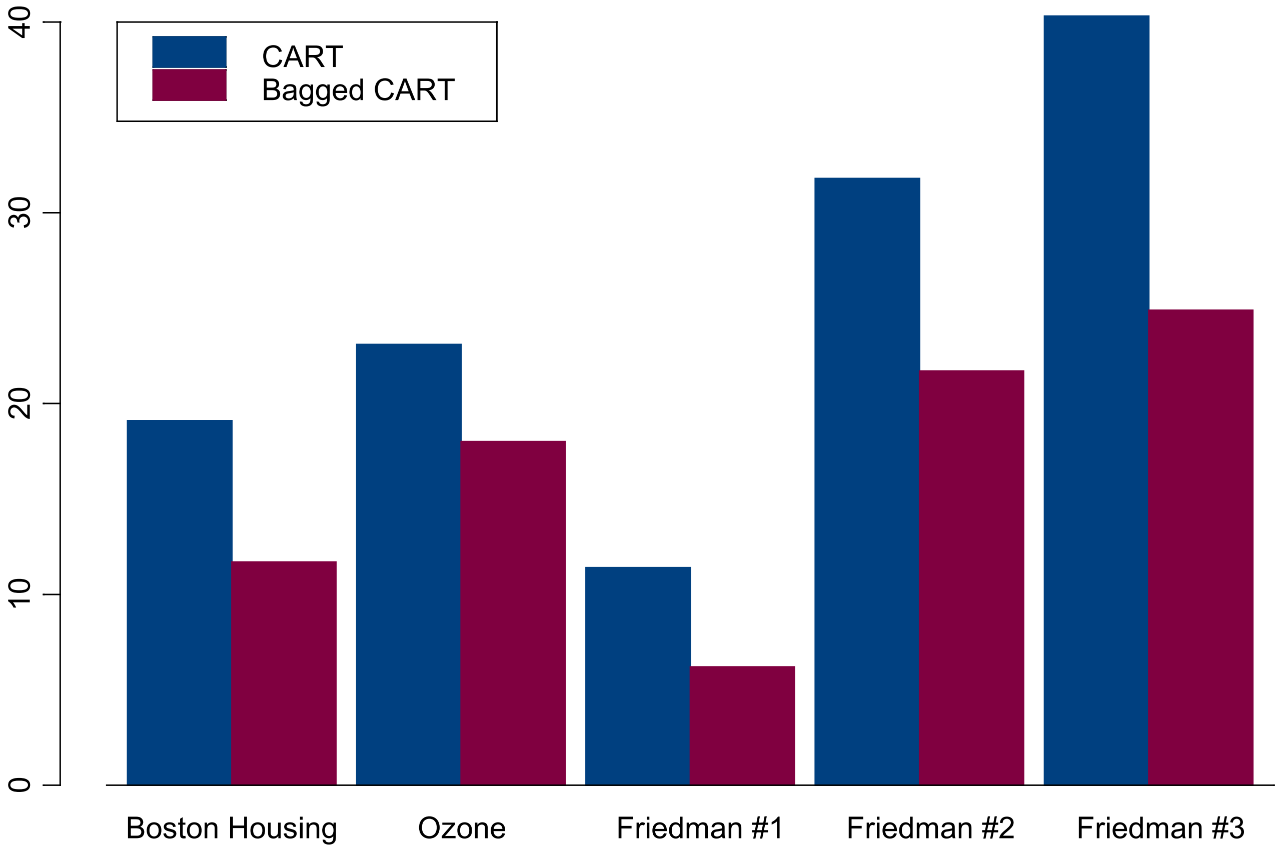


Bagged tree decision boundary



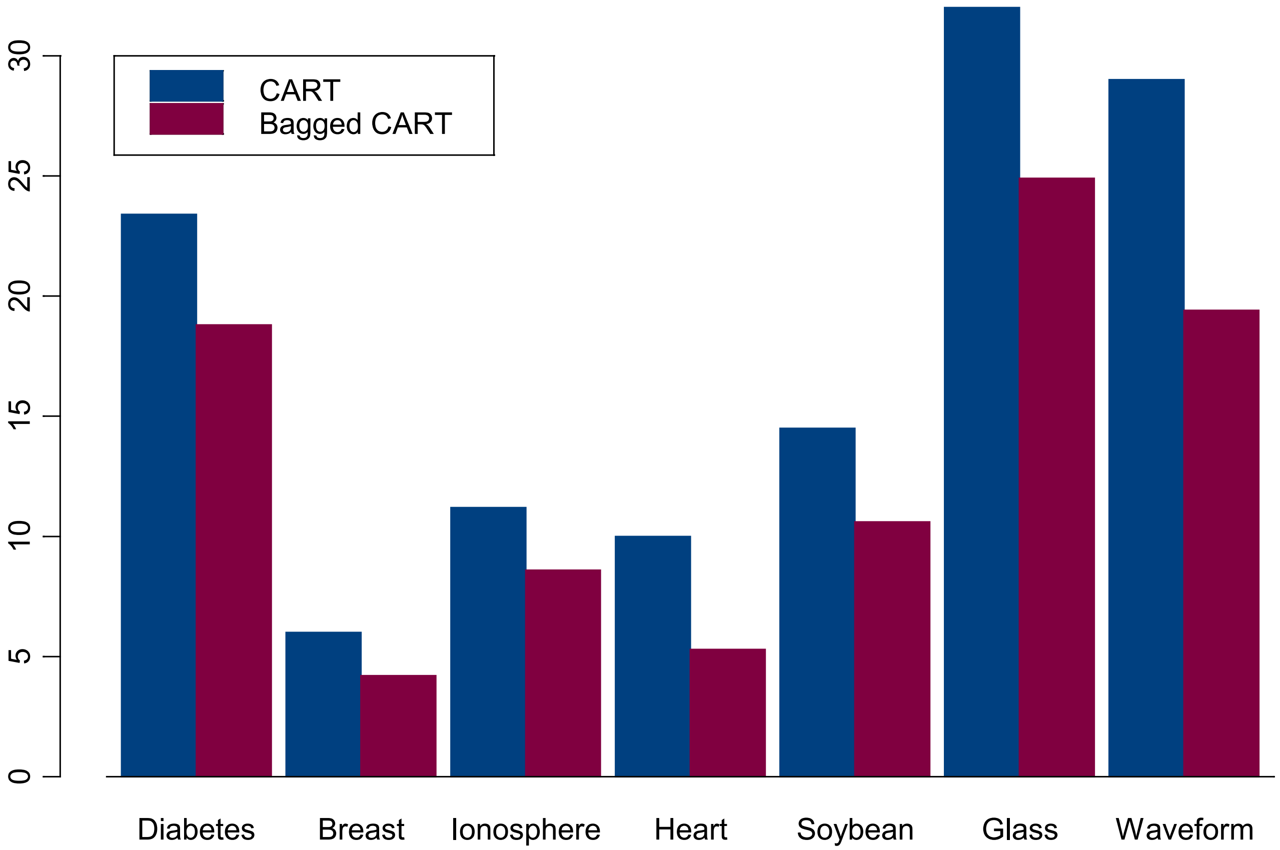
Regression results

Squared error loss



Classification results

Misclassification rates



Bagging References

- Leo Breiman's homepage *www.stat.berkeley.edu/users/breiman/*
- Breiman, L. (1996) "Bagging Predictors," *Machine Learning*, 26:2, 123-140.
- Friedman, J. and P. Hall (1999) "On Bagging and Nonlinear Estimation" *www.stat.stanford.edu/~jhf*

Boosting

Goal: Improve misclassification rates

Method: Sequentially fit models, each more heavily weighting those observations poorly predicted by the previous model

Properties:

- Bias and variance reduction
- Easy to implement
- Theory is not fully (but almost) explained

Origin of Boosting

Classification problems

$$\{y, \boldsymbol{x}\}_i\ ,\ i = 1,\ldots,n$$

$$y \in \{0, 1\}$$

The task - construct a function,

$$F(\boldsymbol{x}) : \boldsymbol{x} \to \{0, 1\}$$

so that F minimizes misclassification error.

Generic boosting algorithm

Equally weight the observations $(y,\boldsymbol{x})_i$

For t in $1,\ldots,T$

  Using the weights, fit a classifier $f_t(\boldsymbol{x}) \rightarrow y$

  Upweight the poorly predicted observations

  Downweight the well-predicted observations

Merge $f_1,\ldots,f_T$ to form the boosted classifier

Real AdaBoost

Schapire & Singer 1998

$y_i \in \{-1,1\}$, $w_i = 1/N$

For t in 1,…,T do

1. Estimate $P_w(y = 1|\boldsymbol{x})$.
2. Set $f_t(\boldsymbol{x}) = \frac{1}{2}\log \frac{\hat{P}_w(y=1\,|\,\boldsymbol{x})}{\hat{P}_w(y=-1\,|\,\boldsymbol{x})}$
3. $\boldsymbol{w}_i \leftarrow w_i \exp\left(-y_i f_t(\boldsymbol{x}_i)\right)$ and renormalize

Output the classifier $\boldsymbol{F}(\boldsymbol{x}) = \operatorname{sign}\left(\sum_t f_t(\boldsymbol{x})\right)$

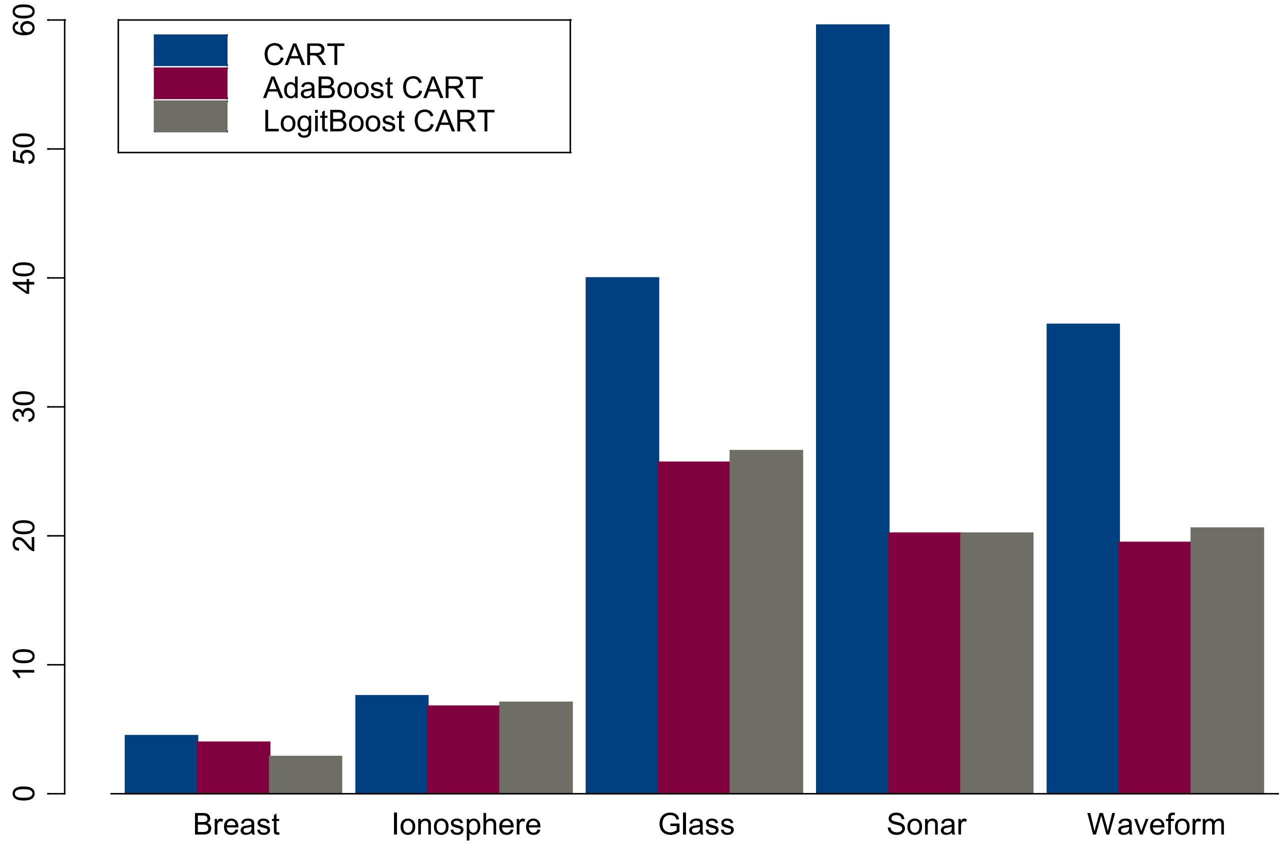
AdaBoost’s Performance

Freund & Schapire [1996]

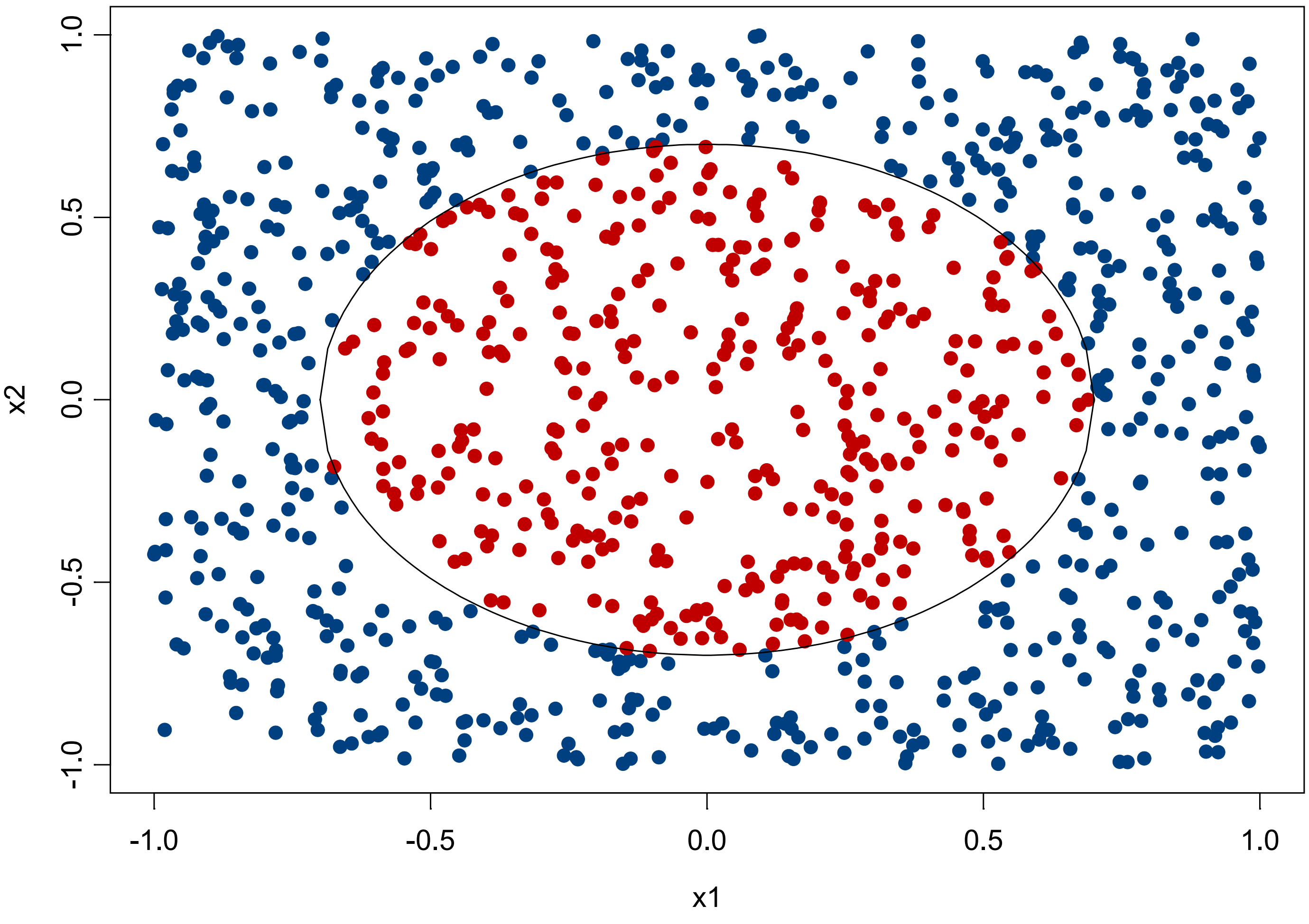
- Leo Breiman - AdaBoost with trees is the “best off-the-shelf classifier in the world.”
- Performs well with many base classifiers and in a variety of problem domains.
- AdaBoost is generally slow to overfit.
- Boosted naïve Bayes tied for first place in the 1997 KDD Cup. (Elkan [1997])
- Boosted naïve Bayes is a scalable, interpretable classifier (Ridgeway, *et al* [1998]).

Misclassification rates

Friedman, Hastie, Tibshirani [1998]

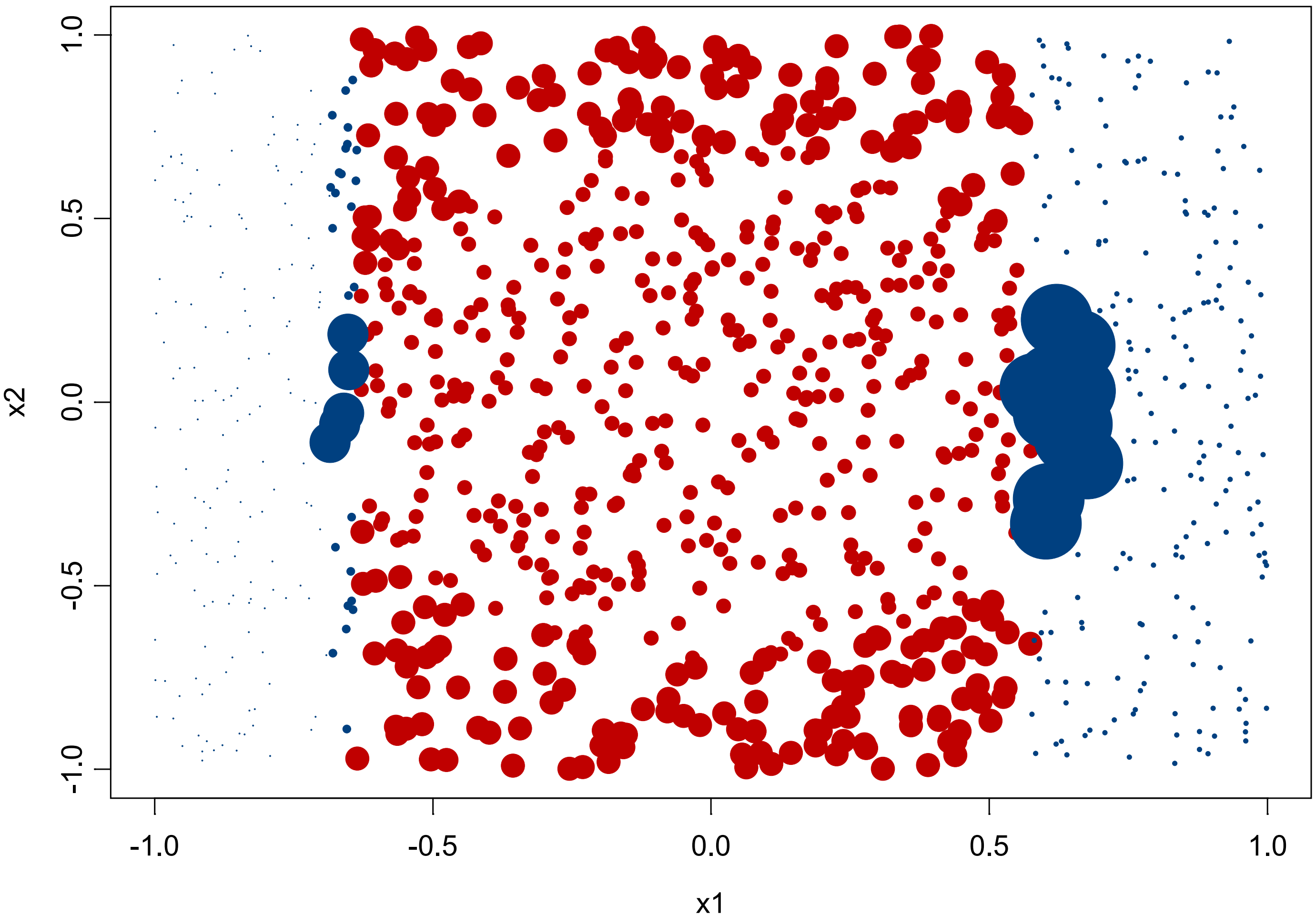


Boosting Example

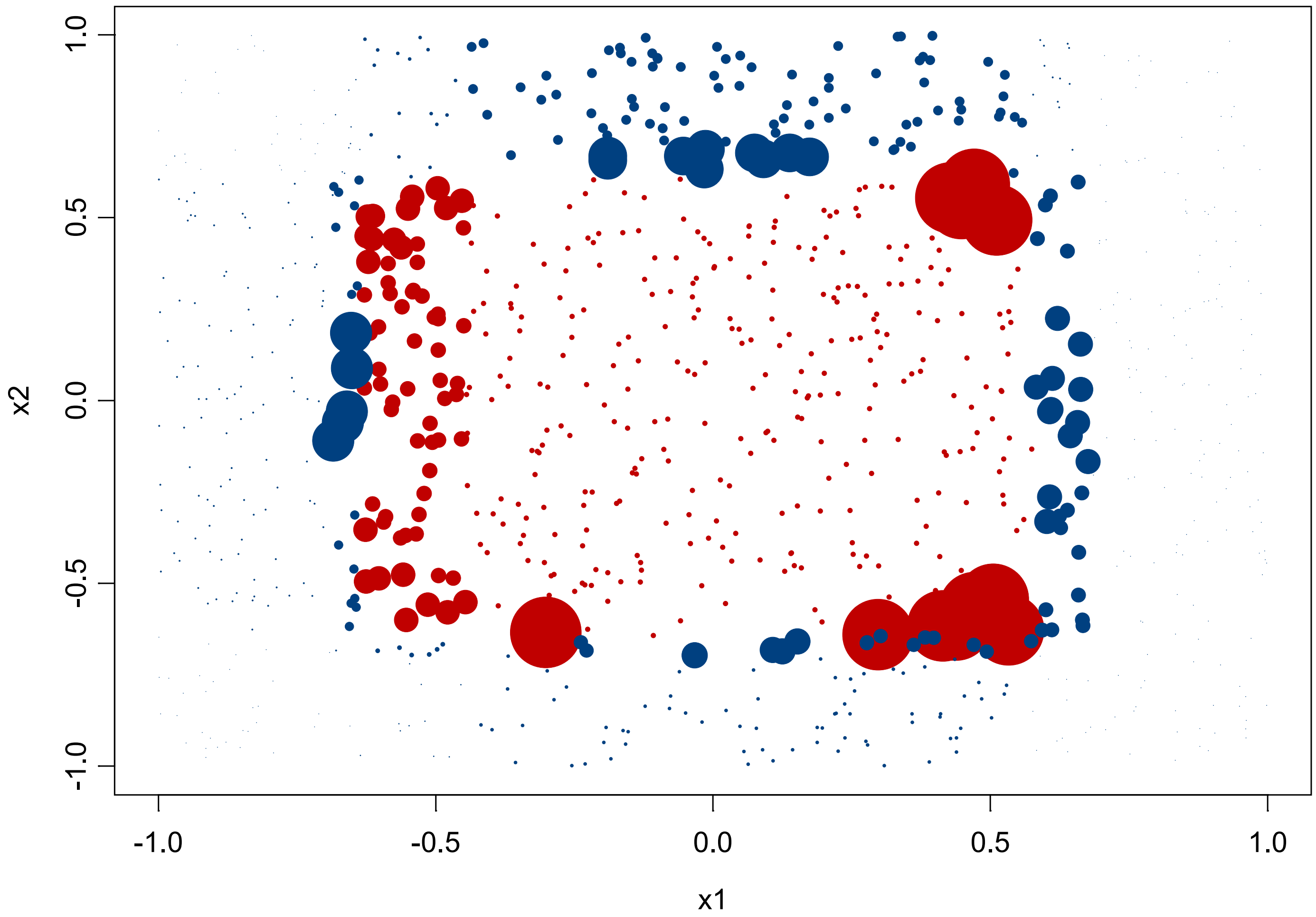


After one iteration

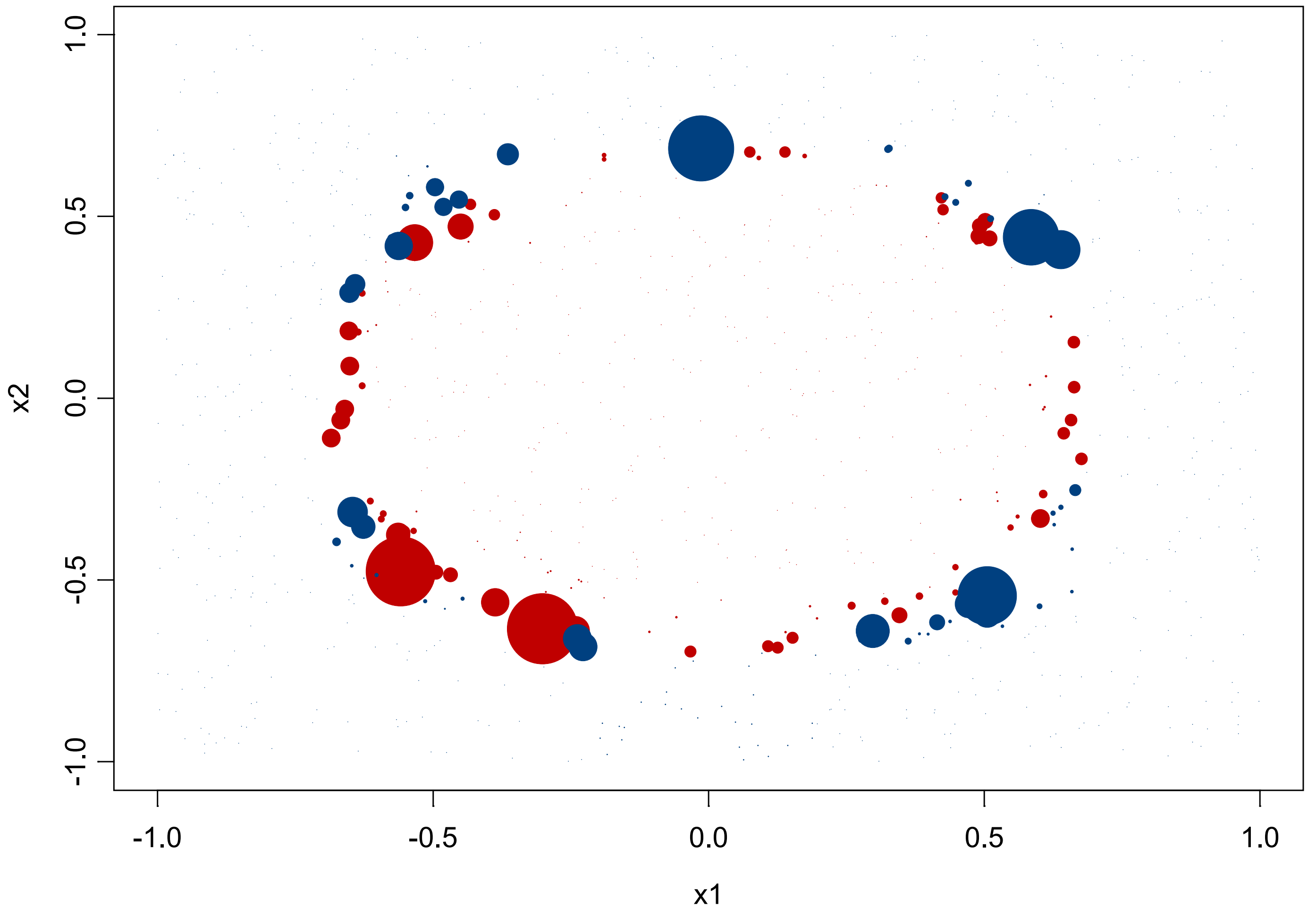
CART splits, larger points have great weight



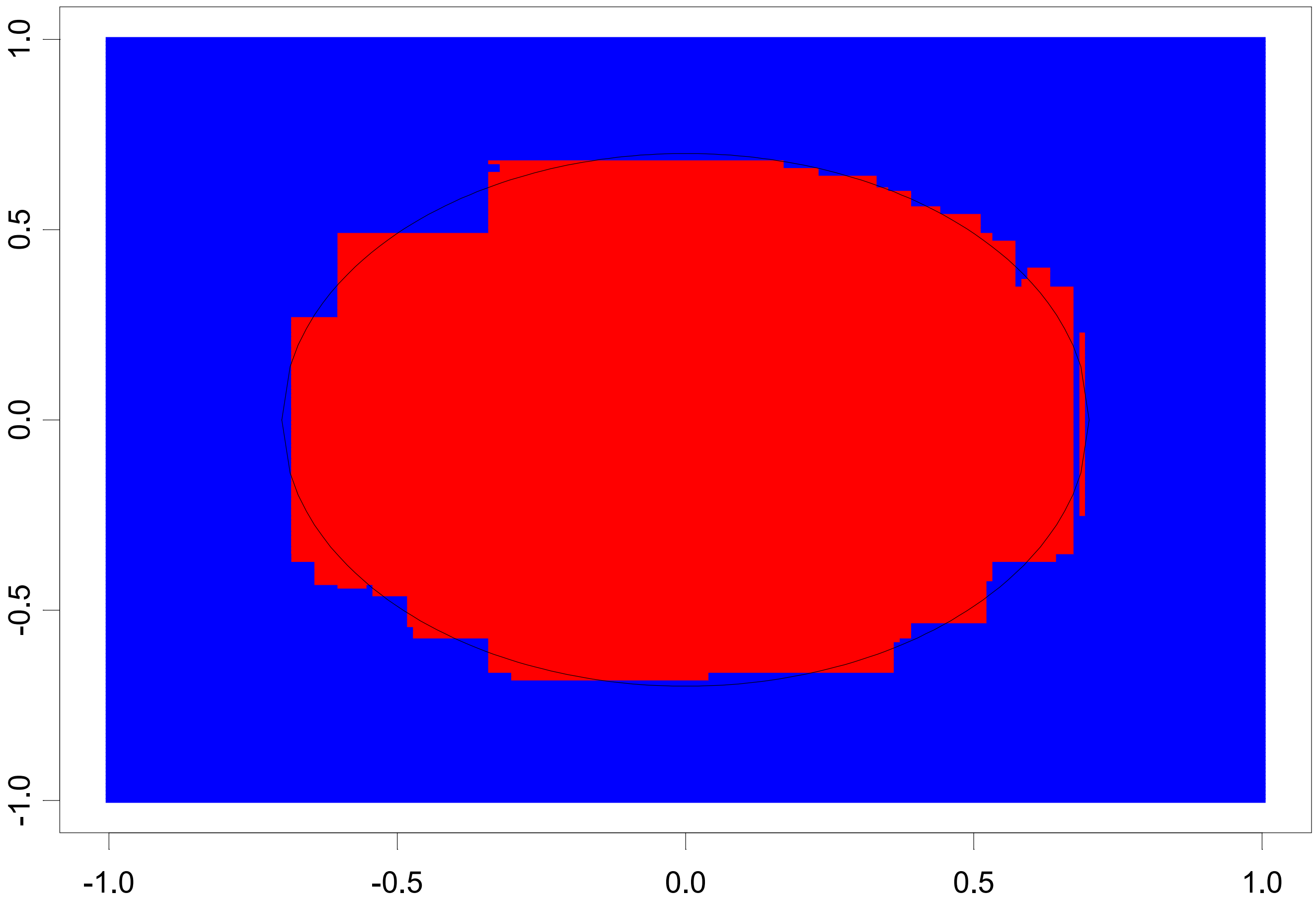
After 3 iterations



After 20 iterations



Decision boundary after 100 iterations



Boosting as optimization

- Friedman, Hastie, Tibshirani [1998] - AdaBoost is an optimization method for finding a classifier.
- Let $y \in \{-1,1\}$, $F(x) \in (-\infty,\infty)$

$$J(F) = E\left(e^{-yF(x)} \mid x\right)$$

$$J(F+f) = E\left(e^{-y(F(x)+f(x))} \mid x\right)$$

Criterion

- $E(e^{-yF(x)})$ bounds the misclassification rate.

$$I(yF(x) < 0) < e^{-yF(x)}$$

- The minimizer of $E(e^{-yF(x)})$ coincides with the maximizer of the expected Bernoulli likelihood.

$$E\big(\ell(p(x), y)\big) = -E\log(1 + e^{-2yF(x)})$$

Boosting References

- Rob Schapire's homepage
  www.research.att.com/~schapire
  - Freund, Y. and R. Schapire (1996). "Experiments with a new boosting algorithm," Machine Learning: Proceedings of the 13th International Conference, 148-156.
- Jerry Friedman's homepage
  www.stat.stanford.edu/~jhf
  - Friedman, J., T. Hastie, R. Tibshirani (1998). "Additive Logistic Regression: a statistical view of boosting," Technical report, Statistics Department, Stanford University.

In general, combining ("bundling") predictions consists of two steps:

1) Constructing varied models, and
2) Combining their predictions

Generate component models by varying:

- Case Weights
- Data Values
- Guiding Parameters
- Variable Subsets

Combine estimates using:

- Estimator Weights
- Voting
- Advisor Perceptrons
- Partitions of Design Space, X

Advanced techniques

- Stochastic gradient boosting
- Adaptive bagging
- Example regression and classification results

Stochastic Gradient Boosting

Goal: Non-parametric function estimation

Method: Cast the problem as optimization and use gradient ascent to obtain predictor

Properties:

- Bias and variance reduction
- Widely applicable
- Can make use of existing algorithms
- Many tuning parameters

Improving boosting

- Boosting usually has the form

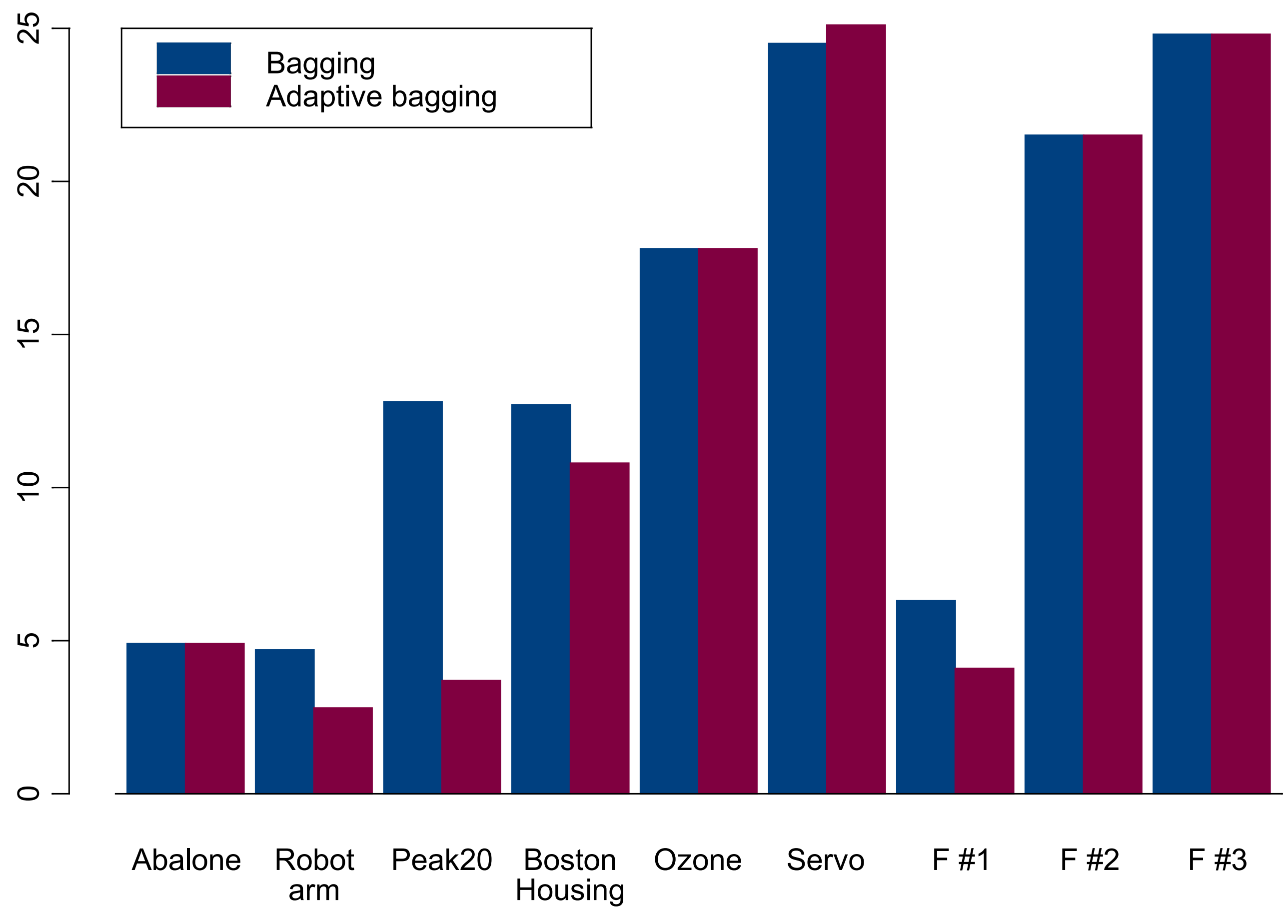
$$F(x) \leftarrow F(x) + E(z \mid x)$$

Improve by...

- Using a bagged estimate of the expectation.
- “Robustifying” the expectation.
- Trimming observations with small weights.
- BMA to compute the expectation

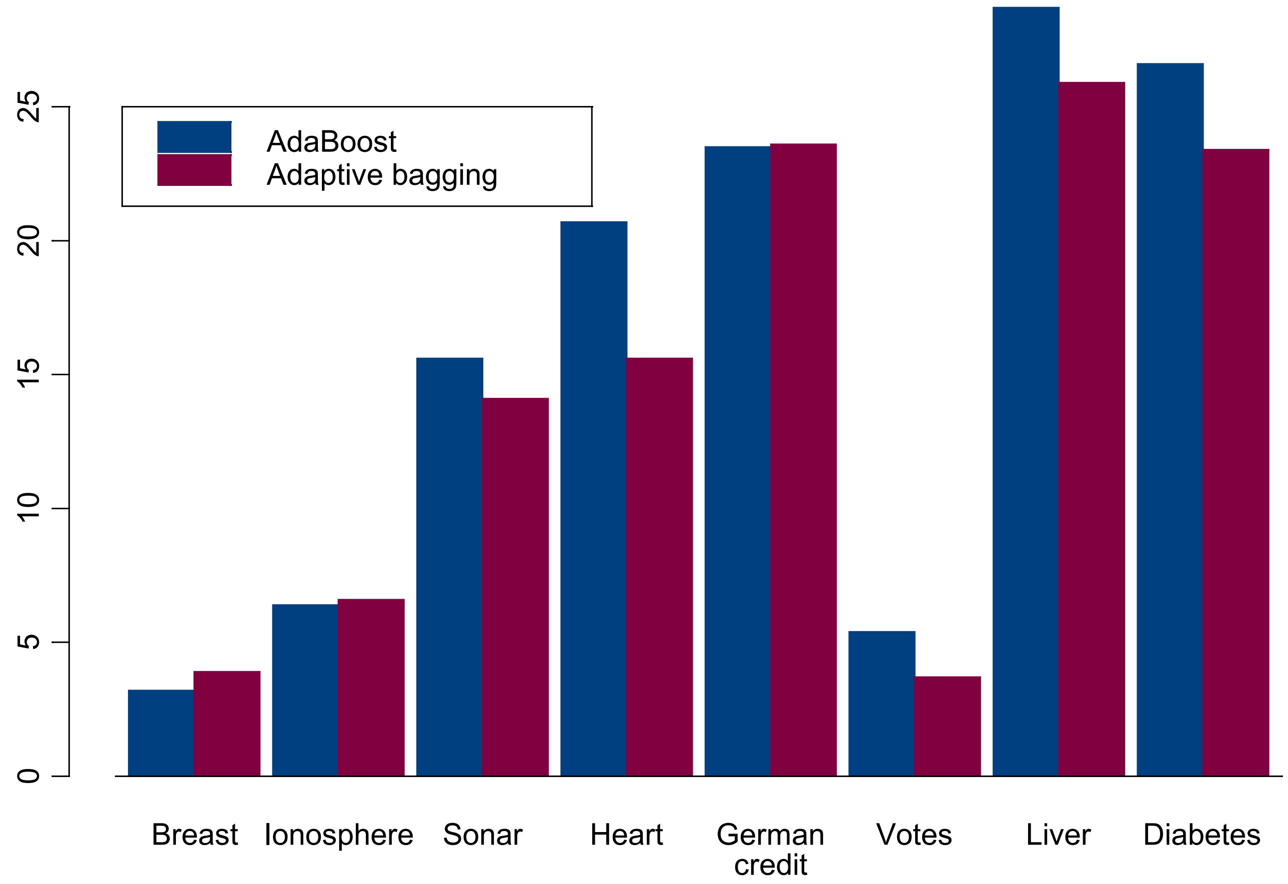
Regression results

Squared error loss



Classification results

Misclassification rates



Stochastic gradient boosting offers...

- Application to likelihood based models (GLM, Cox models)
- Bias reduction - non-linear fitting
- Massive datasets - bagging, trimming
- Variance reduction - bagging
- Interpretability - additive models
- High-dimensional regression - trees
- Robust regression

SGB References

- Friedman, J. (1999). “Greedy function approximation: a gradient boosting machine,” Technical report, Dept. of Statistics, Stanford University.
- Friedman, J. (1999). “Stochastic gradient boosting,” Technical report, Dept. of Statistics, Stanford University.