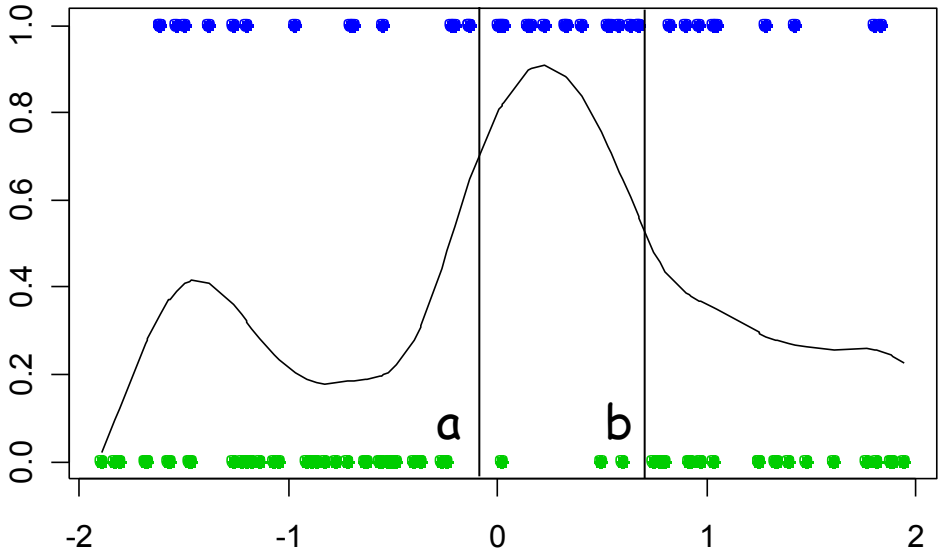
Data Mining Trees
ARF (Active Region Finder)

Naive thought: For the *j*th descriptor variable x_j, an "interesting" subset $\{a<x_{ji}<b\}$ is one such that

$$p = \text{Prob}[\, Z=1 \mid a<x_{ji}<b \,]$$

is much larger than

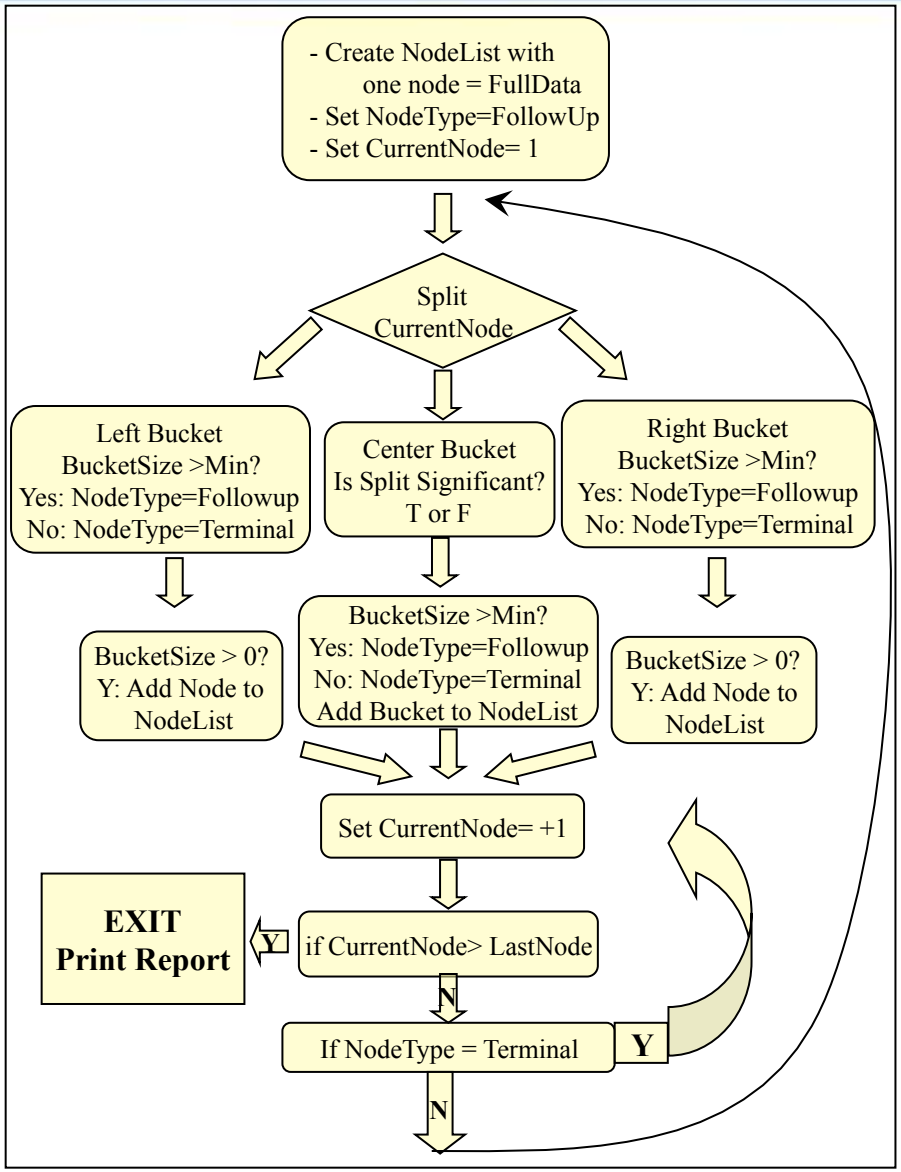
$$\pi = \text{Prob}[\, Z=1 \,]$$



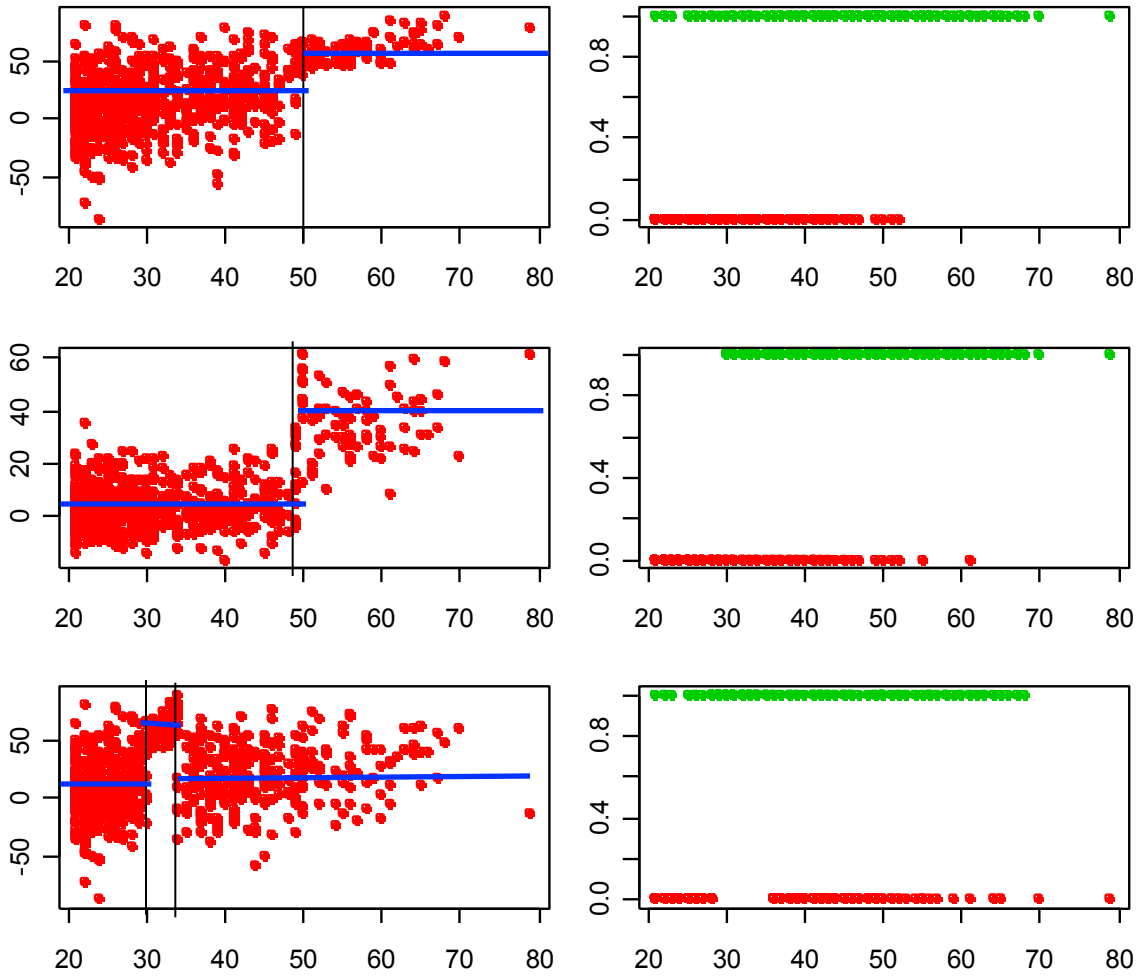
$T = (p-\pi)/\sigma_p$ measures how *interesting* a subset is.

Add a penalty term to prevent selection of subsets that are too small or too large.

ARF algorithm diagram



Comparing CART & ARF



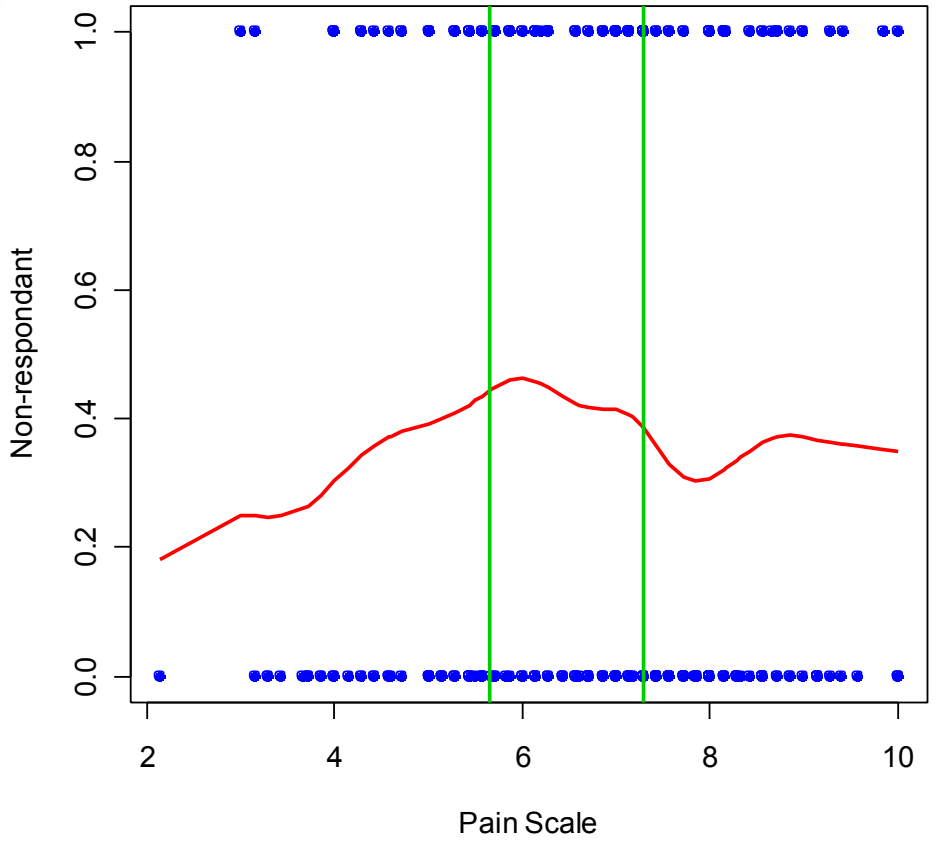
ARF: Captures subset with small variance (but not the rest).

CART Needs both subset with small variance relative to mean diff.

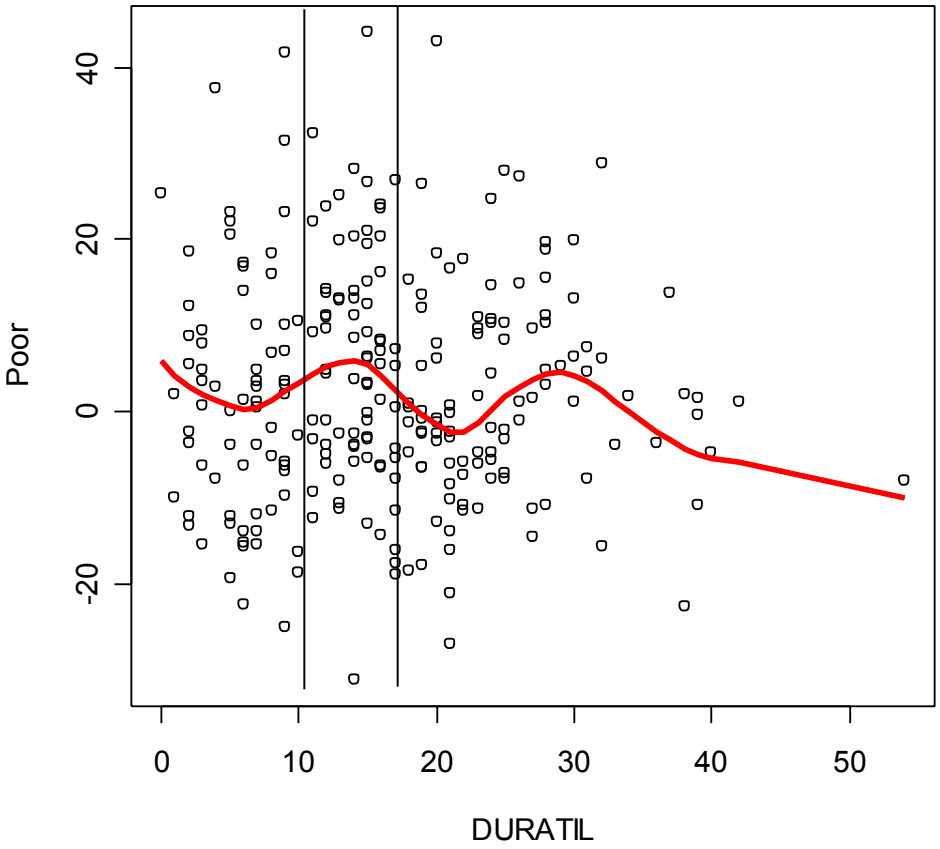
ARF: captures interior subsets.

Two Examples

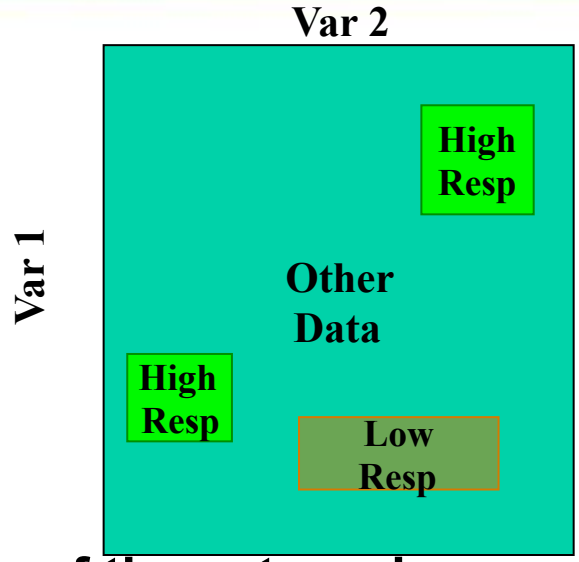
Subset that are hidden in the middle



Point density is important



Methodology



1. **Methodology Objective:**

   The Data Space is divided between
   - **High response subsets**
   - **Low Response subsets**
   - **Other**

2. **Categorical Responses:**

   Subsets that have high response on one of the categories.

   $$T = (p-\pi)/\sigma_p$$

3. **Continuous Responses: High mean response measured by**

   $$Z = (\bar{x} - \mu) / \sigma_{\bar{x}}$$

4. **Statistical significance should be based on the entire tree building process.**
5. **Categorical Predictors**
6. **Data Visualization**
7. **PDF report.**

Report

Simple Tree or Tree sketch : Only statistically significant nodes.

Full Tree: All nodes.

Table of Numerical Outputs: Detailed statistics of each node

List of Interesting Subsets: List of significant subsets

Conditional Scatter Plot (optional): Data Visualization.

How about outliers?

For Regression trees

- Popular belief: Trees are not affected by outlier (are robust)
- Outlier detection:

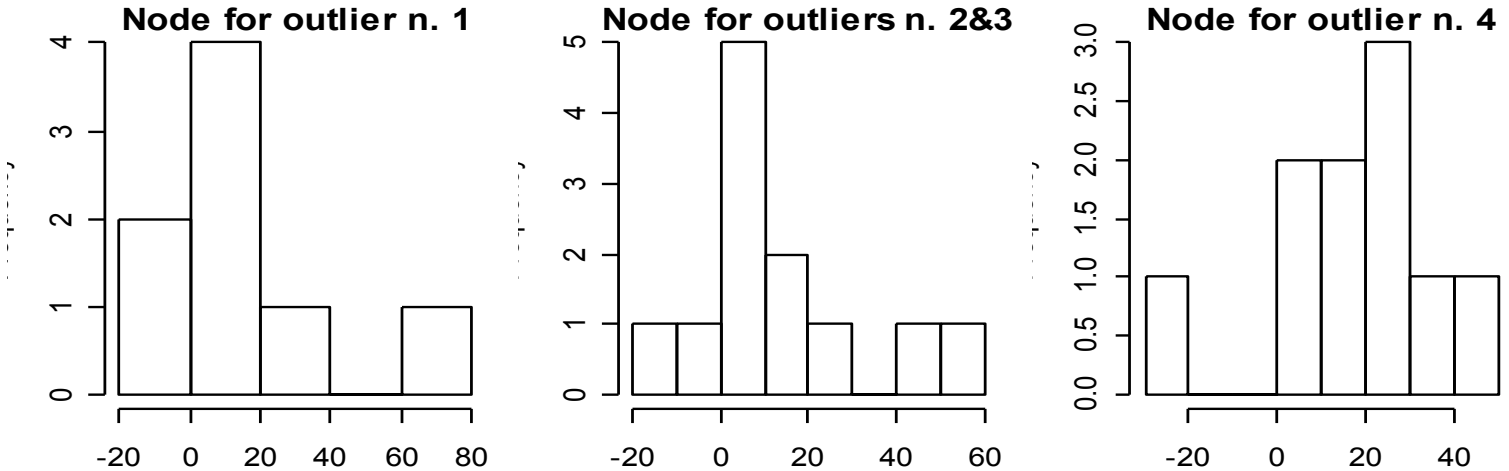
Run the data mining tree allowing for small buckets.

For observation X_i in terminal node j calculate the score

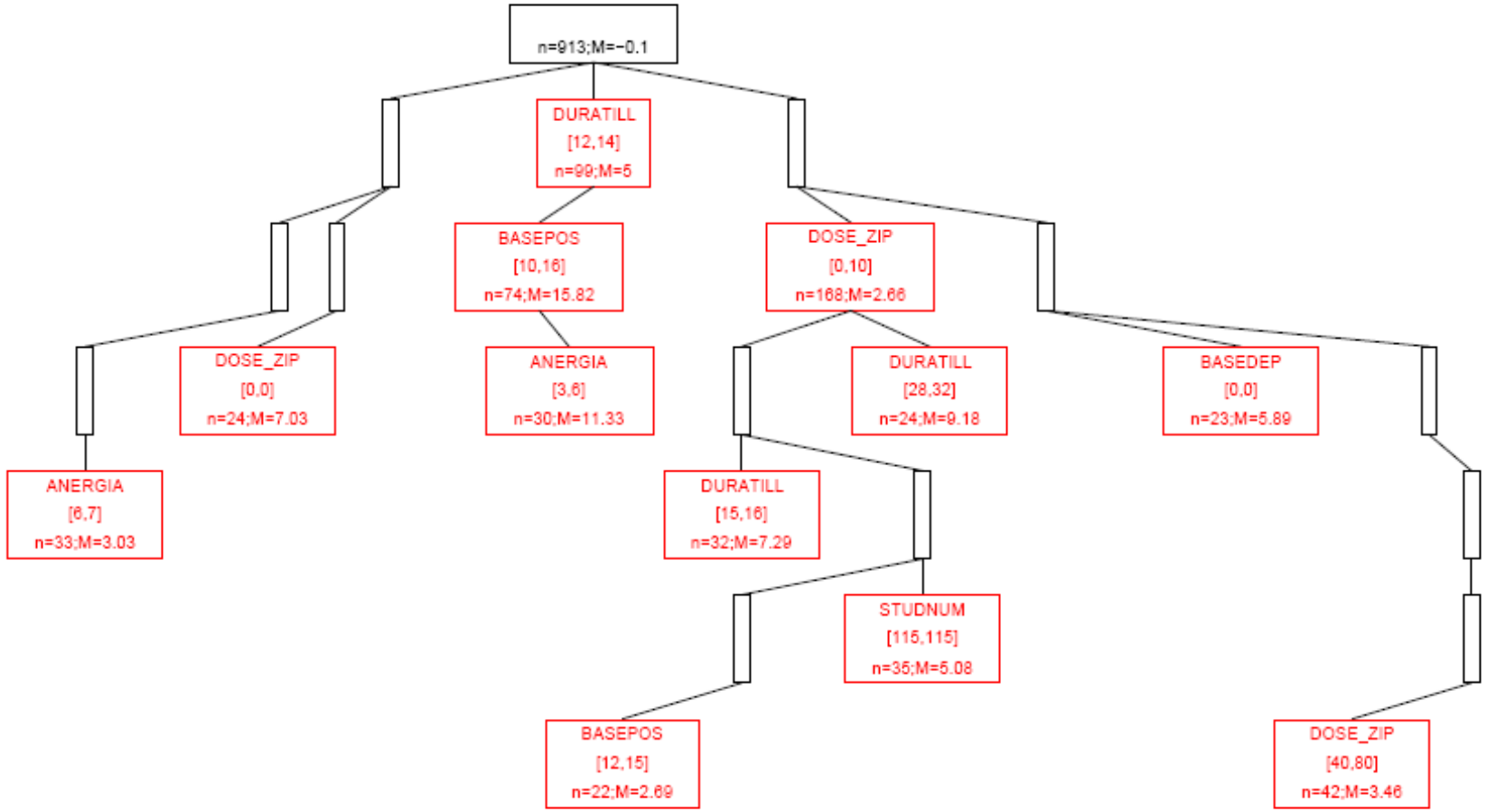
$$Z_i = \frac{|X_i - Median|}{MAD}$$

Z_i is the number of std dev away from the mean

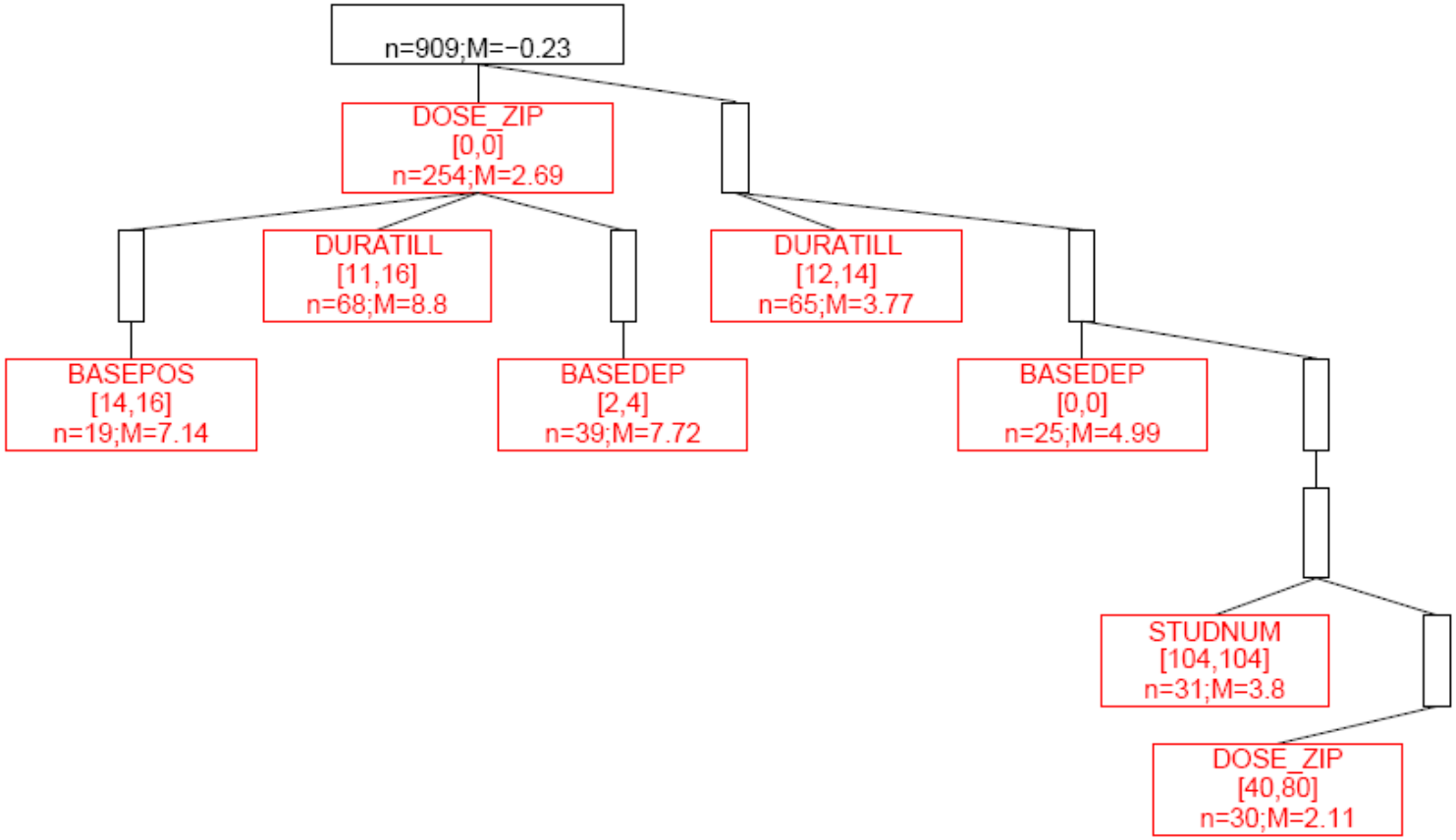
$Z_i > 3.5$ then X_i is noted as an outlier.



Tree with Outliers



After Outlier removal



Robustness issues

ISSUE

In regulatory environments outliers are rarely omitted.
Our method is easily adaptable to robust splits by calculating the robust version of the criterion by replacing the mean and std dev by suitable estimators of location and scale:

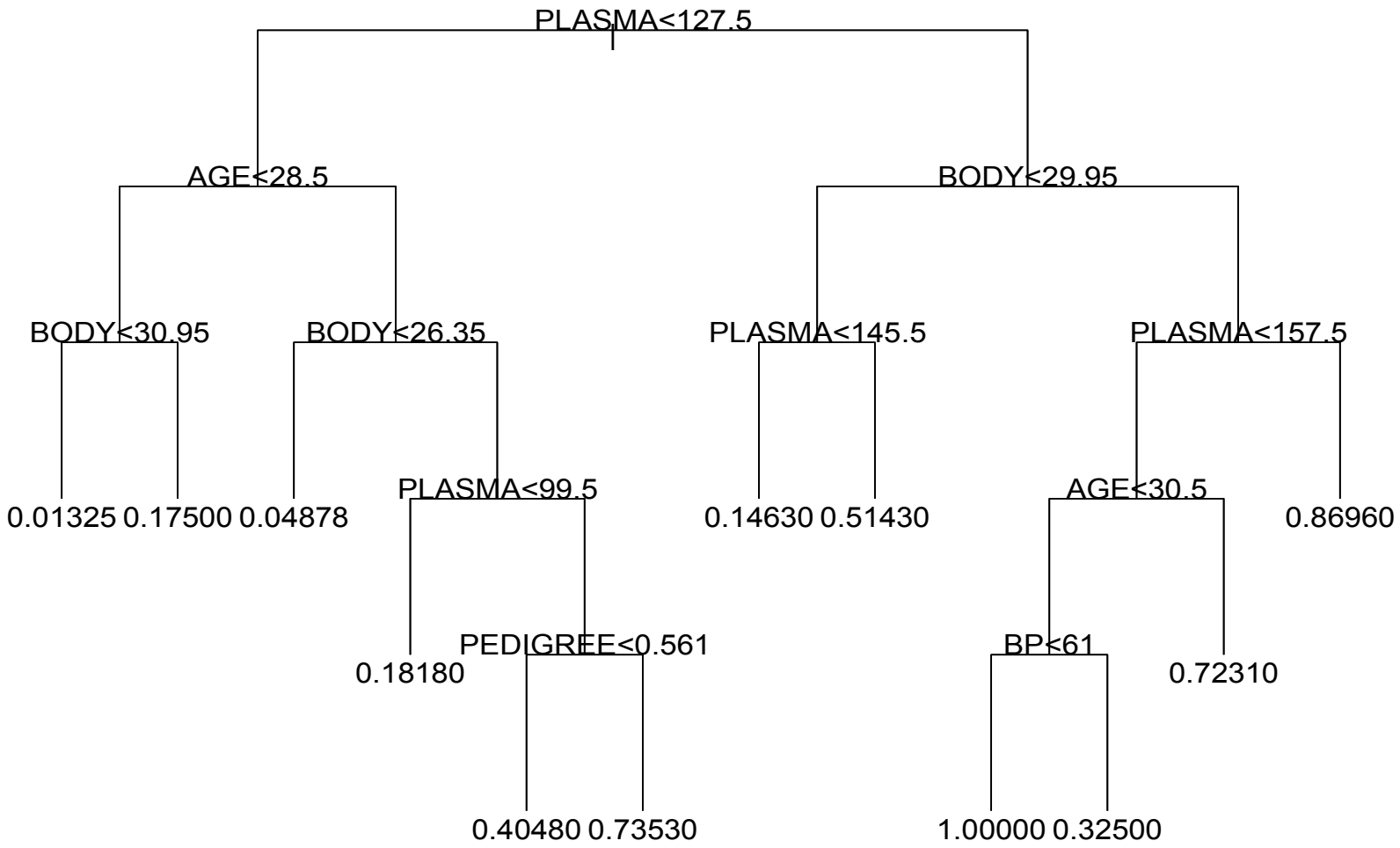
$$Z = (T - \mu_T^R) / \sigma_T^R$$

Binary/Categorical Response

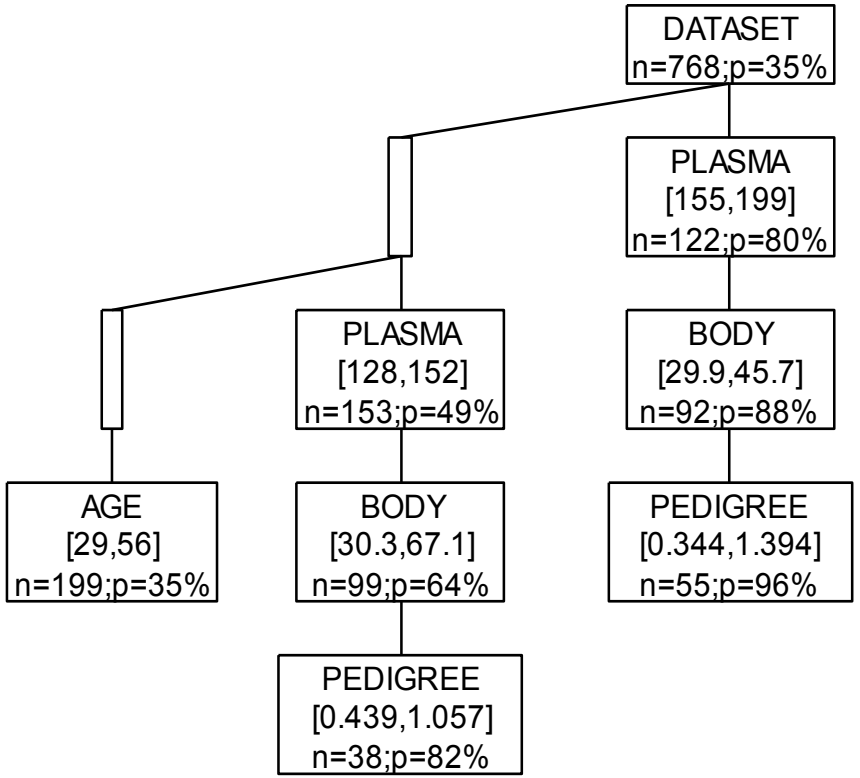
- How do we think about Robustness of trees?
- One outlier might not make any difference.
- 5% , 10% or more outliers could make a difference.

Classic Example of CART: Pima Indians Diabetes

- 768 Pima Indian females, 21+ years old ; 268 tested positive to diabetes
- 8 predictors: PRG, PLASMA, BP, THICK, INSULIN, BODY, PEDIGREE, AGE



ARF applied to Pima Indian data



	Subset	%Success	n
1	PLASMA in [155,199] & BODY in [29.9,45.7] & PEDIGREE in [0.344,1.394]	96.364	55
2	PLASMA in [128,152] & BODY in [30.3,67.1] & PEDIGREE in [0.439,1.057]	81.579	38
3	PLASMA in [0,127] & AGE in [29,56]	35.176	199

Ziprasidone Data Mining
4- & 6-week U.S. trials

- Protocol 104 – 4 weeks N=195
- Protocol 106 – 4 weeks N=132
- Protocol 114 – 6 weeks N=299
- Protocol 115 – 6 weeks N=325

85 subjects on haloperidol excluded

Total N = 951

Ziprasidone Data Mining
N by dose (mg.) and Protocol

	104	*106*	*114*	*115*
PBO	**47**	**47**	**92**	**80**
10	**46**			
40	**55**	**43**		**86**
80	**47**		**104**	
120		**42**		**76**
160			**103**	
200				**83**

Ziprasidone Data Mining Variables

Outcomes: Change in BPRS Total score

Predictors: age, sex, race, protocol, dose, baseline clinical ratings (positive Sx, CGI-S, anergia, depressive Sx, AIMS), duration of illness in years, current smoking status

Ziprasidone Data

Patient Characteristics
Total = 951

	N	%
Male	**700**	**74**
Race		
White	**620**	**65**
Black	**234**	**25**
Other	**97**	**10**
Smoker	**716**	**75**

Patient Characteristics

	Mean	S.D.	Range
BPRS change	**-5.1**	**13.4**	**-58, 55**
Residual change	**0**	**13.1**	**-45, 65**
Baseline BPRS	**35.9**	**11.0**	**14, 86**
Age	**38.7**	**10.1**	**18, 72**
Duration of illness	**16.0**	**9.6**	**0, 54**
Baseline Positive Sx	**12.7**	**3.4**	**4, 24**
Baseline Depression	**5.5**	**3.3**	**0, 17**
Baseline CGI-S	**4.8**	**0.8**	**3, 7**
Baseline Anergia	**6.0**	**3.4**	**0, 18**

Data Definitions

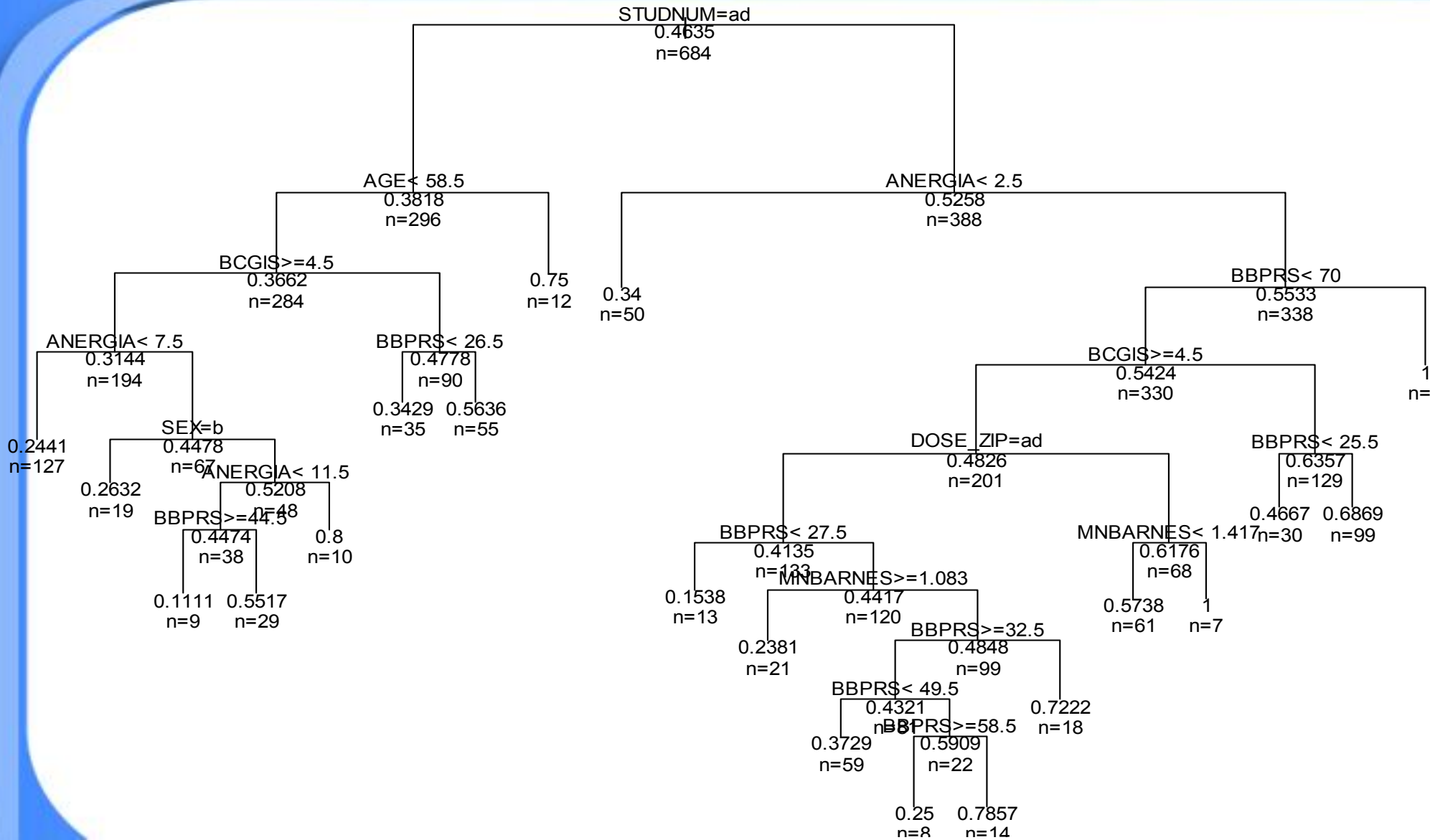
- AIMS = mean of AIMS total/5 and TD severity
- BPRS total & Sx scores (positive, depression, anergia) – absolute minimum is zero (items scored with minimum = 0 and not 1)
- Positive Sx score – sum of conceptual disorganization, hallucinatory behavior, unusual thought content, suspiciousness
- Depression – sum of anxiety, guilt feelings, depressive mood
- Anergia – sum of blunted affect, emotional withdrawal, motor retardation
- ***Continuous Response:*** *Residual BPRS change* – Residual (observed minus predicted) LOCF BPRS total regressed on baseline BPRS
- ***Binary Resp:*** 50% BPRS REDUCTION

Study Details

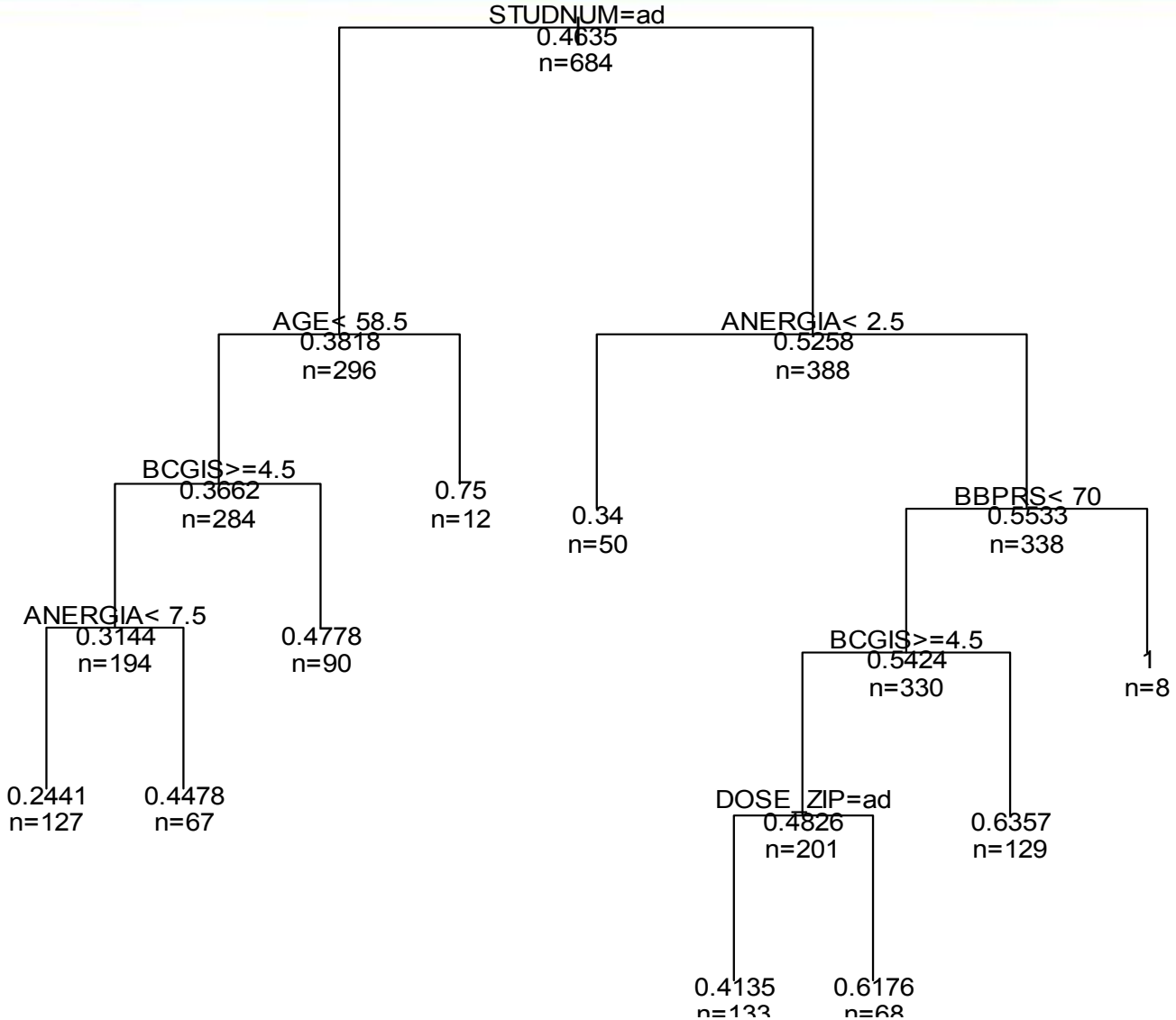
- Prospective, phase IV, multicenter, open-label, observational study.
- Primary care patients randomized to single-dose AZ-ER (2 g) or 10-day A/C (875 mg/125 mg q12h).
- Enrolled patients were ≥ 18 years of age with evidence of uncomplicated Acute Bacterial Sinusitis for ≥ 7 and ≤ 30 days. Evidence included presence of 2 cardinal Sx (facial pain/ pressure and discolored discharge/drainage) and ≥ 2 secondary Sx (fever, frequent coughing, nasal congestion, and post-nasal drainage).
- Patients with chronic/complicated sinusitis, prior use of selected sinusitis treatments, or limitation of immune function, GI absorption, or hepatic/renal function were excluded.

CART tree for 50% Reduction

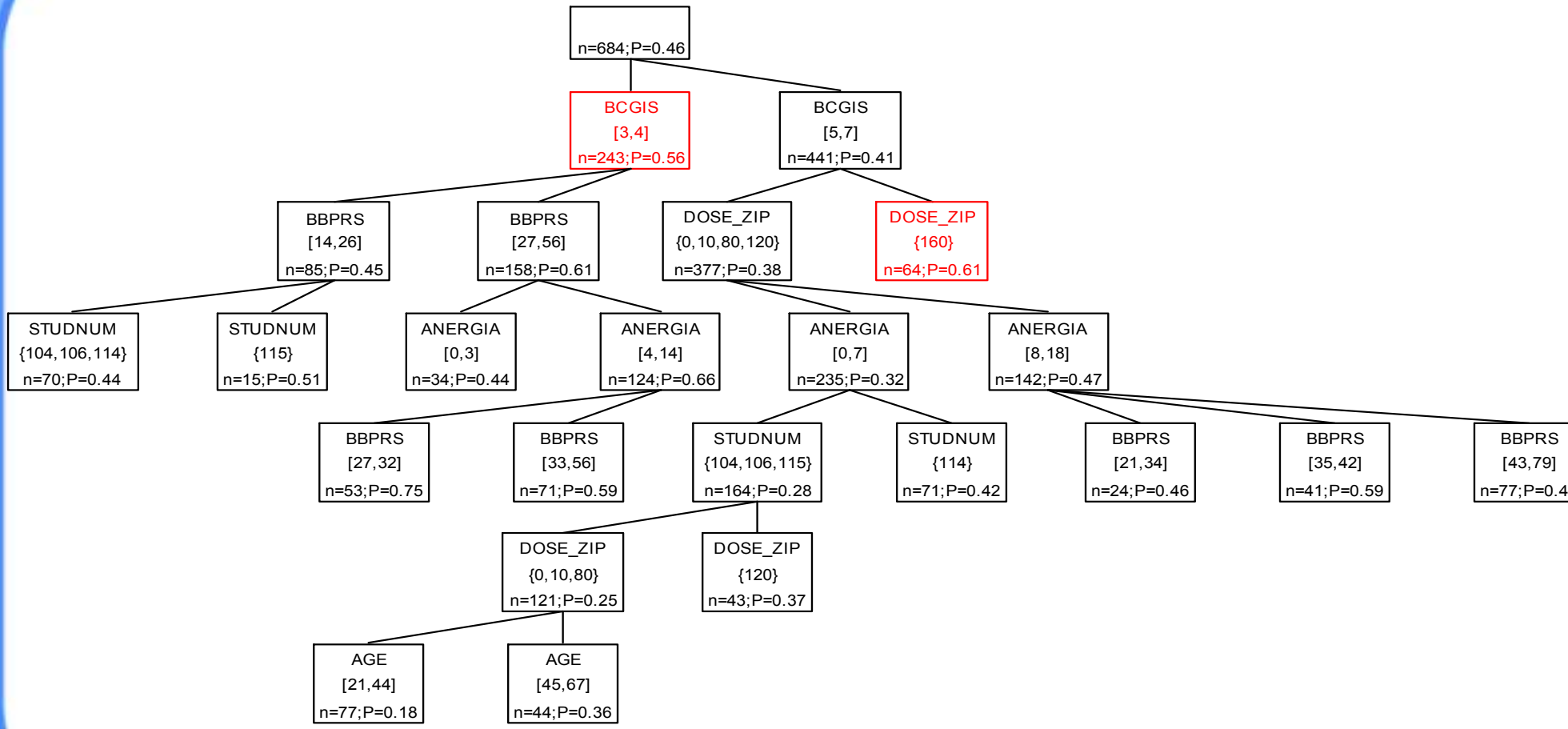
Default pruning



CART tree after more pruning



ARF tree


n=684;P=0.46
BCGIS
[3,4]
n=243;P=0.56
BCGIS
[5,7]
n=441;P=0.41
BBPRS
[14,26]
n=85;P=0.45
BBPRS
[27,56]
n=158;P=0.61
DOSE_ZIP
{0,10,80,120}
n=377;P=0.38
DOSE_ZIP
{160}
n=64;P=0.61
STUDNUM
{104,106,114}
n=70;P=0.44
STUDNUM
{115}
n=15;P=0.51
ANERGIA
[0,3]
n=34;P=0.44
ANERGIA
[4,14]
n=124;P=0.66
ANERGIA
[0,7]
n=235;P=0.32
ANERGIA
[8,18]
n=142;P=0.47
BBPRS
[27,32]
n=53;P=0.75
BBPRS
[33,56]
n=71;P=0.59
STUDNUM
{104,106,115}
n=164;P=0.28
STUDNUM
{114}
n=71;P=0.42
BBPRS
[21,34]
n=24;P=0.46
BBPRS
[35,42]
n=41;P=0.59
BBPRS
[43,79]
n=77;P=0.4
DOSE_ZIP
{0,10,80}
n=121;P=0.25
DOSE_ZIP
{120}
n=43;P=0.37
AGE
[21,44]
n=77;P=0.18
AGE
[45,67]
n=44;P=0.36

A New Paradigm

Main issue: Comparative efficacy – subsets where:

- **the drug is more effective than placebo or other drugs**
- **low dose is better than placebo**
- **high dose is better than a low dose or vice versa**

Changes in Data Analysis Framework:

-The X space is defined by two or more samples.

-We estimate the conditional difference of means or in general a function of the conditional means.

-We extend CART and ARF to the differences between two or more means

A new paradigm-MPART

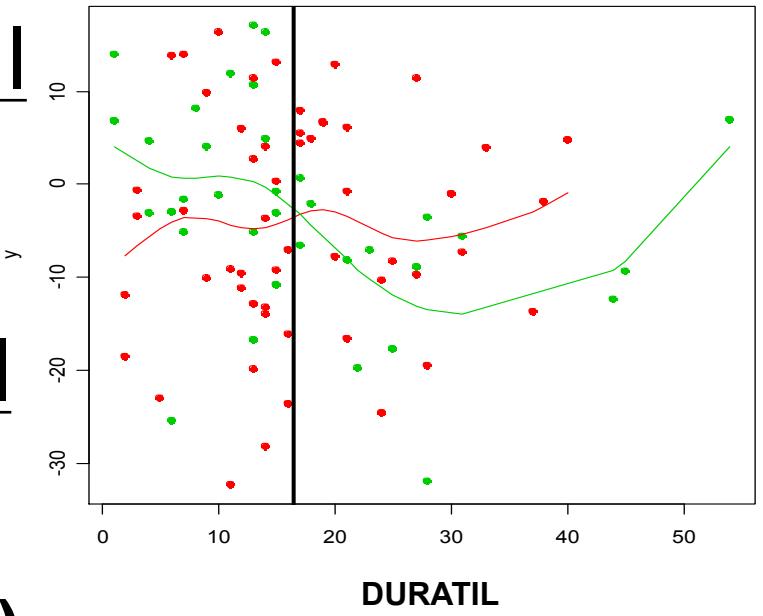
Categorical Responses:

$$T = \frac{|(\bar{p}_2^L - \bar{p}_1^L) - (\bar{p}_2^R - \bar{p}_1^R)|}{SE}$$

Might use odd ratios here

Continuous Responses:

$$T = \frac{|(\bar{x}_2^L - \bar{x}_1^L) - (\bar{x}_2^R - \bar{x}_1^R)|}{SE}$$



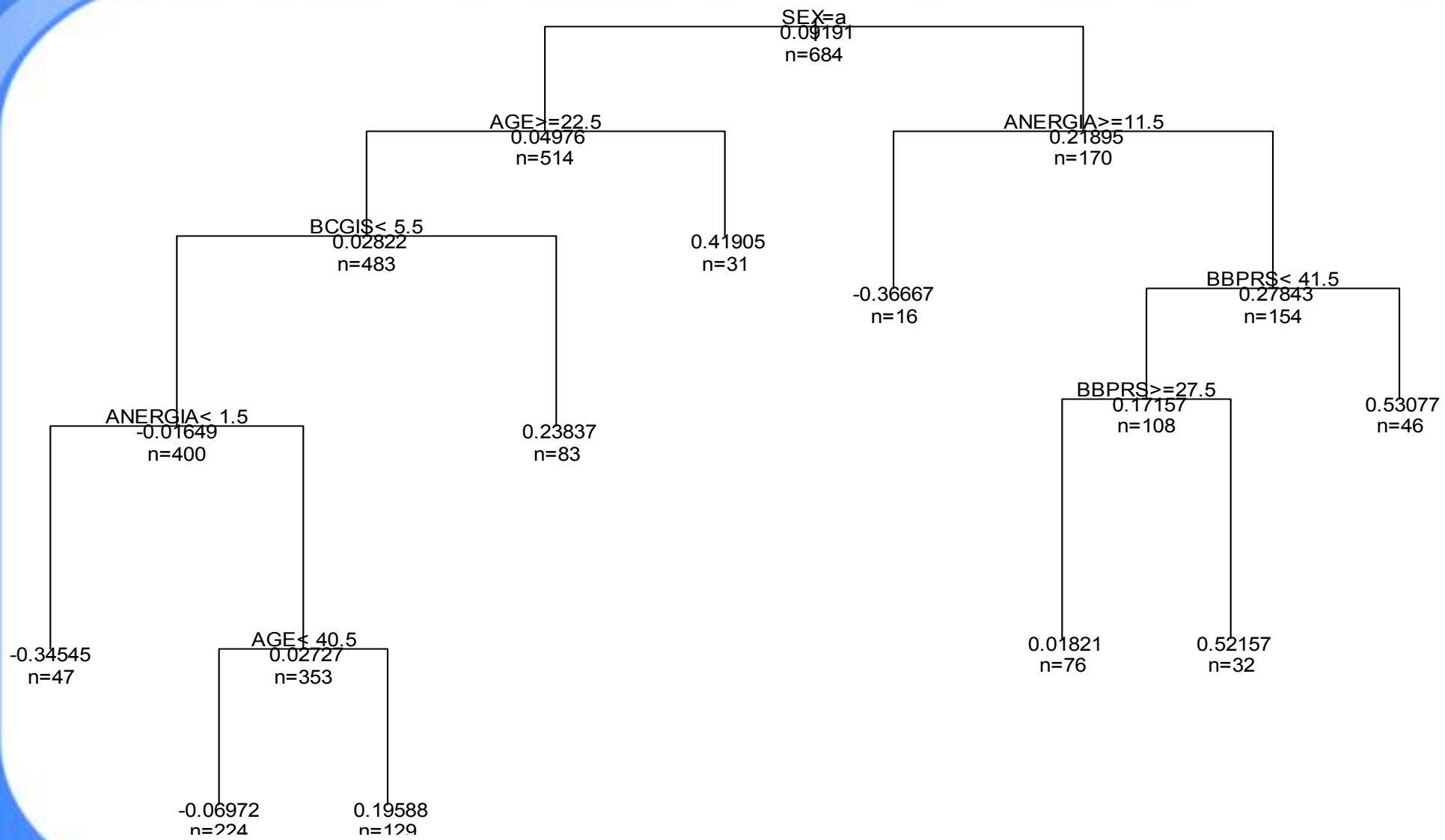
M-ARF criterion:

$$T_1 = \frac{(\bar{x}_2 - \bar{x}_1)}{SE} \quad T_2 = \frac{(\bar{p}_2 - \bar{p}_1)}{SE}$$

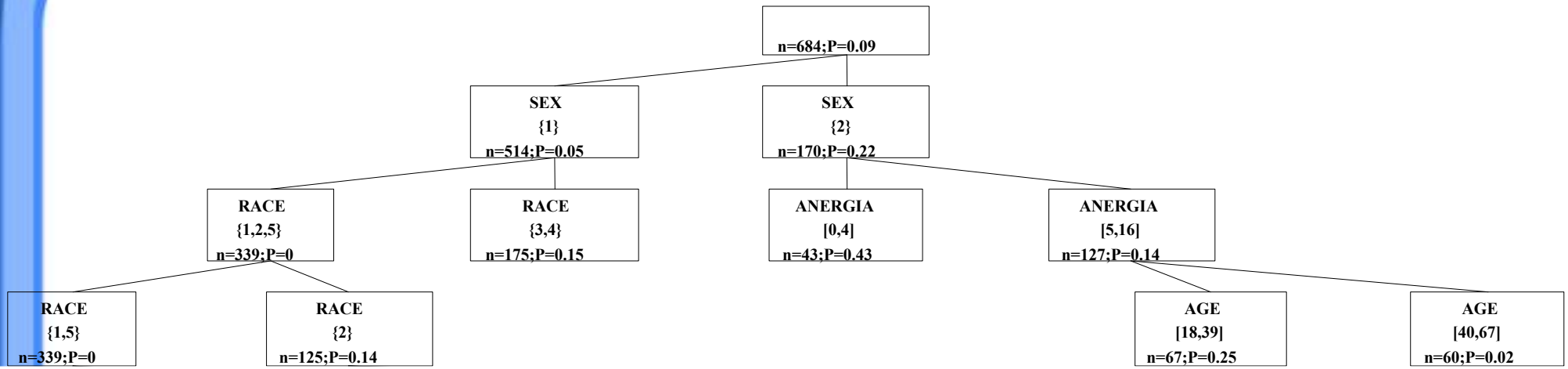
For more than two groups use F-statistic, Tukey's Dunnett's...

Extension to survival modelling

MPART tree for Ziprasidone Ex.



M-ARF tree for Ziprasidone Ex.

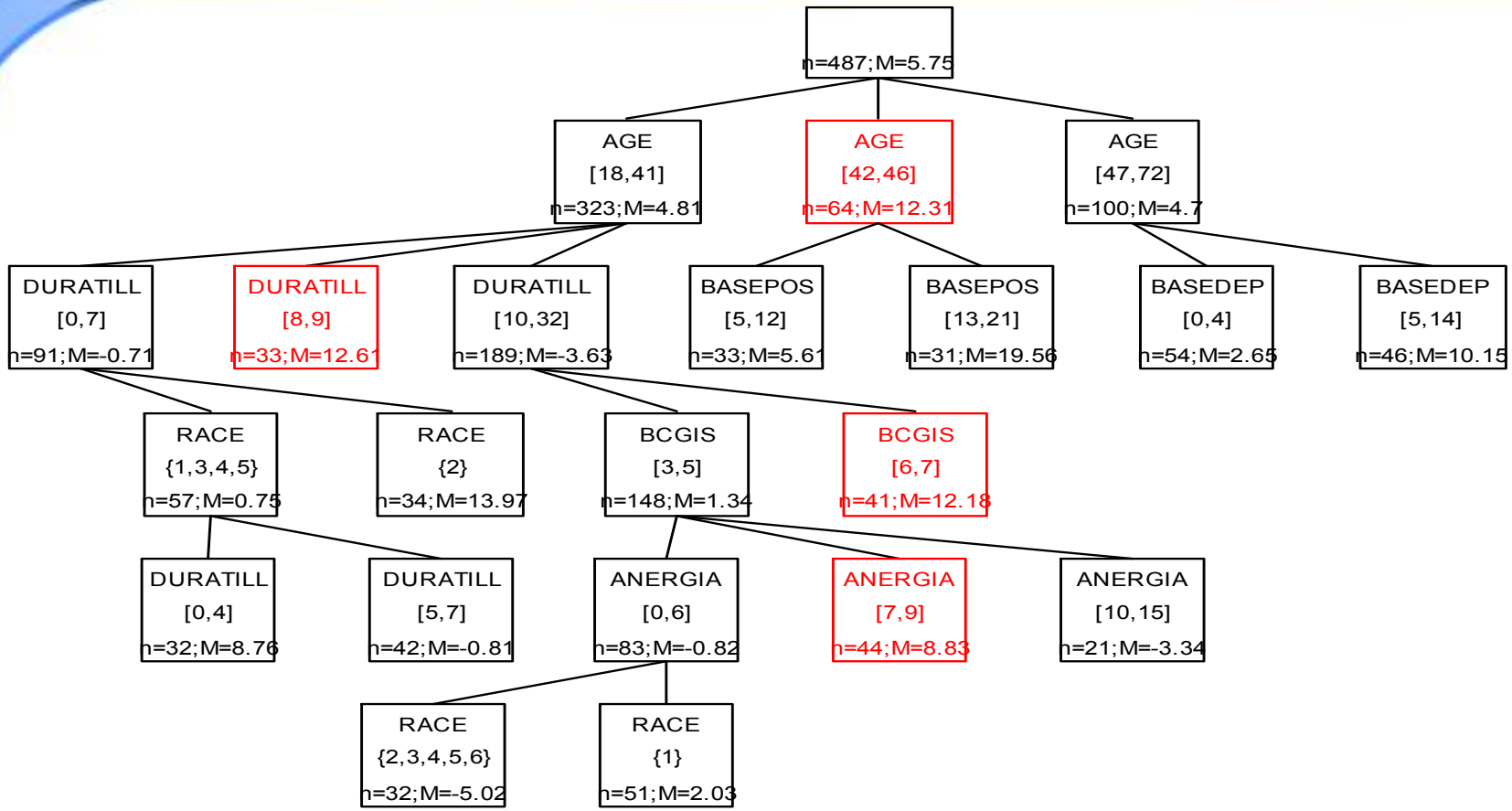


Extension to other methods, ANN, SVM

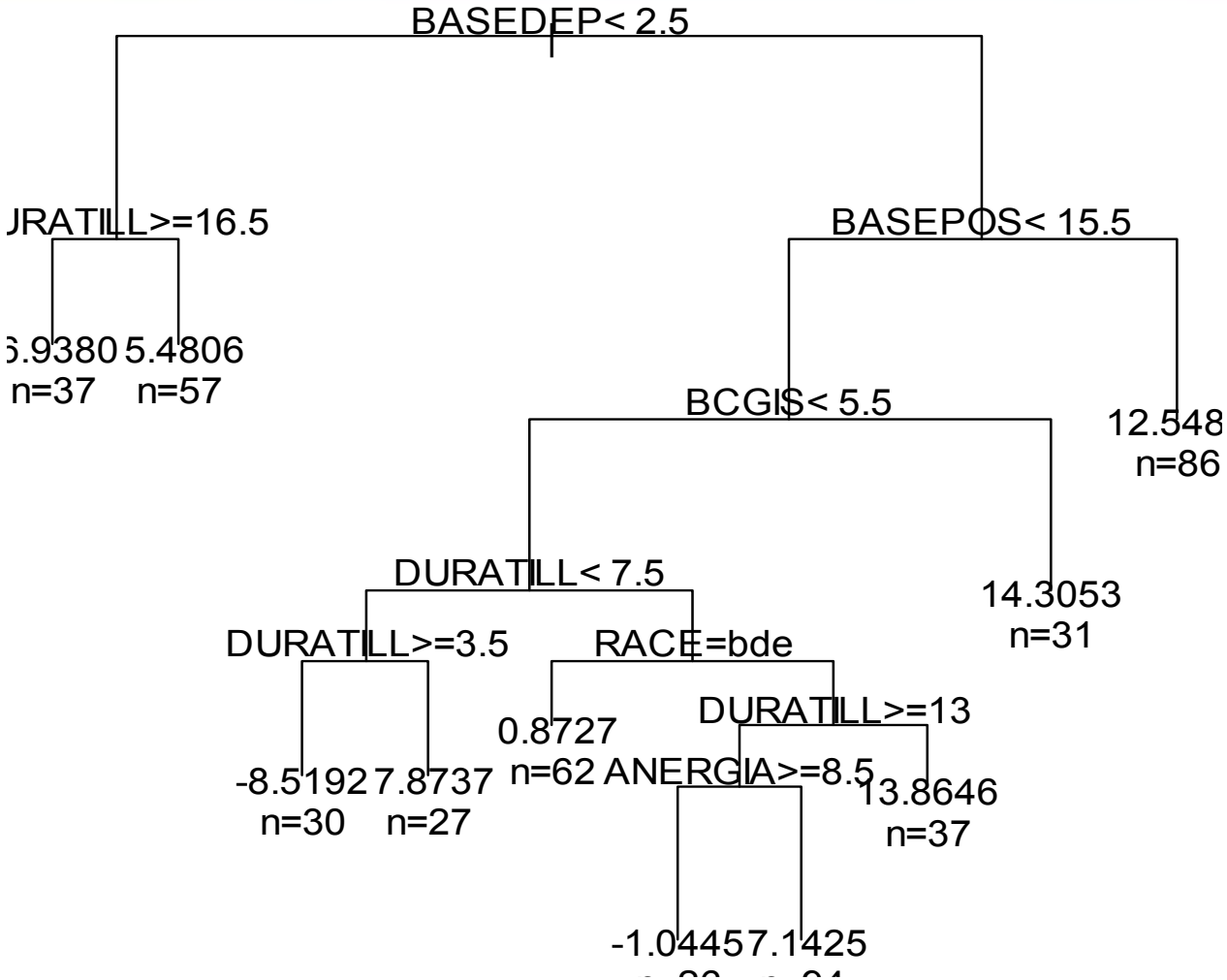
Alvir, Cabrera, Caridi and Nguyen (2007)
Mining Clinical Trial data.
Alvir, Cabrera, Caridi and Nguyen (2008)
Multivariate partitioning.(MPART)

Datamining website, R-package, papers
R-package: www.rci.rutgers.edu

Back to the Ziprasidone example
Compare 120 mg/160 mg Vs Placebo with

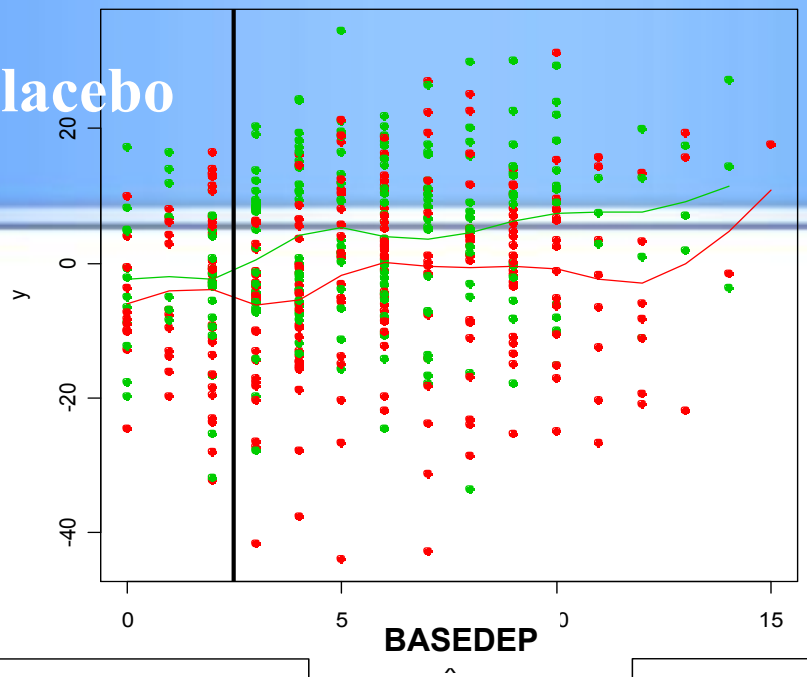


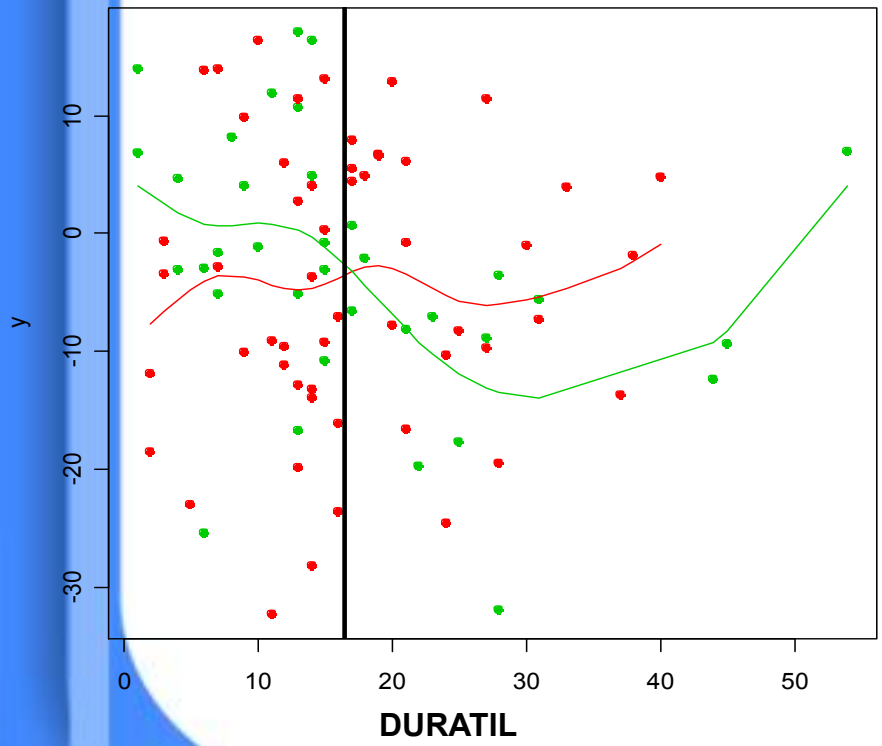
Ziprasidone: 120 mg/160 mg Vs Placebo
MULTIRESPONSE CART

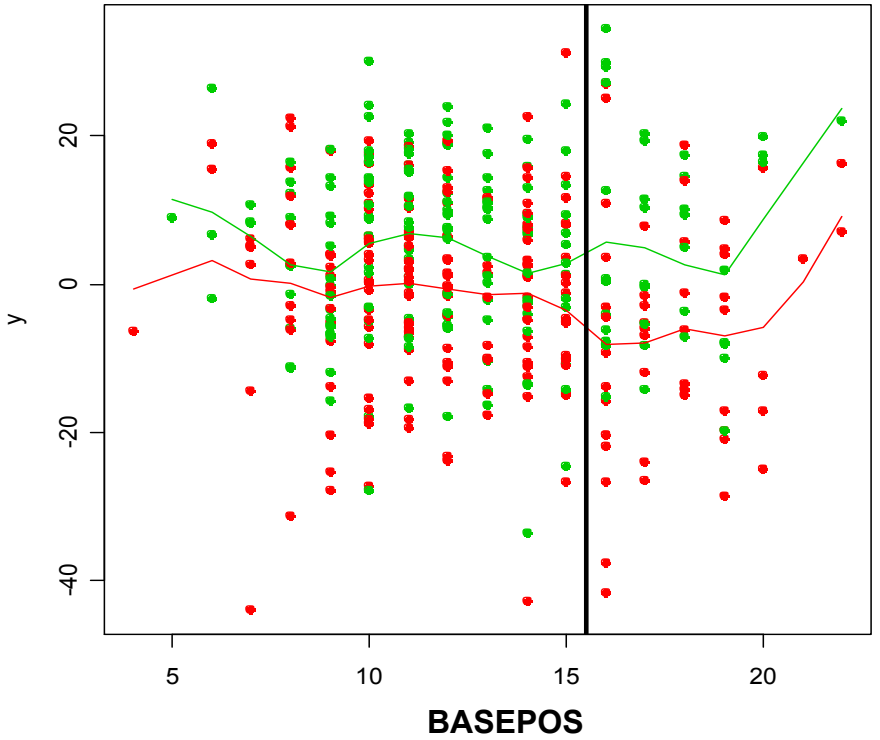


Ziprasidone:
120 mg/160 mg Vs Placebo

TOP THREE SPLITS







47th Interscience Conference on Antimicrobial Agents and Chemotherapy Chicago

Symptom Resolution with Azithromycin Extended Release Versus Amoxicillin/Clavulanate in Patients with Acute Sinusitis in a General Practice Physician Environment

J. F. Piccirillo[1], B. F. Marple[2], C. S. Roberts[3], J. R. Frytak[4], V. F. Schabert[5], J. C. Wegner[4], H. Bhattacharyya[3], S. P. Sanchez[3]

1 Washington University School of Medicine, St Louis, MO
2 University of Texas Southwestern Medical Center, Dallas, TX
3 Pfizer Inc, New York, NY
4 i3 Innovus, Eden Prairie, MN
5 Integral Health Decisions Inc, Santa Barbara, CA

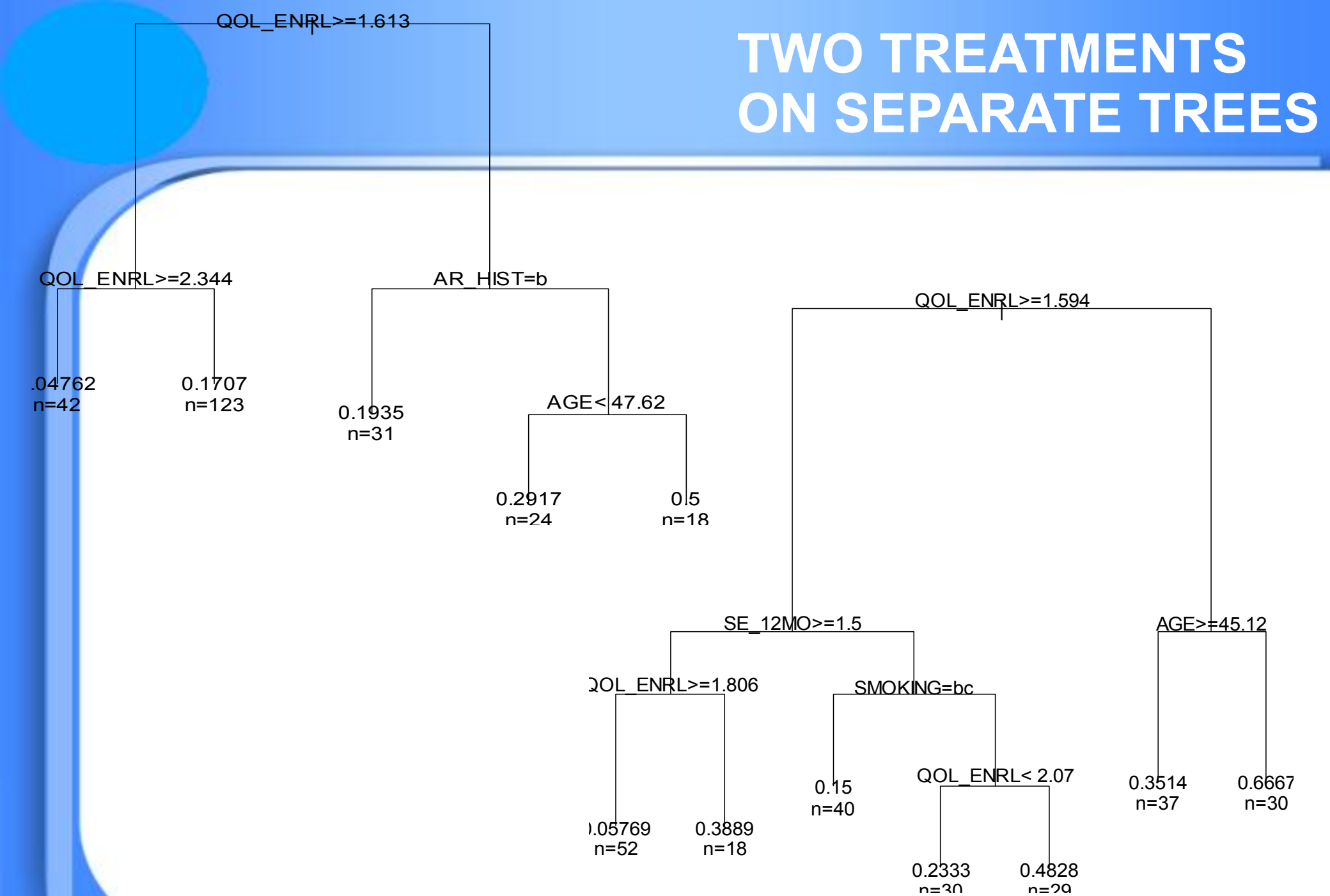
Sample Characteristics

	PP Population	
	AZ-ER (N = 236)	AC (N = 238)
	Mean (SD)	Mean (SD)
Age of Subject	46.4 (13.5)	46.0 (12.4)
BMI	29.3 (7.7)	29.2 (7.4)
	N (%)	N (%)
Gender		
Female	161 (68.2)	162 (68.1)
Male	75 (31.8)	76 (31.9)
Smoking Status		
Never smoked	150 (63.6)	156 (65.6)
Ex-smoker	44 (18.6)	43 (18.1)
Current smoker	42 (17.8)	39 (16.4)
Race		
Black (Non-Hispanic)	8 (3.4)	5 (2.1)
Asian (Non-Hispanic)	3 (1.3)	5 (2.1)
Hispanic	11 (4.7)	7 (2.9)
Other	214 (90.7)	221 (92.9)
History of Allergic Rhinitis		
No	145 (61.4)	152 (63.9)
Yes	91 (38.6)	86 (36.1)
Febrile		
No	174 (73.7)	182 (76.5)
Yes	62 (26.3)	56 (23.5)
Concomitant Meds in Last 30 Days		
Nasal corticosteroid	12 (5.1)	16 (6.7)
Decongestant	70 (30.0)	63 (26.5)
Antihistamine	38 (16.1)	29 (12.2)

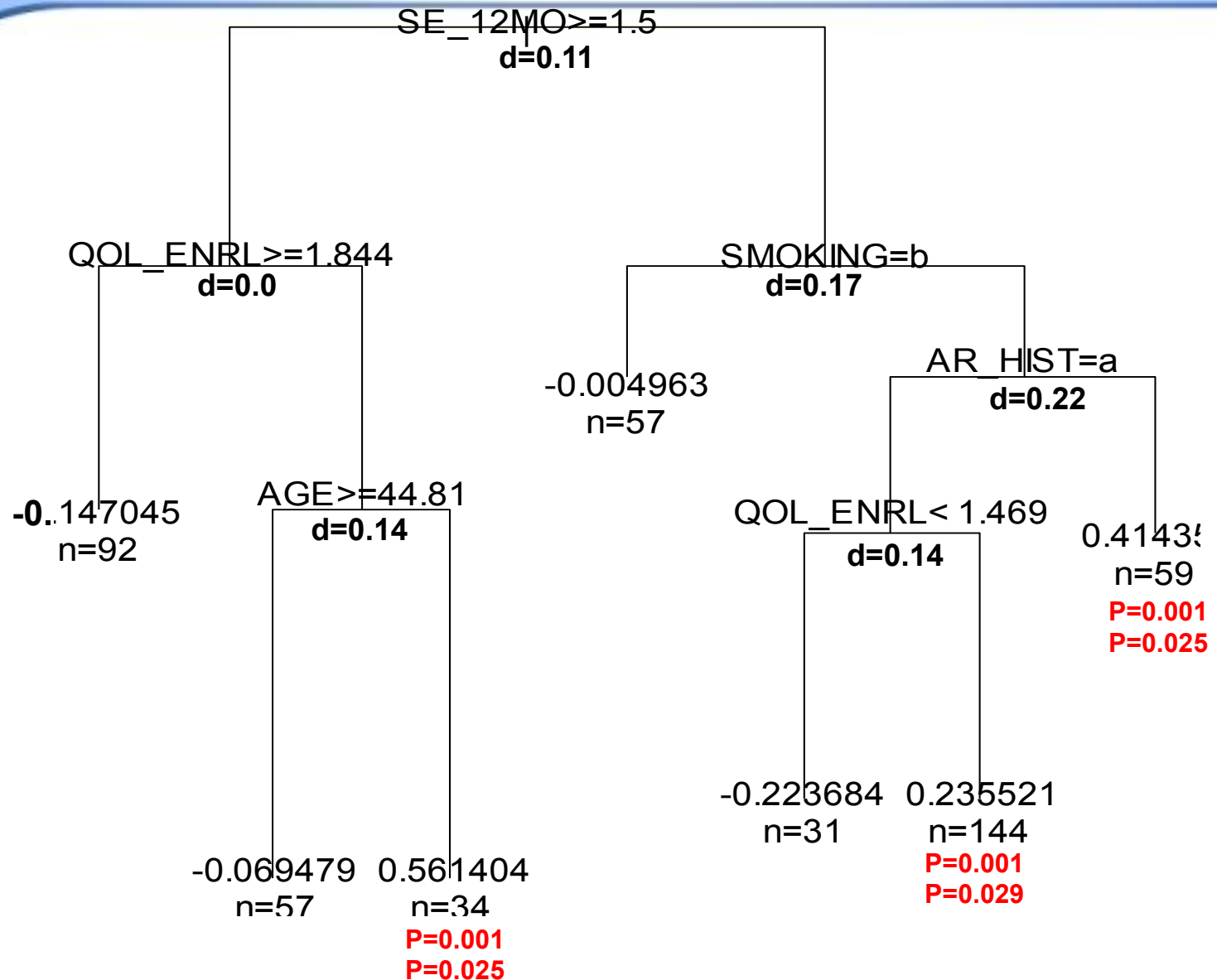
Study details (continued)

- Per-protocol (PP) population was defined as ITT subjects who returned the diary, completed the first telephone interview, and reported taking ≥ 1 dose of the study medication.
- Primary endpoint was self-reported symptom resolution at day 5.
- PP population was designated as the primary analysis population, due to the necessity of diary receipt for assessment of the primary endpoint.
- AZ-ER was superior to A/C on the primary endpoint.
- More AZ-ER patients (29.7%, 70/236) showed symptom resolution before the end of day 5 than did A/C patients (18.9%, 45/238), a difference of 10.8% (95% CI, 3.1%, 18.4%).

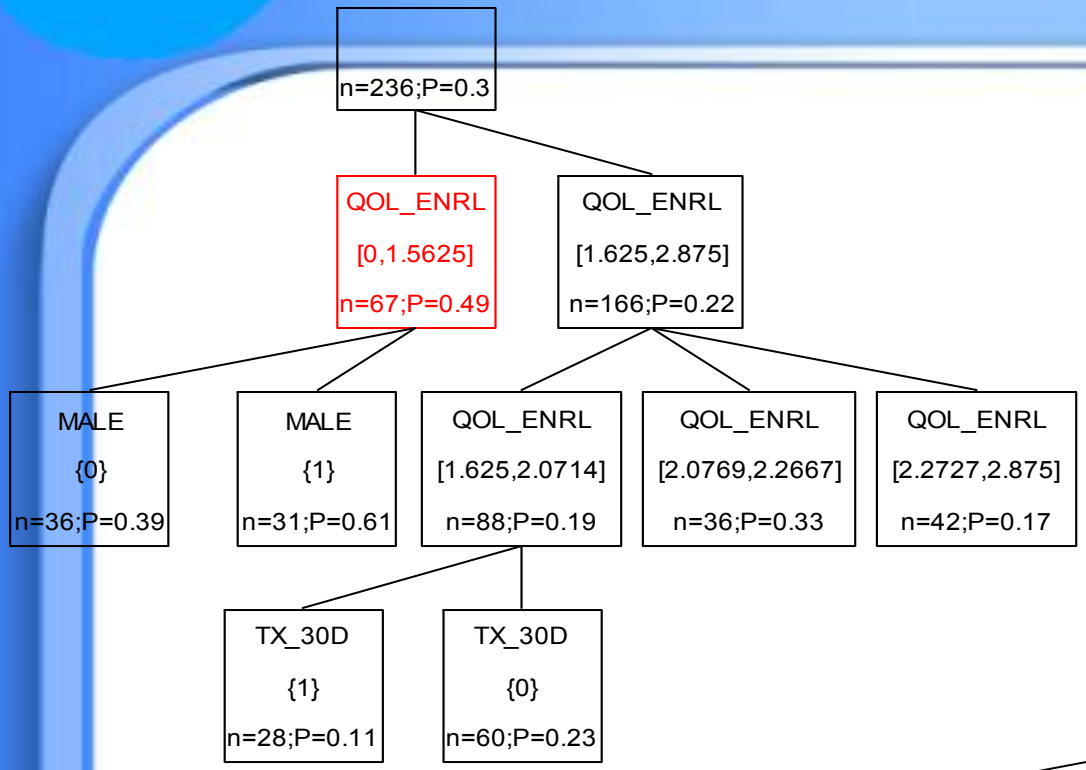
TWO TREATMENTS ON SEPARATE TREES

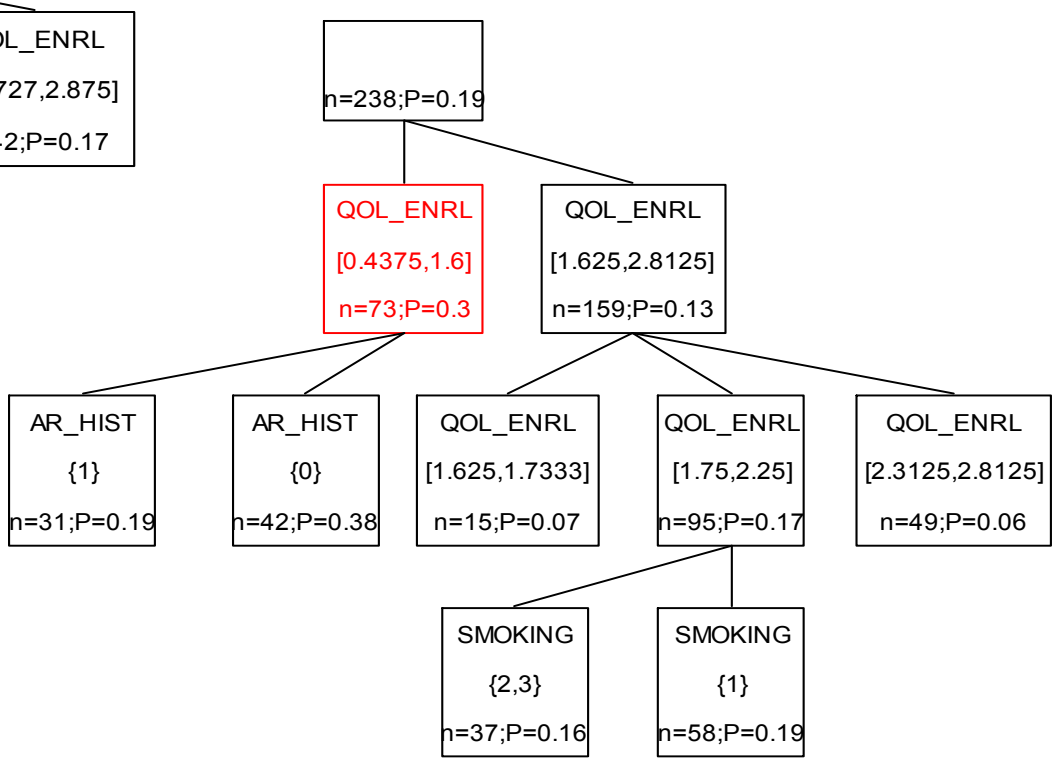


ONE TREE ON THE DIFFERENCE OF TREATMENTS

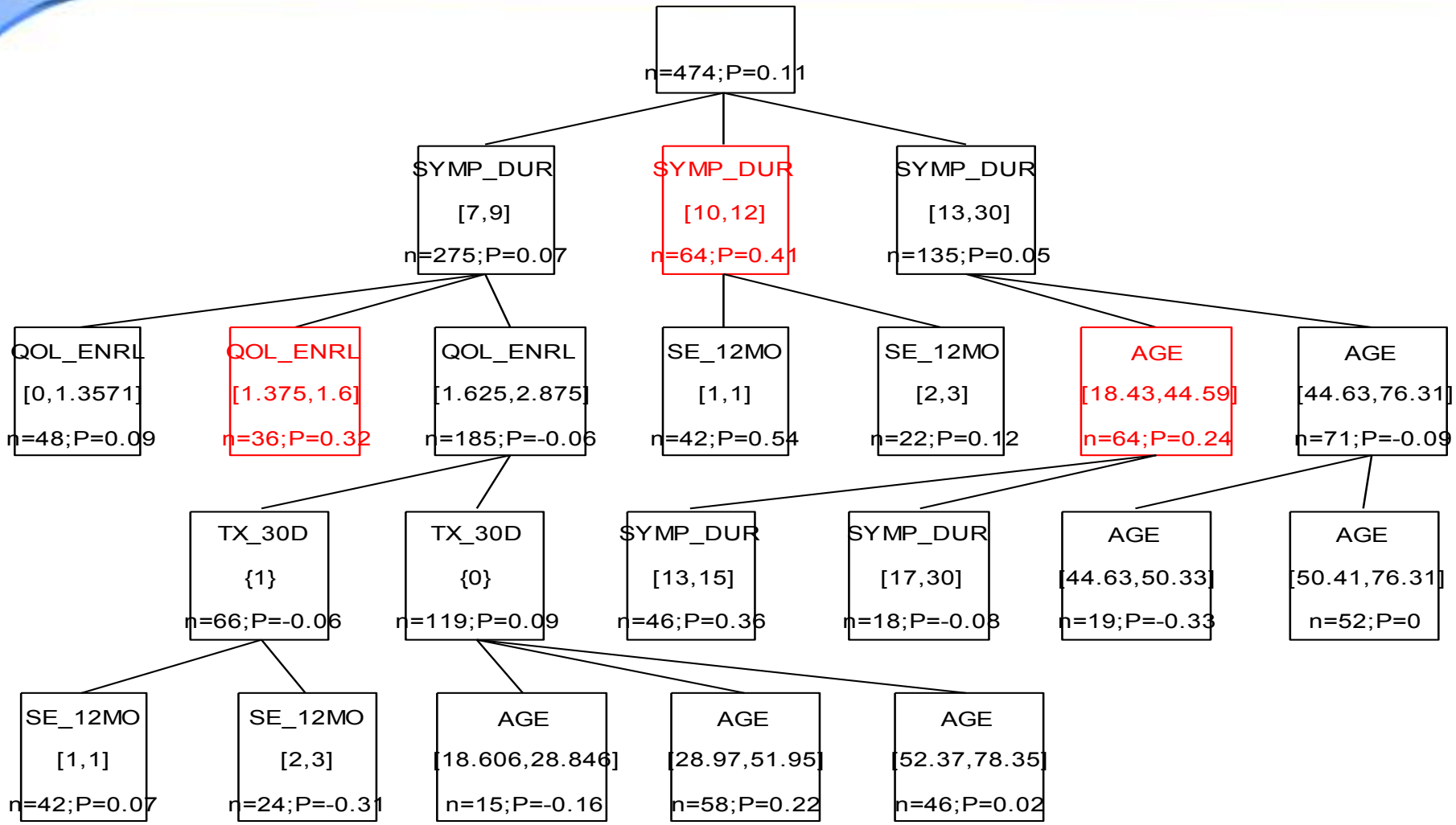


TWO TREATMENTS ON SEPARATE TREES





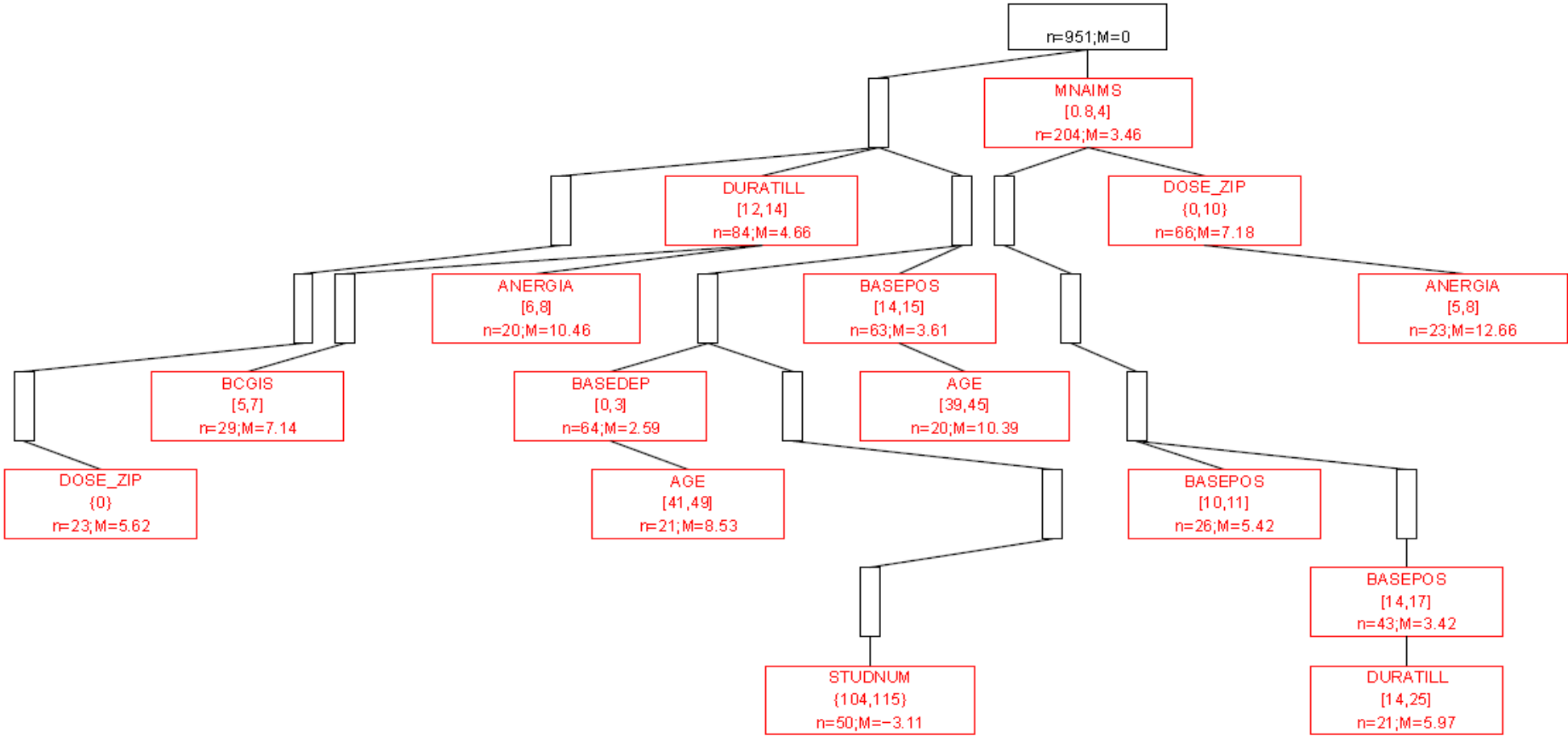
ONE TREE ON THE DIFFERENCE OF TREATMENTS



RESPONSE VARIABLE = POORRESP

PREDICTORS = AGE, ANERGIA, BASEDEP, BASEPOS, BCGIS, DOSE_ZIP, DURATILL, MNAIMS, RACE, STUDNUM. NOT IN USE = SEX, SMOKEYN

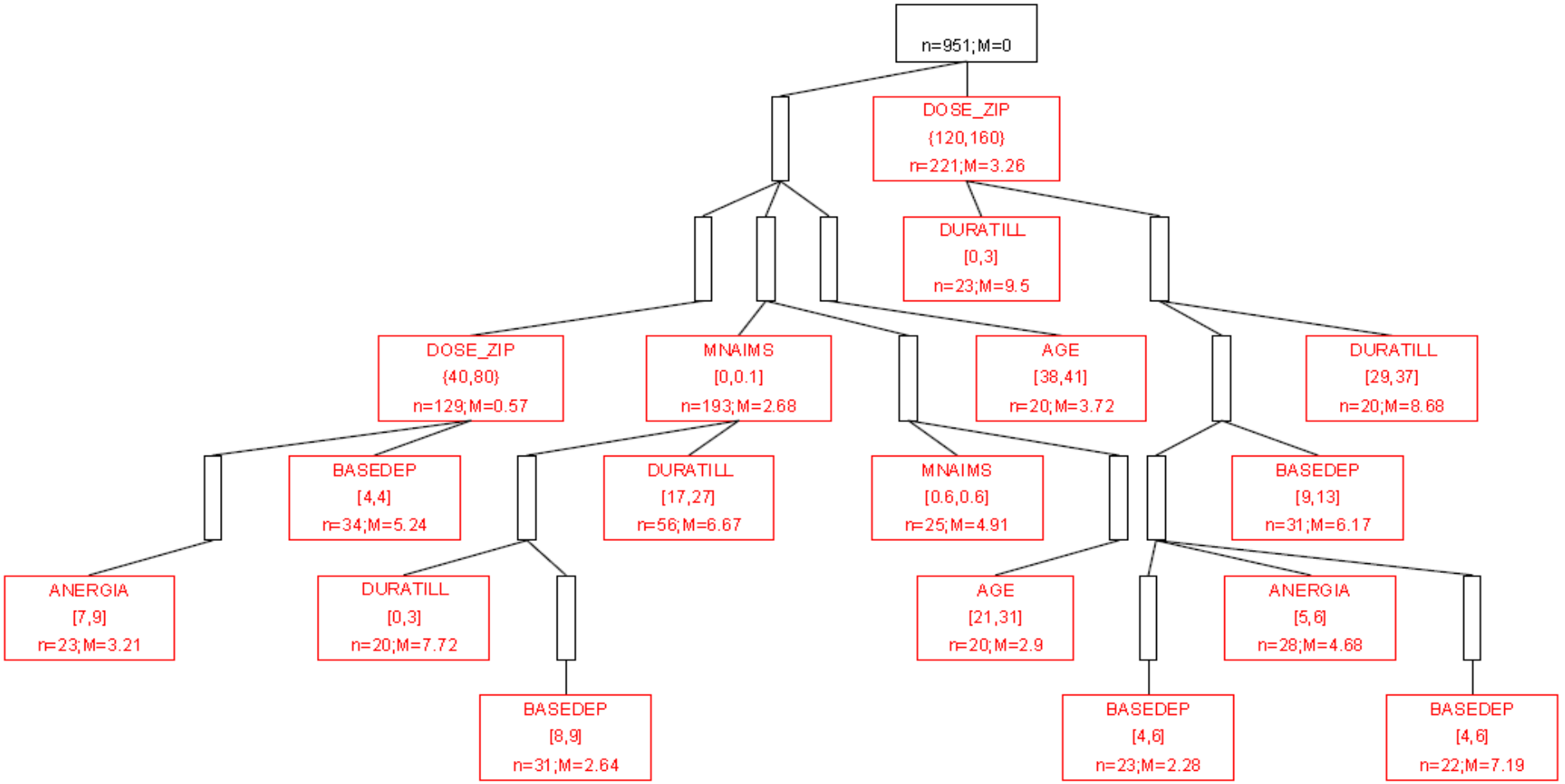
N = 951 = 0



RESPONSE VARIABLE = GOODRESP

PREDICTORS = AGE, ANERGIA, BASEDEP, BASEPOS, DOSE_ZIP, DURATILL, MNAIMS, STUDNUM. NOT IN USE = SEX, RACE, SMOKEYN, BCGIS

N = 951 = 0



MORE ON BAGGING BOOSTING

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, $\hat{f}_D(\boldsymbol{x})$, 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

$$\bar{f}(\boldsymbol{x}) = \mathrm{E}_D\, \hat{f}_D(\boldsymbol{x})$$

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

$$\mathrm{E}_D\, J(\hat{f}, D) = \mathrm{E}\,\boldsymbol{\varepsilon}^2 + \mathrm{E}_x (f(\boldsymbol{x}) - \bar{f}(\boldsymbol{x}))^2 + \mathrm{E}_{x,D} (\hat{f}_D(\boldsymbol{x}) - \bar{f}(\boldsymbol{x}))^2$$
$$= \text{noise} + \text{bias} + \text{variance}$$

Bias-variance decomposition (cont.)

$$\mathrm{E}_D\, J(\hat{f}, D) = \mathrm{E}\,\boldsymbol{\varepsilon}^2 + \mathrm{E}_x (f(\boldsymbol{x}) - \bar{f}(\boldsymbol{x}))^2 + \mathrm{E}_{x,D} (\hat{f}_D(\boldsymbol{x}) - \bar{f}(\boldsymbol{x}))^2$$
$$= \text{noise} + \text{bias} + \text{variance}$$

- The noise cannot be reduced.
- The squared-bias term might be reducible
- The variance term is 0 if we use

$$\hat{f}_D(\boldsymbol{x}) = \bar{f}(\boldsymbol{x})$$

But this requires having an infinite number of datasets

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:

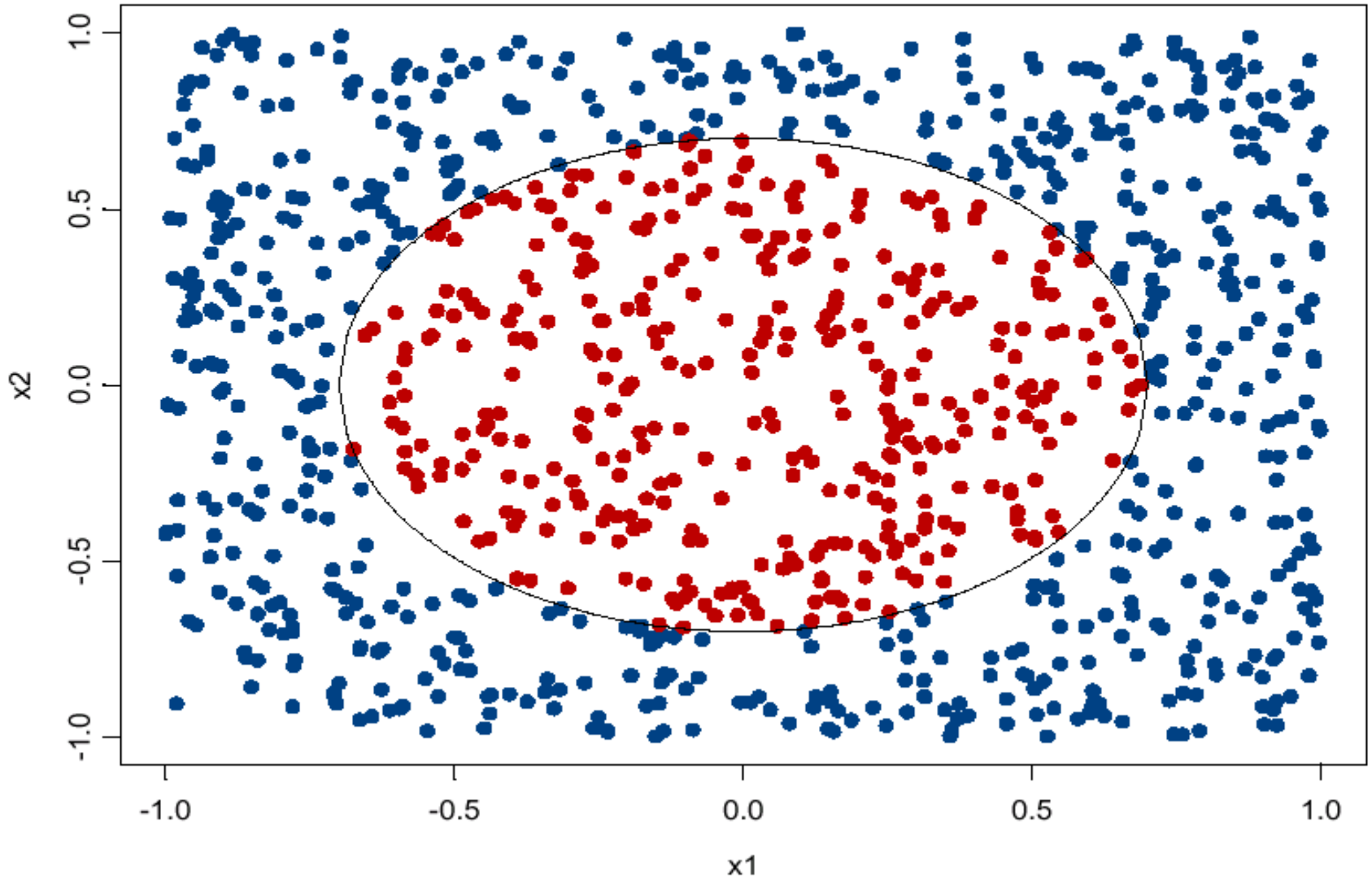
- Stabilizes "unstable" methods
- Easy to implement, parallelizable
- Theory is not fully explained

Bagging algorithm

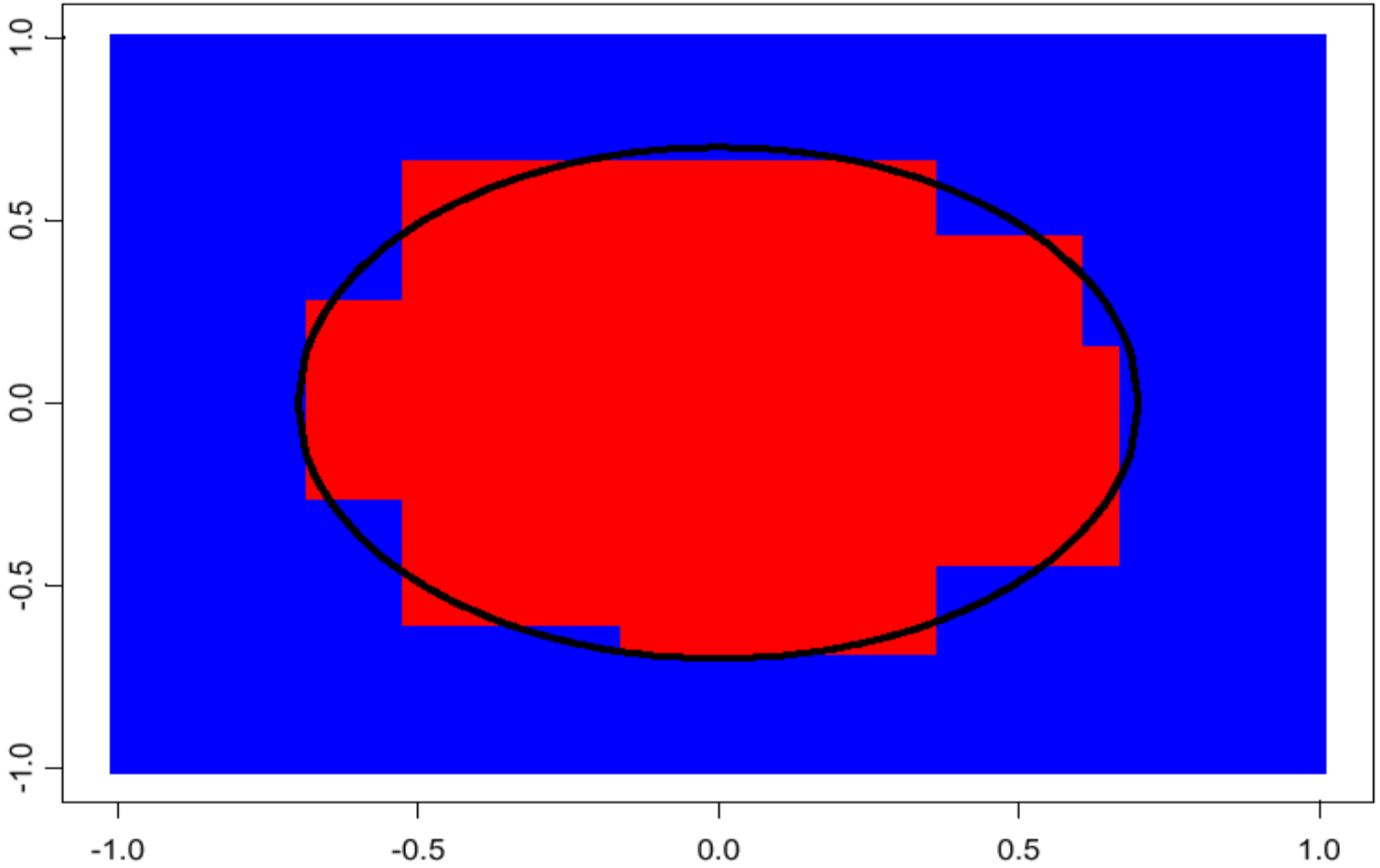
1. Create K bootstrap replicates of the dataset.
2. Fit a model to each of the replicates.
3. Average (or vote) the predictions of the K models.

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

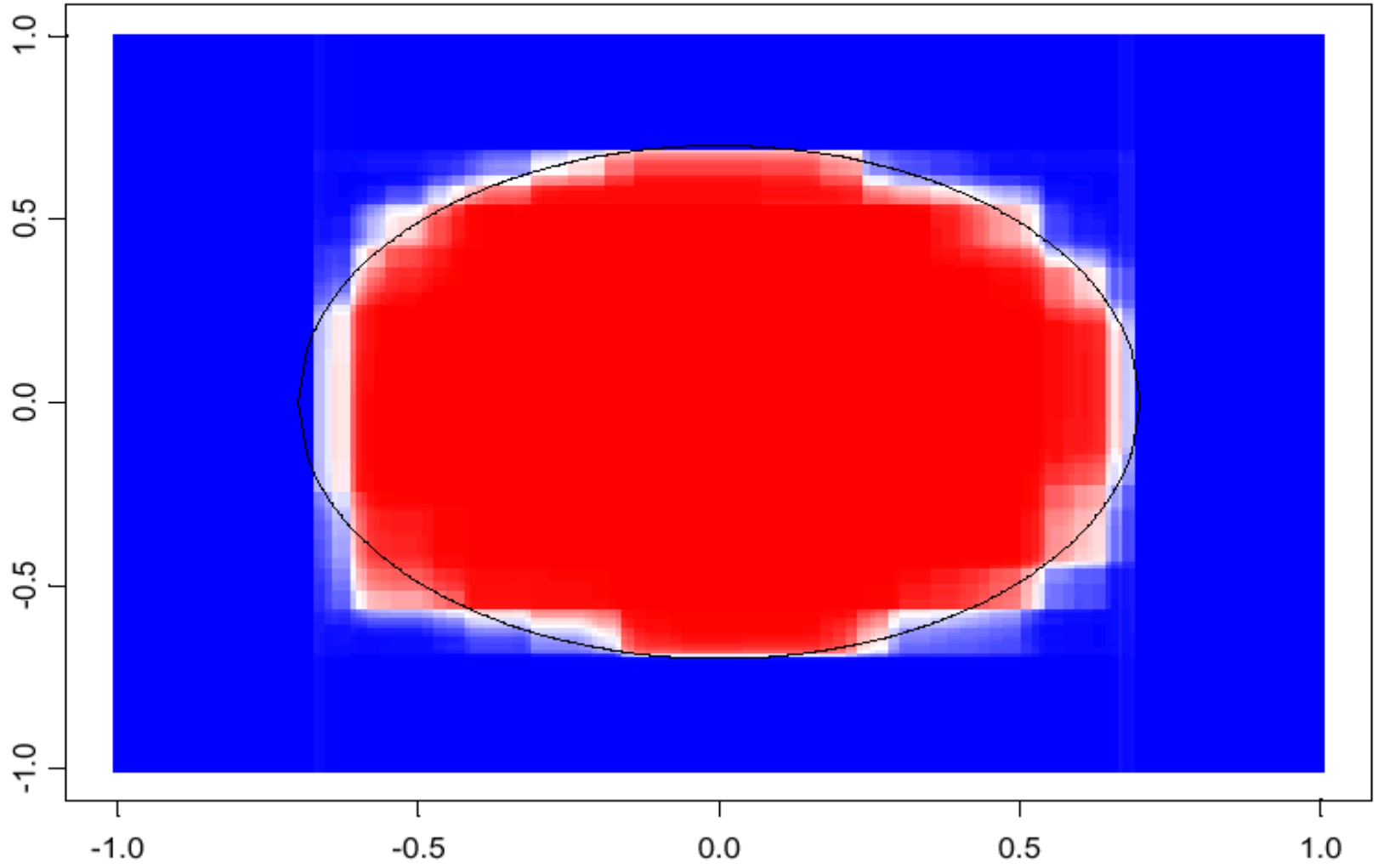
Bagging Example



CART decision boundary

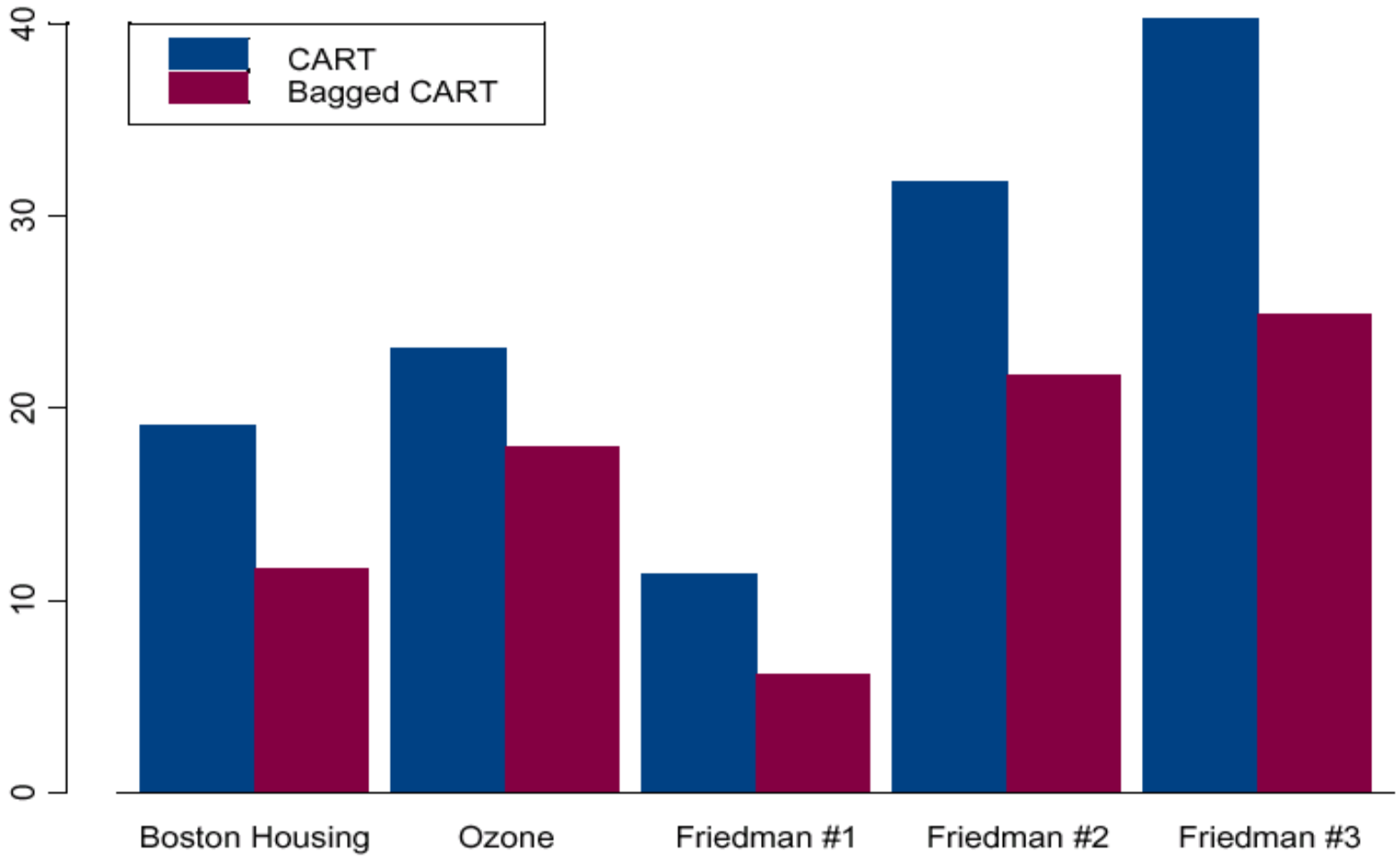


100 bagged trees



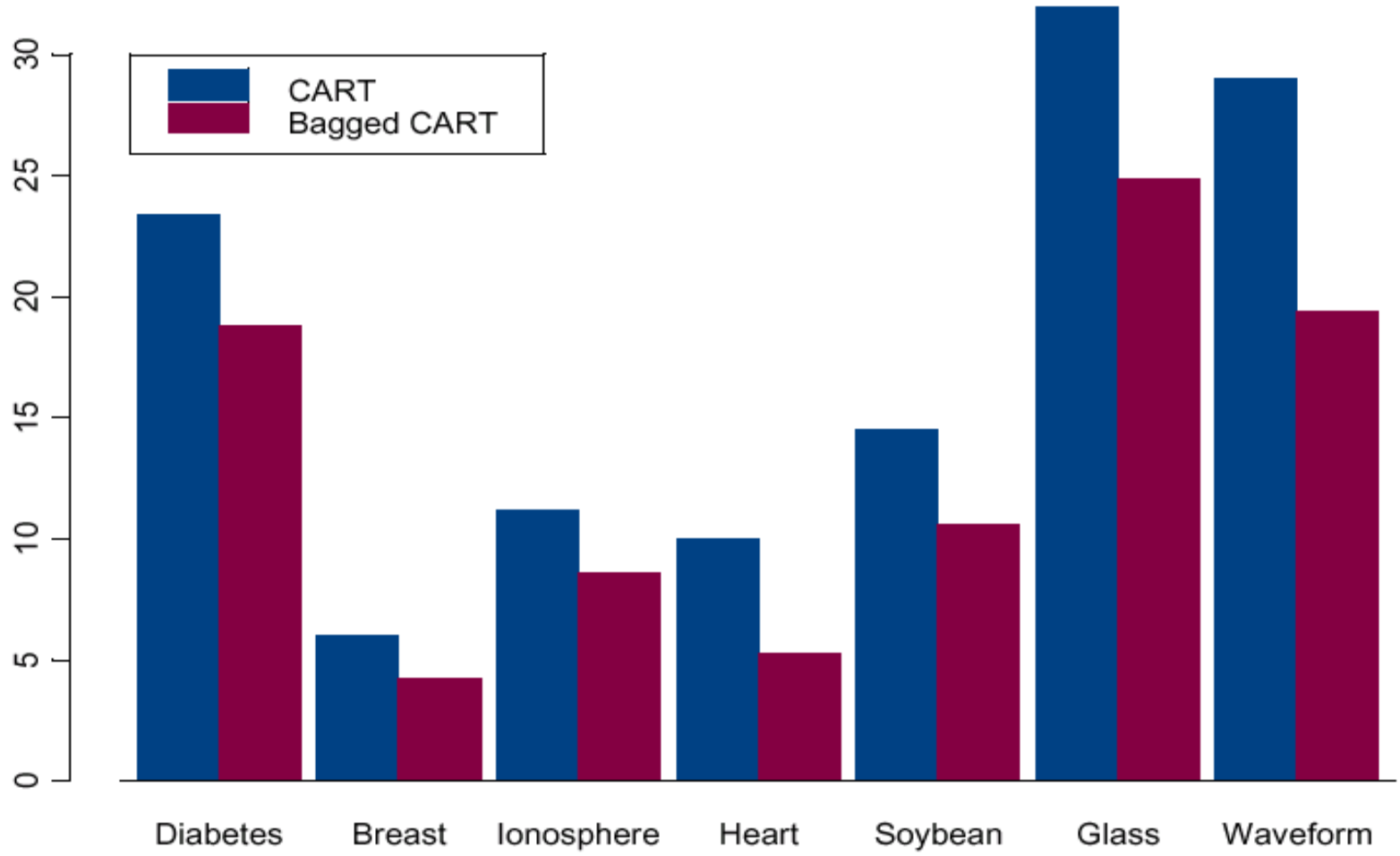
Regression results

Squared error loss



Classification results

Misclassification rates



Random Forests

“ The key to accuracy is low correlation and bias. To keep bias low, trees are grown to maximum depth.

To keep correlation low, the current version uses this randomization.

1) Each tree is grown on a bootstrap sample of the training set.

2) A number m is specified much smaller than the total number of variables M. At each node, m variables are selected at random out of the M, and the split is the best split on these m variables. ”

(see Random Forests , Machine Learning(2001) 45 5-320)

An important feature is that it carries along an internal test set estimate of the prediction error.

For every tree grown, about one-third of the cases are out-of-bag (out of the bootstrap sample). Abbreviated oob.

Put these oob cases down the corresponding tree and get response estimates for them.

For each case n, average or pluralize the response estimates over all time that n was oob to get a test set estimate $\hat{y}_n$ for y_n.

Averaging the loss over all n give the test set estimate of prediction error.

``

Table 3 Test Set Errors (%)

Data Set	Adaboost	Forest-RC		
		Selection	Two Features	One Tree
glass	22.0	24.4	23.5	42.4
breast cancer	3.2	3.1	2.9	5.8
diabetes	26.6	23.0	23.1	32.1
sonar	15.6	13.6	13.8	31.7
vowel	4.1	3.3	3.3	30.4
ionosphere	6.4	5.5	5.7	14.2
vehicle	23.2	23.1	22.8	39.1
German credit	23.5	22.8	23.8	32.6
image	1.6	1.6	1.8	6.0
ecoli	14.8	12.9	12.4	25.3
votes	4.8	4.1	4.0	8.6
liver	30.7	27.3	27.2	40.3
letters	3.4	3.4	4.1	23.8
sat-images	8.8	9.1	10.2	17.3
zip-code	6.2	6.2	7.2	22.7
waveform	17.8	16.0	16.1	33.2
twonorm	4.9	3.8	3.9	20.9
threenorm	18.8	16.8	16.9	34.8
ringnorm	6.9	4.8	4.6	24.6

Adaptive Bagging

Goal: Bias and variance reduction

Method: Sequentially fit *bagged* models, where each fits the current residuals

Properties:

- Bias and variance reduction
- No tuning parameters

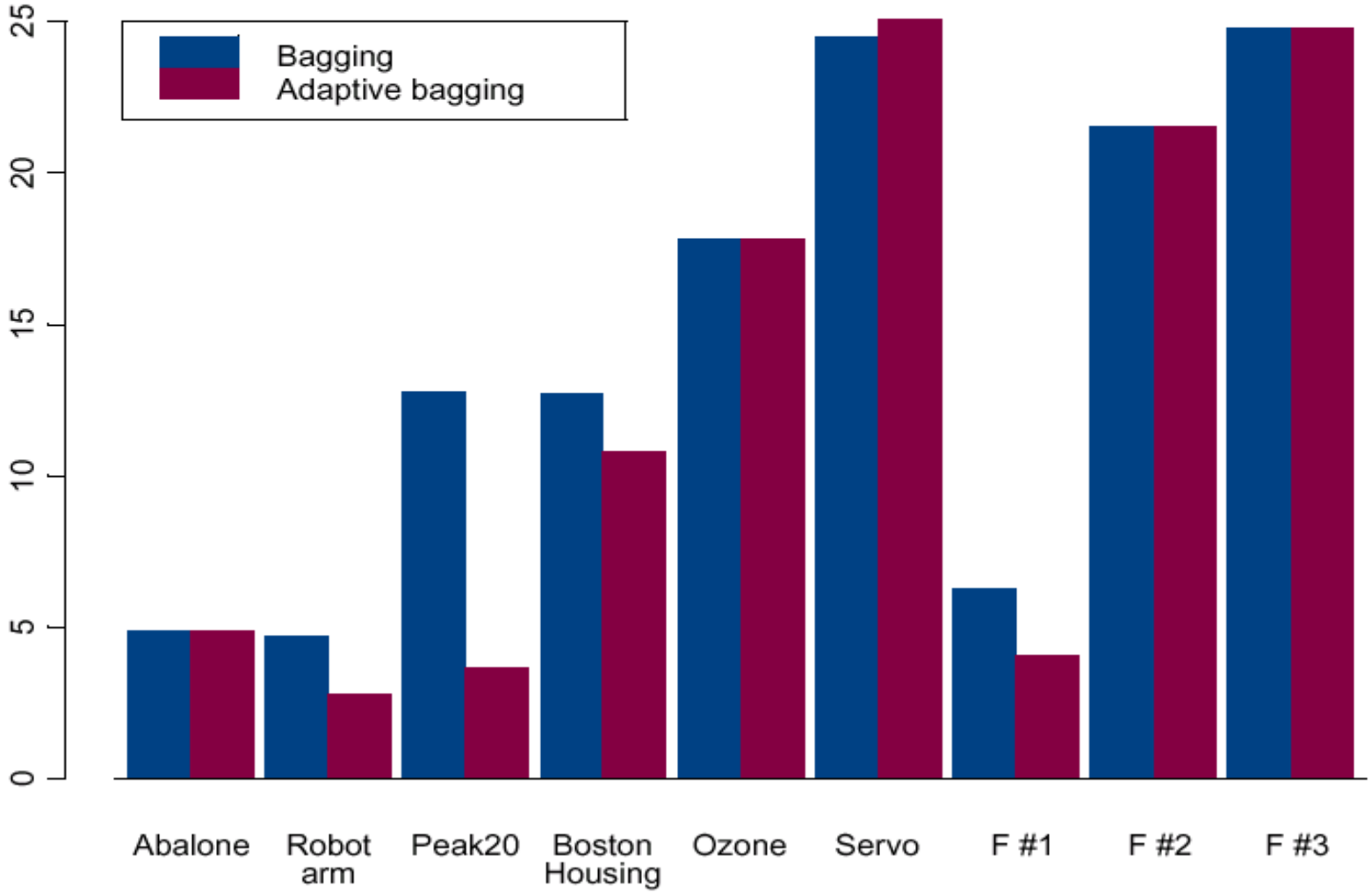
Adaptive bagging algorithm

1. Fit a bagged regressor to the dataset D.
2. Predict "out-of-bag" observations.
3. Fit a new bagged regressor to the bias (error) and repeat.

For a new observation, sum the predictions from each stage.

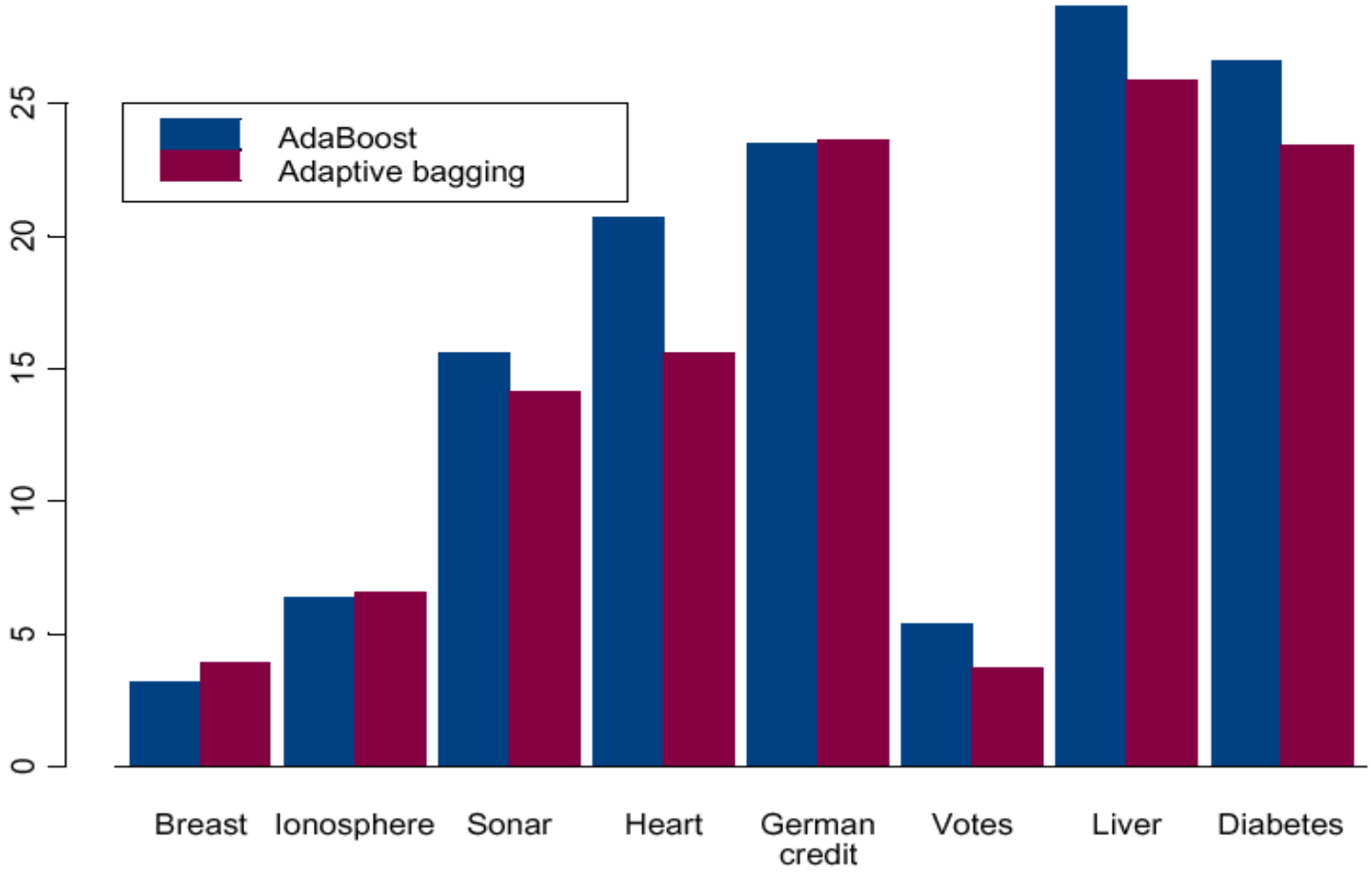
Regression results

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

Peter Buhlmann and Bin Yu. Explaining bagging. Can be downloaded from http://stat.ethz.ch/~buhlmann/bibliog.html, September 2000.

J.H. Friedman and O. Hall. On bagging and nonlinear estimation. Can be downloaded from http://www-stat.stanford.edu/~jhf/#reports, May 2000.

Andreas Buja's home page:

``The Effect of Bagging on Variance, Bias and Mean Squared Error''
A. Buja, W. Stuetzle.
Bootstrap aggregation ("bagging") is a device for reducing the variance of learning algorithms. We give a complete second-order analysis of the effect of bagging on finite sums of U-statistics.

``Smoothing Effects of Bagging''
A. Buja, W. Stuetzle.
An short note on bagging. It relates the von Mises expansion of a bagged statistical functional to the Efron-Stein ANOVA expansion of the unbagged functional to show that the bagged functional is always smooth.

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

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

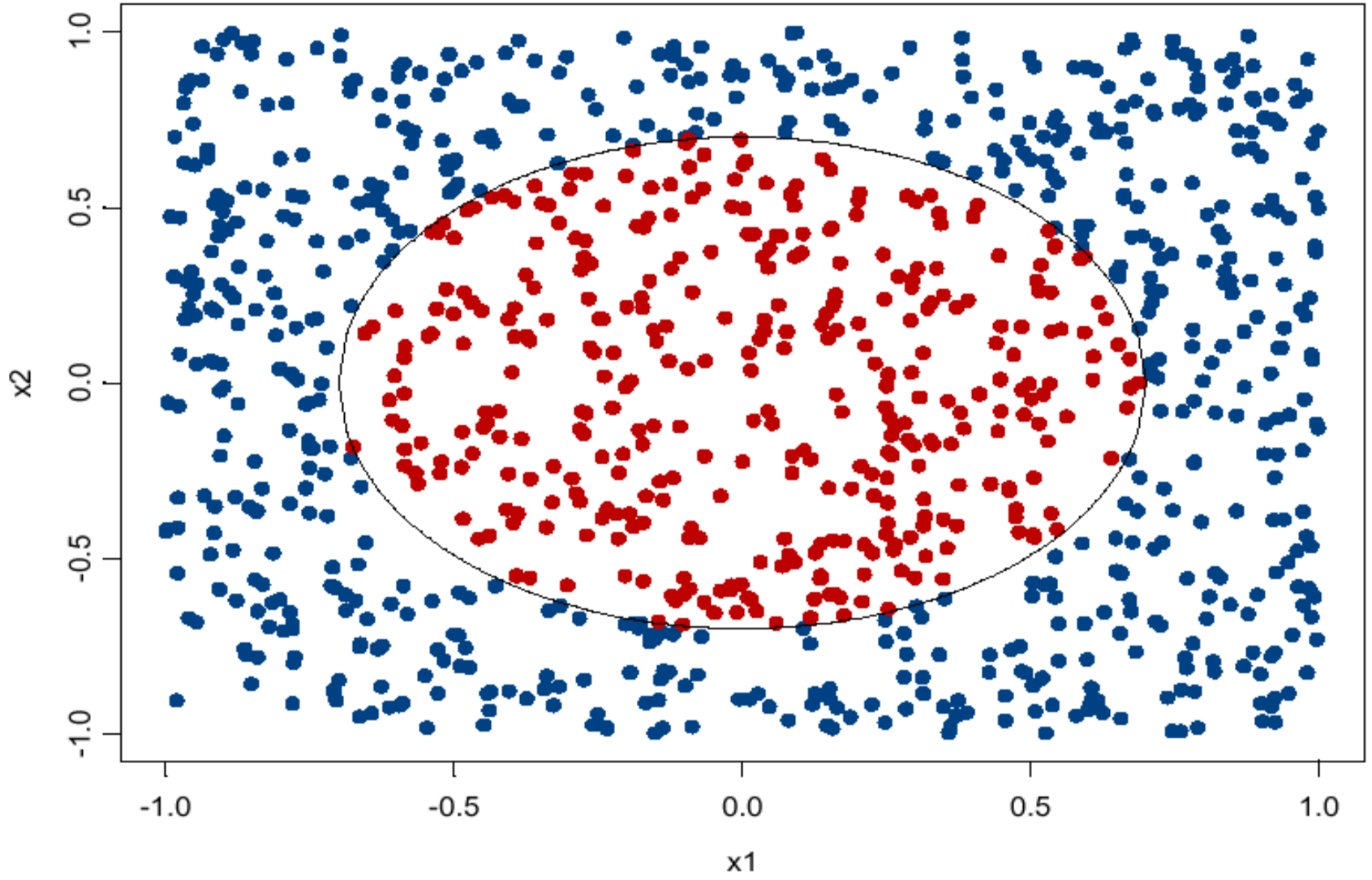
$y_i \in \{-1,1\}$, $w_i = 1/N$

For t in $1,\ldots,T$ do

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

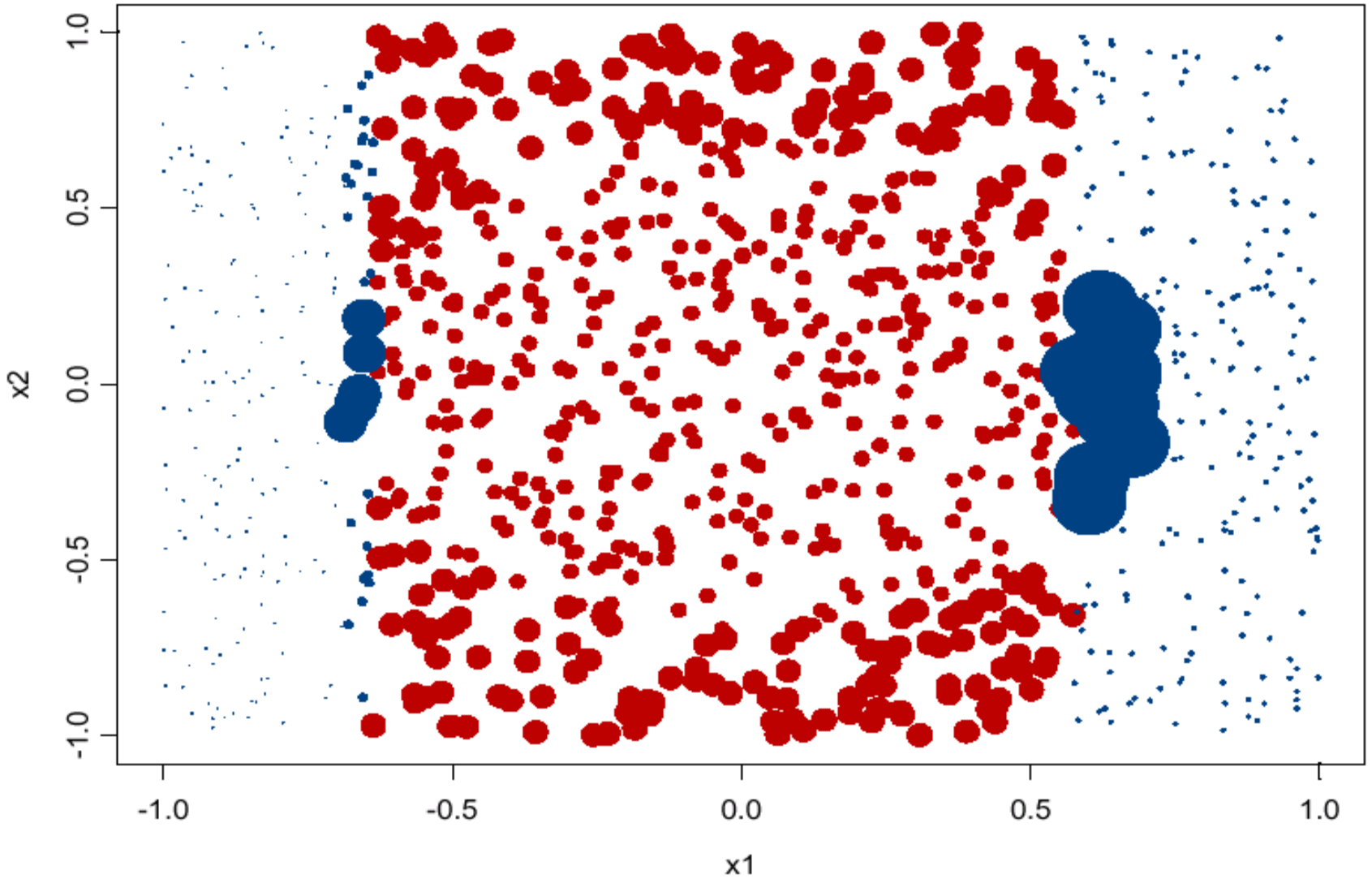
Output the classifier $\boldsymbol{F}(\boldsymbol{x}) = \mathrm{sign}\left(\sum f_t(\boldsymbol{x})\right)$

Boosting Example

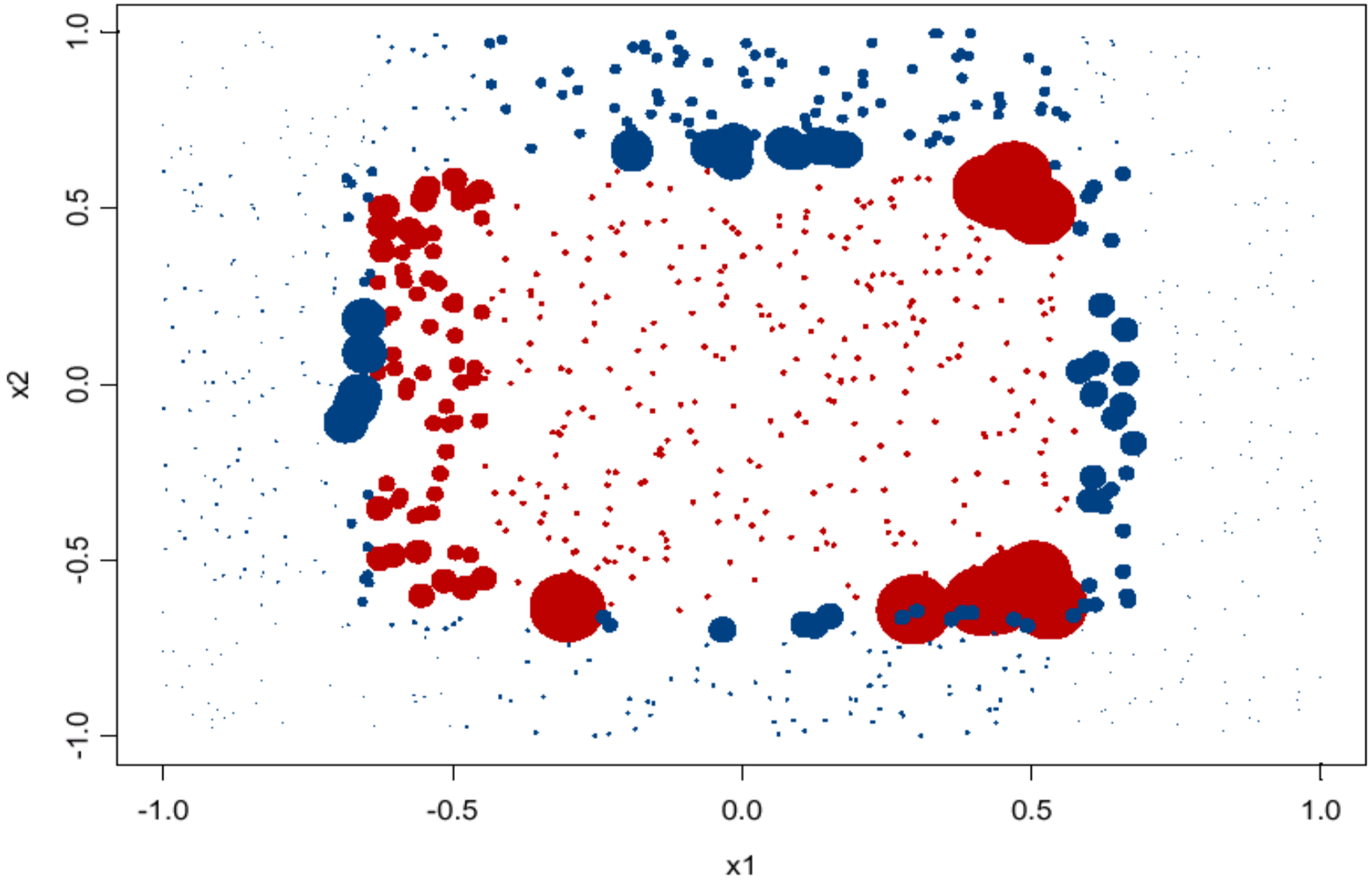


After one iteration

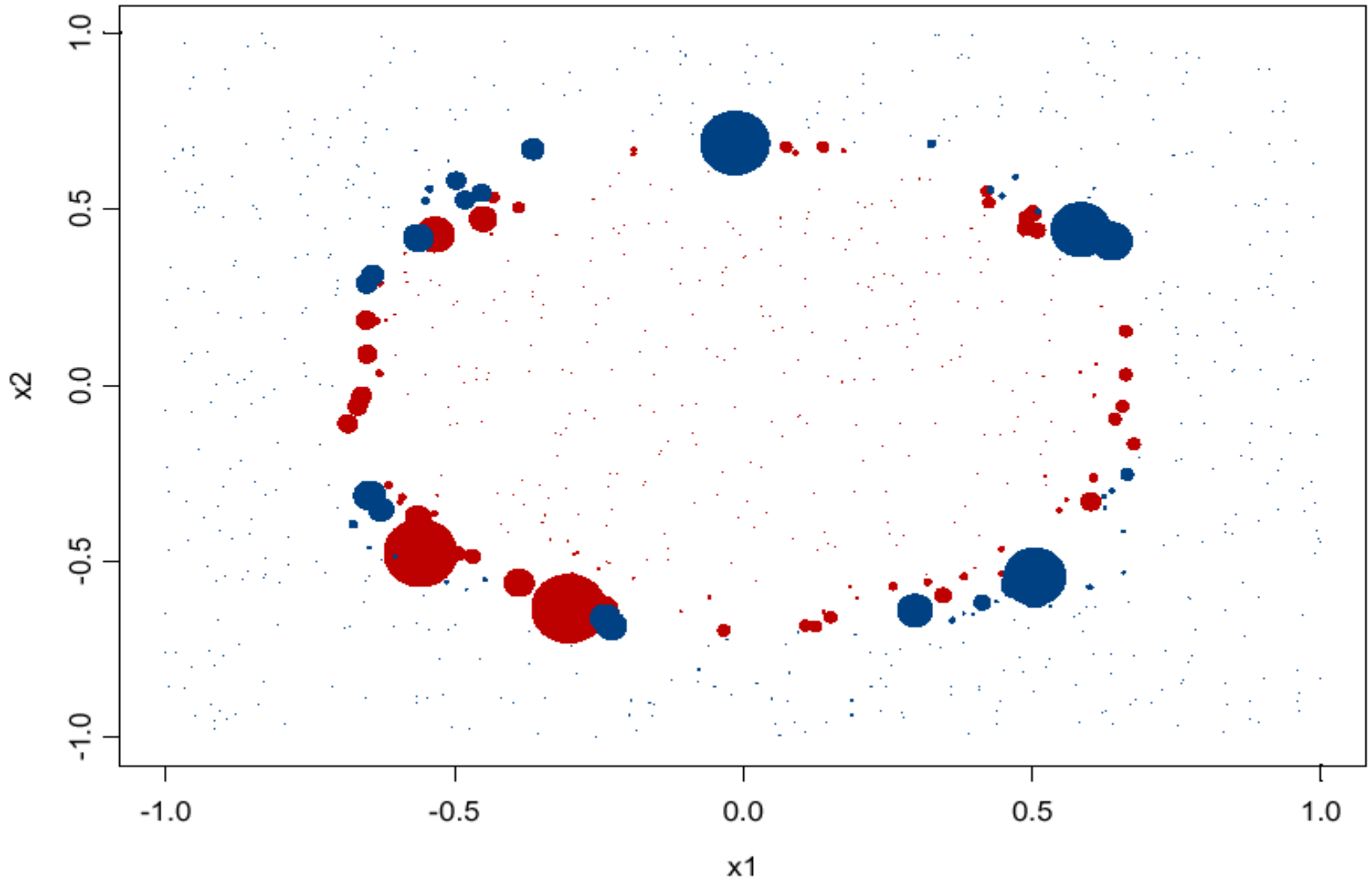
CART splits, larger points have great weight



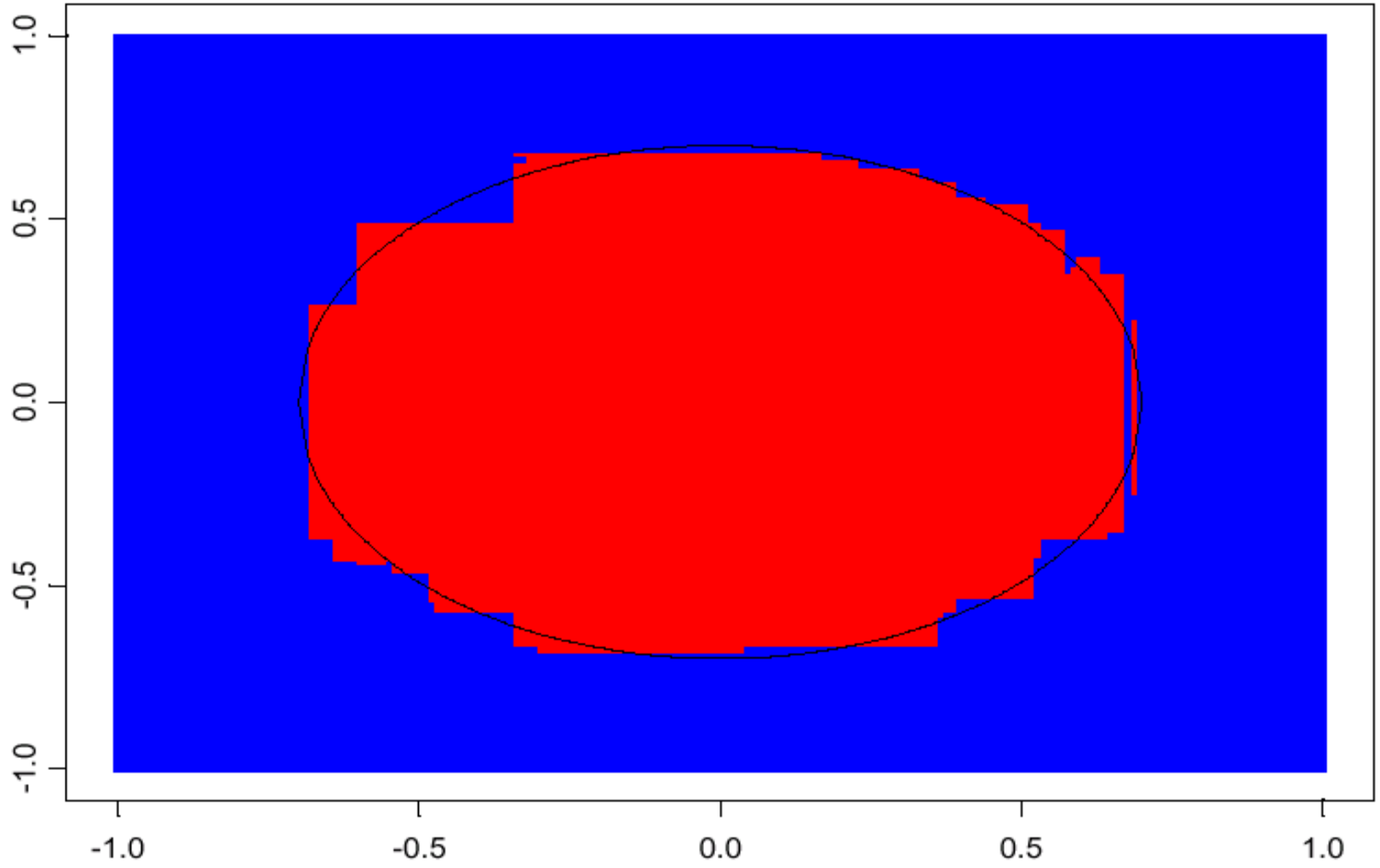
After 3 iterations



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 $y \in \{-1,1\}$, $F(x) \in (-\infty,\infty)$

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

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.

$$J(F) = \mathrm{E}\ell(F) = \mathrm{E}\left[y^* F(\mathbf{x}) - \log\left(1 + e^{F(\mathbf{x})}\right) | \mathbf{x}\right]$$

$$y^* = \tfrac{1}{2}(1+y) \in \{0, 1\}$$

Optimization step

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

- Select f to minimize J…

$$F^{(t+1)} \leftarrow F^{(t)} + \tfrac{1}{2}\log\frac{E_w[I(y=1)\mid x]}{1-E_w[I(y=1)\mid x]}$$

$$w(x,y) = e^{-yF^{(t)}(x)}$$

Let $J(F) = E[e^{-yF(x)}]$. Suppose we have a current estimate $F(x)$ and seek an improved estimate $F(x) + cf(x)$. For fixed c (and x), we expand $J(F(x) + cf(x))$ to second order about $f(x) = 0$

$$\begin{aligned} J(F + cf) &= E[e^{-y(F(x)+cf(x))}] \\ &\approx E[e^{-yF(x)}(1 - ycf(x) + c^2y^2f(x)^2/2)] \\ &= E[e^{-yF(x)}(1 - ycf(x) + c^2/2)] \end{aligned}$$

since y^2=1 and $f(x)^2 = 1$. Minimizing pointwise with respect to $f(x) \in \{-1, 1\}$, we write

$$f(x) = \arg\min_f E_w(1 - ycf(x) + c^2/2|x) \tag{16}$$

Here the notation $E_w(\cdot|x)$ refers to a *weighted conditional expectation*, where $w = w(x, y) = e^{-yF(x)}$, and

$$E_w[g(x, y)|x] \stackrel{\text{def}}{=} \frac{E[w(x, y)g(x, y)|x]}{E[w(x, y)|x]}.$$

For $c > 0$, minimizing (16) is equivalent to maximizing

$$E_w[yf(x)] \tag{17}$$

The solution is

$$f(x) = \begin{cases} 1 & \text{if } E_w(y|x) = P_w(y = 1|x) - P_w(y = -1|x) > 0 \\ -1 & \text{otherwise} \end{cases} \tag{18}$$

LogitBoost

Friedman, Hastie, Tibshirani [1998]

- Logistic regression

$$y = \begin{cases} 1 & \text{with probability } p(x) \\ 0 & \text{with probability } 1 - p(x) \end{cases}$$

$$p(x) = \frac{1}{1 + e^{-F(x)}}$$

- Expected log-likelihood of a regressor, $F(x)$

$$\mathrm{E}\,\ell(F) = \mathrm{E}\left(yF(x) - \log(1 + e^{F(x)}) \mid x\right)$$

Newton steps

$$J(F+f) = E\left(y(F(x)+f(x)) - \log(1+e^{F(x)+f(x)}) \mid x\right)$$

- Iterate to optimize expected log-likelihood.

$$F^{(t+1)}(x) \leftarrow F^{(t)}(x) - \frac{\frac{\partial}{\partial f} J(F^{(t)}+f)\Big|_{f=0}}{\frac{\partial^2}{\partial f^2} J(F^{(t)}+f)\Big|_{f=0}}$$

LogitBoost, continued

- Newton steps for Bernoulli likelihood

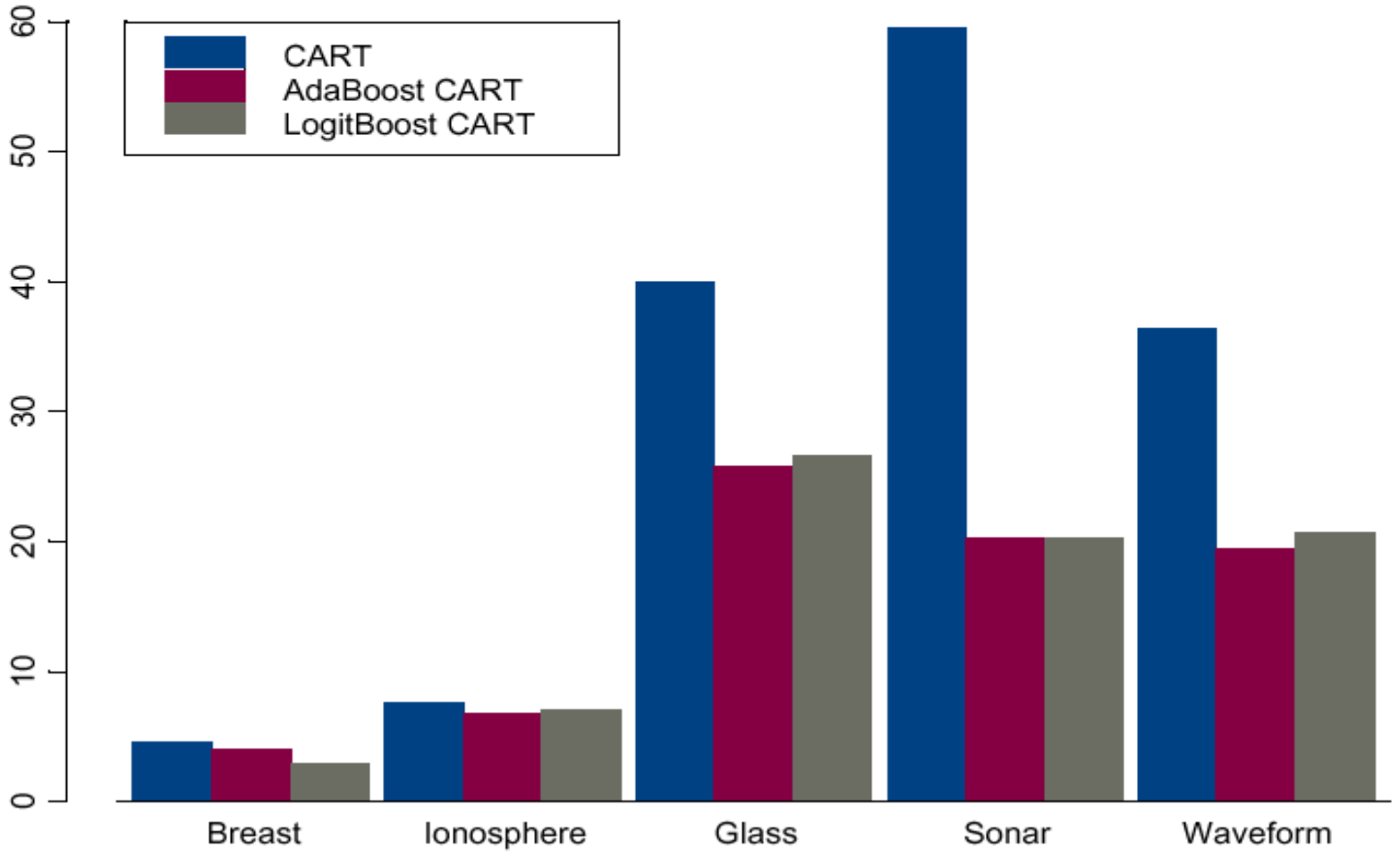
$$F(x) \leftarrow F(x) + E_w\left(\frac{y - p(x)}{p(x)(1-p(x))} \middle| x \right)$$

$$w(x) = p(x)(1-p(x))$$

- In practice the $E_w(\bullet|x)$ can be any regressor - trees, smoothers, etc.
- Trees are adaptive and work well for high dimensional data.

Misclassification rates

Friedman, Hastie, Tibshirani [1998]



Naïve Bayes Classification

Probabilistic Classification

$$P(Y = y \mid X_1 = x_1, \ldots, X_d = x_d) = \frac{P(\underline{X} \mid Y = y)P(Y = y)}{P(\underline{X})}$$

The naïve Bayes assumption

$$P(\underline{X} \mid Y = y) = P(X_1 = x_1 \mid Y = y) \cdots P(X_d = x_d \mid Y = y)$$

Estimation

- Probability estimates are trivial

$$\hat{P}(X_j = x_j \mid Y = y) = \frac{\text{count}(X_j = x_j \cap Y = y)}{\text{count}(X_j = x_j)}$$

- Estimation is linear in the number of predictors and the number of observations

Interpretability

Consider the log-odds in favor of Y=1

$$\log \frac{P(Y=1 \mid \underline{X})}{P(Y=0 \mid \underline{X})} = w_0 + \sum_{j=1}^{d} w_j(X_j)$$

- Positive w_j are evidence in favor of Y=1
- Negative w_j are evidence in favor of Y=0

Evidence balance sheets

Evidence in favor of knee surgery		**Evidence against knee surgery**	
Female	+8	Prior evidence	-10
Knee is unstable	+88	Age 50	-12
Knee locks	+172	No effusion	-62
Tender med JL	+49	Negative McMurray's	-38
Total positive evidence	+317	Total negative evidence	-122
Total evidence		+195	
Probability of knee surgery		88%	

Boosting algorithms

1. Learn a classifier from the data
2. Upweight observations poorly predicted, downweight observations well predicted
3. Refit the model using the new weighting
4. After T iterations, have each model vote on the final prediction.

AdaBoost algorithm

Freund & Shapire (1997)

- AdaBoost defines a particular reweighting scheme and a voting method for merging the classifiers
- AdaBoost decreases bias and variance in many settings - Bauer and Kohavi [1998]
- Boosted naïve Bayes tied for first place in the 1997 KDD Cup

AdaBoost

- Extremely dense voting scheme

$$P(Y=1\mid x)=\frac{1}{1+\prod_{t=1}^{T}\beta_t^{2r(x)-1}} \qquad r(x)=\frac{\sum_{t=1}^{T}(\log\frac{1}{\beta_t})P_t(Y=1\mid x)}{\sum_{t=1}^{T}(\log\frac{1}{\beta_t})}$$

- Destroys interpretability

Regaining Interpretability

Rewriting the voting scheme…

$$\log \frac{P(Y=1 \mid X)}{P(Y=0 \mid X)} = \sum_{t=1}^{T} (\log \beta_t) \left(1 - 2 \left(1 + e^{-\log \frac{P_t(Y=1|X)}{P_t(Y=0|X)}} \right)^{-1} \right)$$

Substitute Taylor expansion…

$$\frac{1}{1+e^{-x}} = \tfrac{1}{2} + \tfrac{1}{4} x - \tfrac{1}{48} x^3 + O\left(x^5\right)$$

Regained Interpretability

$$\sum_{t=1}^{T} \alpha_t \log \frac{P_t(Y=1)}{P_t(Y=0)} + \sum_{j=1}^{d} \sum_{t=1}^{T} \alpha_t \log \frac{P_t(X_j \mid Y=1)}{P_t(X_j \mid Y=0)}$$

$$= \text{boosted prior weight of evidence} +$$

$$\sum_{j=1}^{d} \text{boosted weight of evidence from } X_j$$

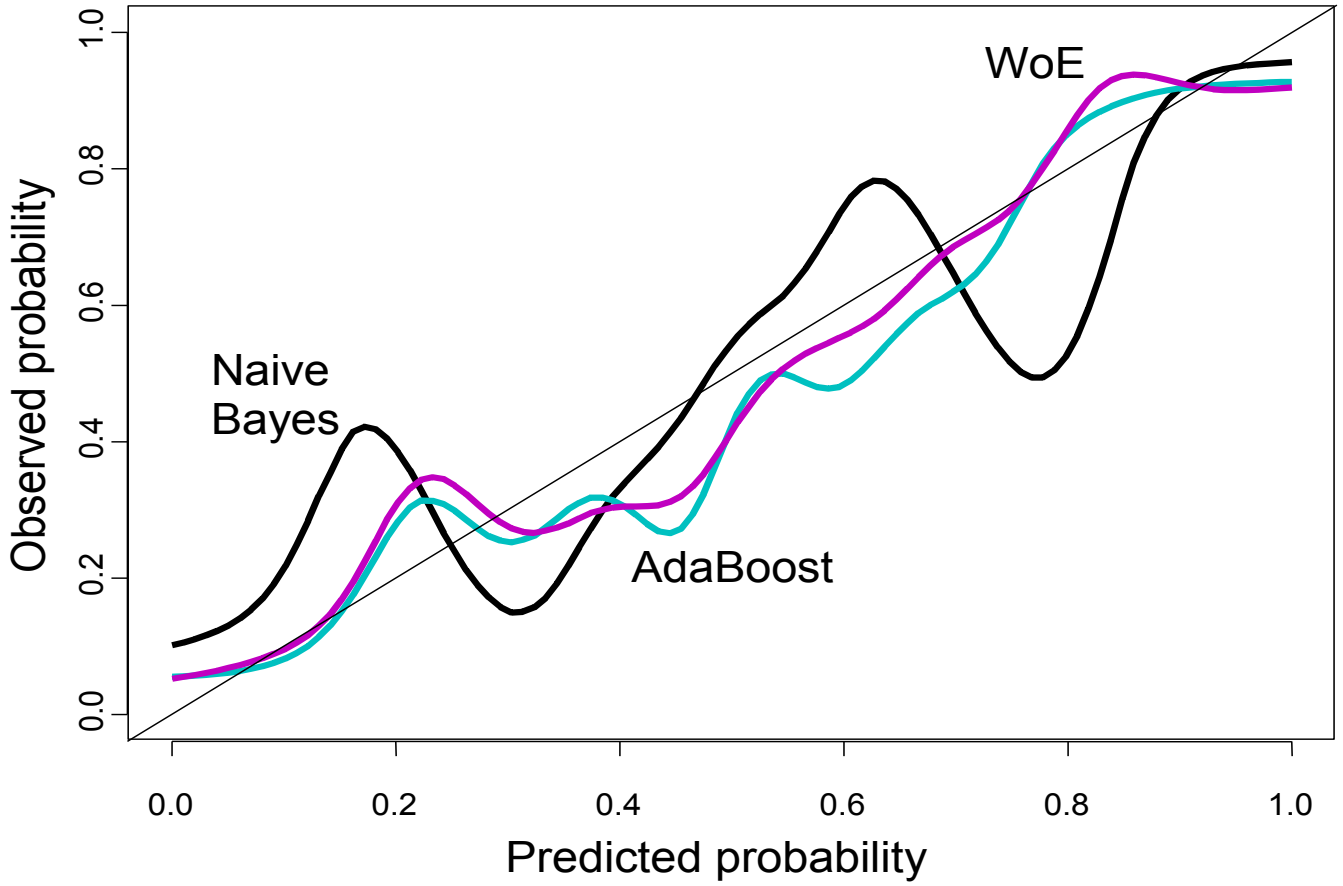
- **Boosting biases parameter estimates**
- **Adjusts naïve Bayes' for over-optimism**

Misclassification rates

	Naïve Bayes	AdaBoost	Weight of evidence
Knee diagnosis	14.0%	13.8%	13.4%
Diabetes	25.0%	24.4%	24.4%
Credit approval	16.8%	15.5%	15.5%
CAD	18.4%	18.3%	18.3%
Breast tumors	3.9%	3.8%	3.8%

- Boosting offers modest improvement
- Actual AdaBoost and approximation are close

Calibration



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.