Prediction in the Era of Massive Datasets

Greg Ridgeway

RAND Statistics Group

Santa Monica, California, USA

http://gregridgeway.homestead.com

RAND

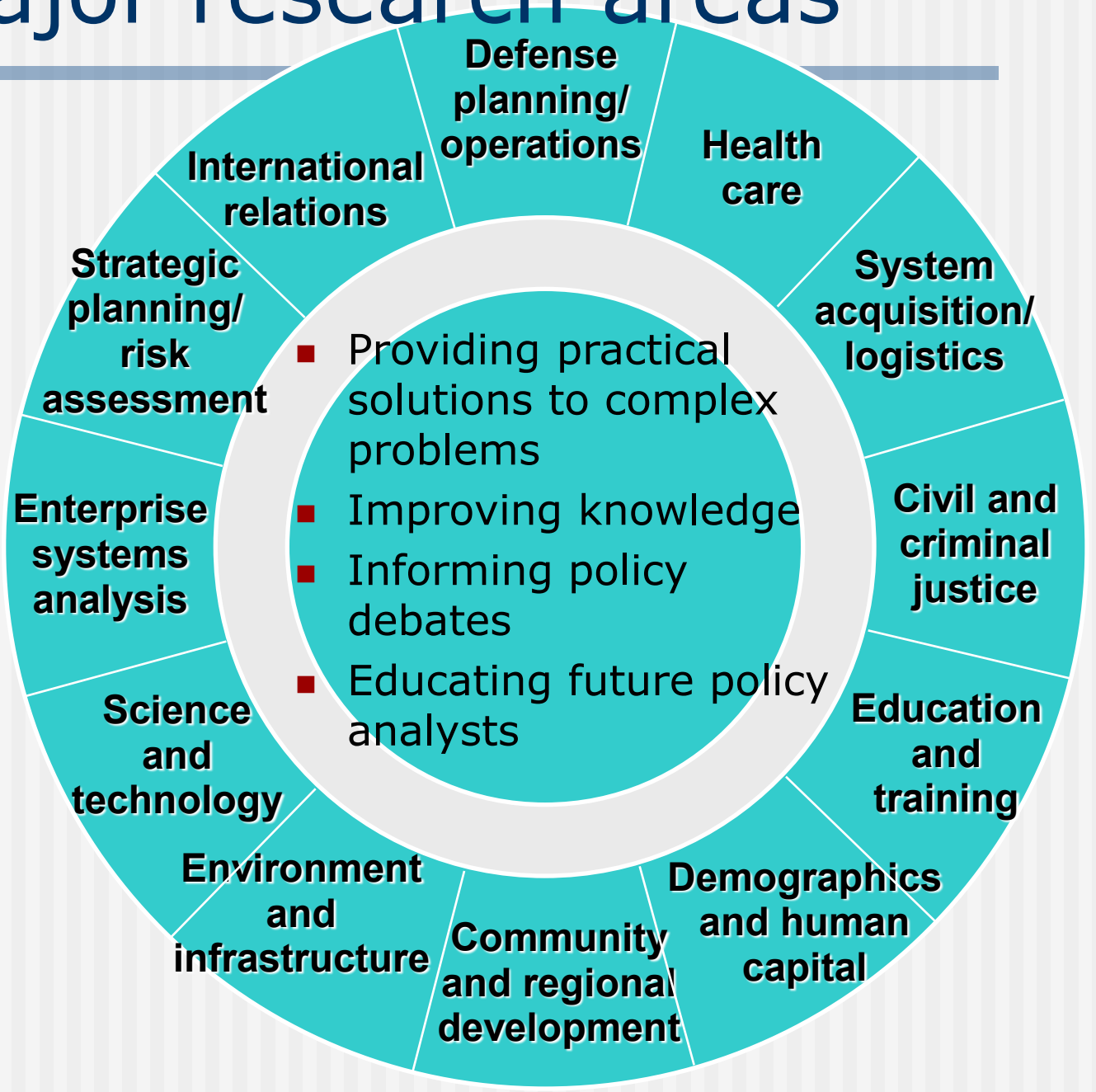
Statistics Group

25th Anniversary

1976 — 2001

www.rand.org/centers/stat

Major research areas



RAND's research and operating locations are worldwide

Statistics examples

Design the new Medicare payment system for rehabilitation hospital care

Use experimental design and spatial statistics to plan computer runs of complex models

Analyze a group randomized experiment of a drug prevention program for middle school students

Load balancing of electrical power generation under deregulation

Conduct an assessment of domestic terrorism preparedness

Analysis of quality of Internet surveys and survey methodology

RAND Also Conducts Private Sector Work That Is in the Public Interest

- Global risk evaluation for overseas capital investments
- Load balancing of electrical power generation under deregulation
- Supply chain management
- Health care plan criteria in the U.S. automobile industry
- Safety options for Amsterdam's Schiphol airport

Outline

- Prediction problems
  - model complexity
  - data access complexity
- Decision trees
  - An algorithm
  - Accuracy, efficiency, and interpretation
  - Overfitting
- Recent innovations

Prediction problems

- Symptoms → Disease
- Credit application → profit
- Assessment at age 12 → high school graduation
- Transaction record → fraud
- Books purchased → other books to purchase
- Criminal record → time until repeat crime

Data mining is…

- Data analysis
- With datasets that are generally
  - Massive, cannot fit into a computer's main memory
  - Observational
  - Retrospective
  - Noisy
  - High-dimensional
  - Unstructured

Data mining is not…

- a replacement for carefully thought out data analysis.
- able to magically make amazing discoveries.
- a stand-alone process but rather is a step in the process involving data collection, data management, data analysis, and *thought*.

Goals of prediction

- From a training dataset
  - Independent observations with observed features and an observed outcome
- learn a function that takes in the features and outputs a prediction
- That minimizes the risk on future observations
  - Misclassification = probability of mislabeling
  - Squared-error loss =

    average of $(\text{actual} - \text{predicted})^2$
  - Absolute loss = average of $|\text{actual} - \text{predicted}|$

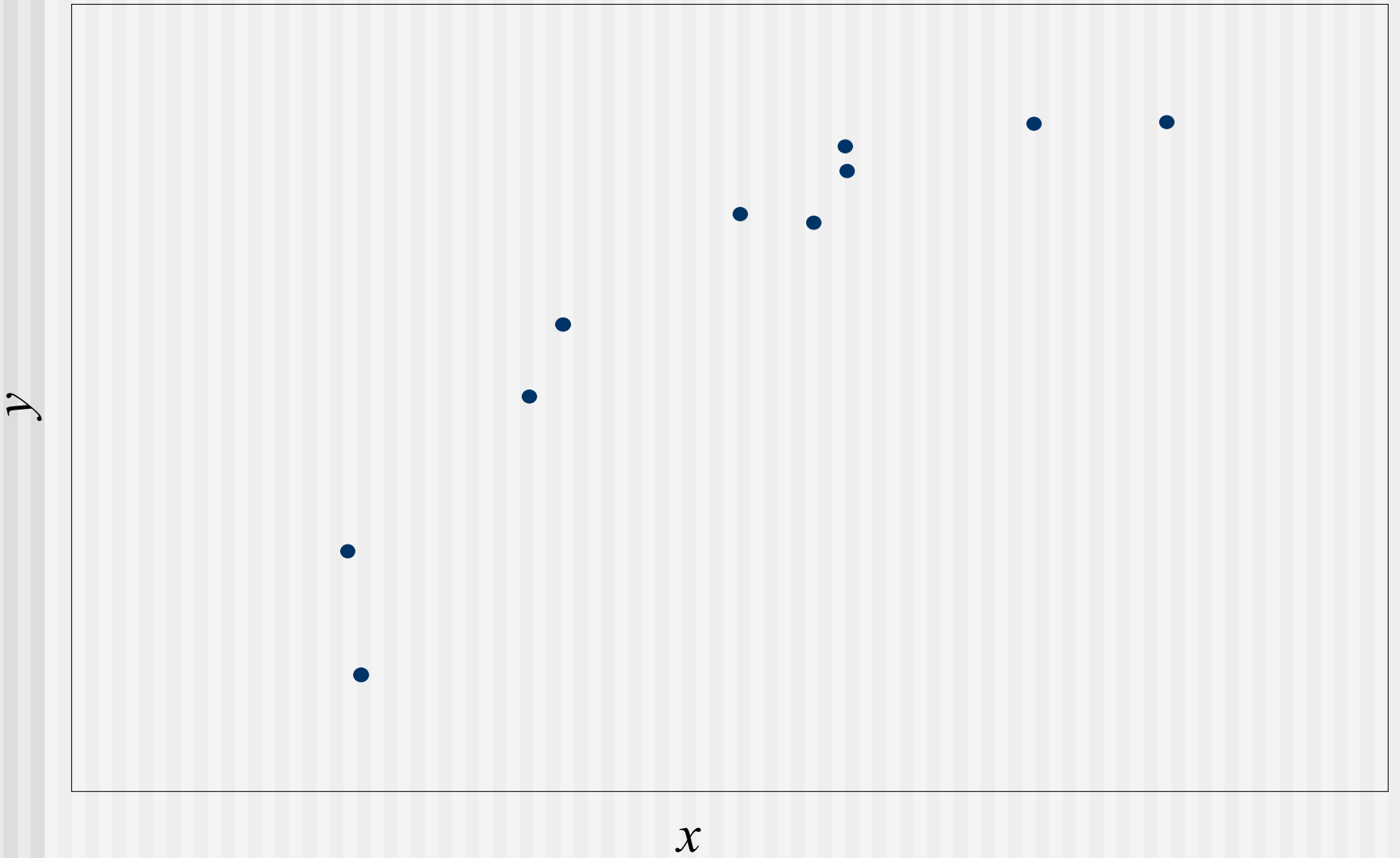
Complexity

- Having a massive dataset
  - allows us to use more complex models and, therefore, make more accurate predictions
  - causes data access complexity since scanning the disk *one million times* slower than scanning memory

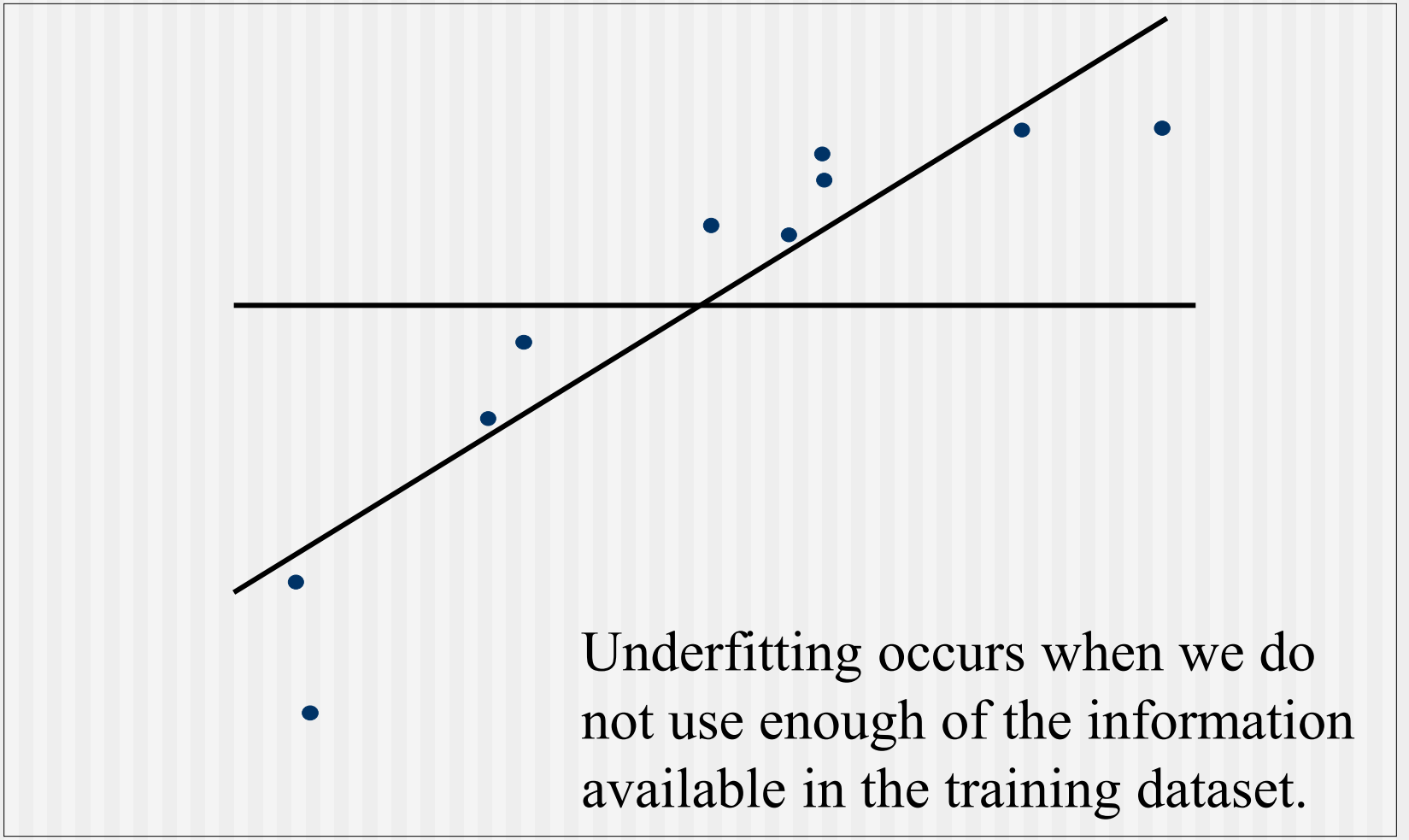
model complexity &

data access complexity

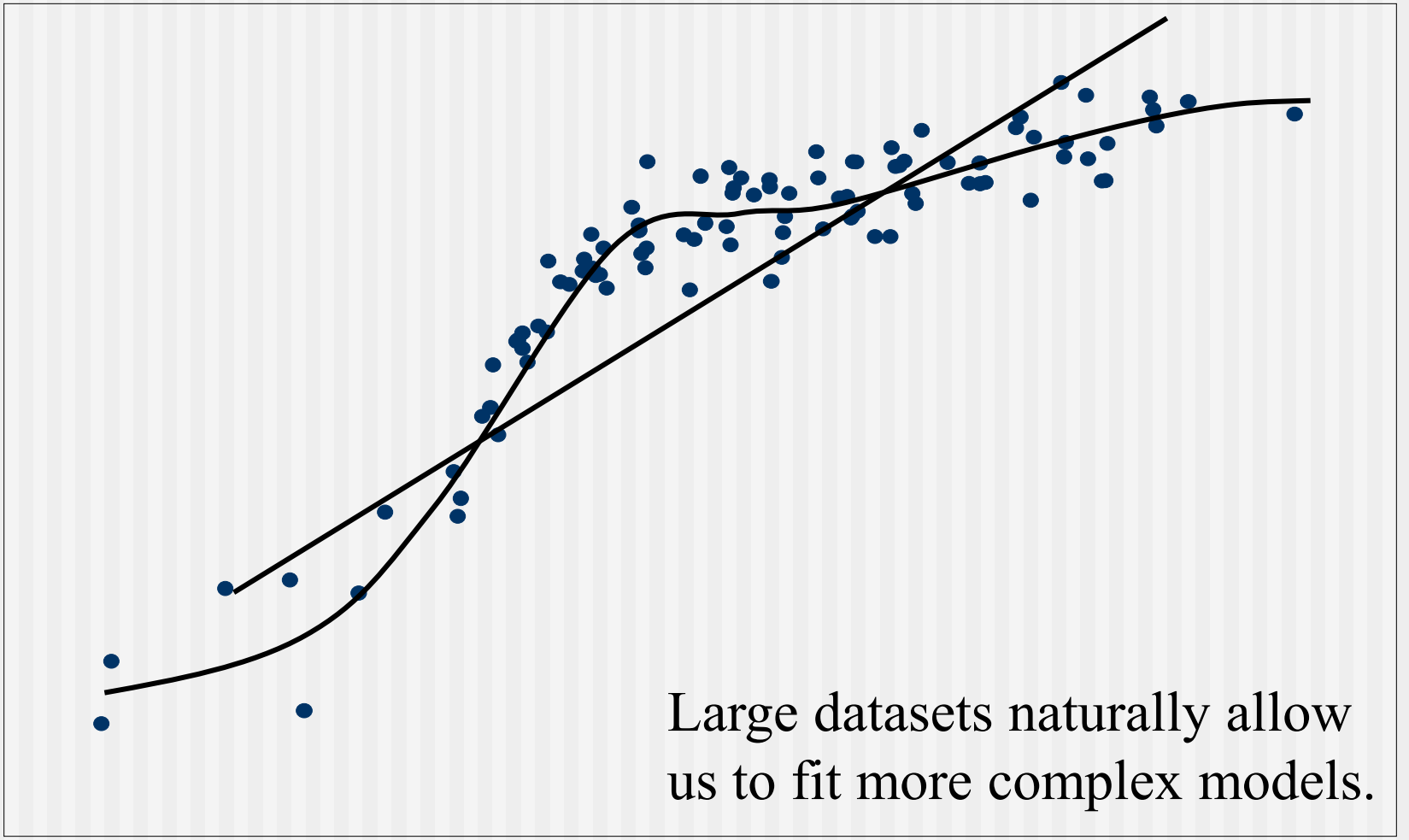
Model complexity example



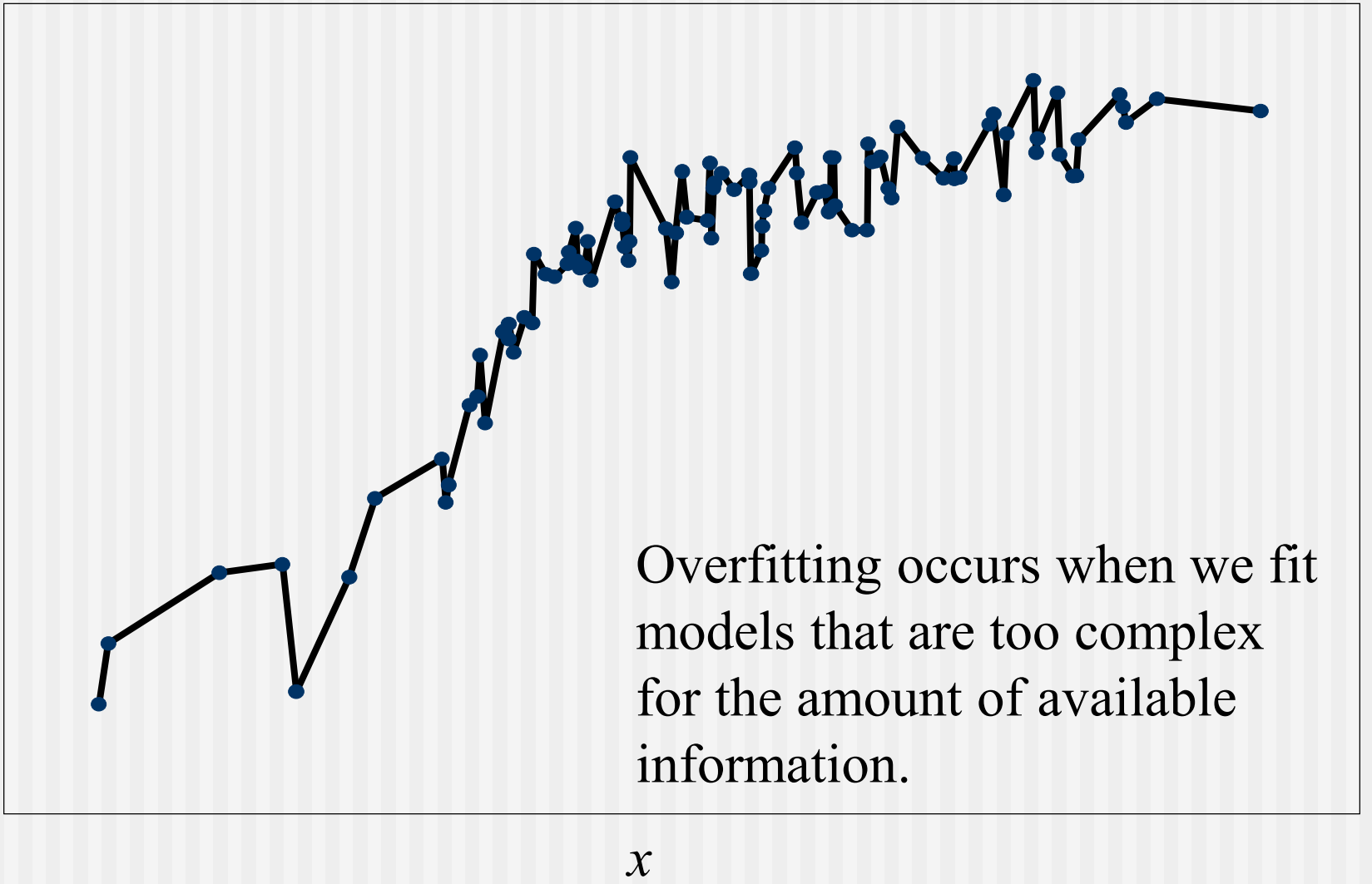
Underfitting



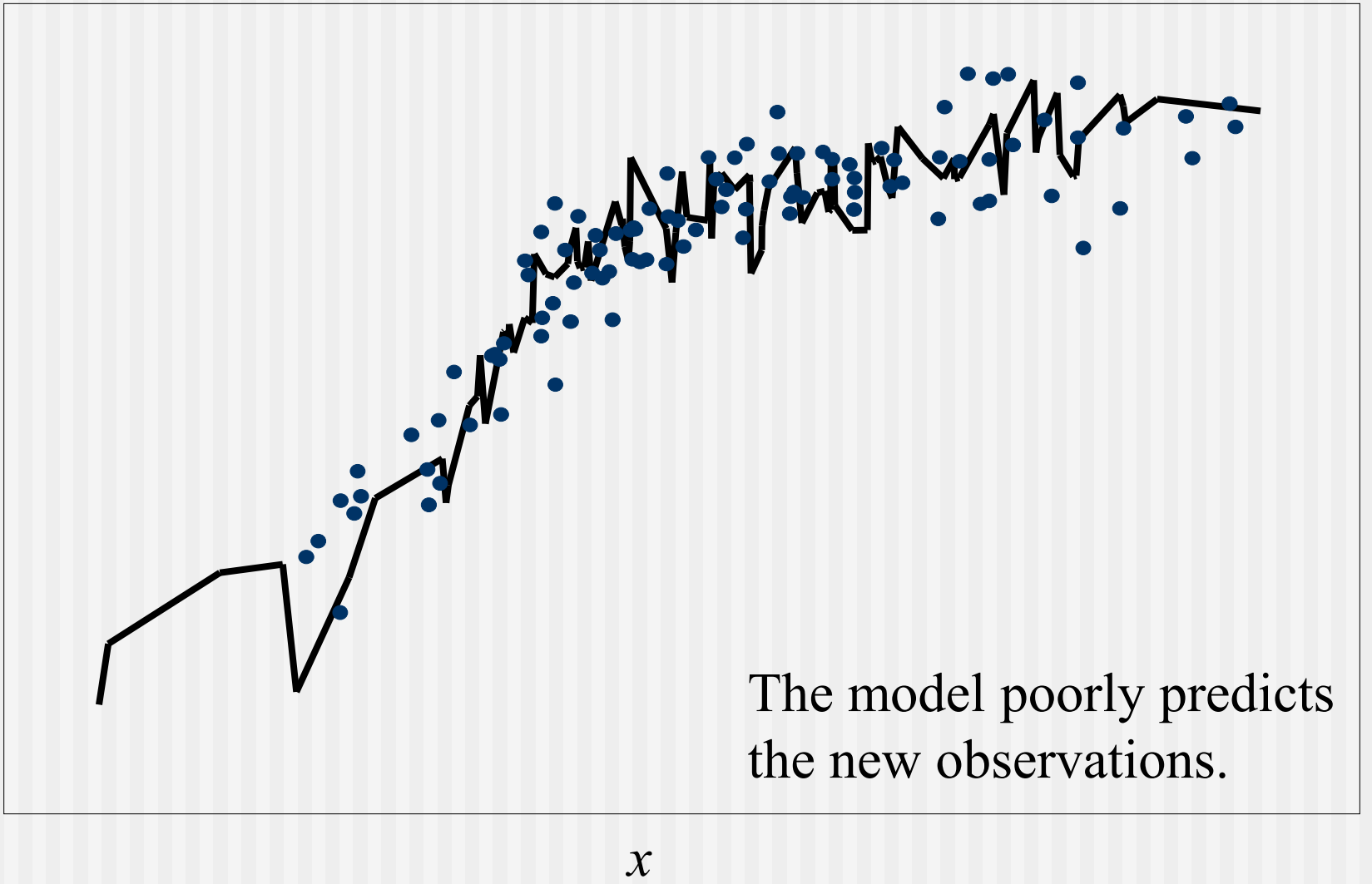
Underfitting



Overfitting



Overfitting



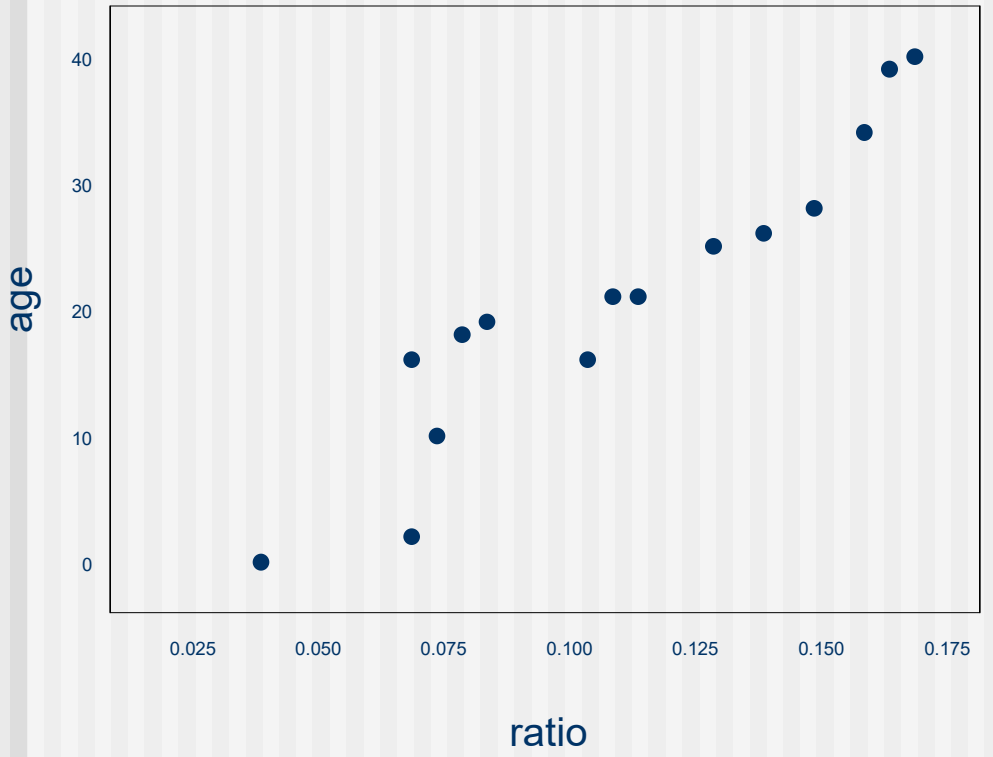
Underfitting and Overfitting

- Underfitting – not absorbing all of the information into the model
- Overfitting – learning the training dataset so well that the model cannot generalize

In between the underfit and overfit model lies the model that minimizes risk.

Predict age at death

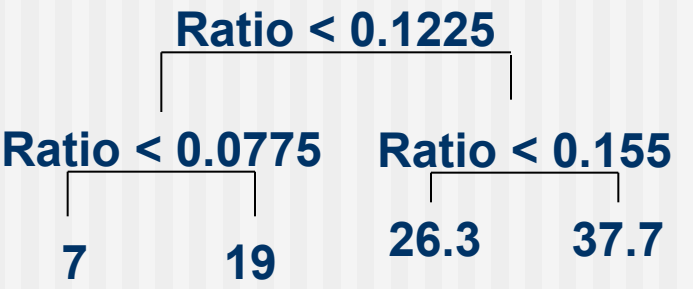
- During aging, L-aspartic acid transforms into its D-form. Researchers obtained bone specimens from 15 human skulls with known age at death and measured the ratio of D-aspartic to L-aspartic acid.



Ratio of D-aspartic to L-aspartic	Age
0.040	0
0.070	2
0.070	16
0.075	10
0.080	18
0.085	19
0.105	16
0.110	21
0.115	21
0.130	25
0.140	26
0.150	28
0.160	34
0.165	39
0.170	40

Decision Trees

- Extract from the data a decision tree
  1. which predictor variables to use
  2. where to split, and
  3. what value to output for each terminal node.



- Algorithms search over tree configurations to find one that produces accurate outputs on the observed data.
  1. Low misclassification error
  2. Low average squared bias

Regression trees

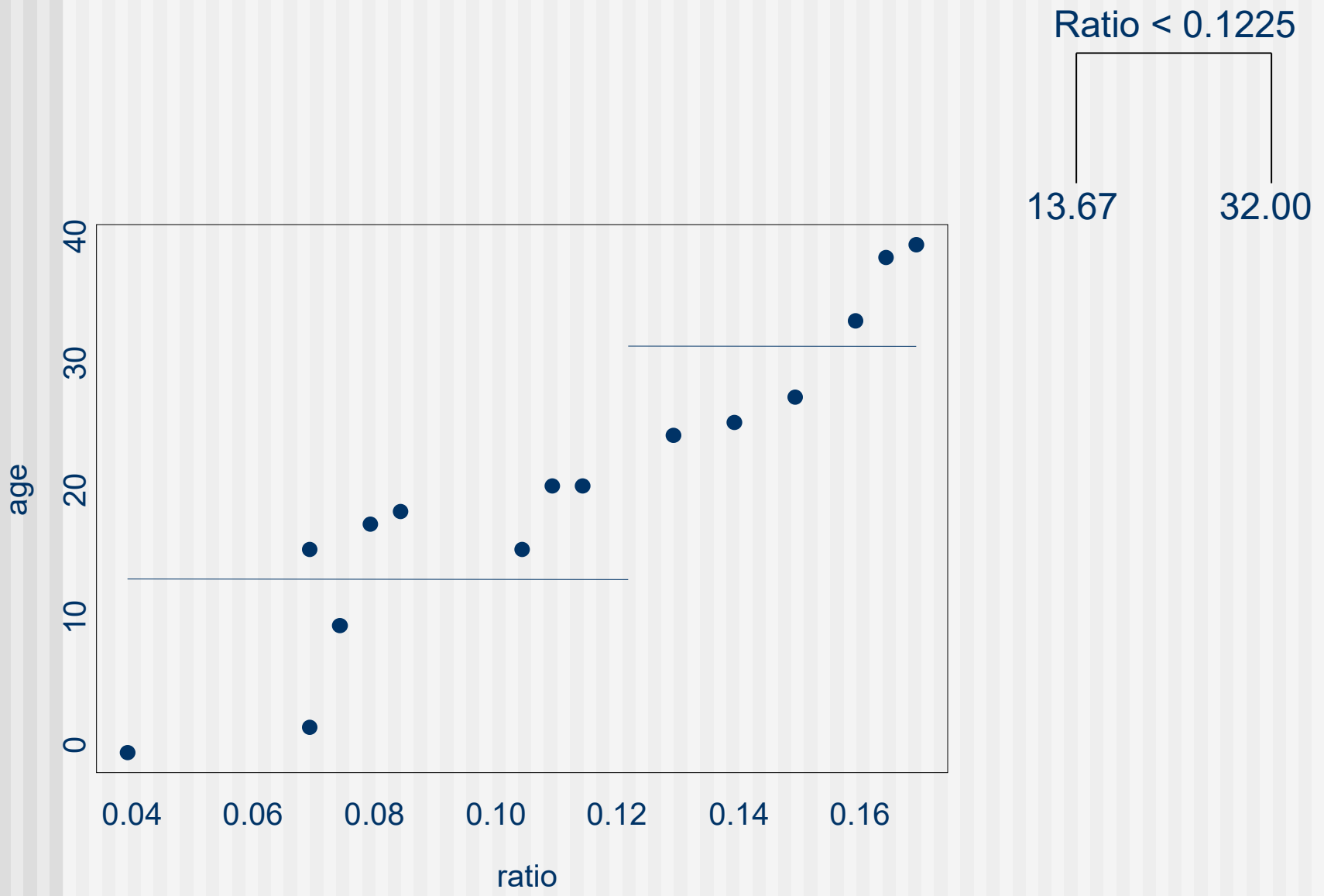
- A **regression tree** is a tree-structured prediction model that has, as an output, a continuous variable.
- Idea:
  1. Start with all observations in a root node.
  2. Split the dataset into two homogenous groups.
  3. Predict the average output of each group.
  4. Repeat this recursively down the tree until the number of observations in a terminal node is too small.

Choosing the split

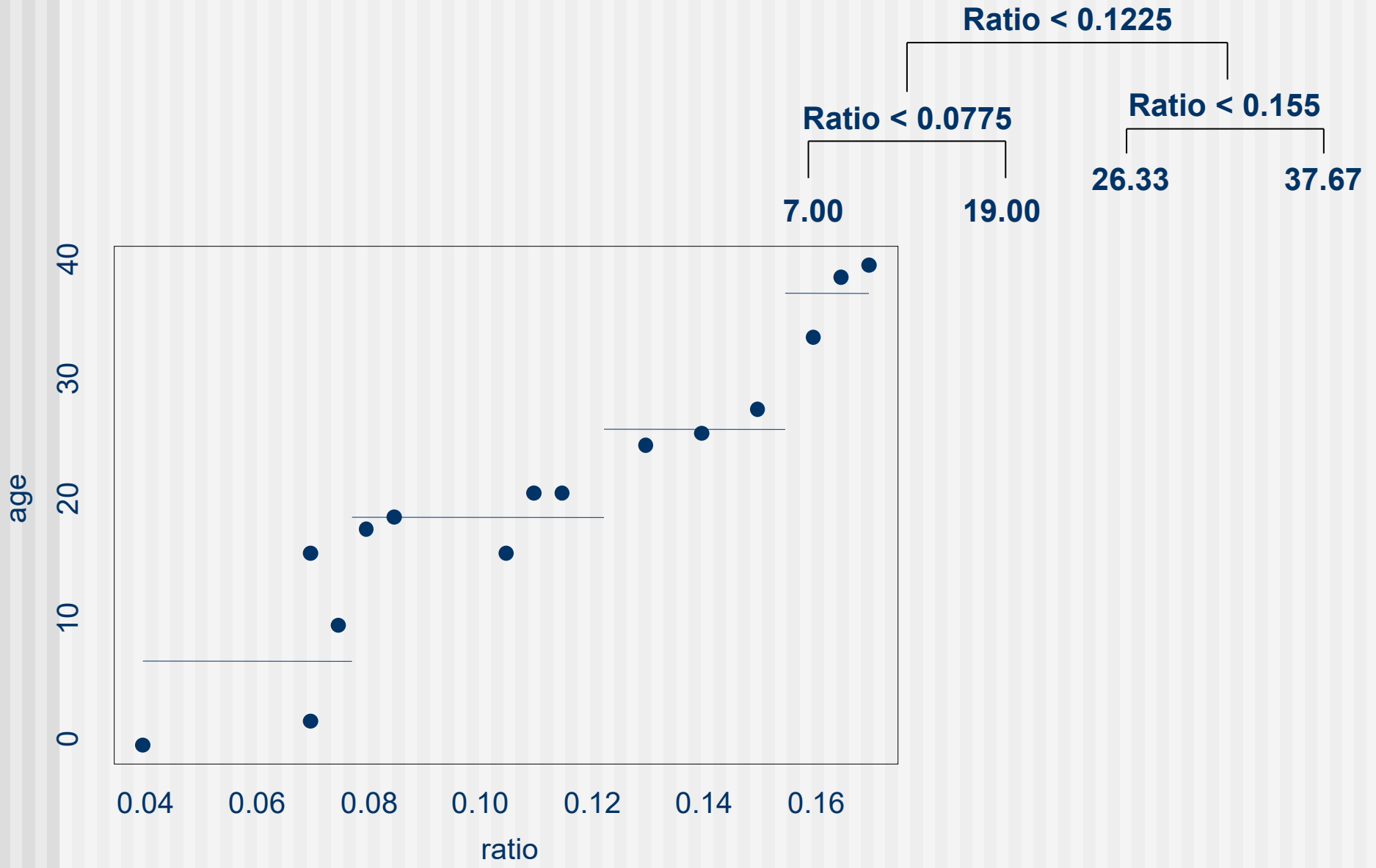
Ratio of D-aspartic and L-aspartic	Age
0.040	0
0.070	2
0.070	16
0.075	10
0.080	18
0.085	19
0.105	16
0.110	21
0.115	21
0.130	25
0.140	26
0.150	28
0.160	34
0.165	39
0.170	40

Split	Prediction left	Prediction right	squared-error
$x \leq 0.0550$	0	22.5	97.2
$x \leq 0.0725$	6	24.08	72.8
$x \leq 0.0775$	7	26.09	57.4
$x \leq 0.0825$	9.2	26.9	59.0
$x \leq 0.0950$	10.83	27.78	59.8
$x \leq 0.1075$	11.57	29.25	50.9
$x \leq 0.1125$	12.75	30.43	50.9
$x \leq 0.1225$	**13.67**	**32**	**48.0**
$x \leq 0.1350$	14.8	33.4	51.8
$x \leq 0.1450$	15.82	35.25	54.8
$x \leq 0.1550$	16.83	37.67	59.2
$x \leq 0.1625$	18.15	39.5	76.0
$x \leq 0.1675$	19.64	40	102.9

CART after one split



Recursing…


Ratio < 0.1225
Ratio < 0.0775
Ratio < 0.155
7.00
19.00
26.33
37.67
age
0
10
20
30
40
0.04
0.06
0.08
0.10
0.12
0.14
0.16
ratio

Classification trees

Refractive index	Na %	Window glass
1.51590	13.24	1
1.51613	13.88	0
1.51673	13.30	1
1.51786	12.73	1
1.51811	12.96	1
1.51829	14.46	0
1.52058	12.85	0
1.52152	13.12	1
1.52171	11.56	0
1.52369	13.44	0

A good split

Sorted

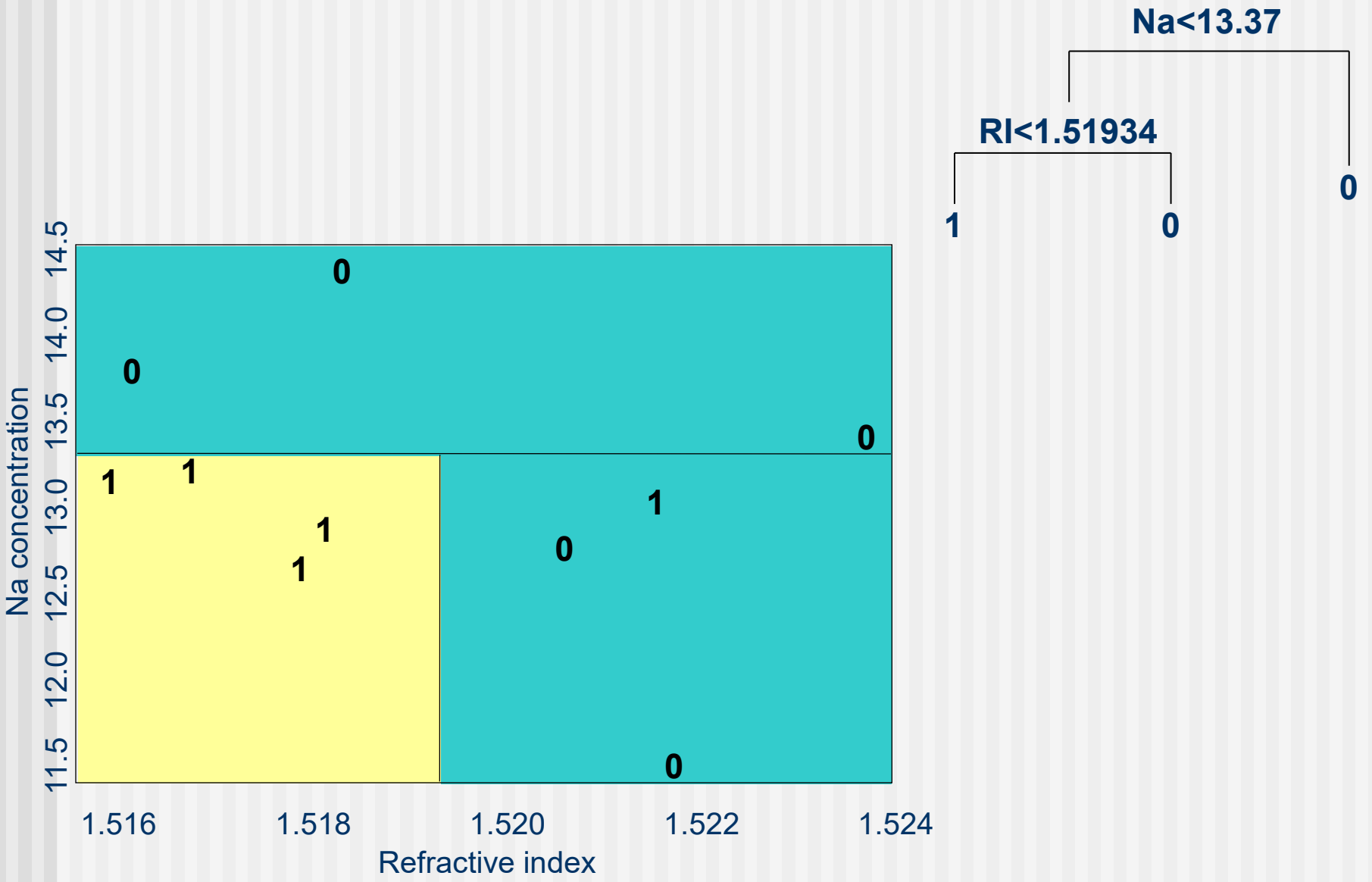
Classification trees

Refractive index	Na %	Window glass
1.52171	11.56	0
1.51786	12.73	1
1.52058	12.85	0
1.51811	12.96	1
1.52152	13.12	1
1.51590	13.24	1
1.51673	13.30	1
1.52369	13.44	0
1.51613	13.88	0
1.51829	14.46	0

A good split

Sorted

Classifying window glass



Prediction Priorities

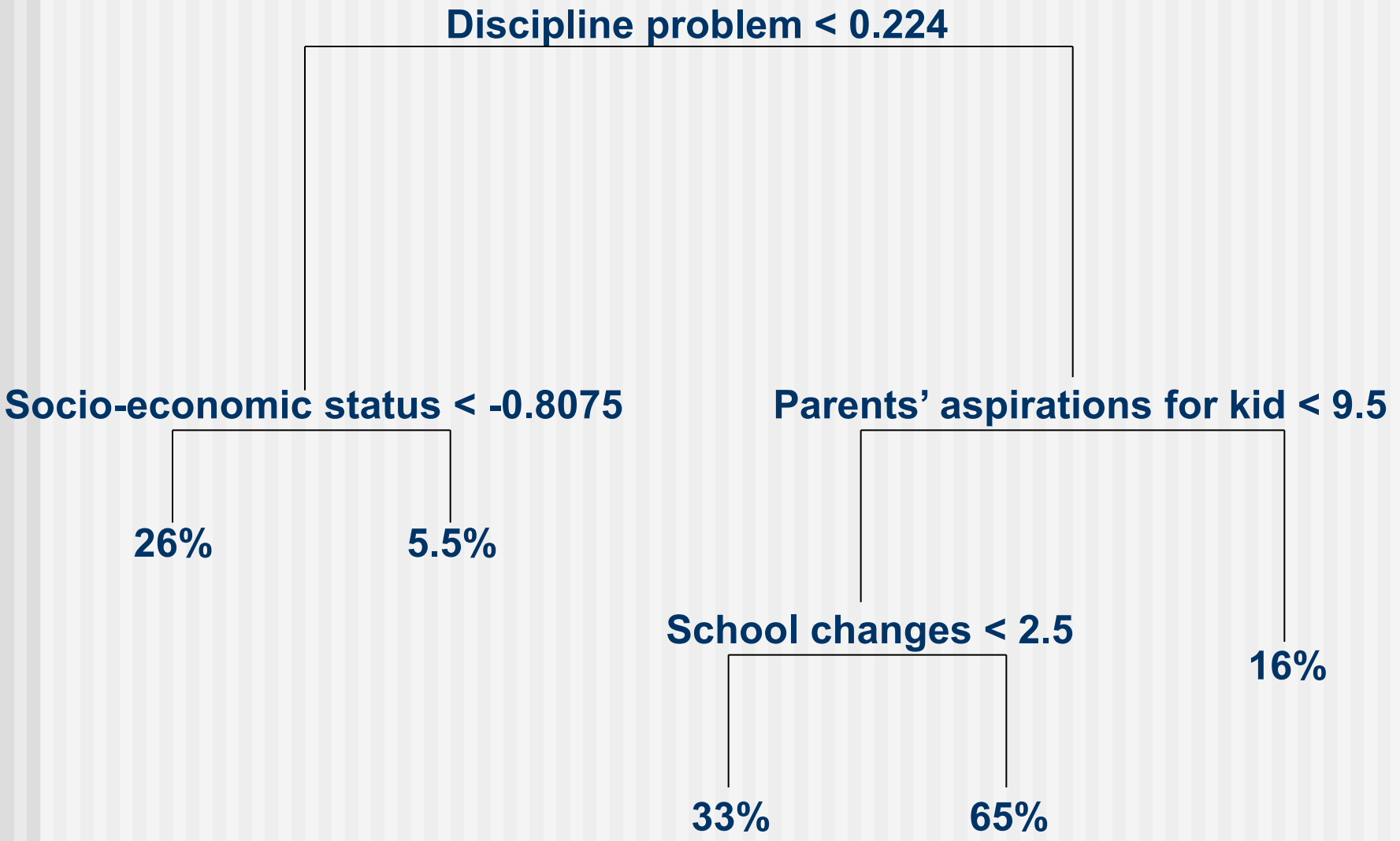
1. Accuracy – Obtain the model that predicts the best on future observations.
2. Efficiency – Find algorithms that learn efficiently from massive datasets.
3. Interpretability – Try to understand why the best model reasons as it does.

Trees and data mining

- Accuracy
  - Trees fit non-linear models with interactions
  - Other methods are more stable and more accurate.
- Efficiency
  - The partitioning strategy reduces the number of observations that we need in memory.
  - Trees handle continuous, nominal, ordinal, and missing input variables
  - Predicting for future observations is efficient.
- Interpretability
  - Trees appear interpretable… very deceiving.

NELS 1988

predict high school drop-out



Estimate generalization error

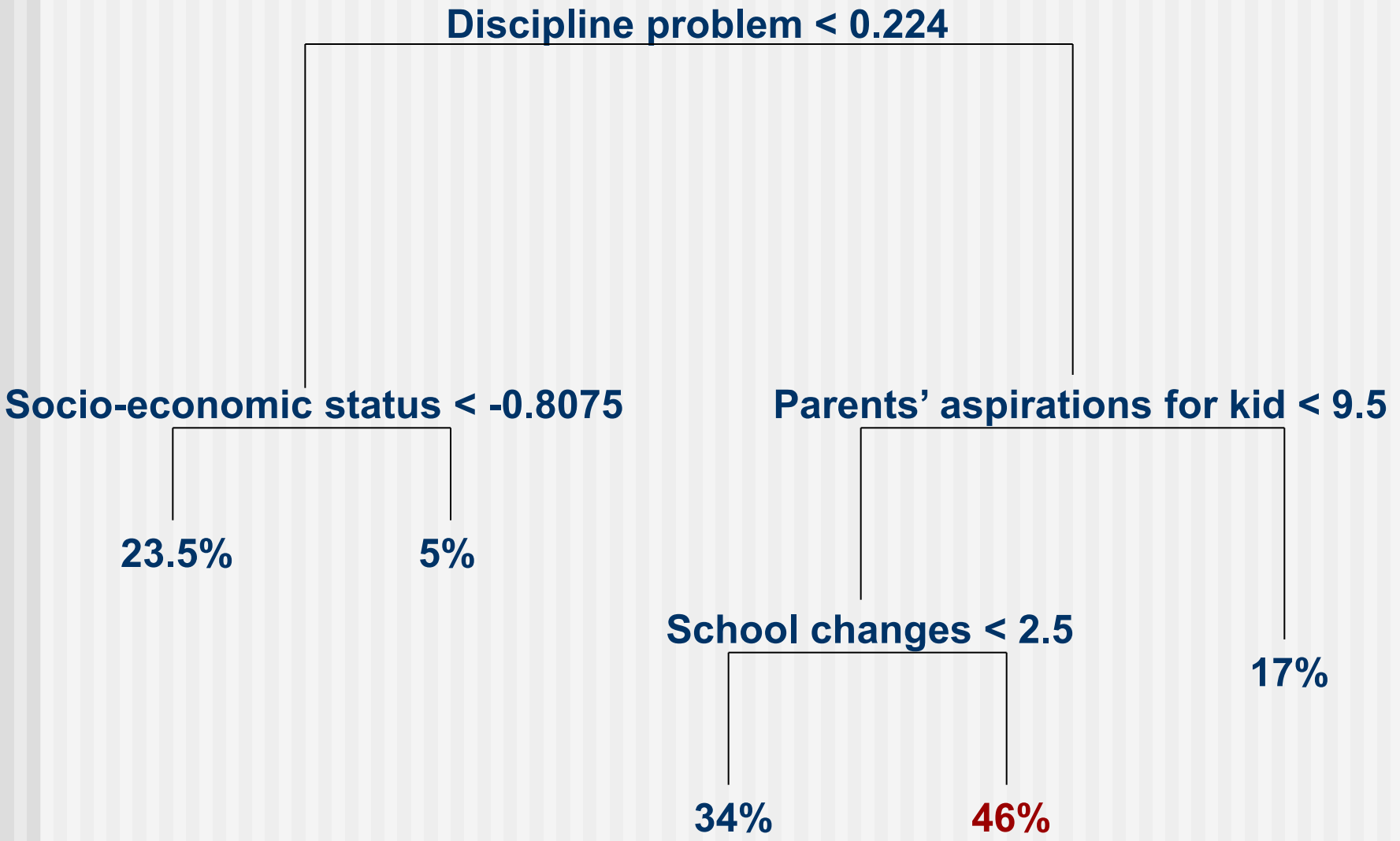
- Randomly split the dataset in half.
- Use half as a training set.
- Use the other half to assess the predictive performance of the method.
  - Gives an unbiased estimate of generalization error.
  - Try multiple splits of the dataset to understand the variability of the estimate of generalization error.
  - Also provides unbiased estimates of the node probabilities.

Misclassification

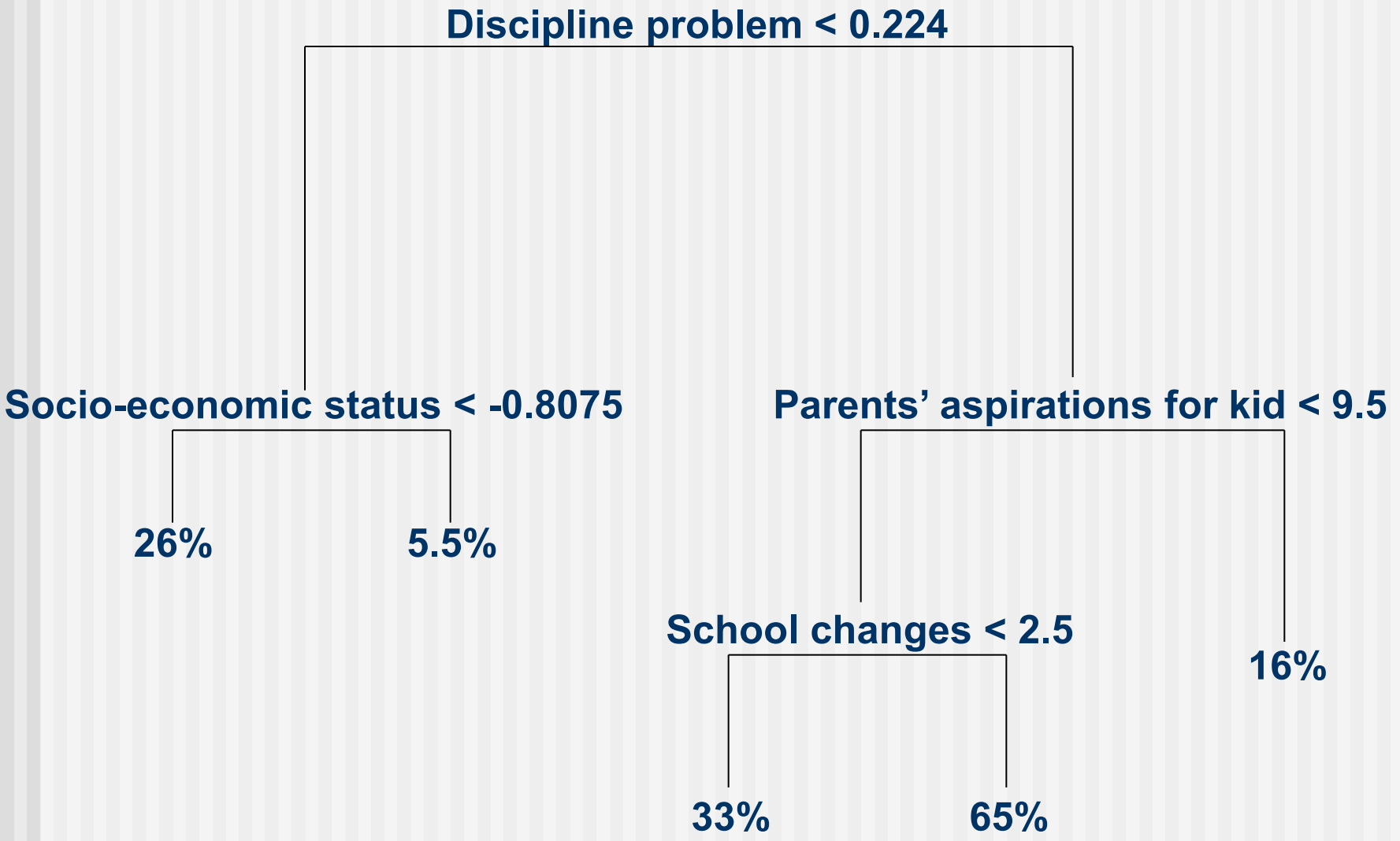
- Classify all students as graduates
  - Misclassification rate = 16.5%
- Using CART
  - Misclassification rate = 15.9%

- Cost of misclassification is almost always an important factor in determining predictive performance.

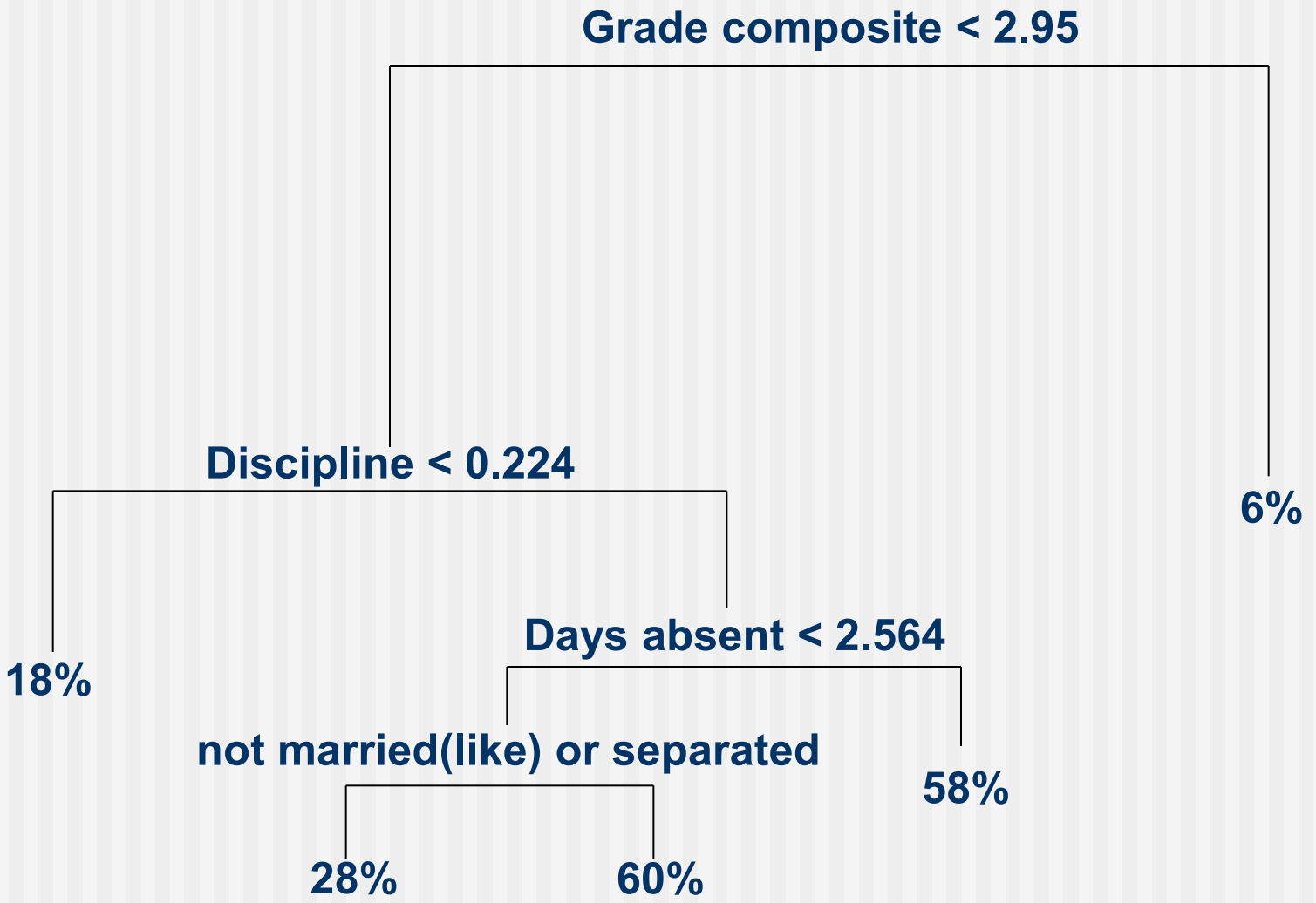
Re-estimate probabilities



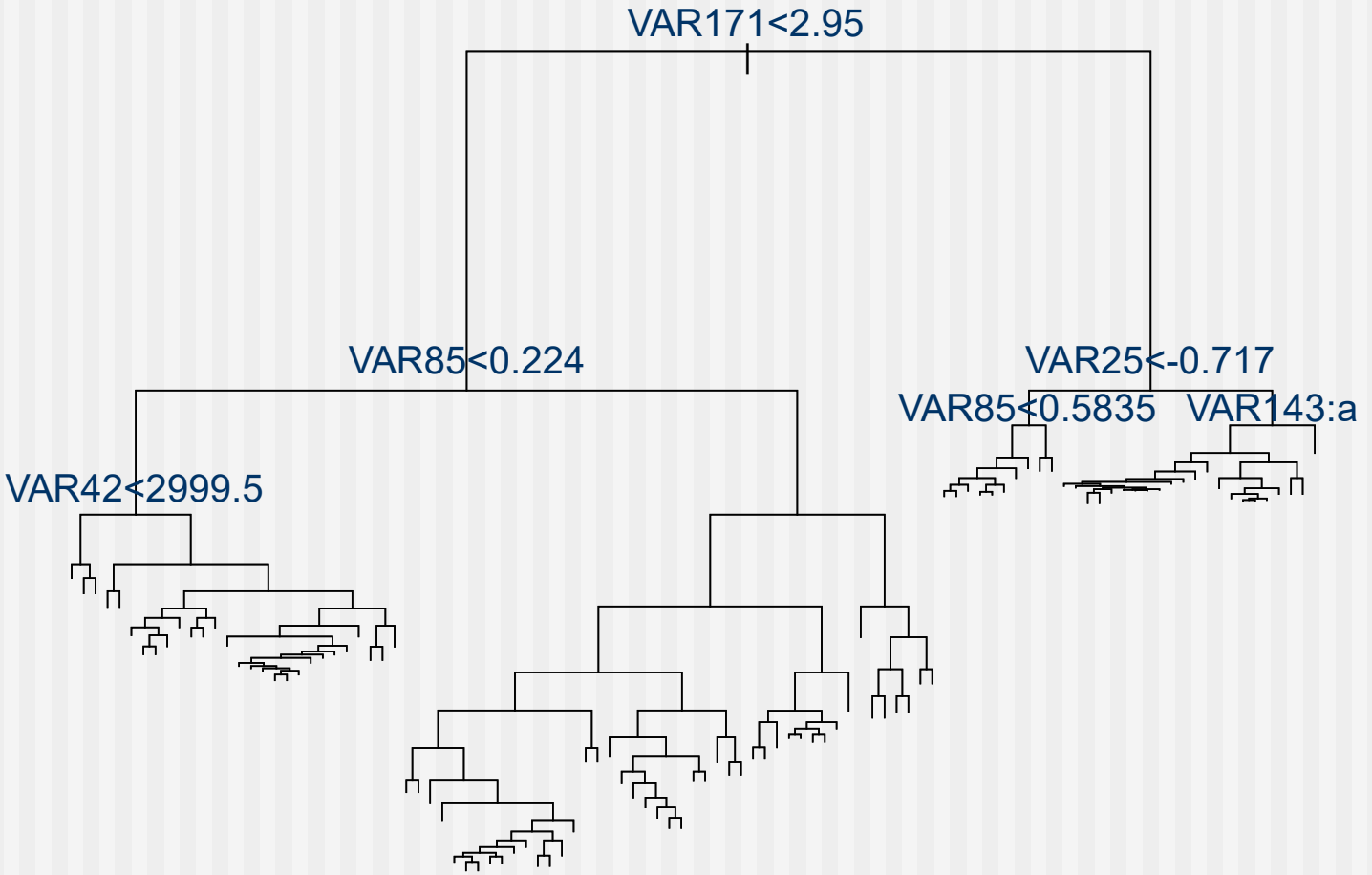
Trained on one half



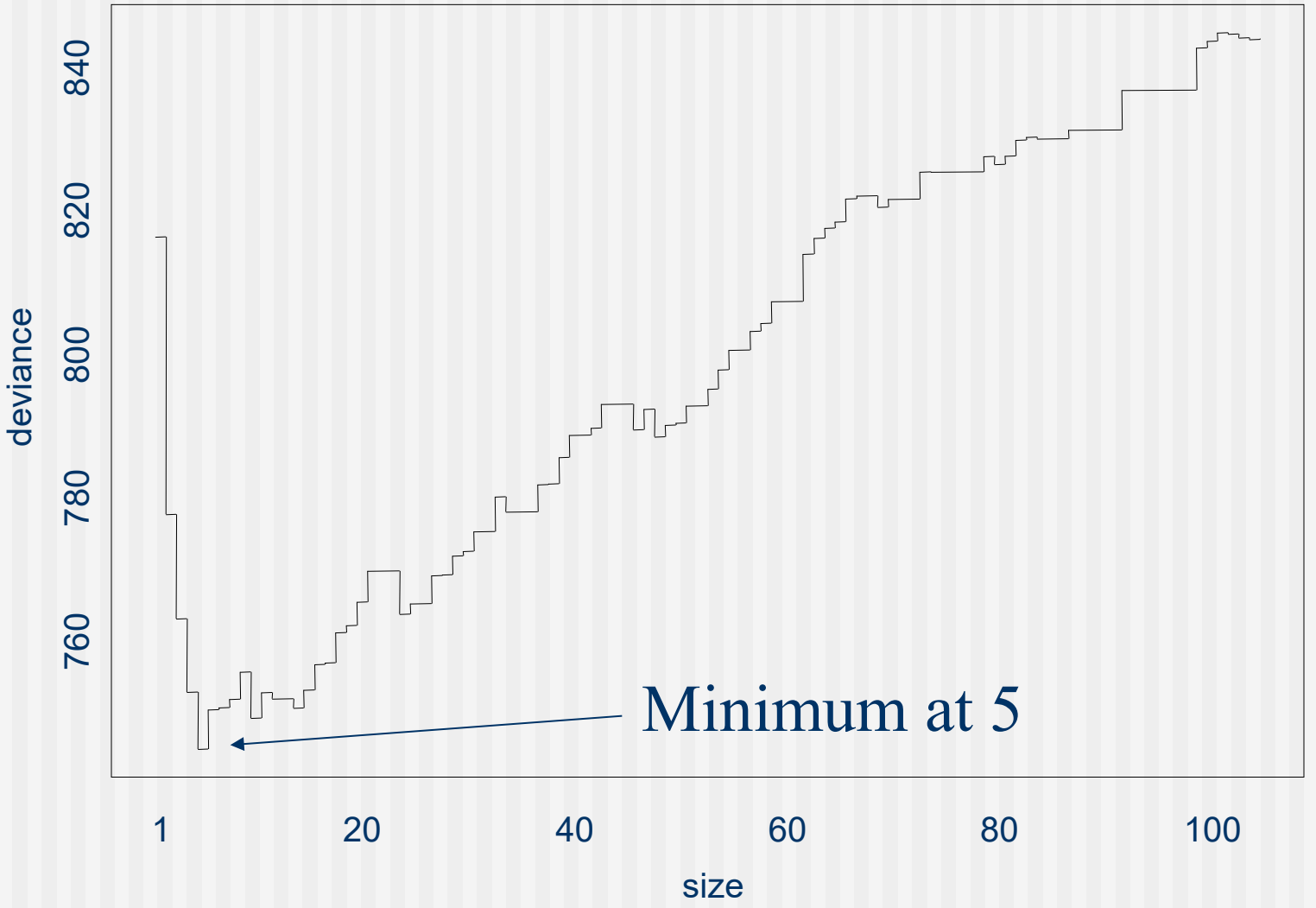
Trained on the other half



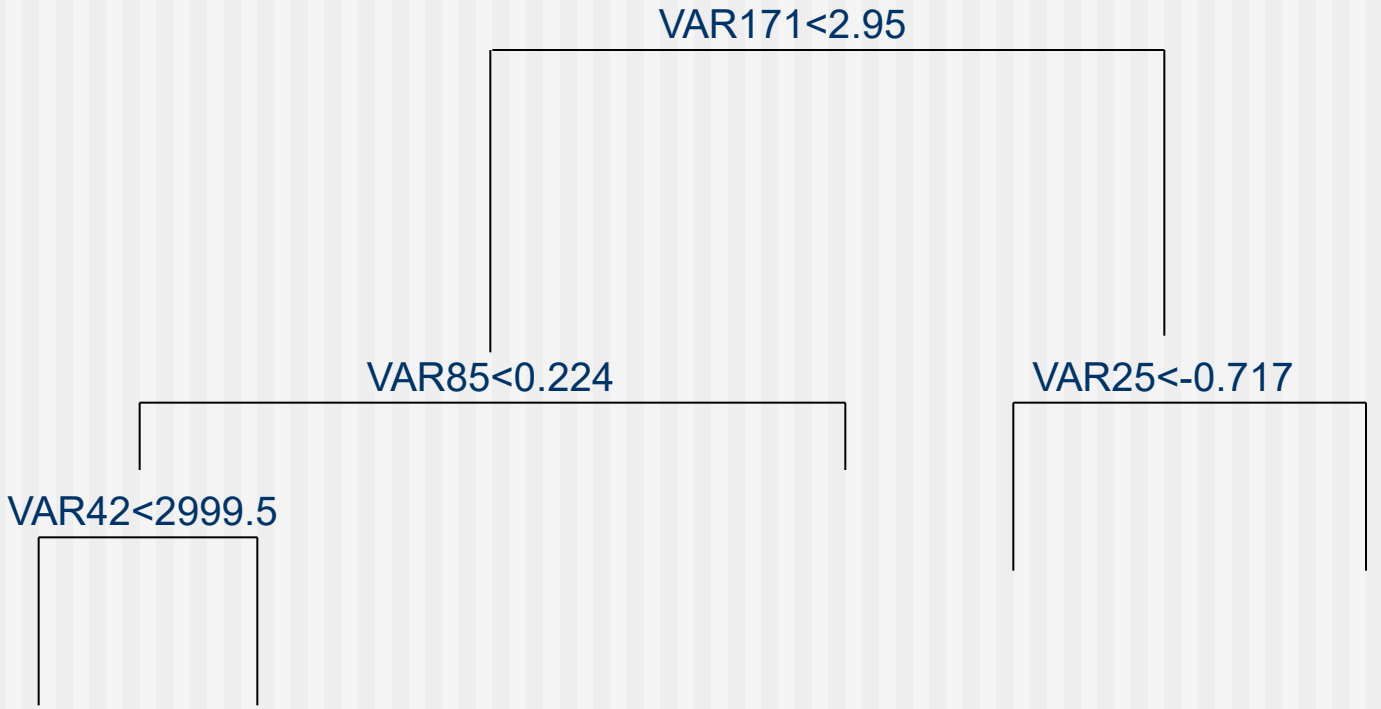
The out-of-control tree



Tree size vs. Generalization error



Optimal tree



Prediction and massive datasets

- Prospective prediction is the primary goal.
- Mined datasets are almost always retrospective.
  - Interpretation has to be a secondary goal.
- Policy is often based on the prediction alone.
  - Example: Children from broken homes are more likely to drop out. The intervention does not fix the home but acts on the predicted risk of dropping out.
- Designed experiments are outside the scope of data mining.

Precautions

- Situations that limit the predictor's future performance
  - Changes in the composition of the target population
  - Biased selection of the training dataset
  - Ignoring patterns of missing data

Innovations in prediction

- **Bagging** – Average multiple models to control variance.
  - **Pasting** – Averaging models constructed on small subsamples
- **Boosting** – Incrementally learn the predictor.
  - **Gradient boosting** – Generalizes boosting and incorporates

Bagging (*B*ootstrap *Agg*regating)

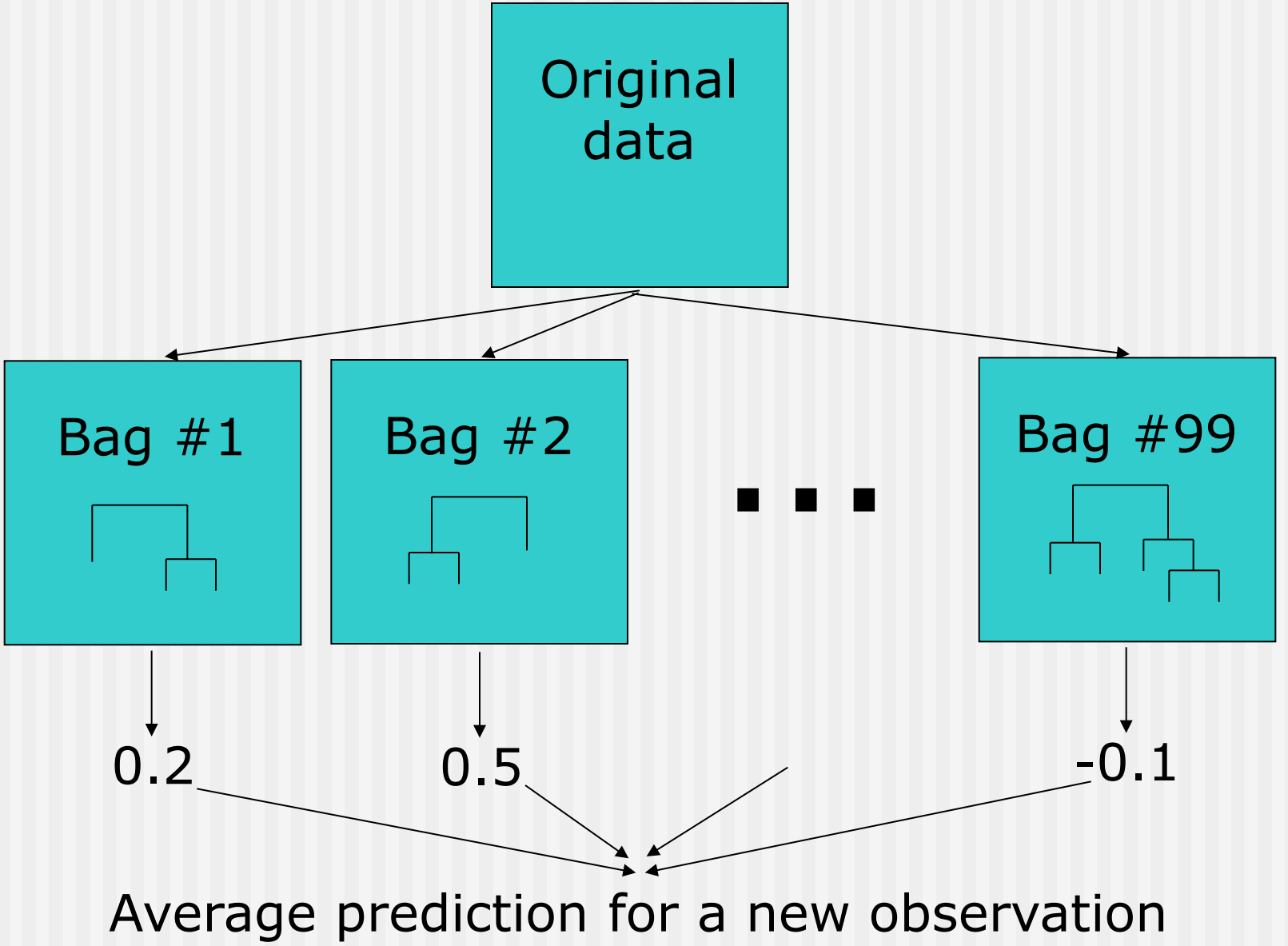
Goal: Variance reduction

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

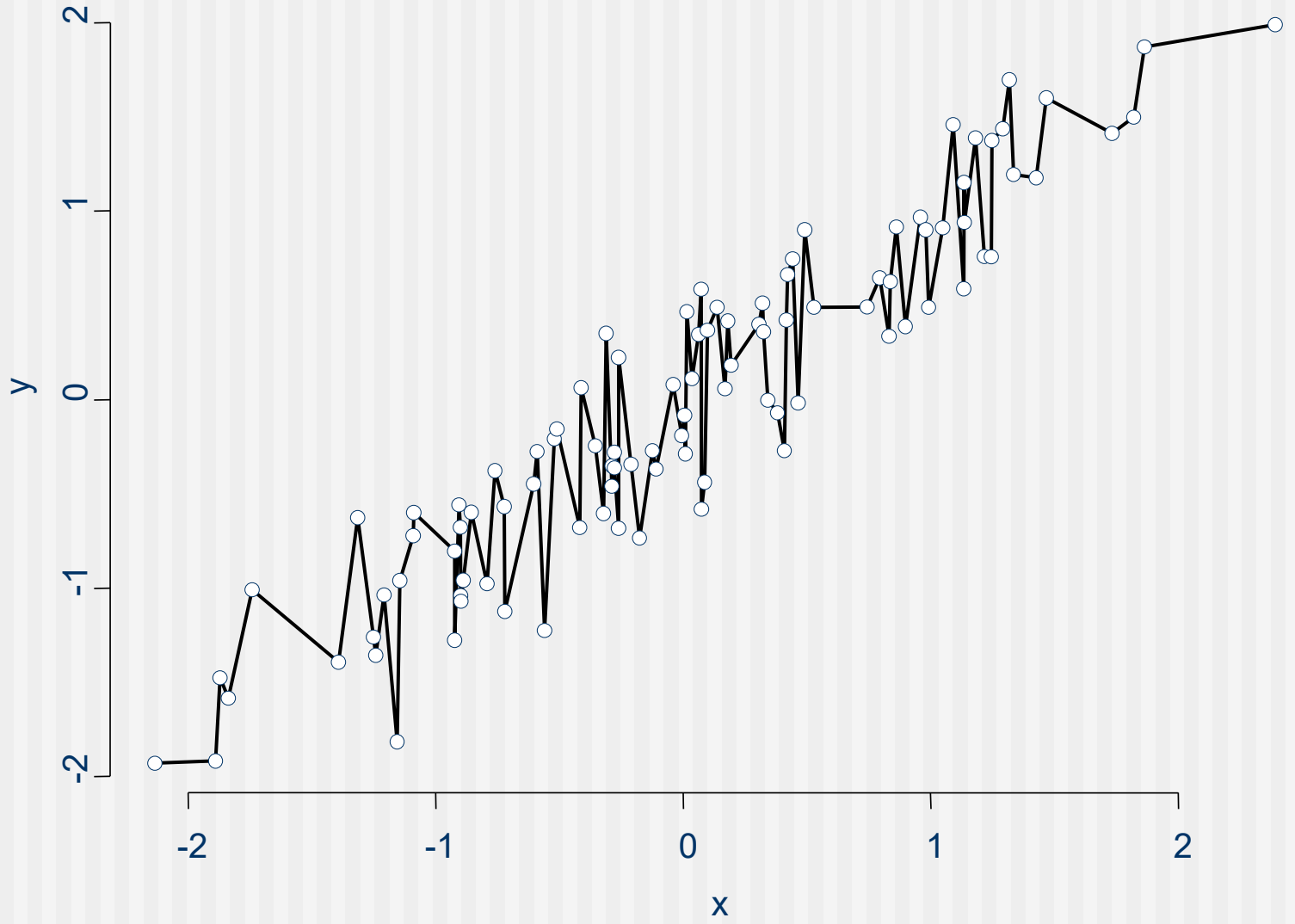

Original data
Bag #1
Bag #2
Bag #99
0.2
0.5
-0.1
Average prediction for a new observation

Bagging algorithm

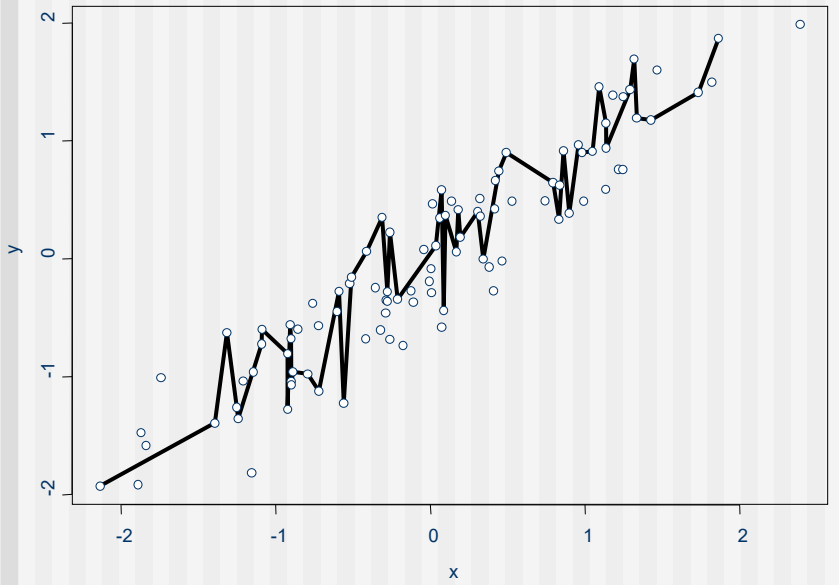
1. Create *K* bootstrap replicates of the dataset.
2. Fit a model to each of the replicates.
3. Average (or vote) the predictions of the *K* models.

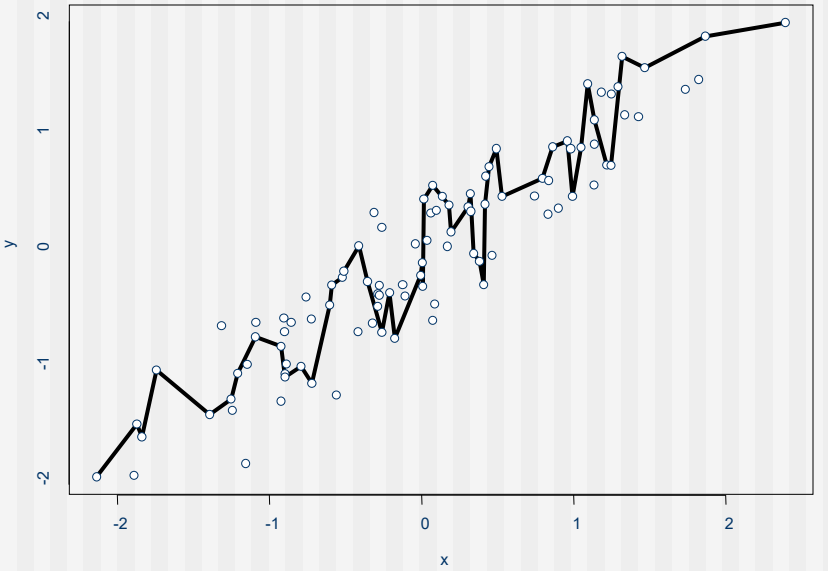
Bootstrapping simulates (approximately) an infinite stream of datasets.

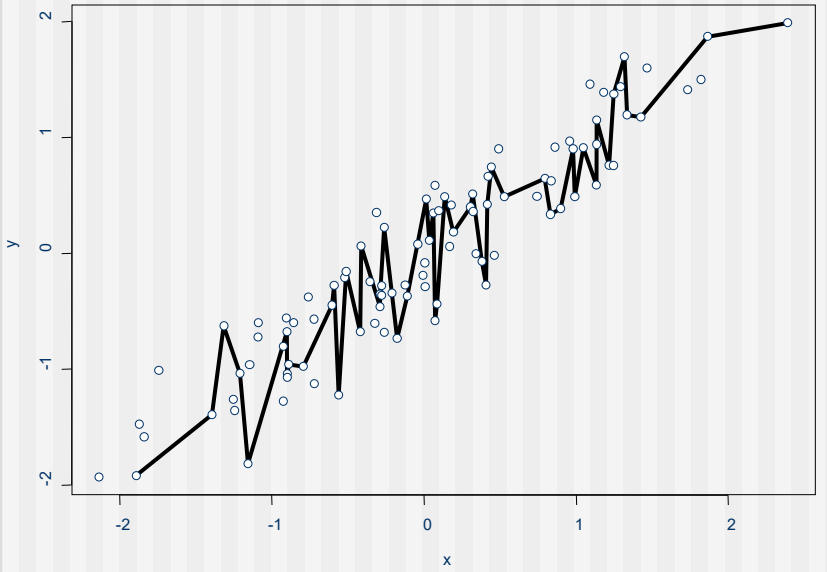
Connect-the-dots predictor

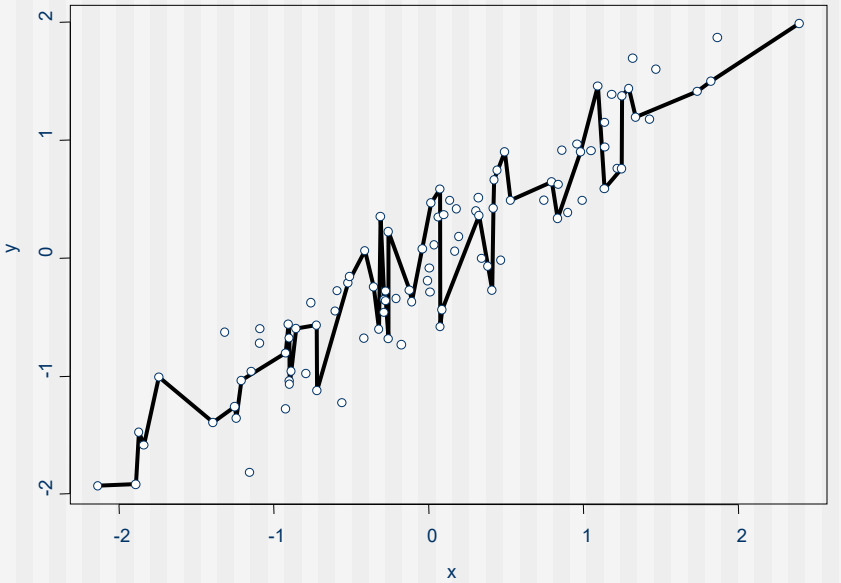


On half-samples…

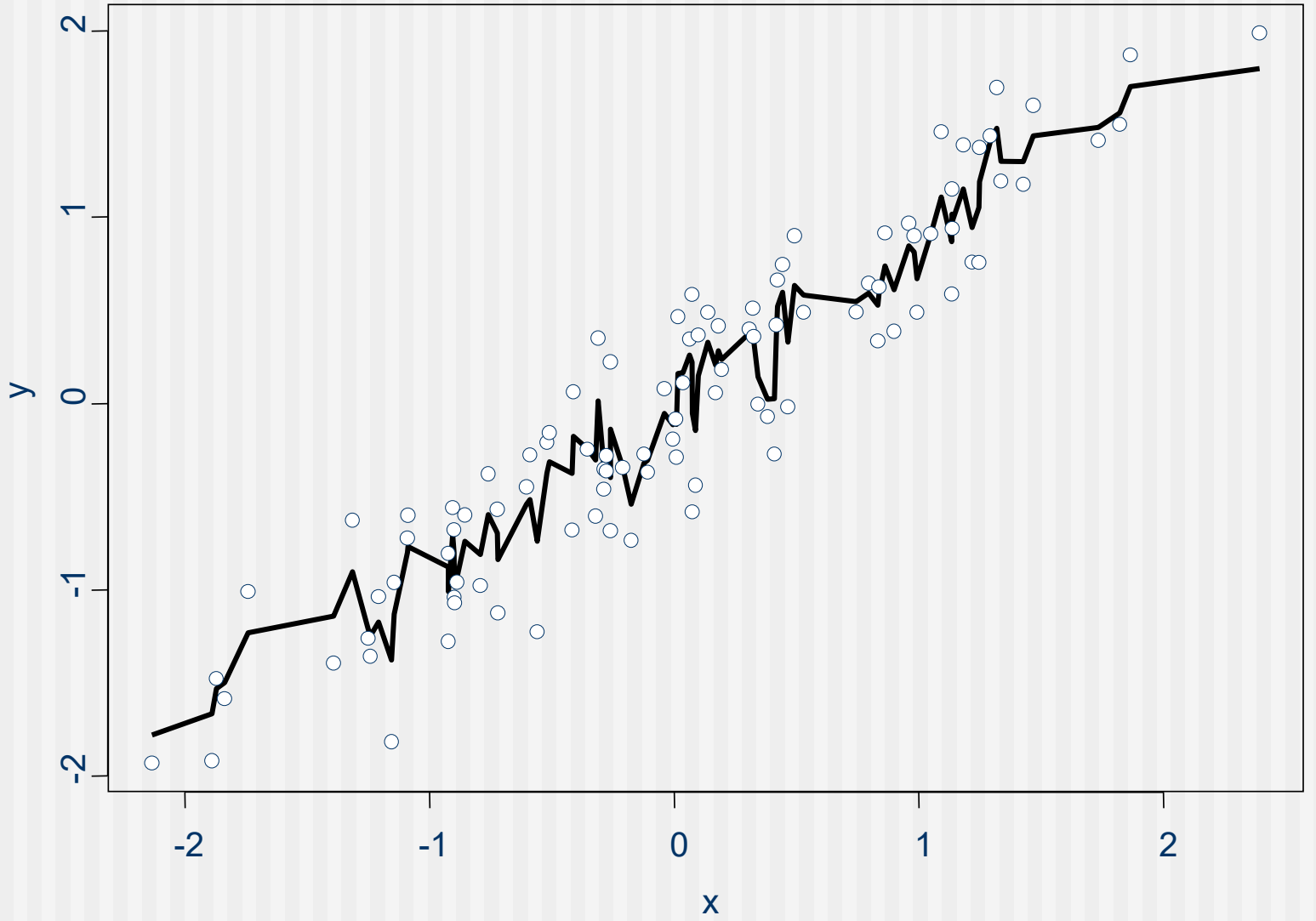




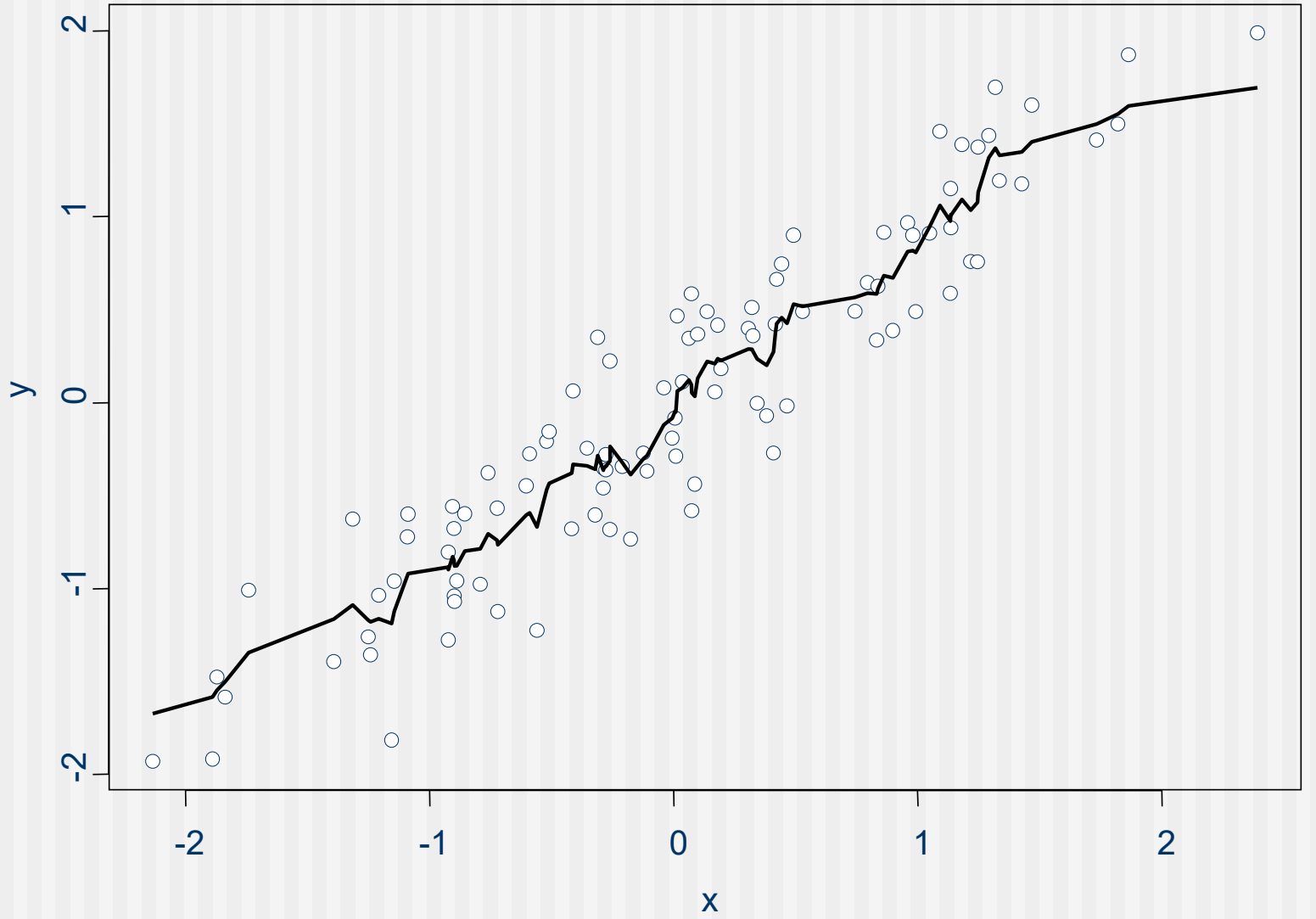




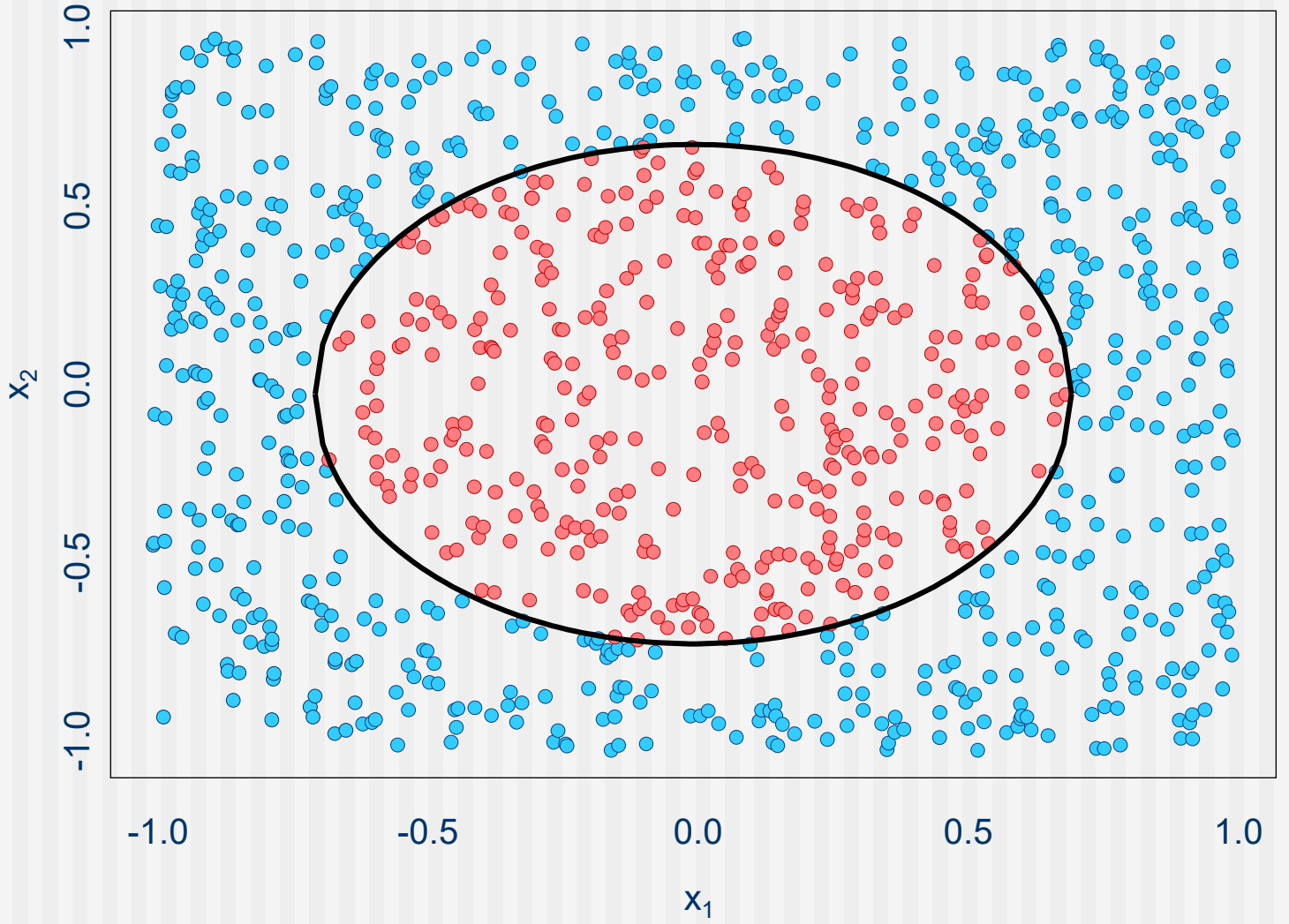
Average over half-samples



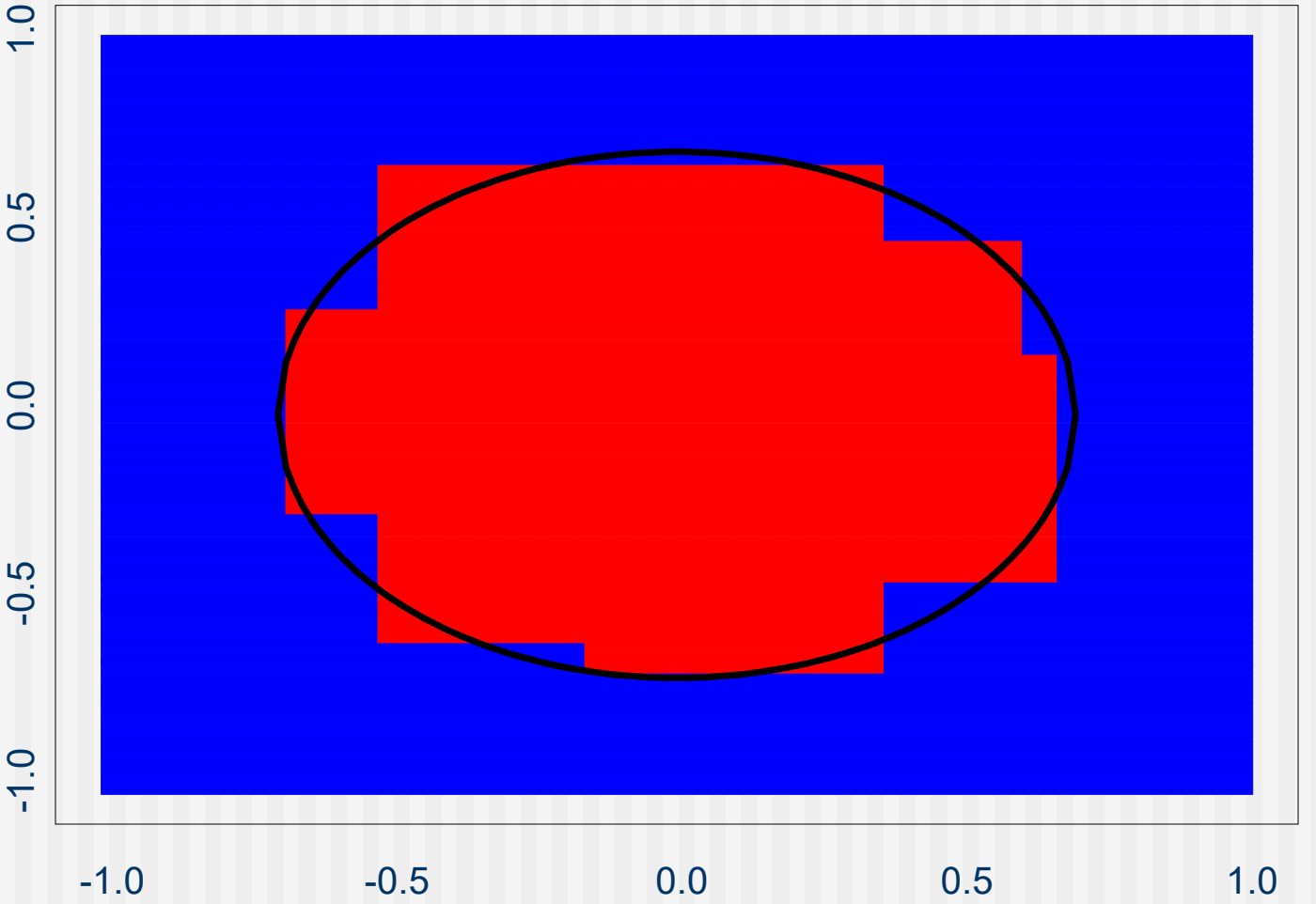
Average over quarter-samples



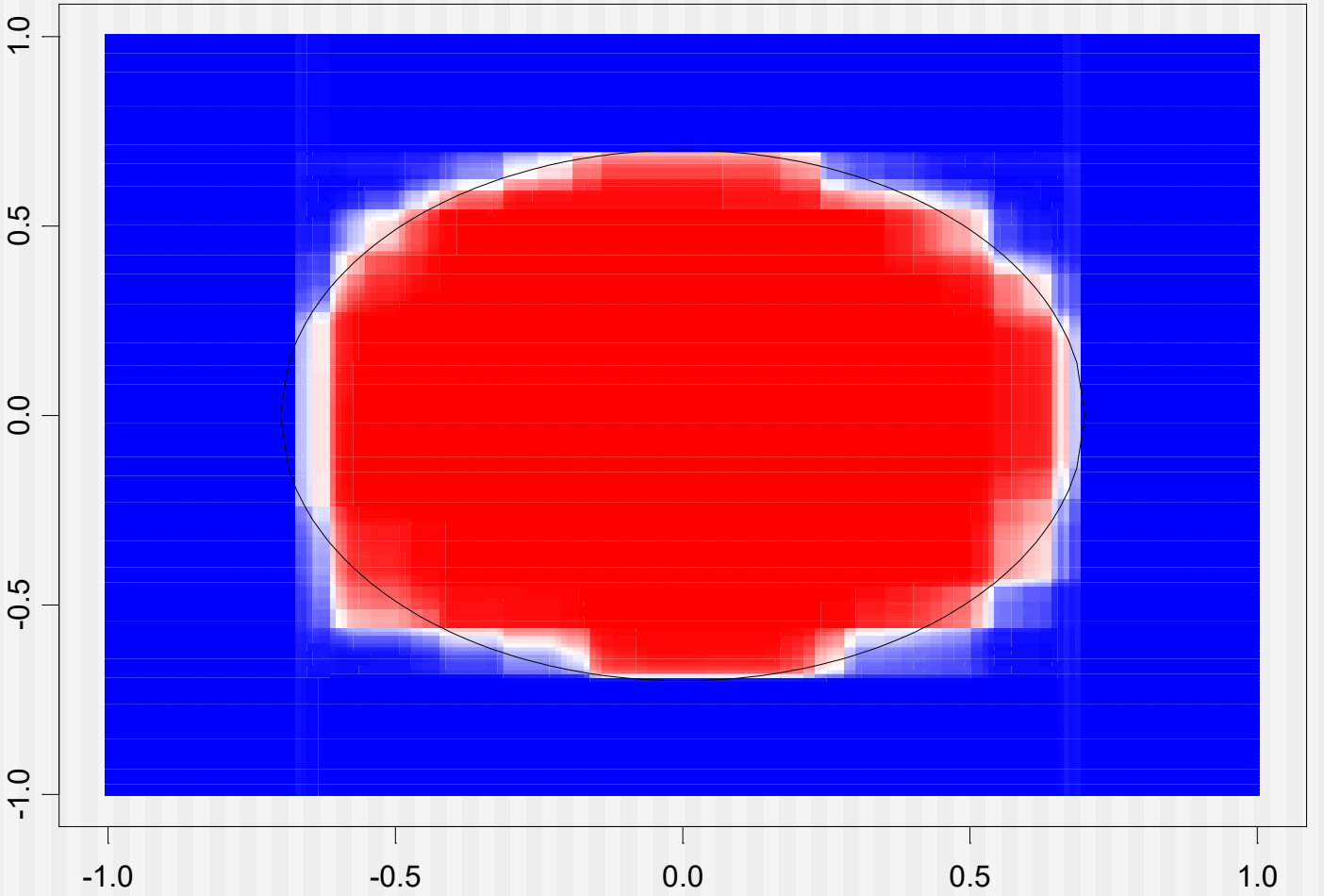
Bagging Example



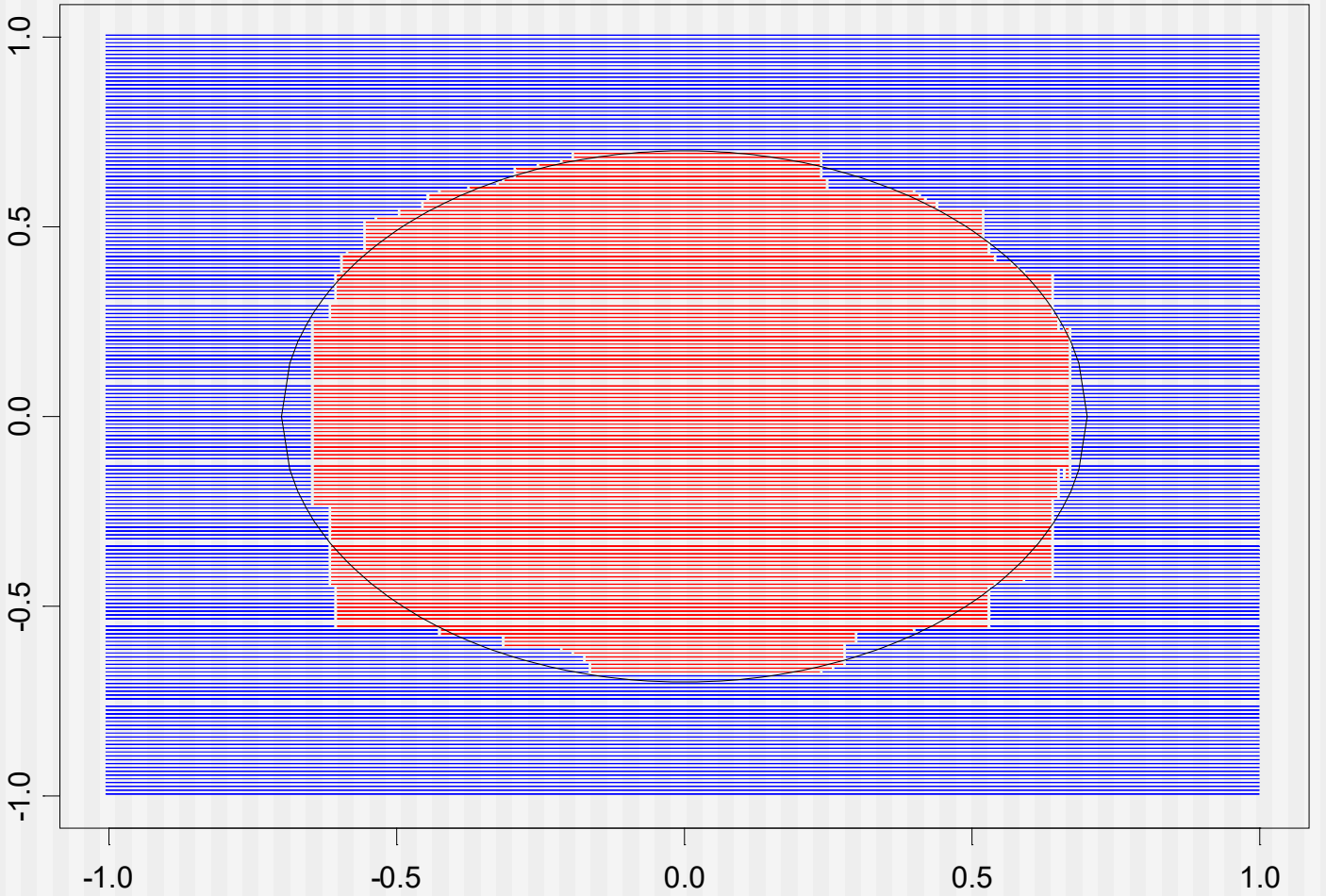
CART decision boundary



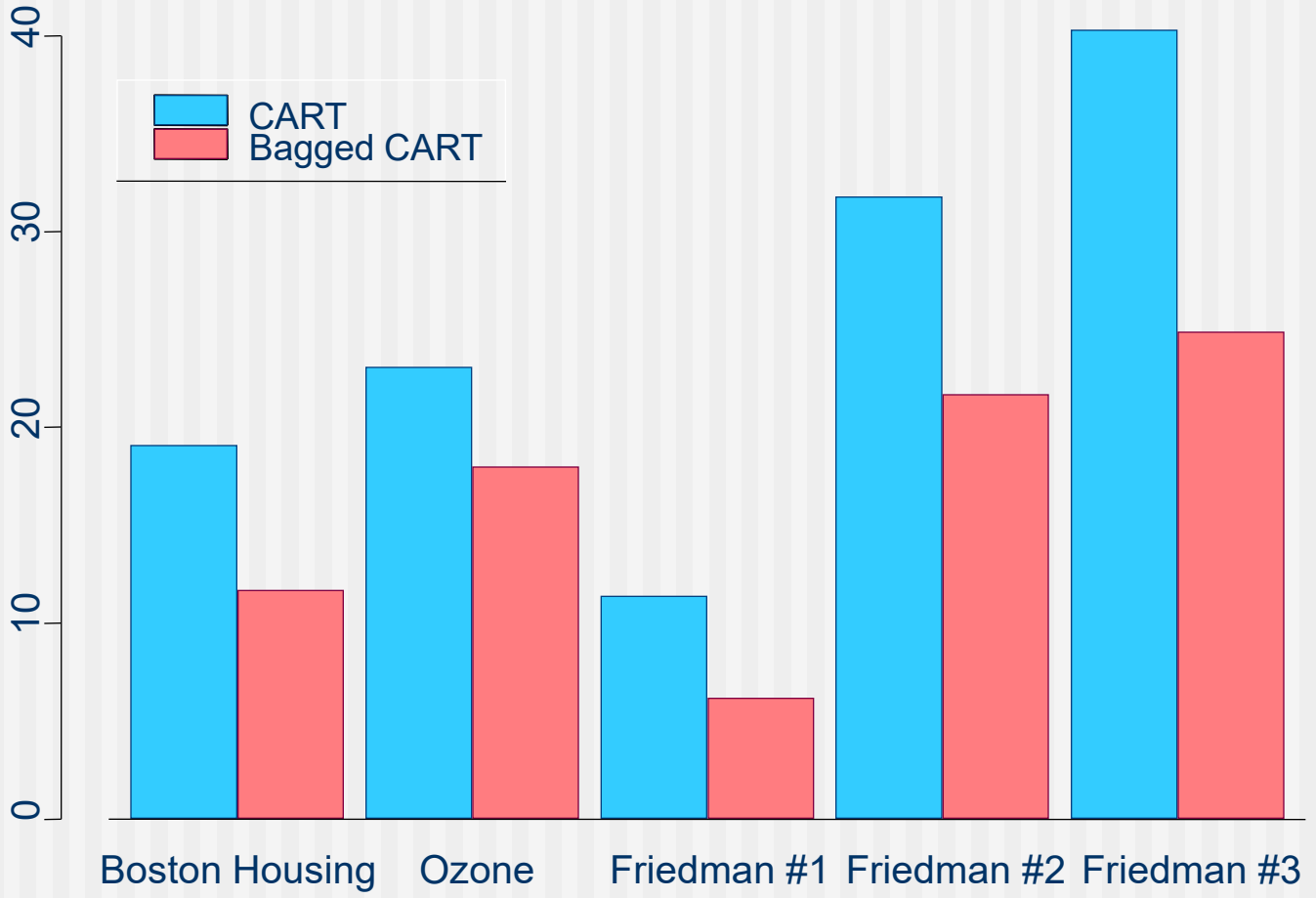
100 bagged trees



Bagged tree decision boundary

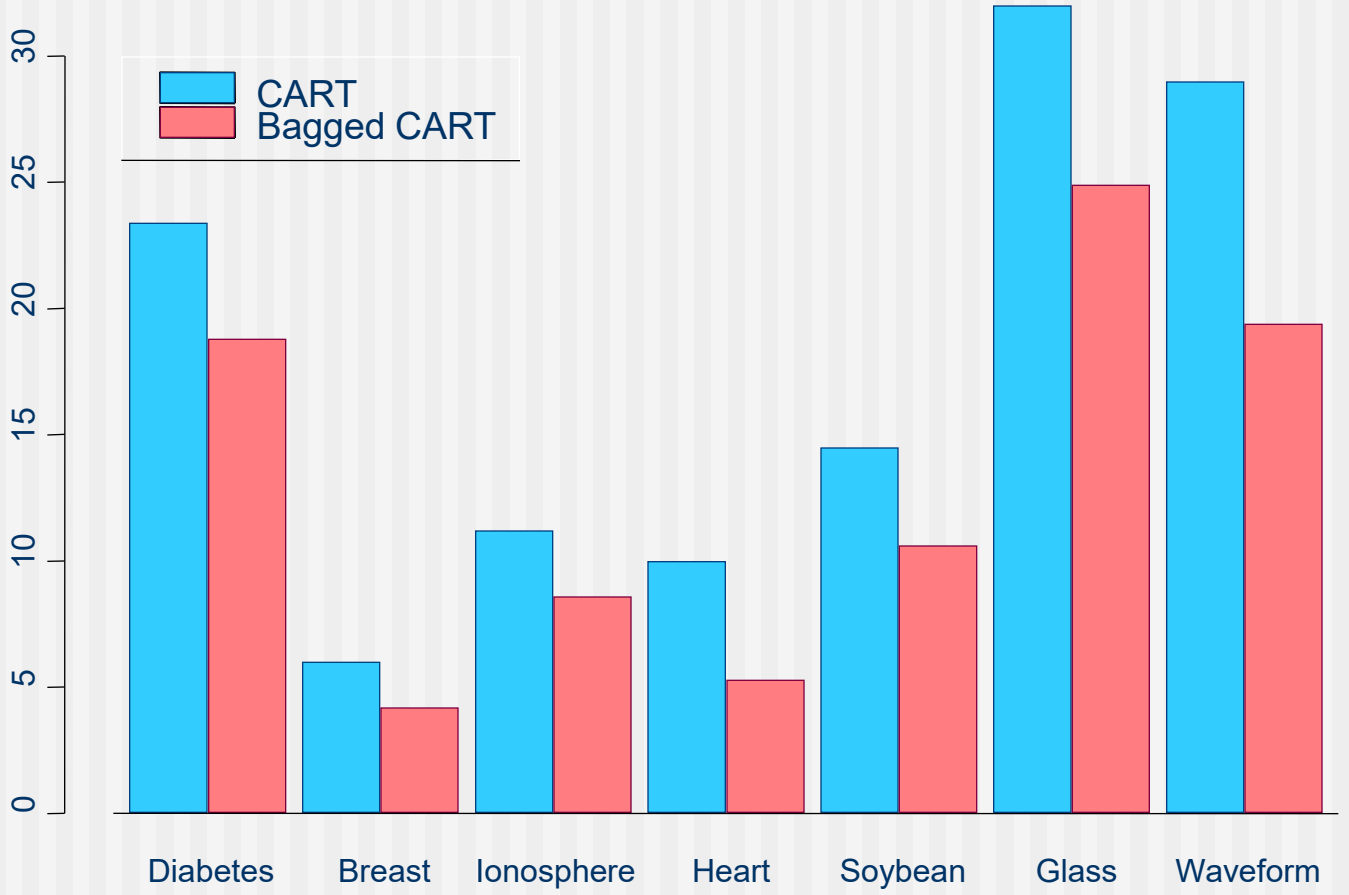


Regression results
Squared error loss



Classification results

Misclassification rates



Gradient Boosting
Automating the process

- Initialize the predictor to be the average output.

$$f(x) = \bar{y}$$

- Propose an additive improvement, $g(x)$, to $f(x)$ using the dataset.

$$f(x) \leftarrow f(x) + g(x)$$

- $g(x)$ may be a tree.

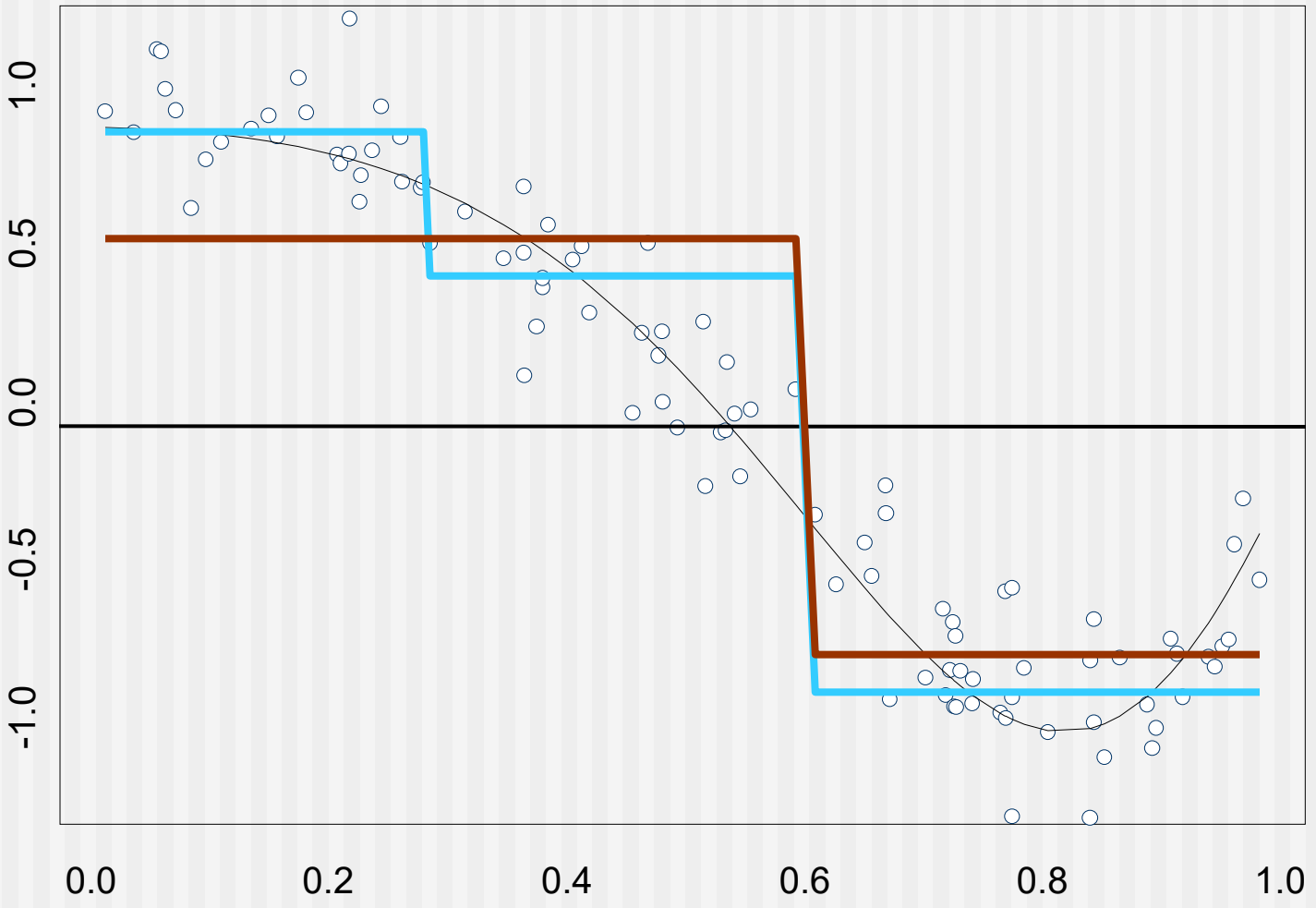
Proposing an improvement

$$\sum_{i \in \text{training}} (y_i - \bar{y})^2 \Rightarrow \sum_{i \in \text{training}} [y_i - (\bar{y} + g(x_i))]^2$$

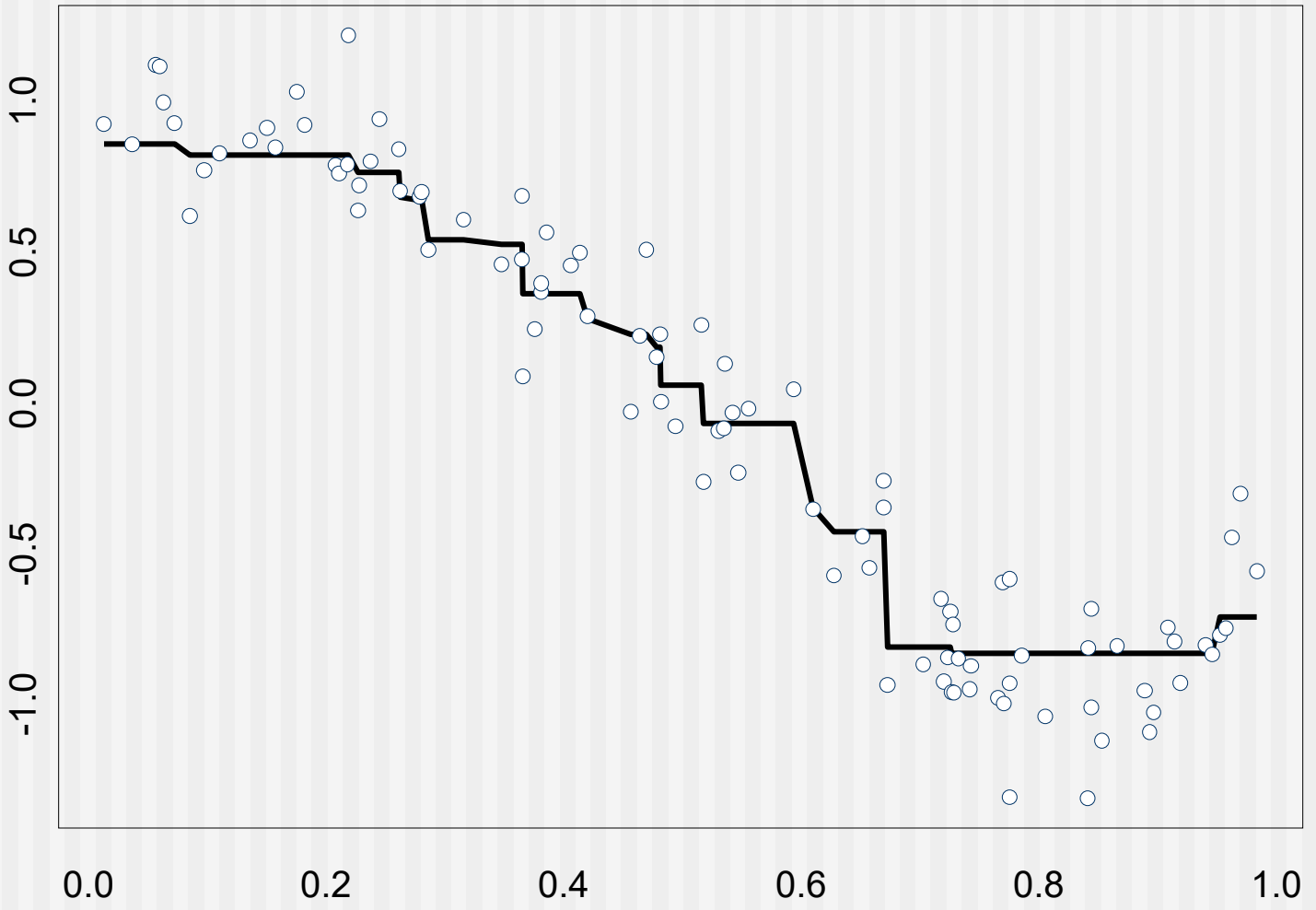
$$= \sum_{i \in \text{training}}^{n} [(y_i - \bar{y}) - g(x_i)]^2$$

- This shows that we should choose $g(x)$ that predicts the residuals using least squares.

Geometric view



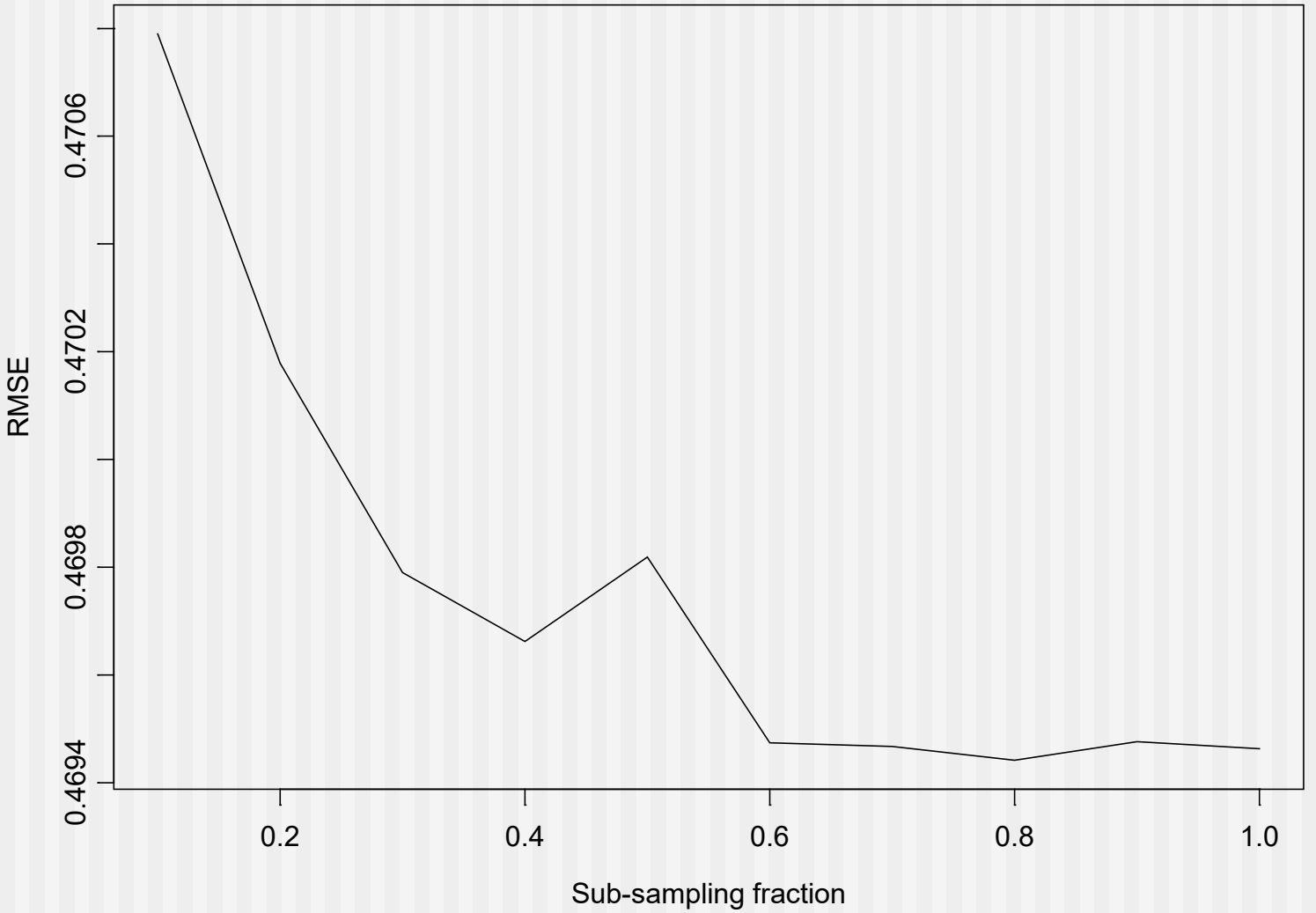
After several iterations



Further innovations

- Use a random subsample to make a proposal.
  - Requires fewer observations in memory
  - Actually improves performance!
- Make conservative moves
  - Use small trees
  - Shrink the predictions toward zero
- Use the “out-of-bag” observations to judge improvement

Effect of subsampling
Predicting cost of stroke care



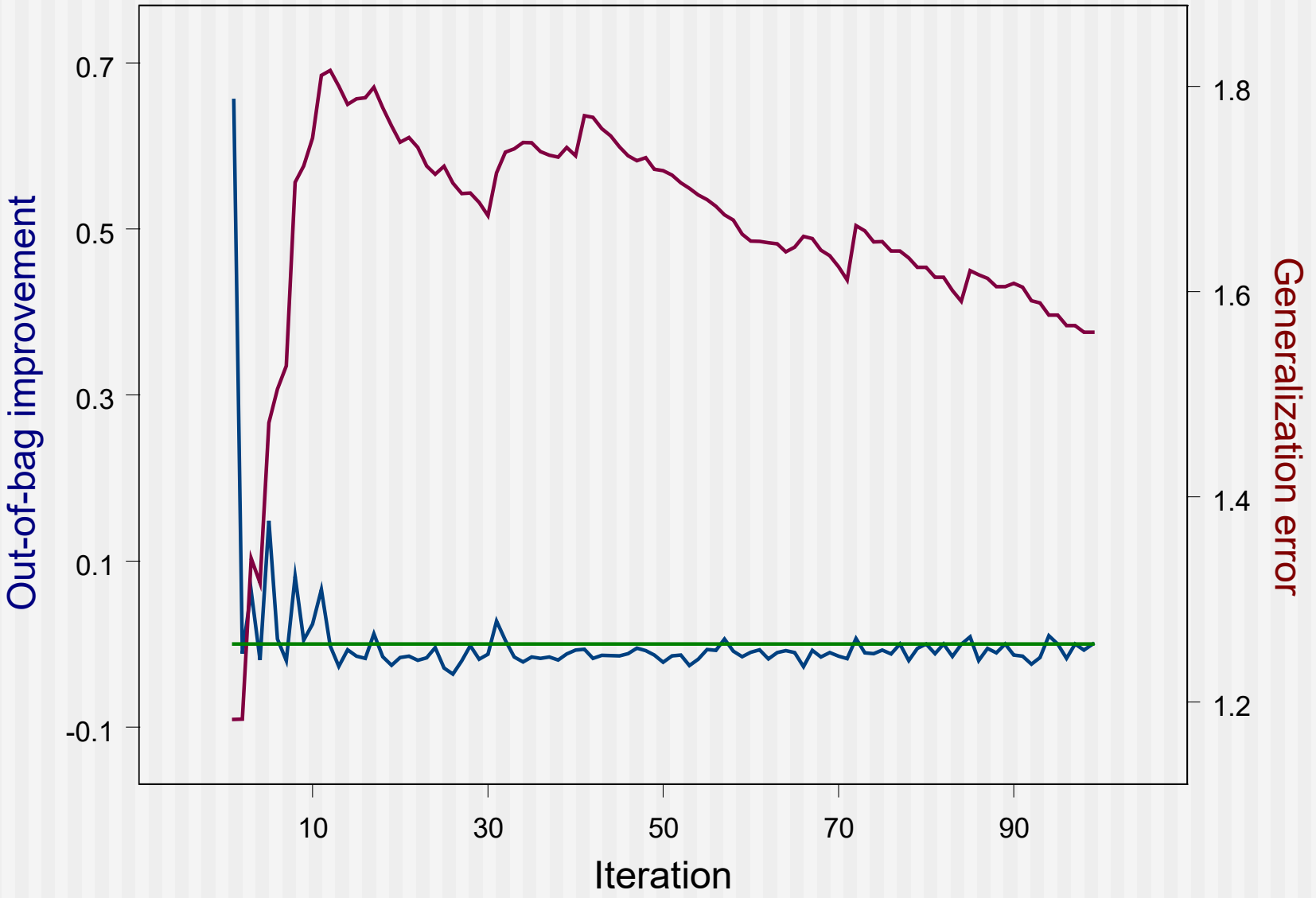
Accepting the proposal

- Let's use half of the dataset to suggest a modification of the predictor.
- We can use the second half to estimate, in a nearly unbiased fashion, if this proposal improves generalization error.

$$\text{Error accepting proposal} - \text{Error with out proposal}$$

$$= \sum_{i \in \text{validation}} \left[y_i - (f(x_i) + g(x_i)) \right]^2 - \left[y_i - f(x_i) \right]^2$$

Out-of-bag estimation



An algorithm

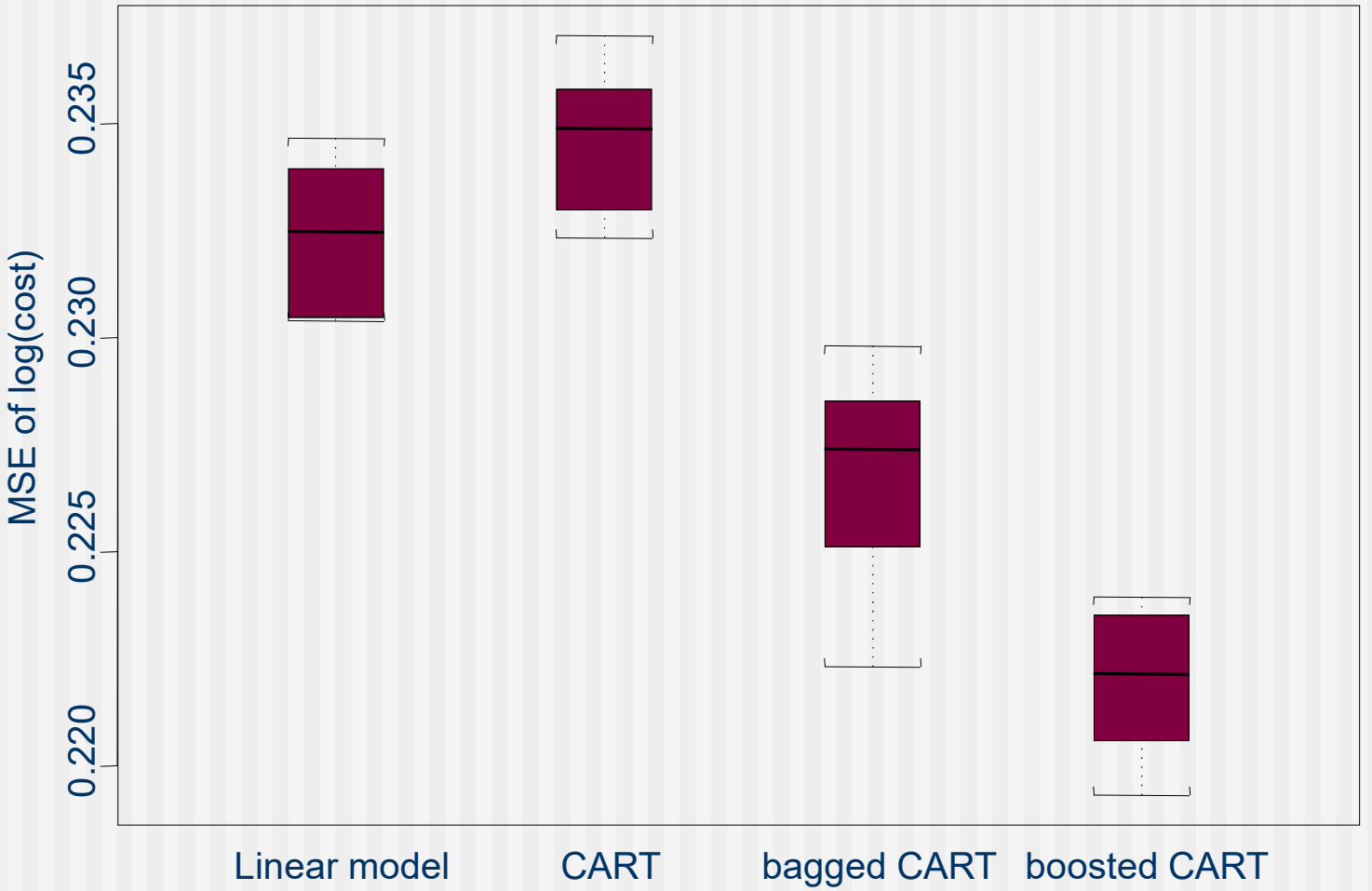
1. Set $f(x)$ to be the average y.
2. Iterate
   a. Set $z_i = y_i - f(x_i)$
   b. Fit a regression tree, $g(x_i)$, predicting z_i from the features x_i using only half of the dataset.
   c. Estimate the improvement $g(x)$ makes in generalization error using the other half of the dataset.

$$\sum_{i \in \text{validation}} \left[y_i - (f(x_i) + g(x_i))\right]^2 - \left[y_i - f(x_i)\right]^2$$

   d. If the improvement is positive then update $f(x)$ and return to (a).

$$f(x) \leftarrow f(x) + c \cdot g(x)$$

Predicting cost of treating stroke patients



Conclusions

- Building predictive models from massive datasets involves
  - model complexity
  - data access complexity
- A new generation of algorithms uses
  - recursive partitioning strategies as a base for efficiency
  - resampling methods
  - out-of-bag estimates to
    - control overfitting and
    - estimate variable effectiveness

References

- J.F. Elder and D. Pregibon (1996). "A Statistical Perspective on Knowledge Discovery in Databases," Chapter 4 in *Advances in Knowledge Discovery and Data Mining*. Available at http://www.datamininglab.com/resources.html

- I.H. Witten and E. Frank (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*.

- Friedman, J. H. (1999). "Greedy Function Approximation: A Gradient Boosting Machine." http://www-stat.stanford.edu/~jhf/