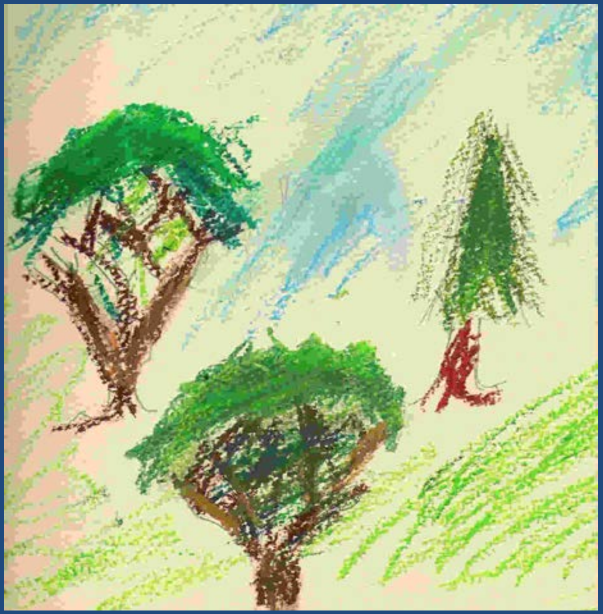
Trees and Random Forests



Adele Cutler
Professor, Mathematics and Statistics
Utah State University

This research is partially supported by NIH 1R15AG037392-01

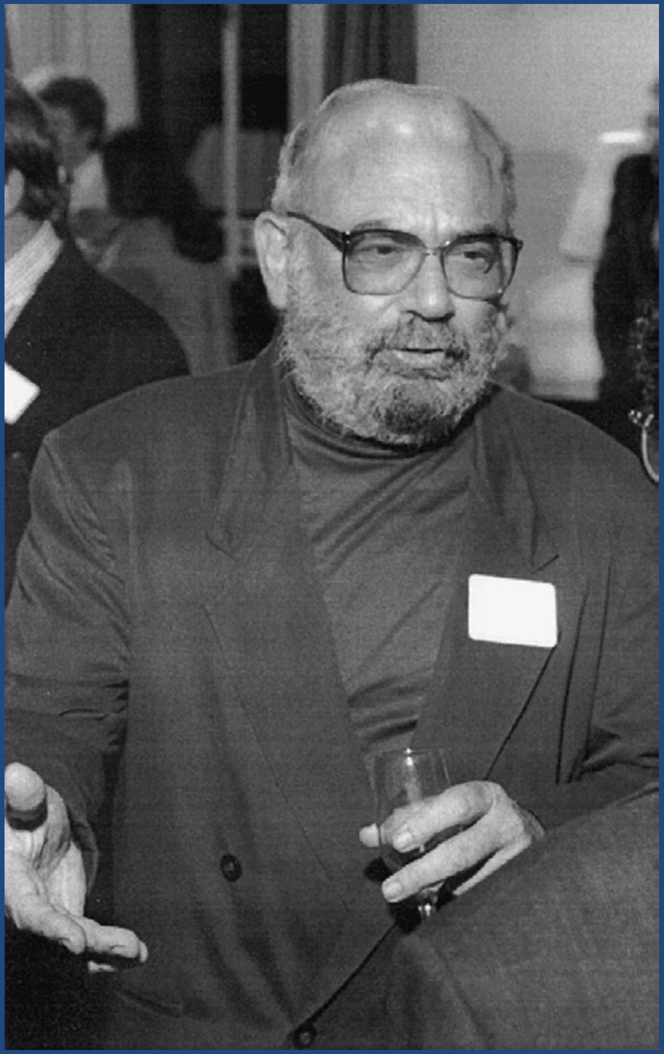
Cache Valley, Utah

October 3, 2013

Utah State University

October 3, 2013

Leo Breiman, 1928 - 2005



1954 PhD Berkeley (mathematics)

1960 -1967 UCLA (mathematics)

1969 -1982 Consultant

1982 - 1993 Berkeley (statistics)

1984 “Classification & Regression Trees”
(with Friedman, Olshen, Stone)

1996 “Bagging”

2001 “Random Forests”

Regression

Given predictor variables **x**, and a continuous response variable y, build a model for:

- Predicting the value of y for a new value of **x**
- Understanding the relationship between **x** and y

e.g. predict a person's systolic blood pressure based on their age, height, weight, etc.

Classification

Given predictor variables **x**, and a categorical response variable y, build a model for:

- Predicting the value of y for a new value of **x**
- Understanding the relationship between **x** and y

e.g. predict a person's 5-year-survival (yes/no) based on their age, height, weight, etc.

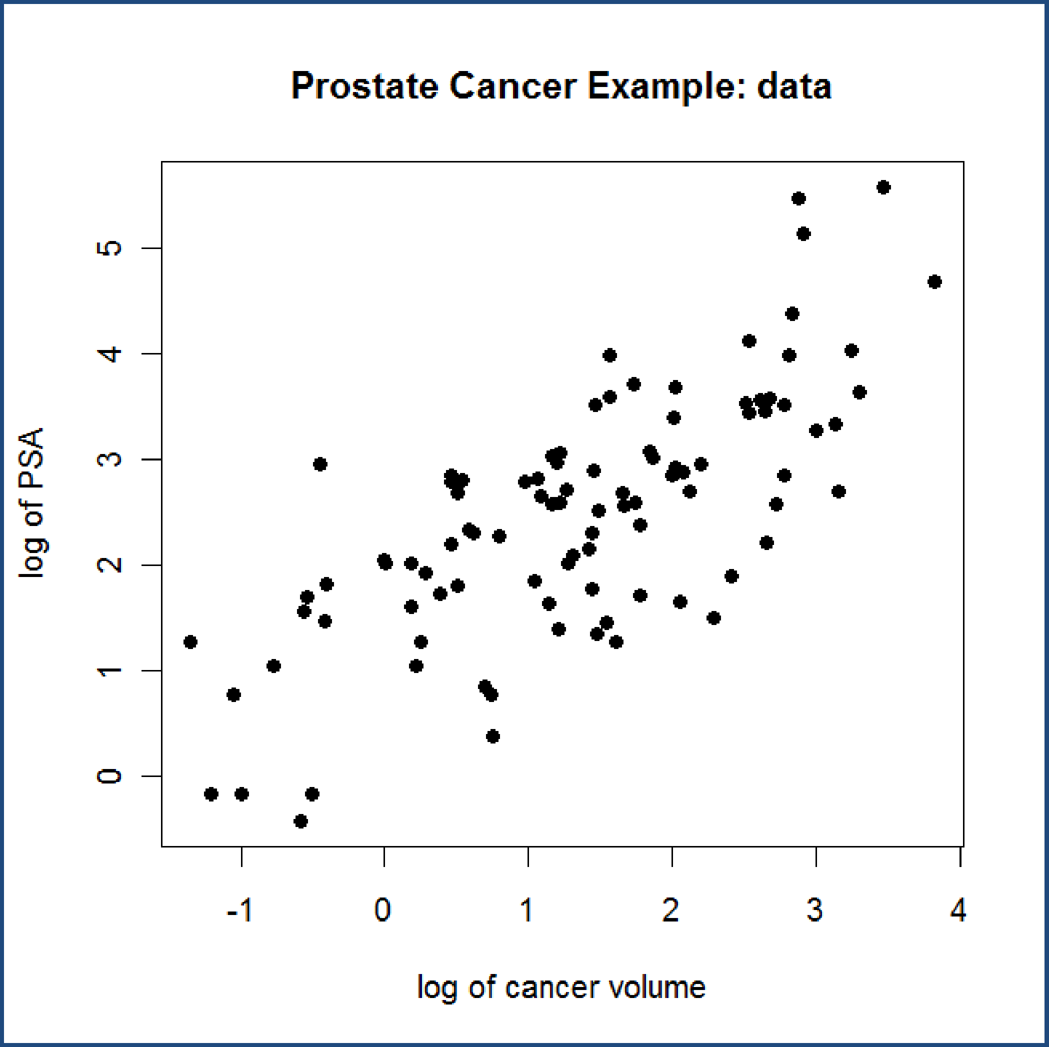
Regression Methods

- Simple linear regression
- Multiple linear regression
- Nonlinear regression (parametric)
- Nonparametric regression:
  - Kernel smoothing, spline methods, wavelets
  - Trees (1984)
- Machine learning methods:
  - Bagging
  - Random forests
  - Boosting

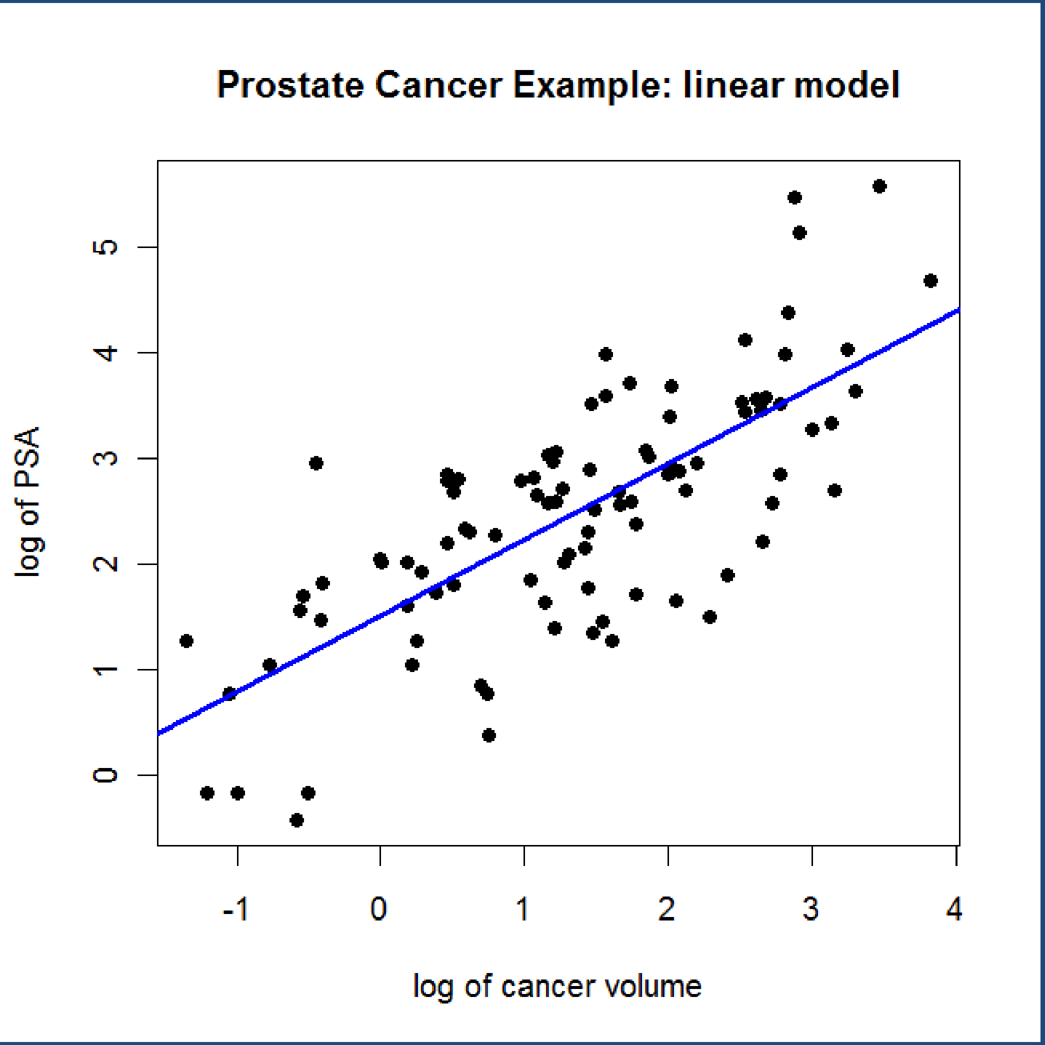
Classification Methods

- Linear discriminant analysis (1930's)
- Logistic regression (1944)
- Nonparametric methods:
  - Nearest neighbor classifiers (1951)
  - Trees (1984)
- Machine learning methods:
  - Bagging
  - Random forests
  - Support vector machines

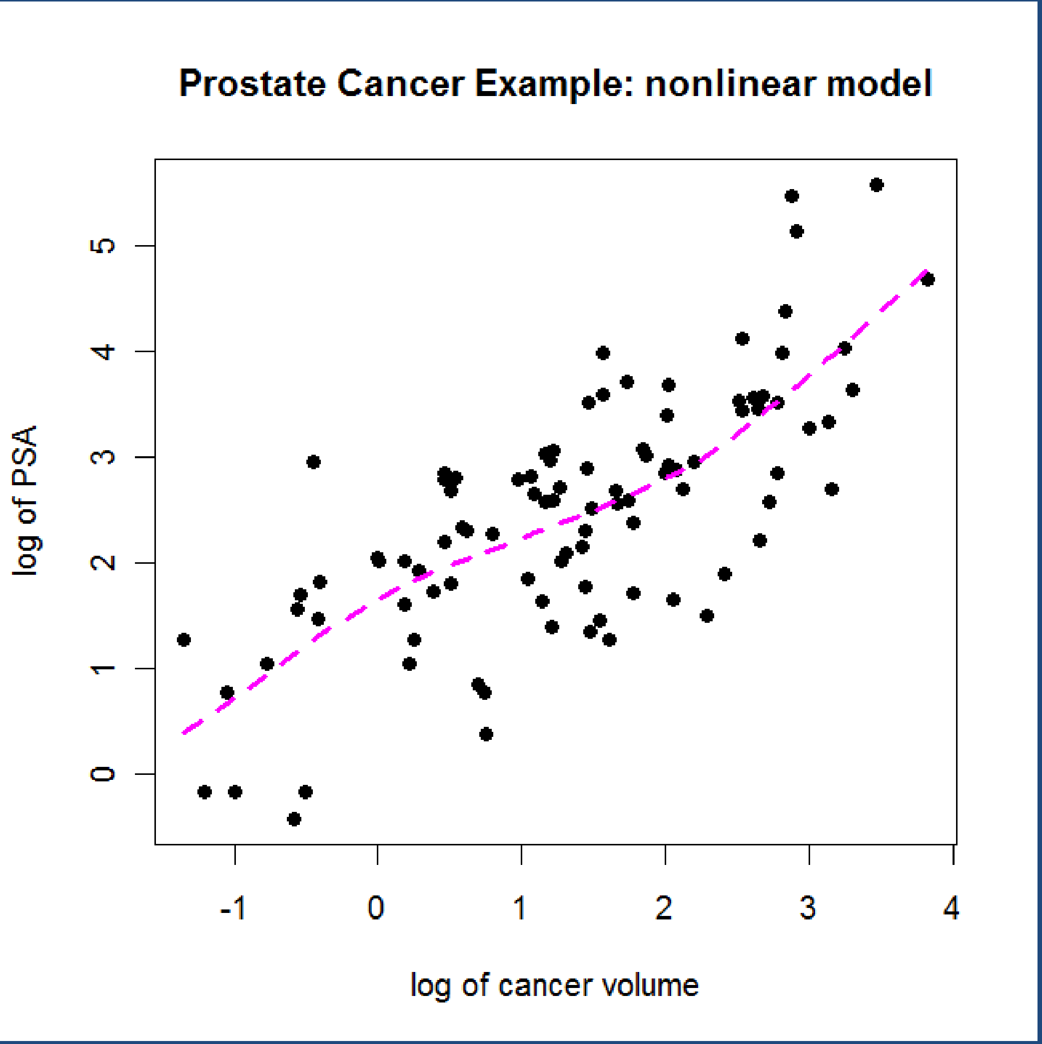
Regression Picture



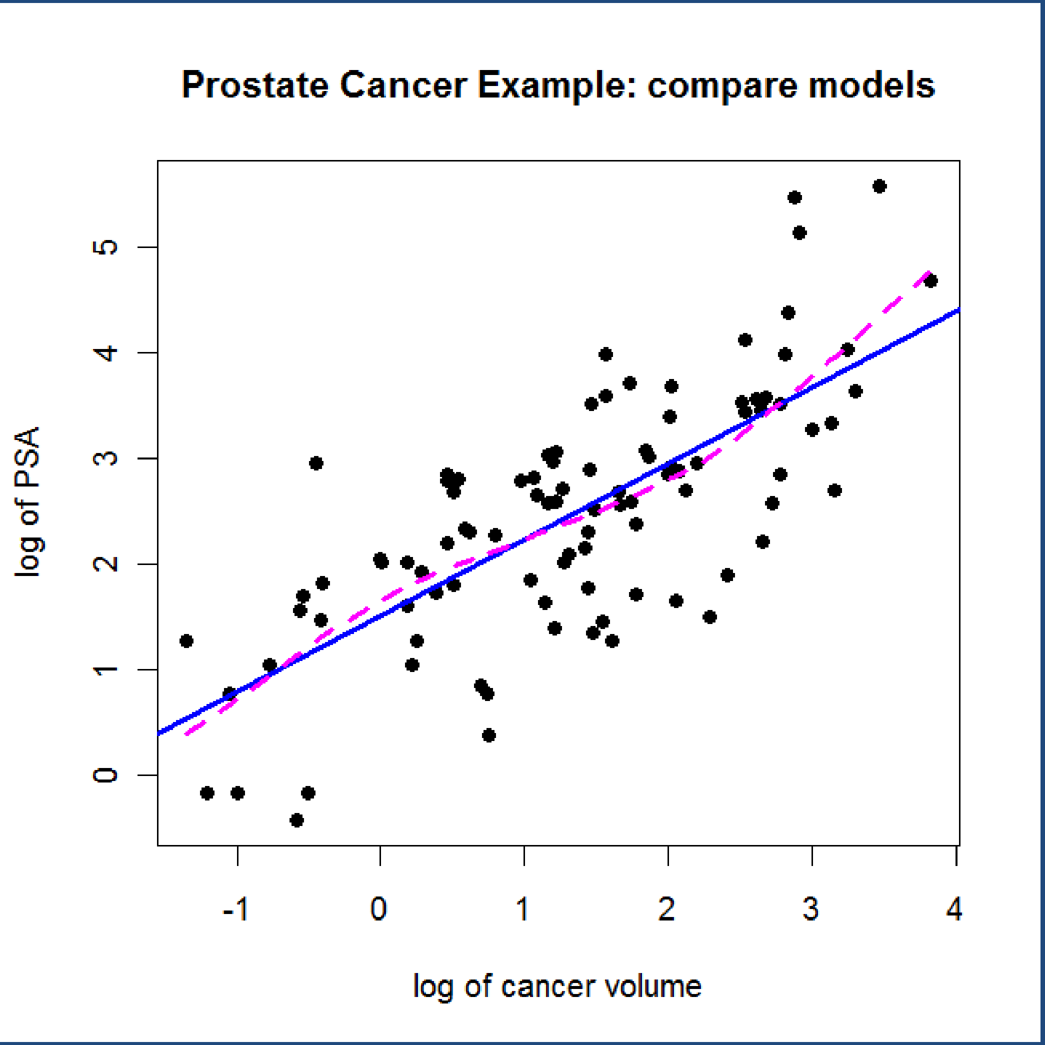
Regression Picture



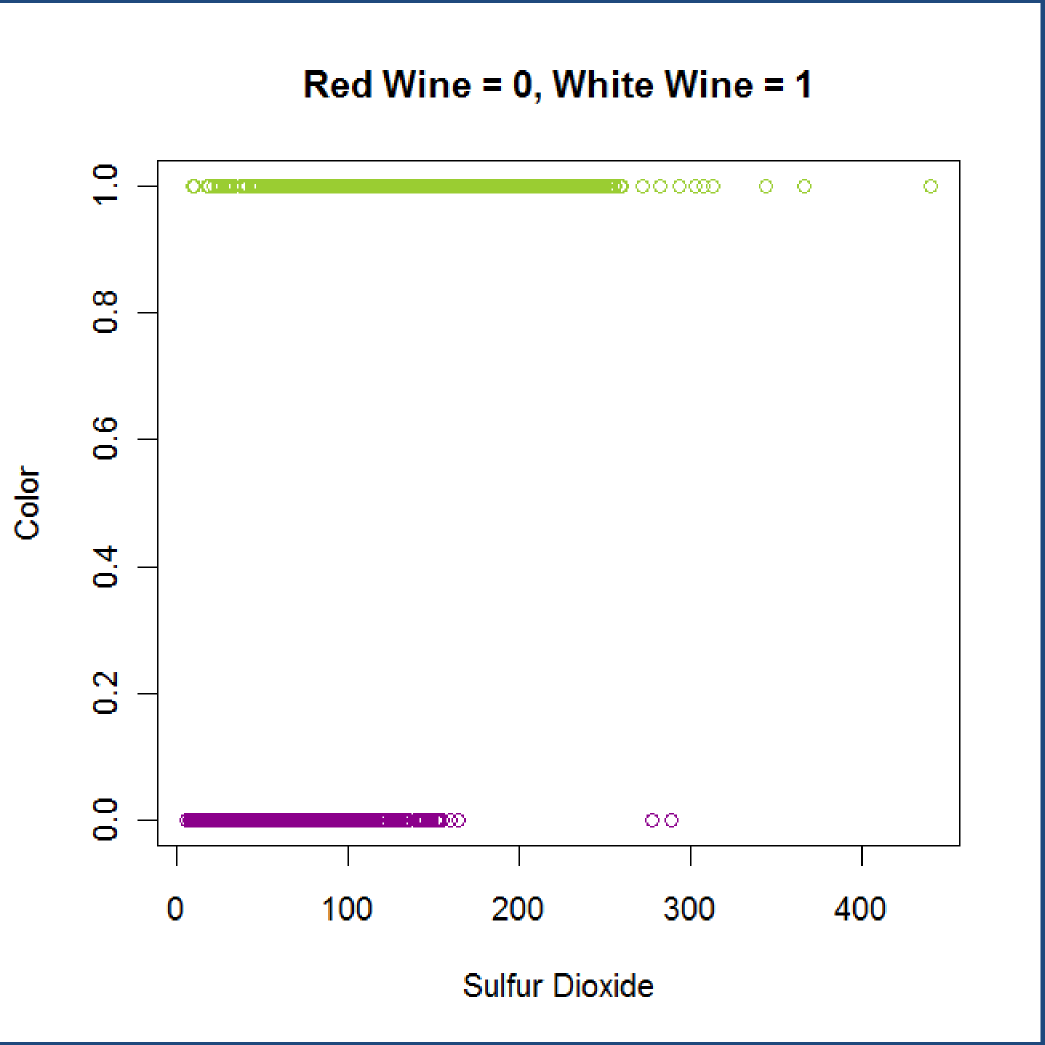
Regression Picture



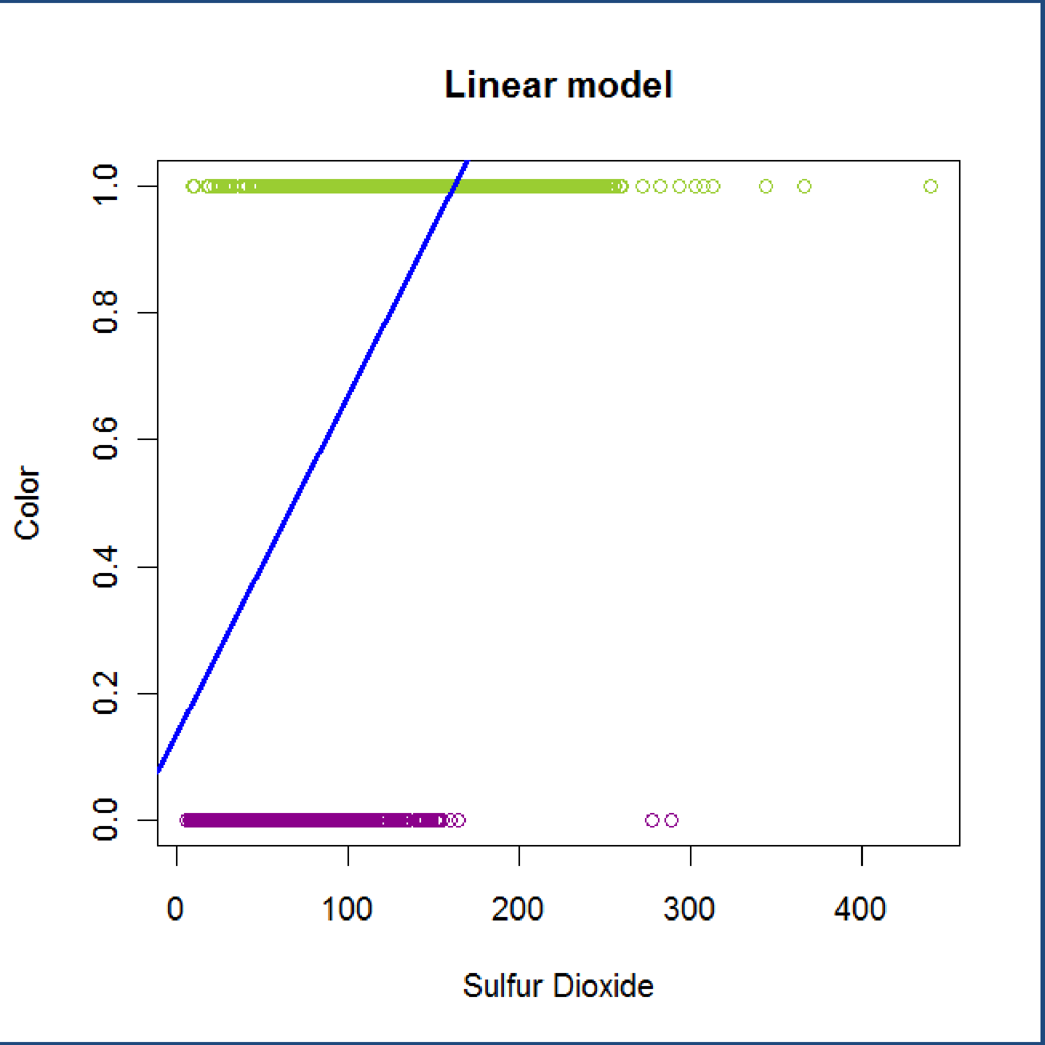
Regression Picture



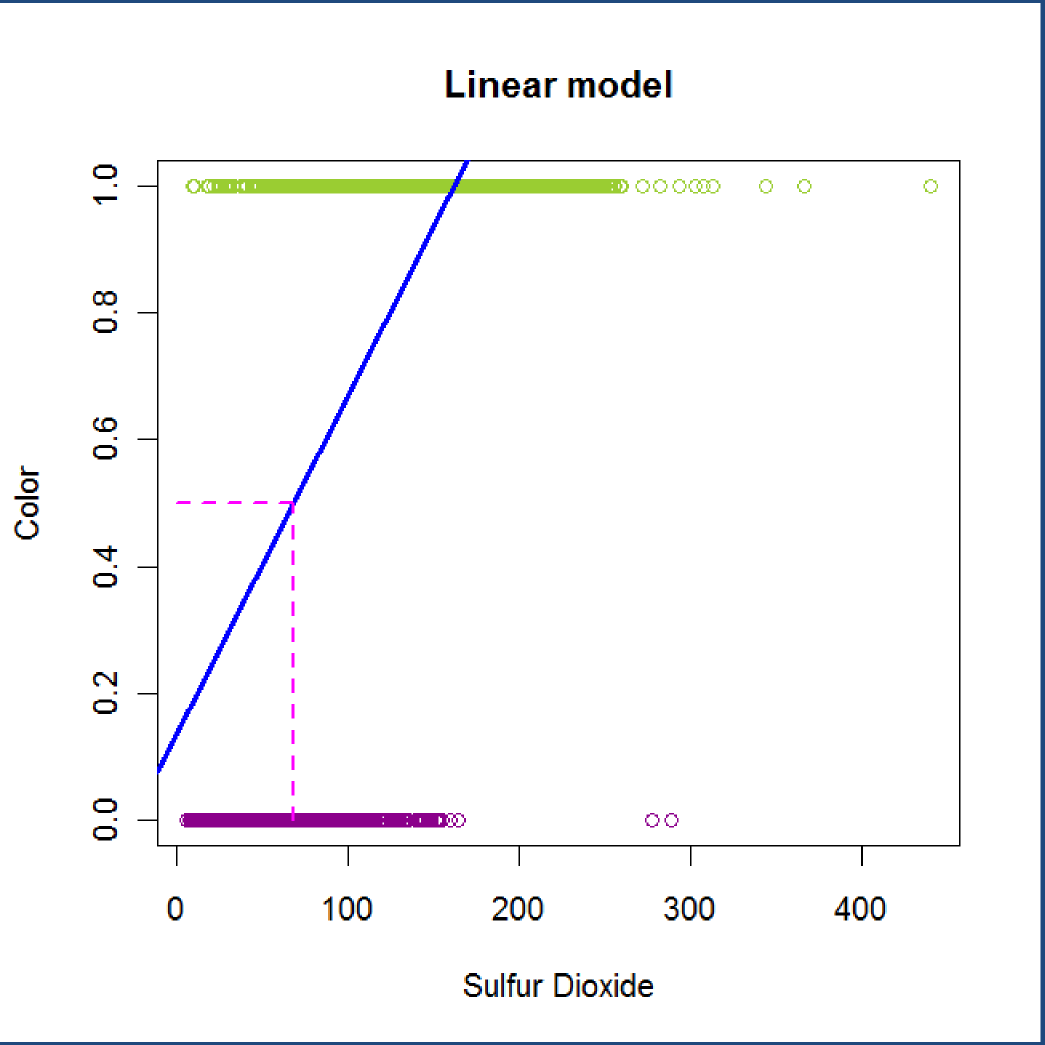
Classification Picture



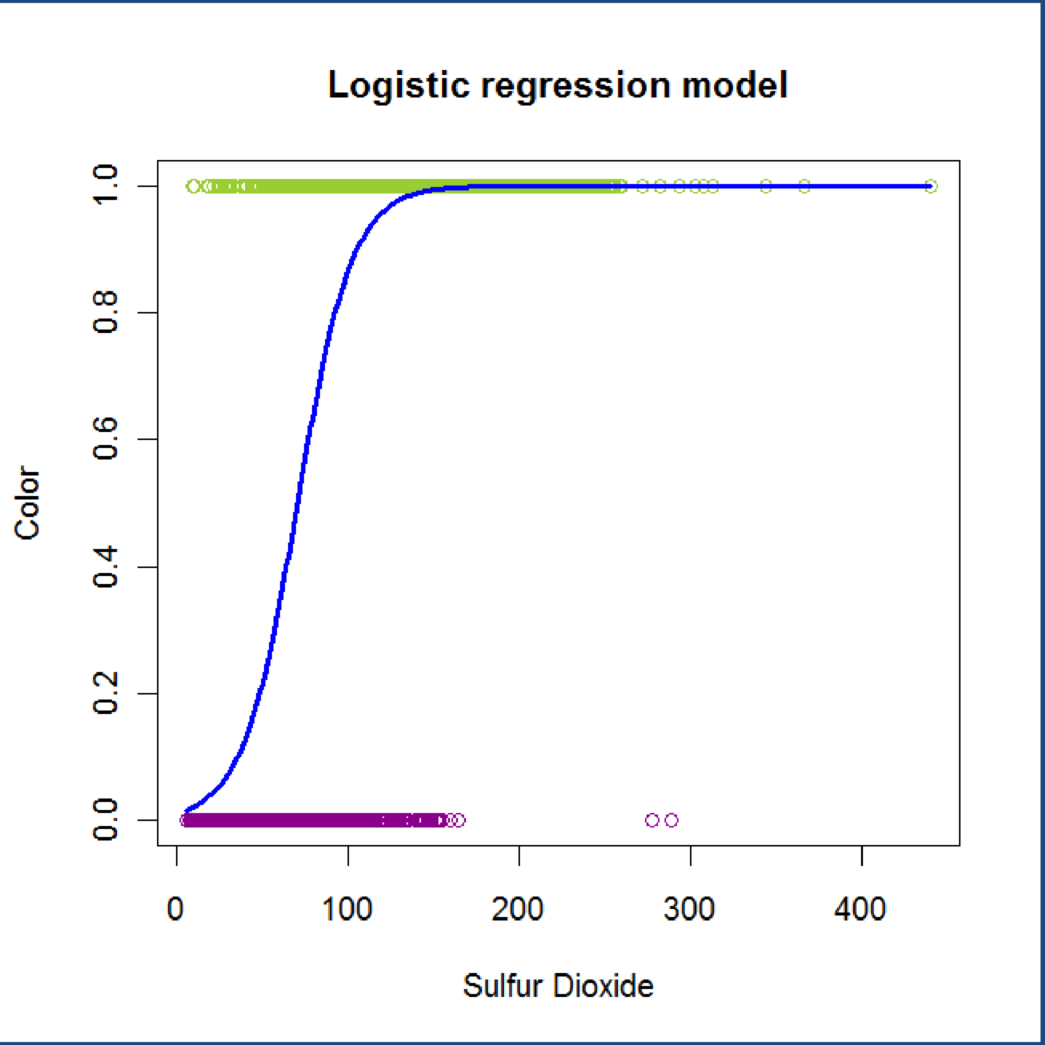
Classification Picture



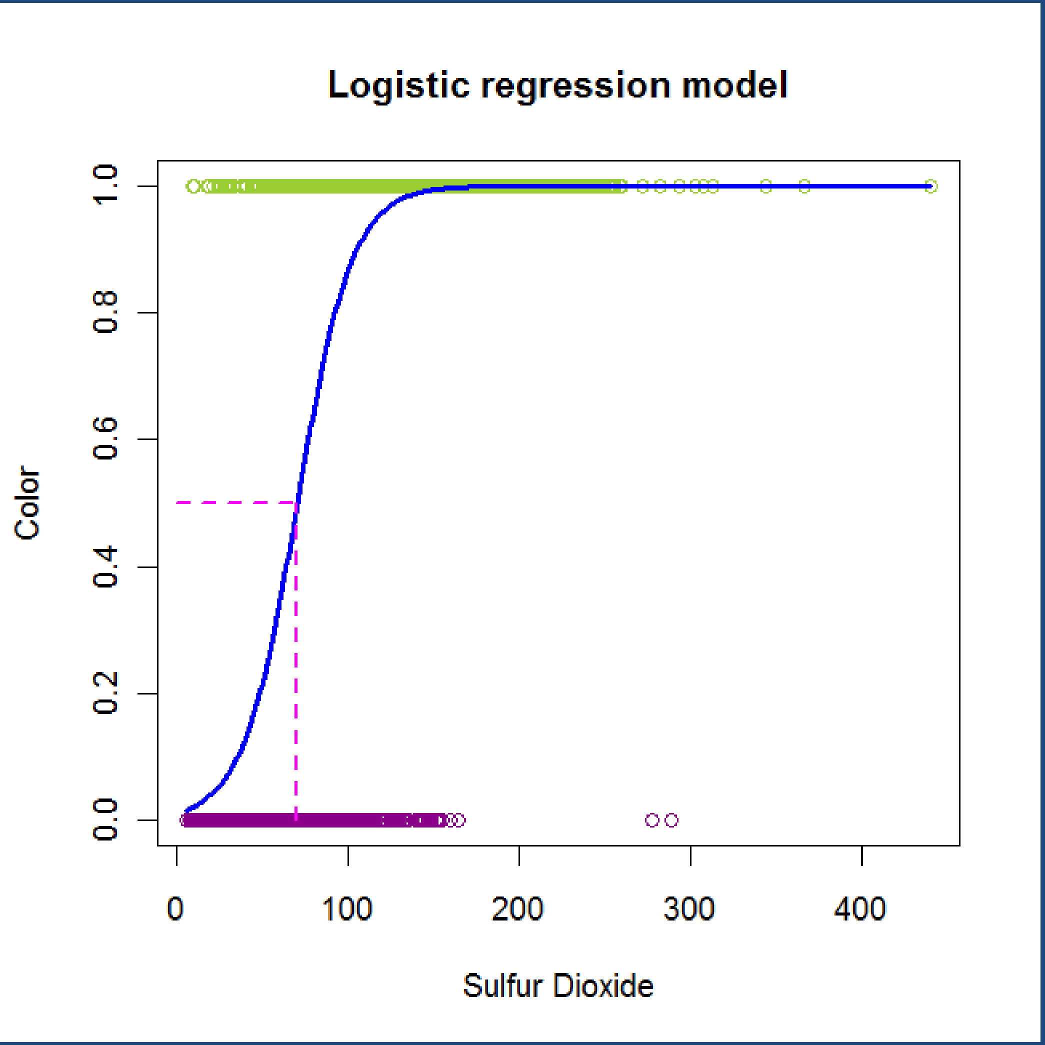
Classification Picture



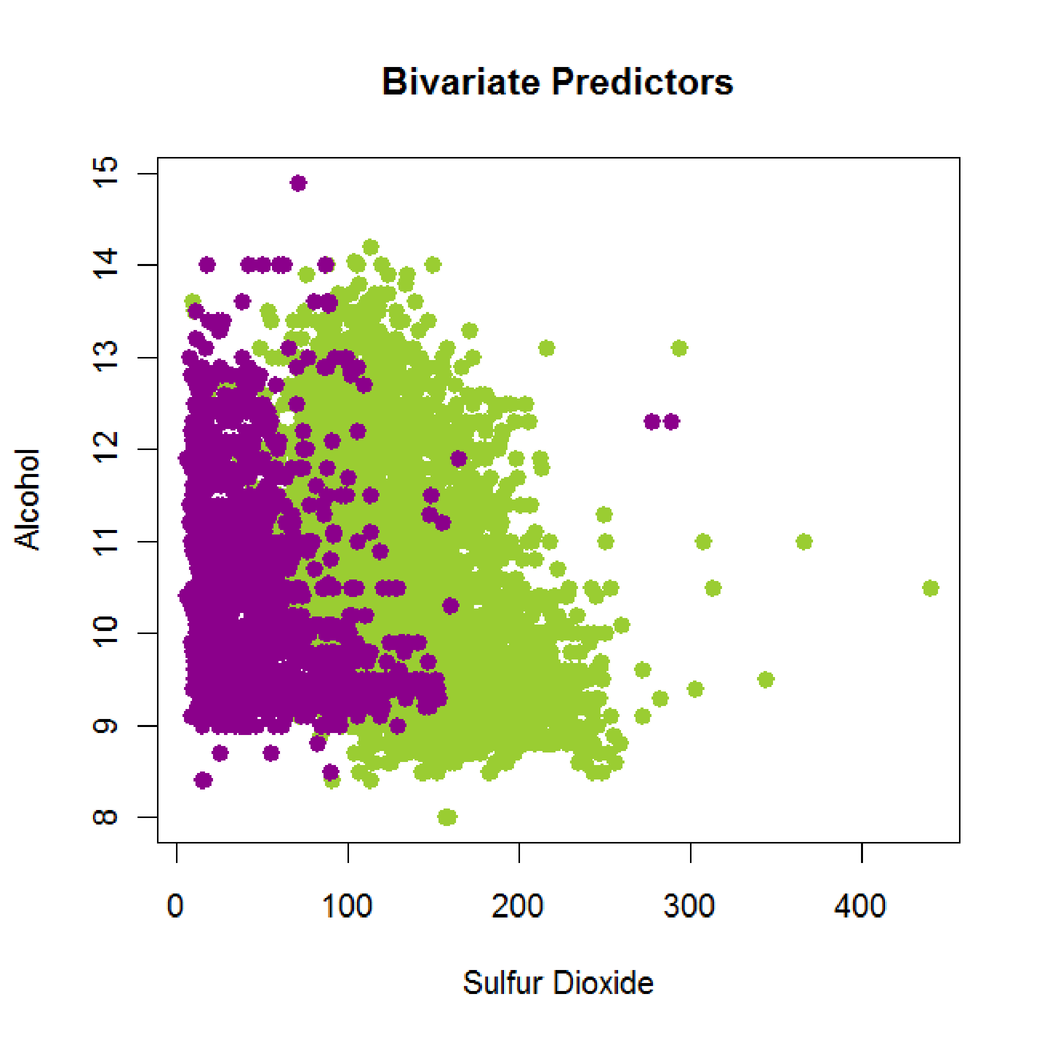
Classification Picture



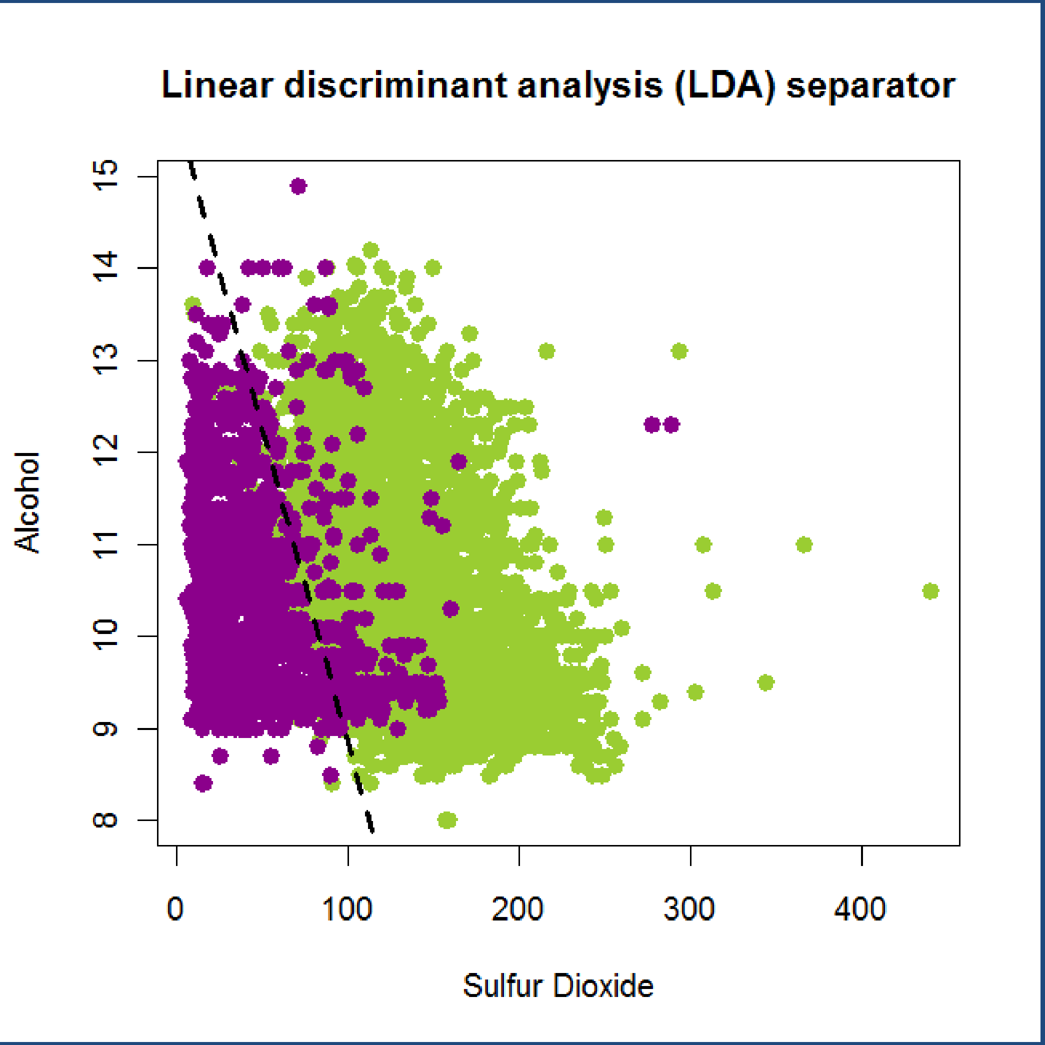
Classification Picture



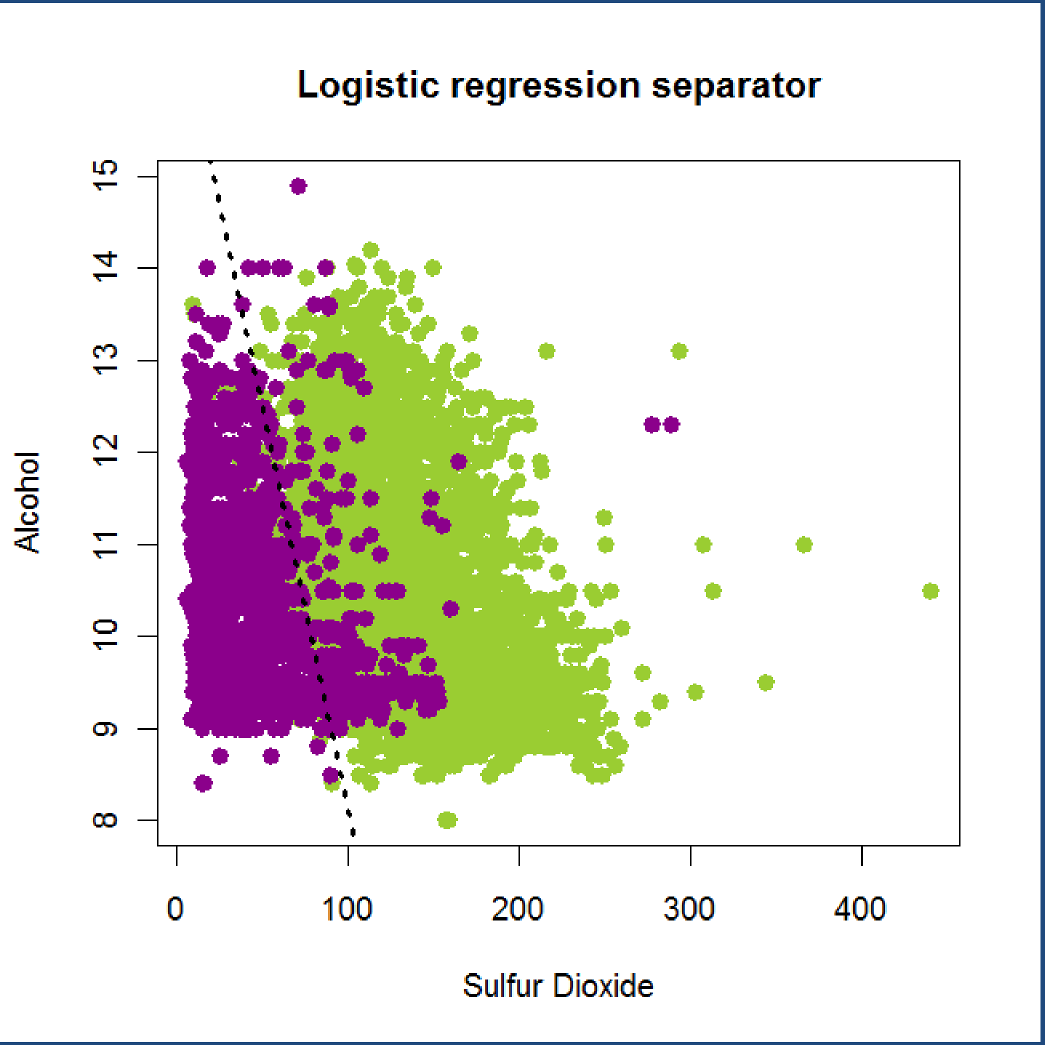
Classification Picture



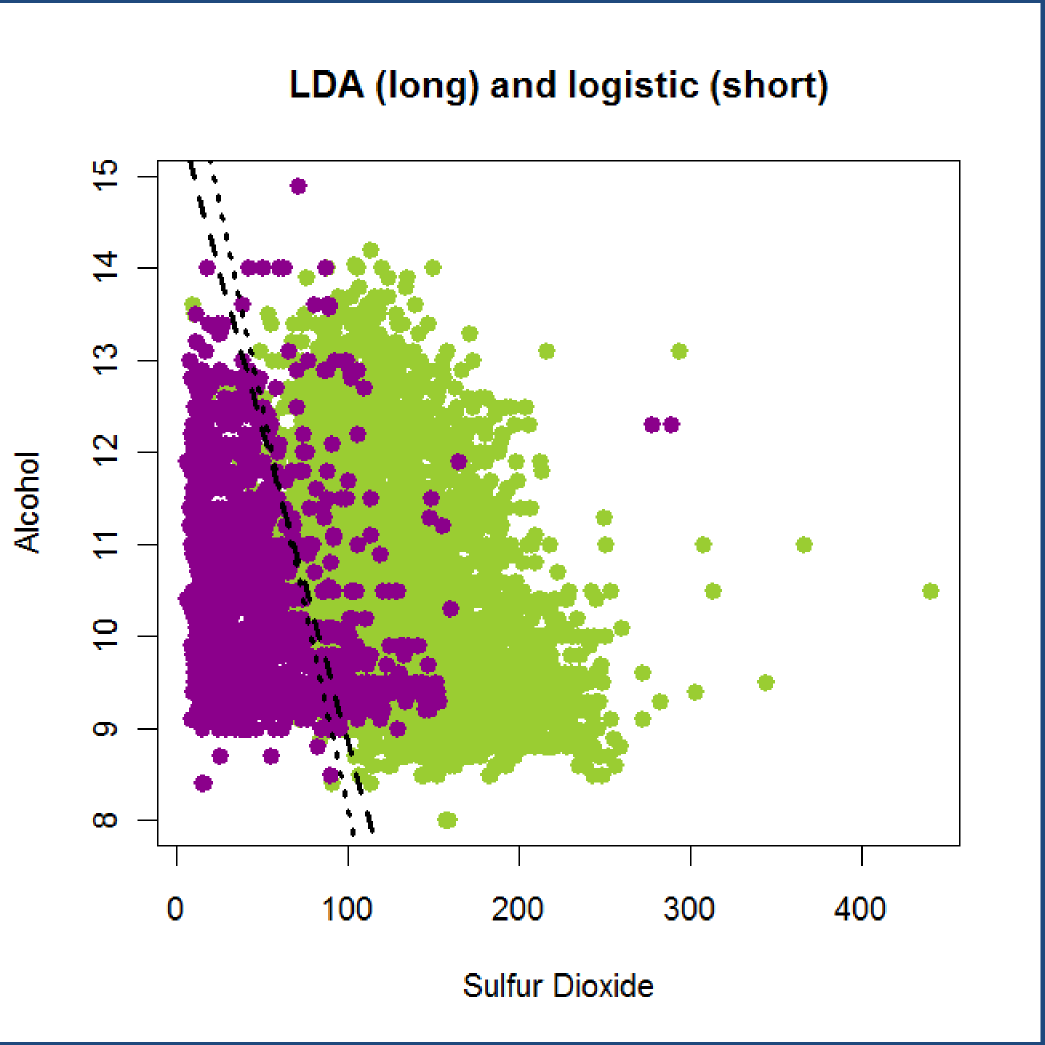
Classification Picture



Classification Picture



Classification Picture



Predictive Modeling

($\mathbf{x}_1$,y_1), … ($\mathbf{x}_n$,y_n), assumed to be independent, find a "model" for:

- Predicting the value of y for a new value of **x**
  - Expected mean squared error (regression)
  - Expected (class-wise) error rate (classificaiton)
- Understanding the relationship between **x** and y
  - Which predictors are useful? How? Where?
  - Is there "interesting" structure?

Estimates of Predictive Accuracy

- Resubstitution
  – Use the accuracy on the training set as an estimate of generalization error
- AIC etc
- Cross-validation
  – Randomly select a training set, use the rest to estimate accuracy
  – 10-fold cross-validation

10-Fold Cross-validation

Divide the data at random into 10 pieces, $D_1,\ldots,D_{10}$

- Fit the predictor to D_2, D_3, … D_{10}, predict D_1
- Fit the predictor to D_1, D_3, … D_{10}, predict D_2
- …
- Fit the predictor to D_1, D_2, … D_9, predict D_{10}

Estimate accuracy using the assembled predictions

Estimates of Predictive Accuracy

- Resubstitution estimates can be very optimistic
- AIC etc:
  - Make assumptions about data (distributions)
  - Only possible for simple situations
- Cross-validation estimates tend to be slightly pessimistic (smaller samples)
- Random Forests has its own way of estimating predictive accuracy (“out-of-bag” estimates)

Accuracy in Classification

Confusion matrix

		Predicted Class 0	Predicted Class 1	Total
Actual Class	0	a	b	a + b
Actual Class	1	c	d	c + d
Total		a + c	b + d	n

Specificity = a/(a + b)

Sensitivity = d/(c + d)

Error rate:

- b/(a + b) for class 0
- c/(c + d) for class 1
- (c + b)/n overall

Classification and Regression Trees

Pioneers:

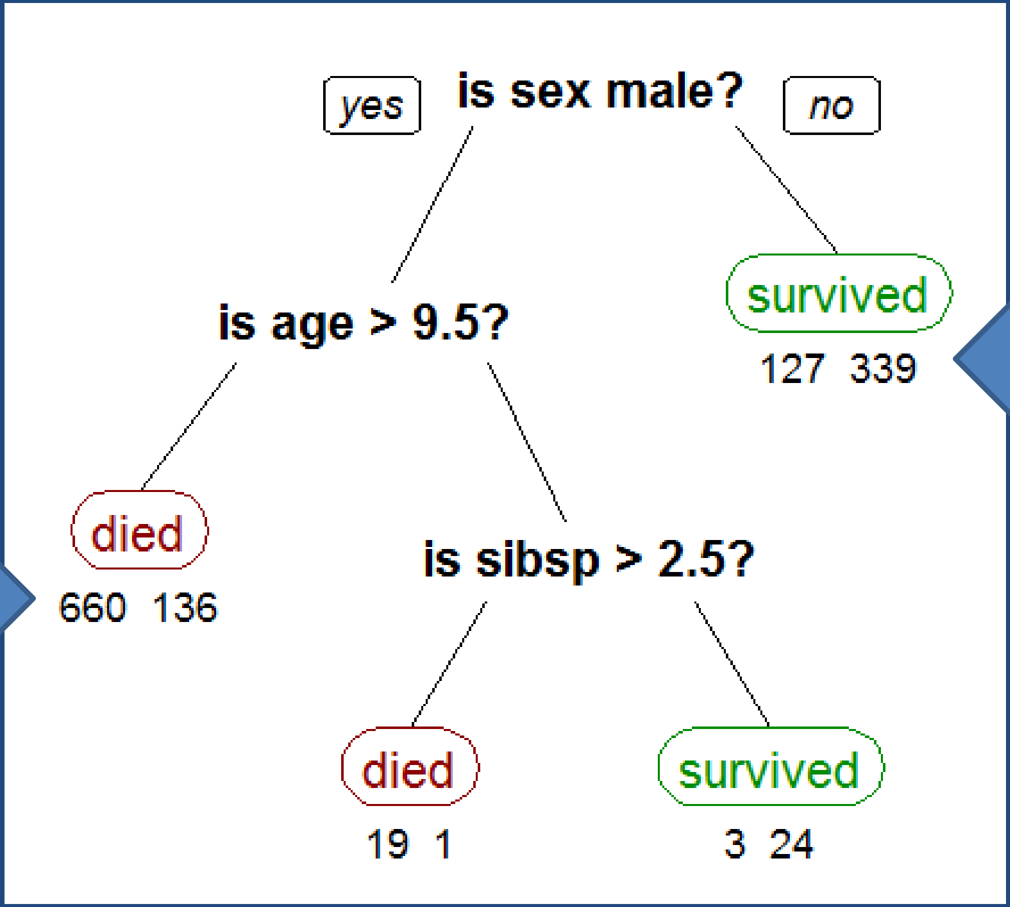
- Morgan and Sonquist (1963).
- Breiman, Friedman, Olshen, Stone (1984). ***CART***
- Quinlan (1993). *C4.5*




A Classification Tree

yes

go left



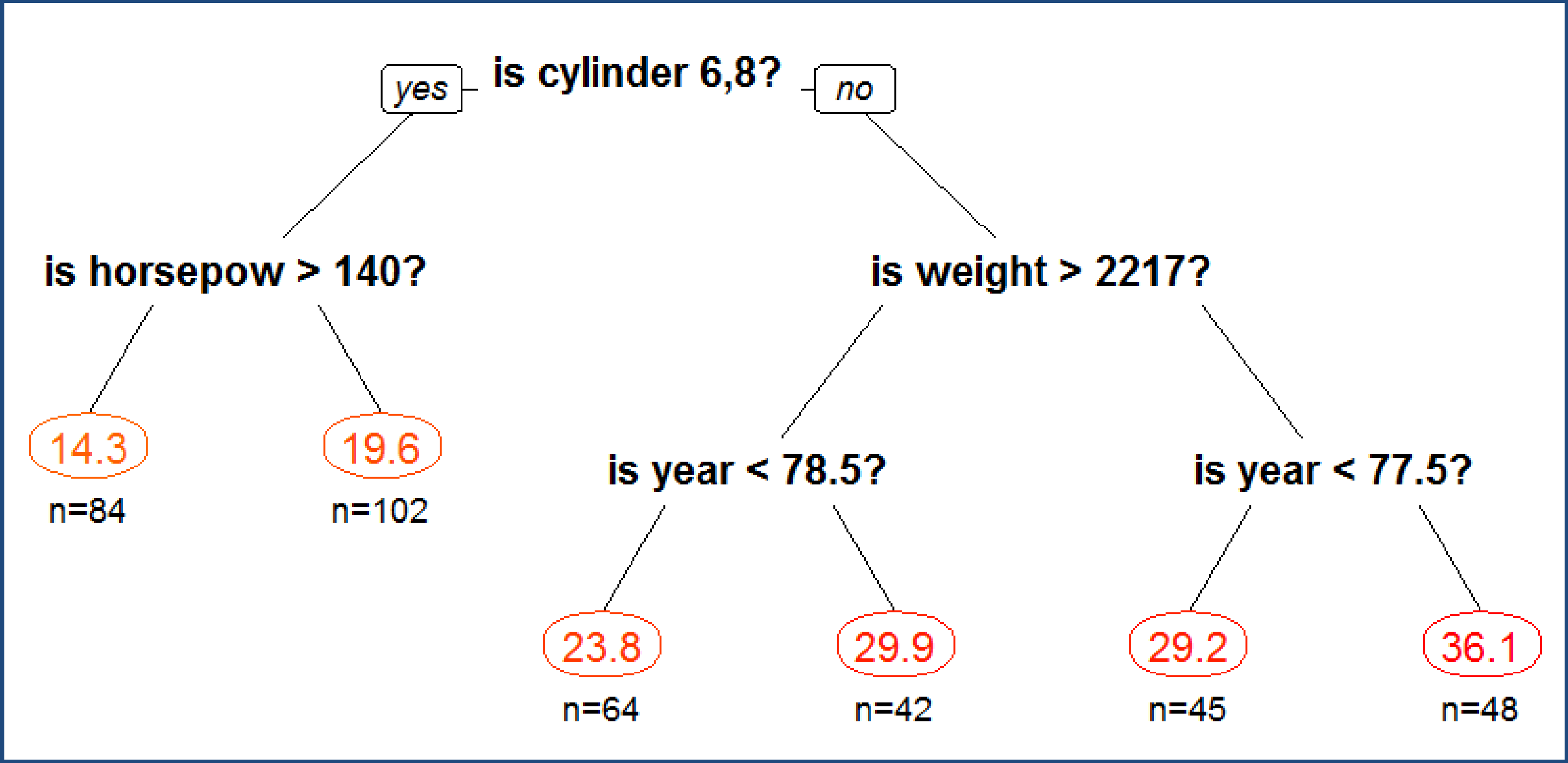
127 died

339 survived

660 died

136 survived

A Regression Tree



Splitting criteria

- **Regression**: residual sum of squares

$$RSS = \sum_{left} (y_i - y_L^*)^2 + \sum_{right} (y_i - y_R^*)^2$$

where y_L^* = mean y-value for left node

y_R^* = mean y-value for right node

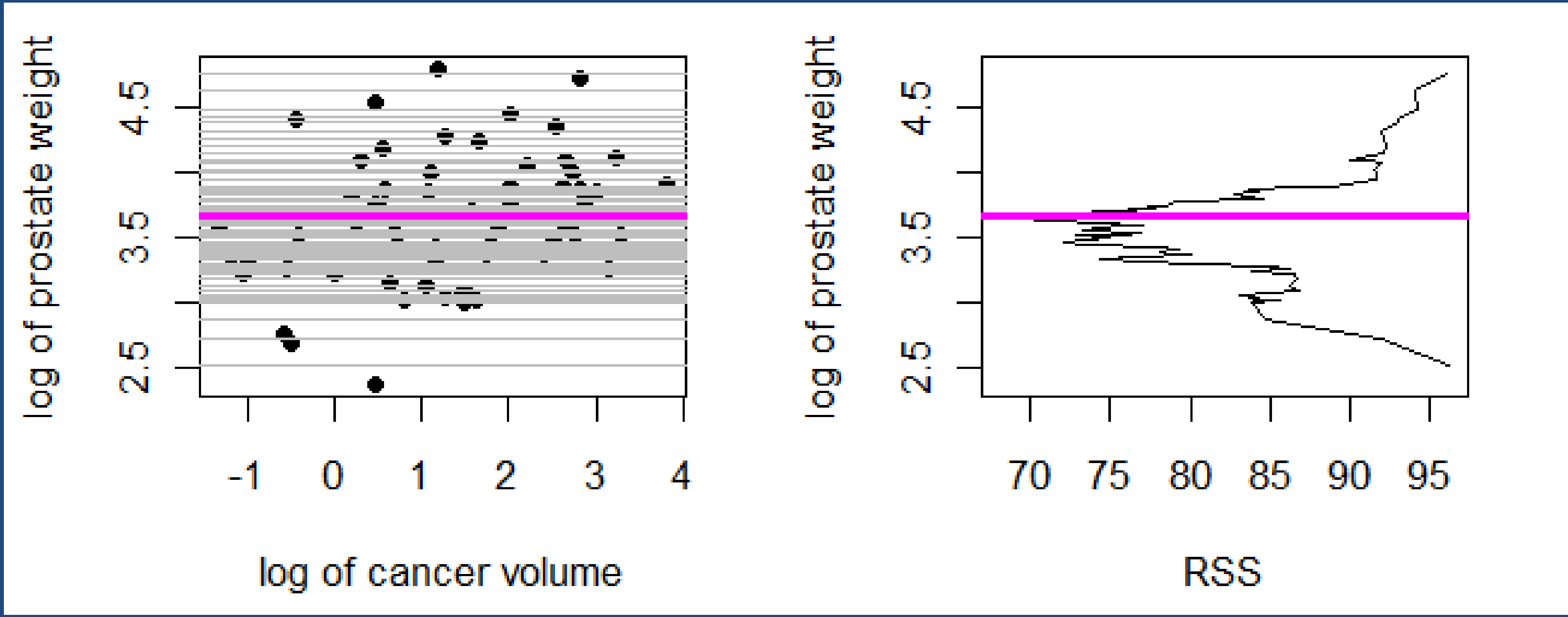
- **Classification**: Gini criterion

$$Gini = n_L \sum_{k=1,\ldots,K} p_{kL} (1- p_{kL}) + n_R \sum_{k=1,\ldots,K} p_{kR} (1- p_{kR})$$

where p_{kL} = proportion of class k in left node

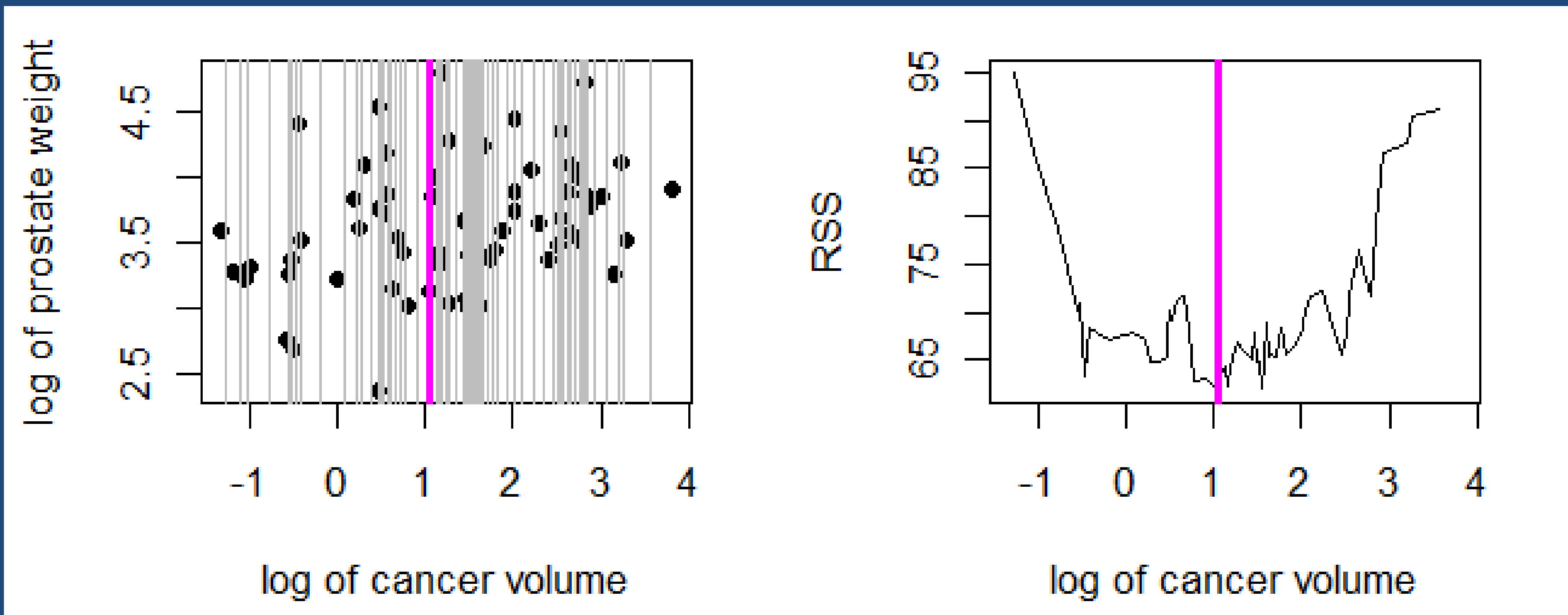
p_{kR} = proportion of class k in right node

Regression tree (prostate cancer)



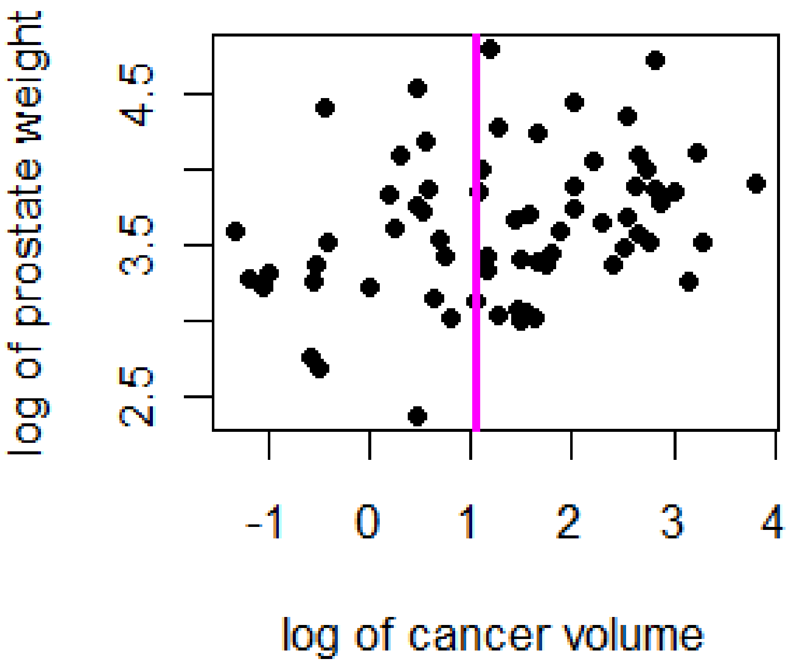
Best horizontal split is at 3.67 with RSS = 68.1

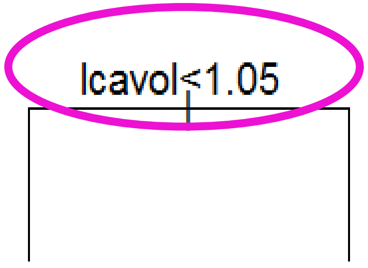
Regression tree (prostate cancer)



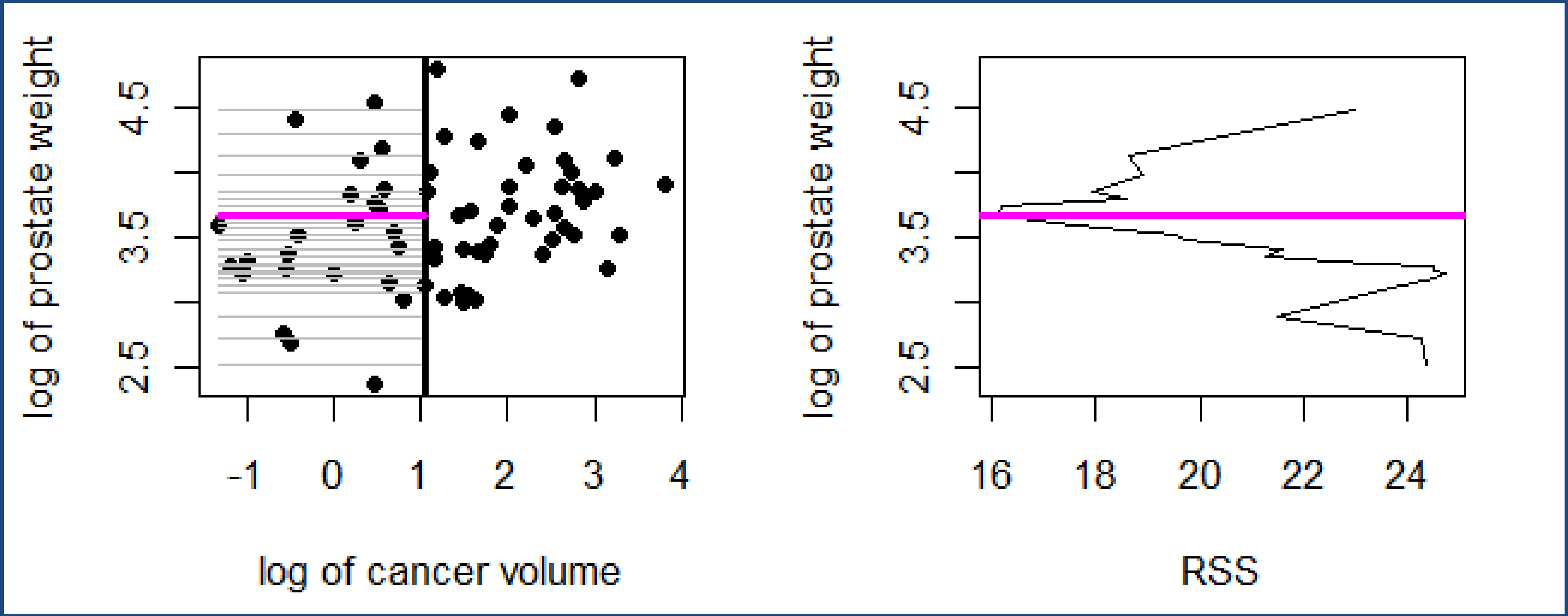
Best vertical split is at 1.05 with RSS = 61.8

Regression tree (prostate cancer)



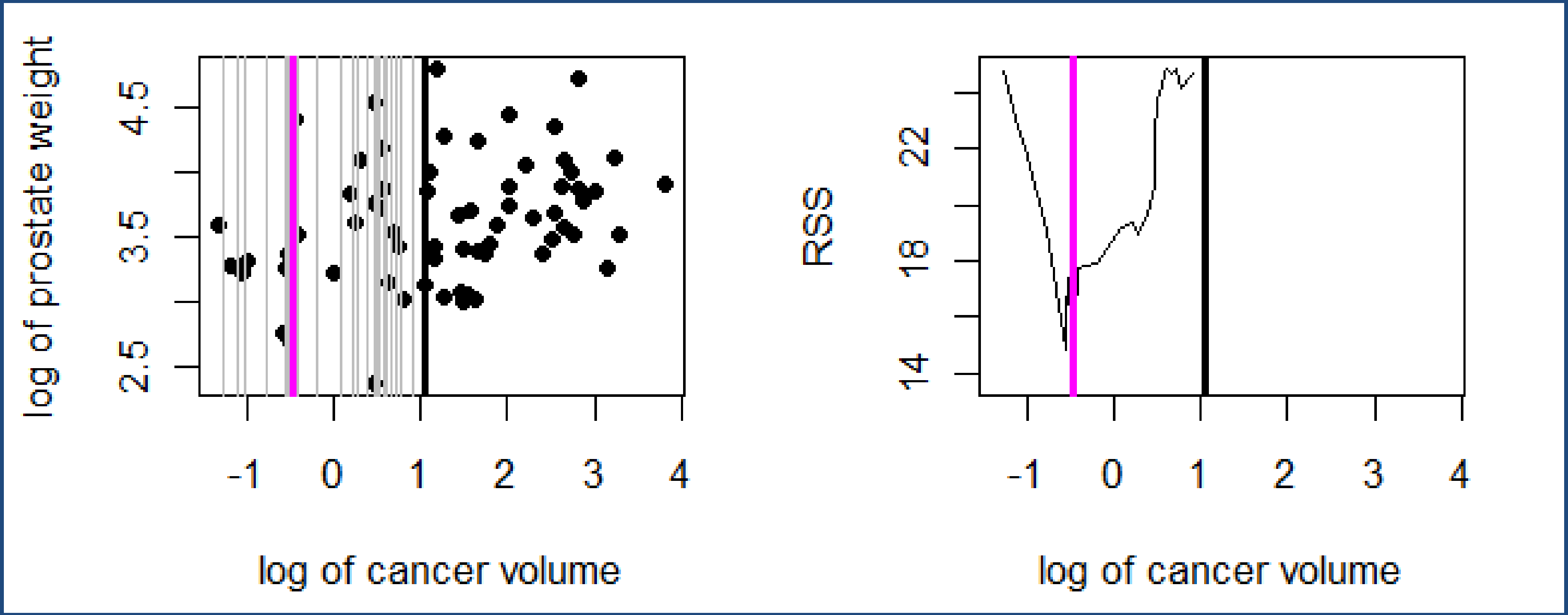


Regression tree (prostate cancer)



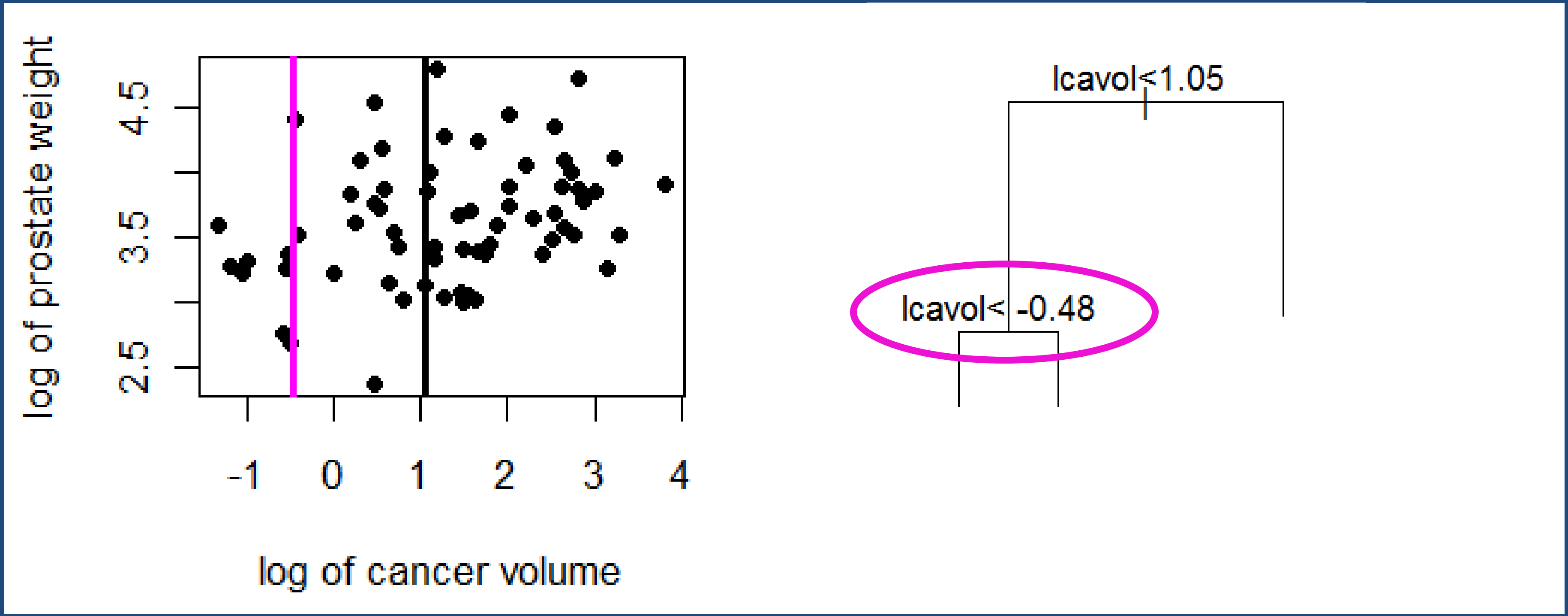
Best horizontal split is at 3.66 with RSS = 16.1

Regression tree (prostate cancer)

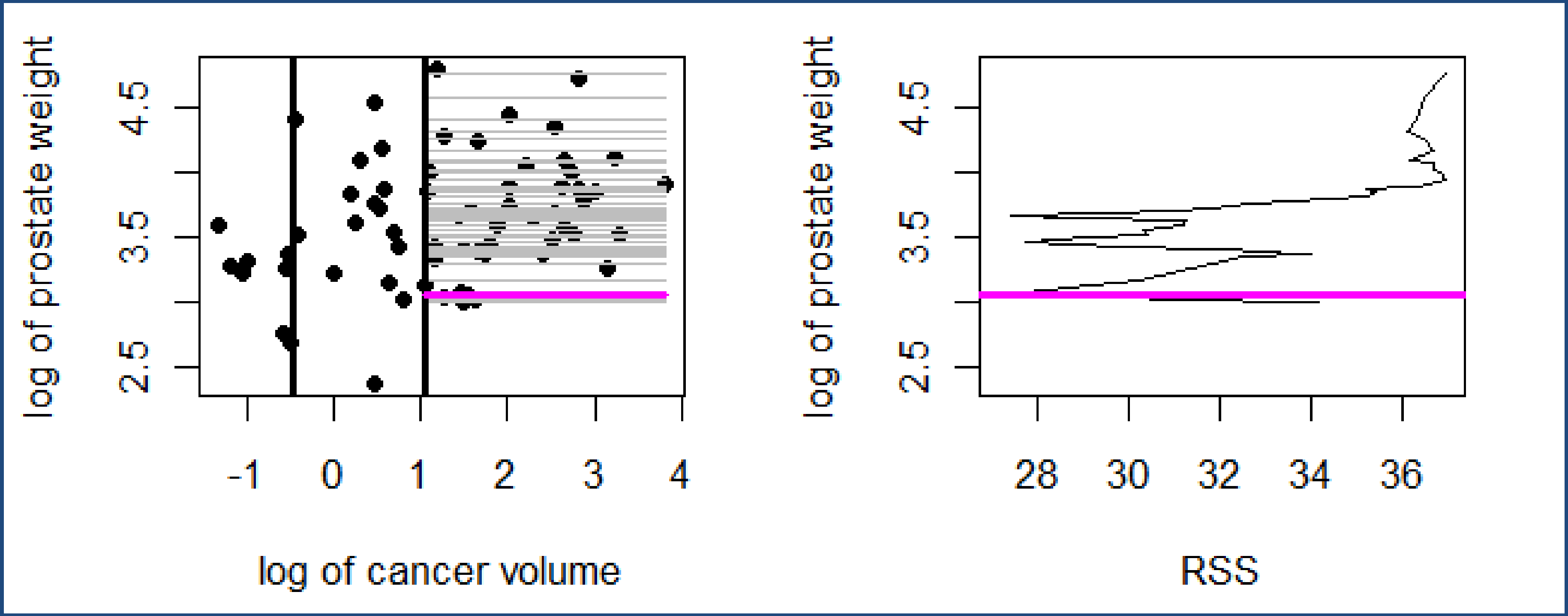


Best vertical split is at -.48 with RSS = 13.6

Regression tree (prostate cancer)

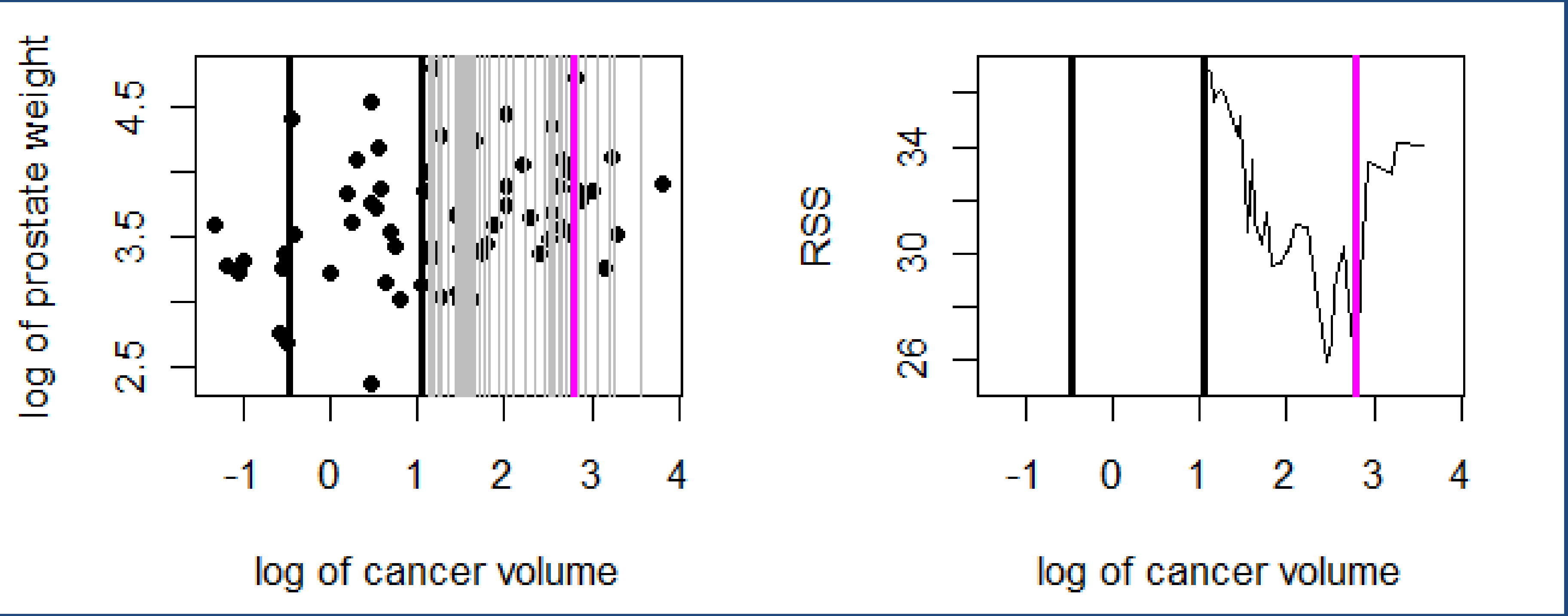


Regression tree (prostate cancer)



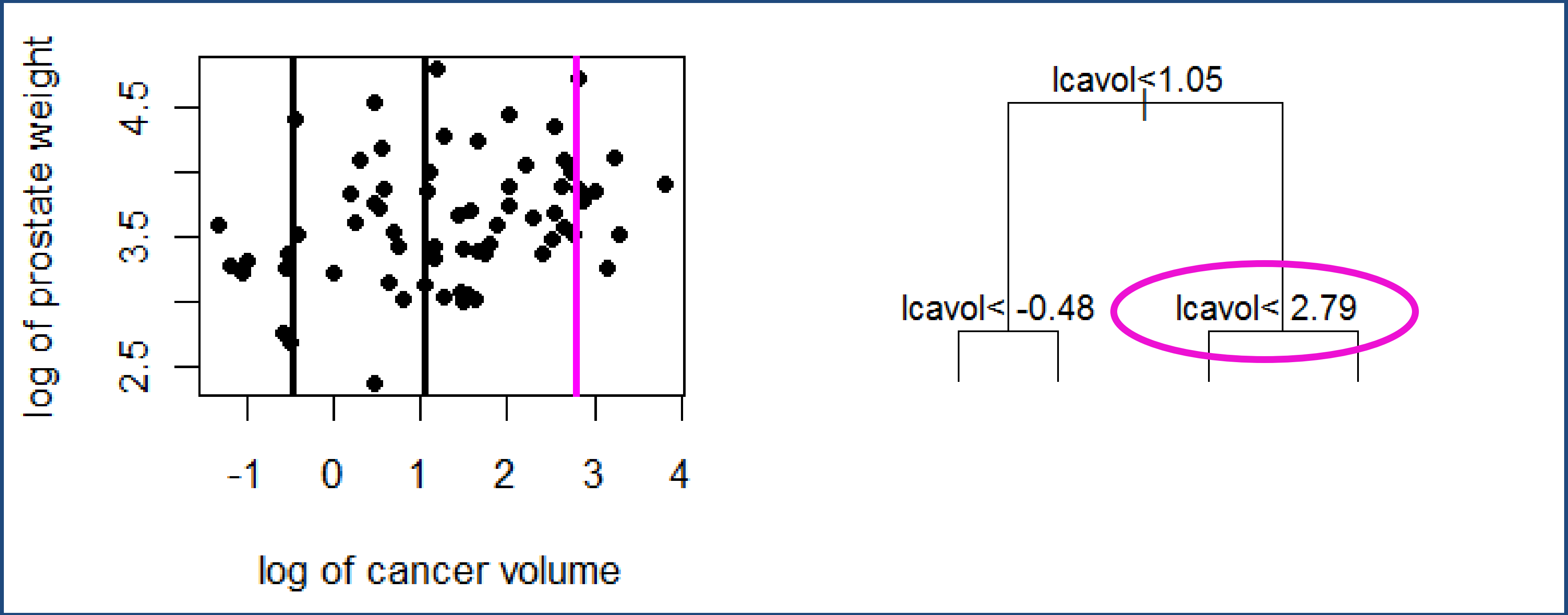
Best horizontal split is at 3.07 with RSS = 27.1

Regression tree (prostate cancer)

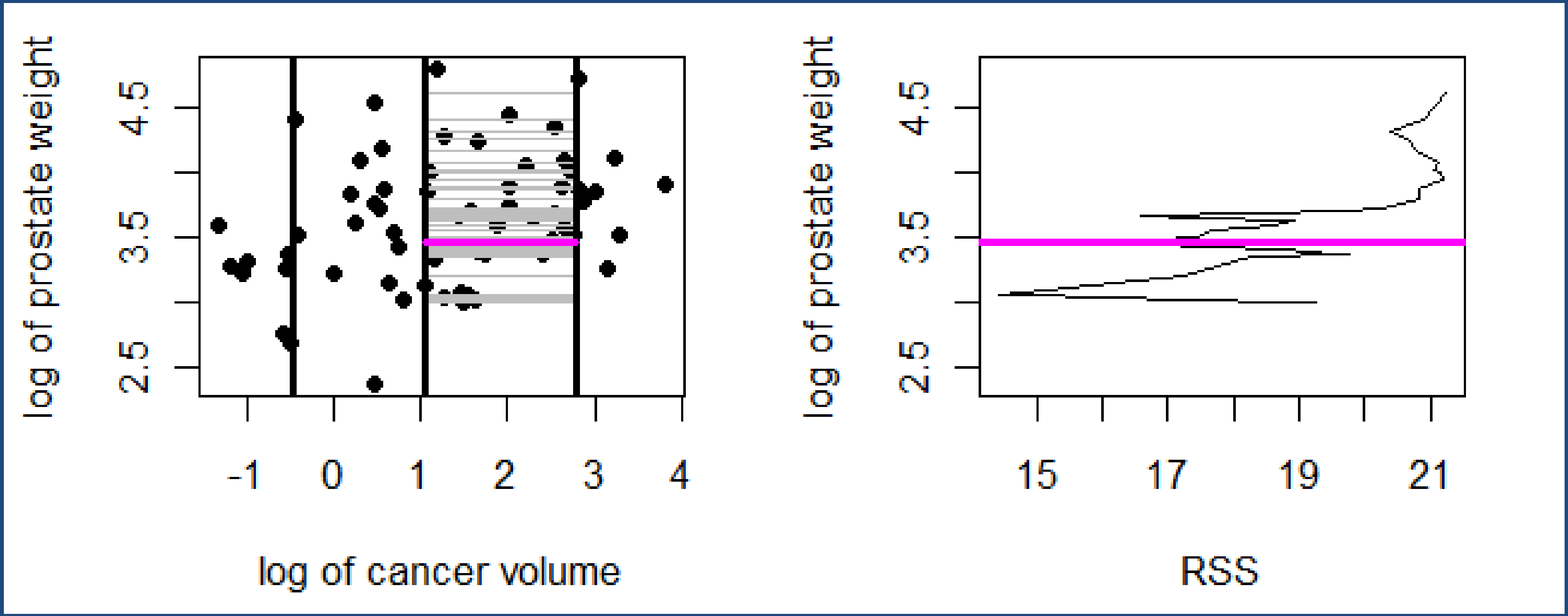


Best vertical split is at 2.79 with RSS = 25.1

Regression tree (prostate cancer)

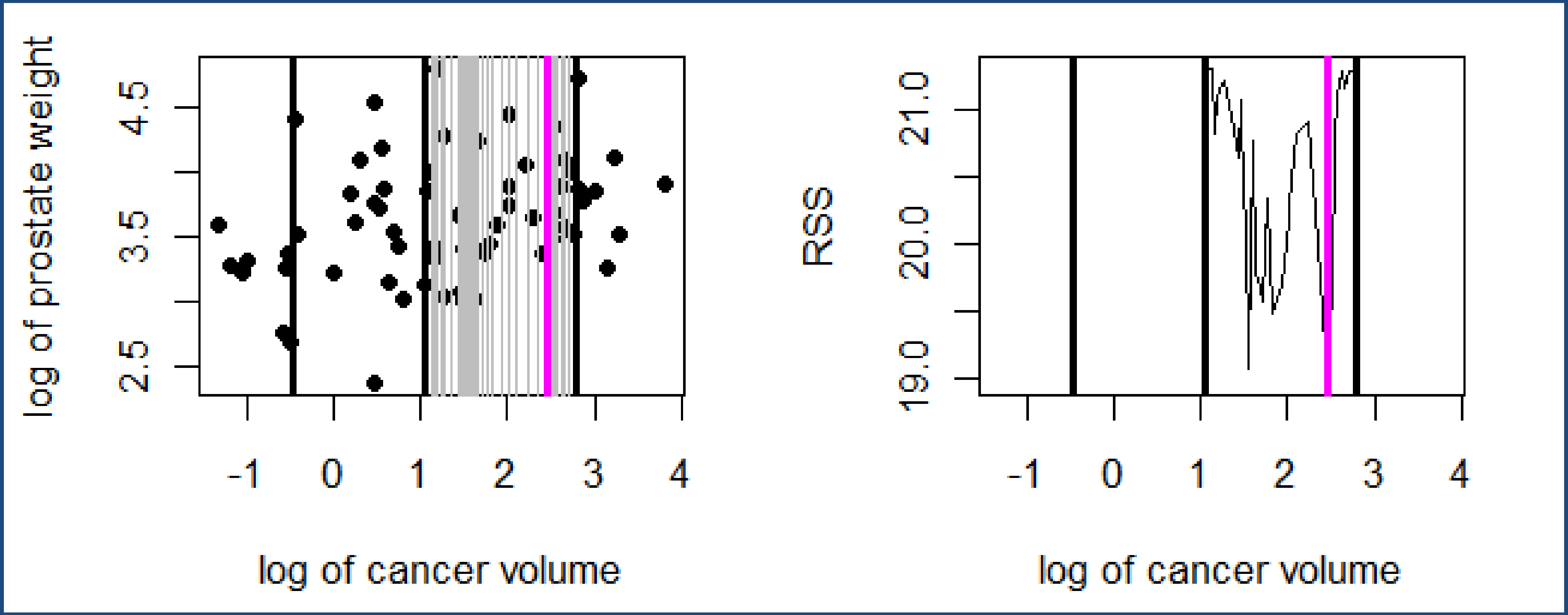


Regression tree (prostate cancer)



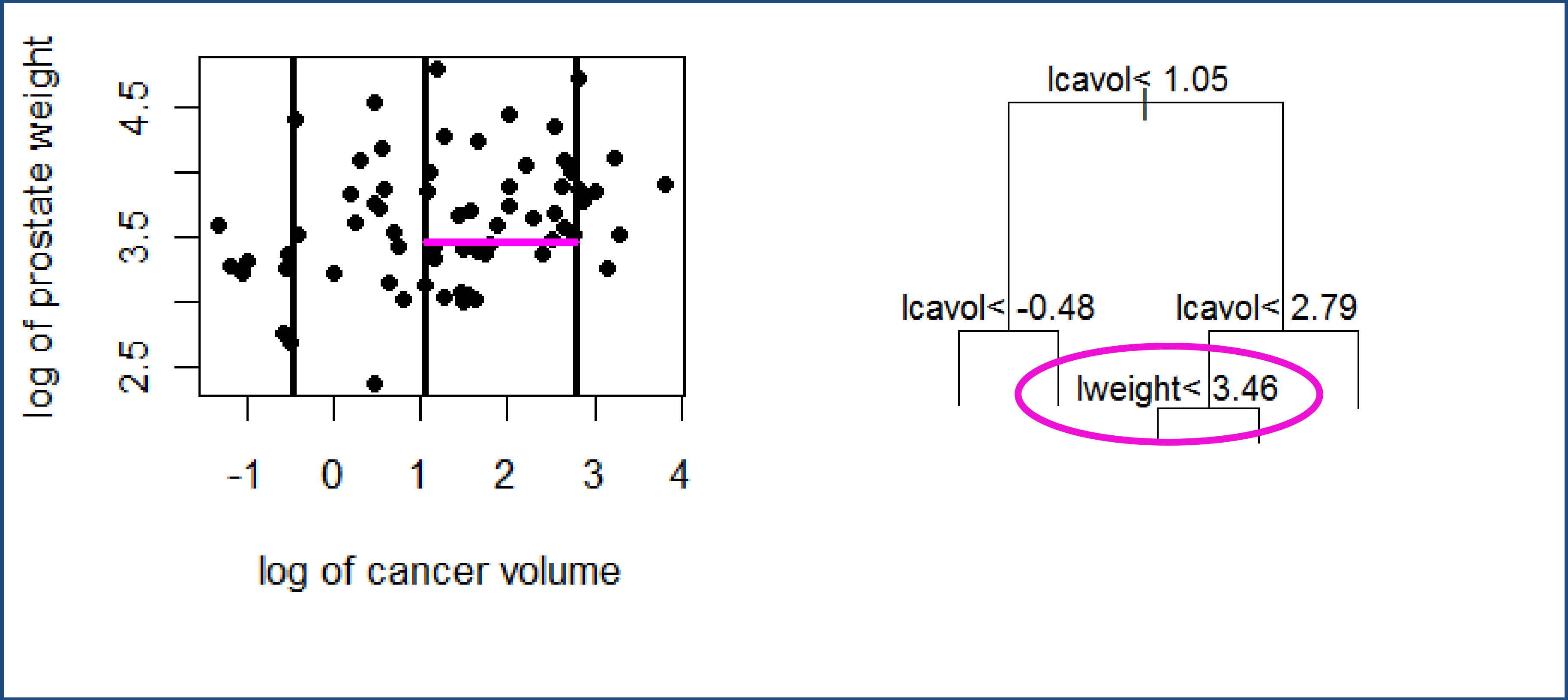
Best horizontal split is at 3.46 with RSS = 16.1

Regression tree (prostate cancer)

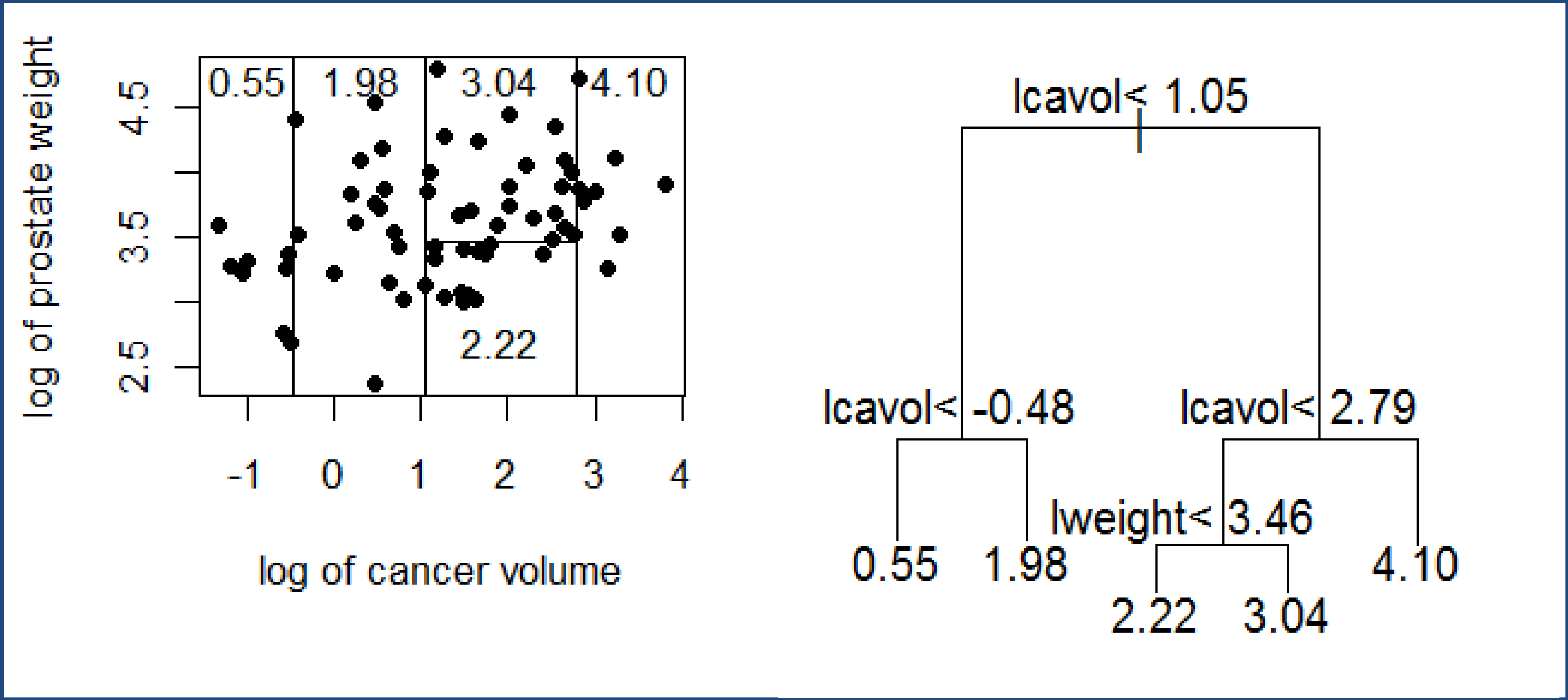


Best vertical split is at 2.46 with RSS = 19.0

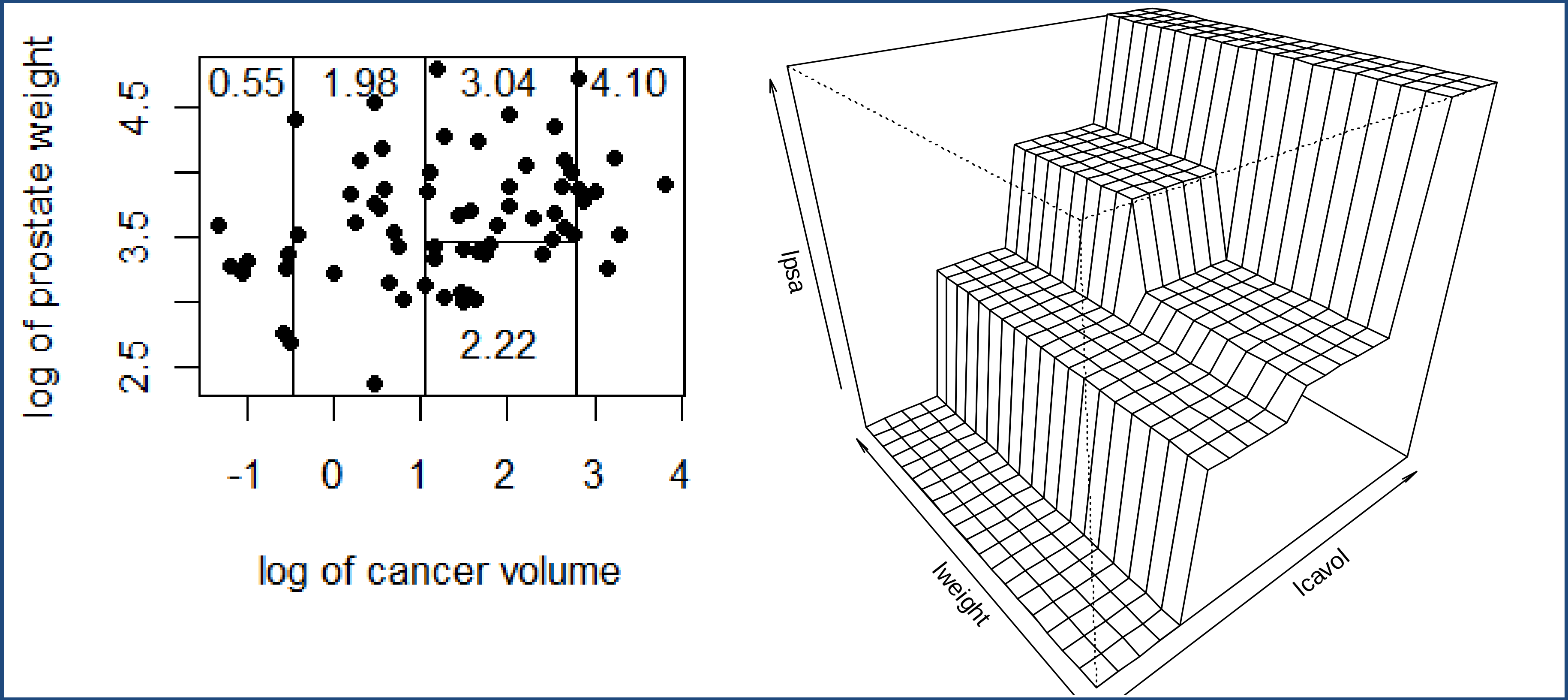
Regression tree (prostate cancer)



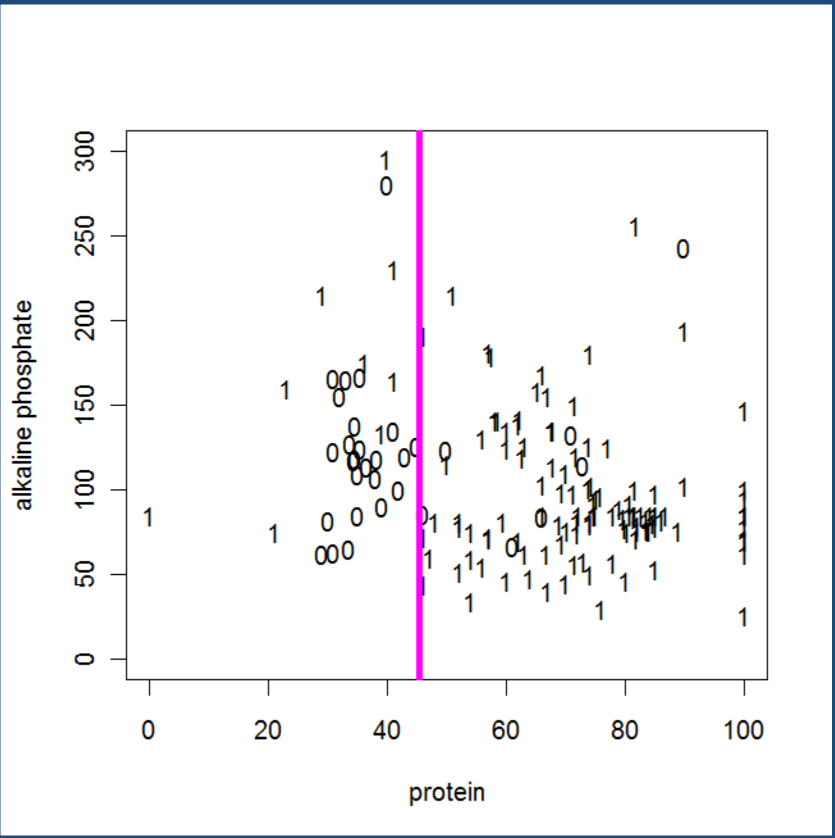
Regression tree (prostate cancer)

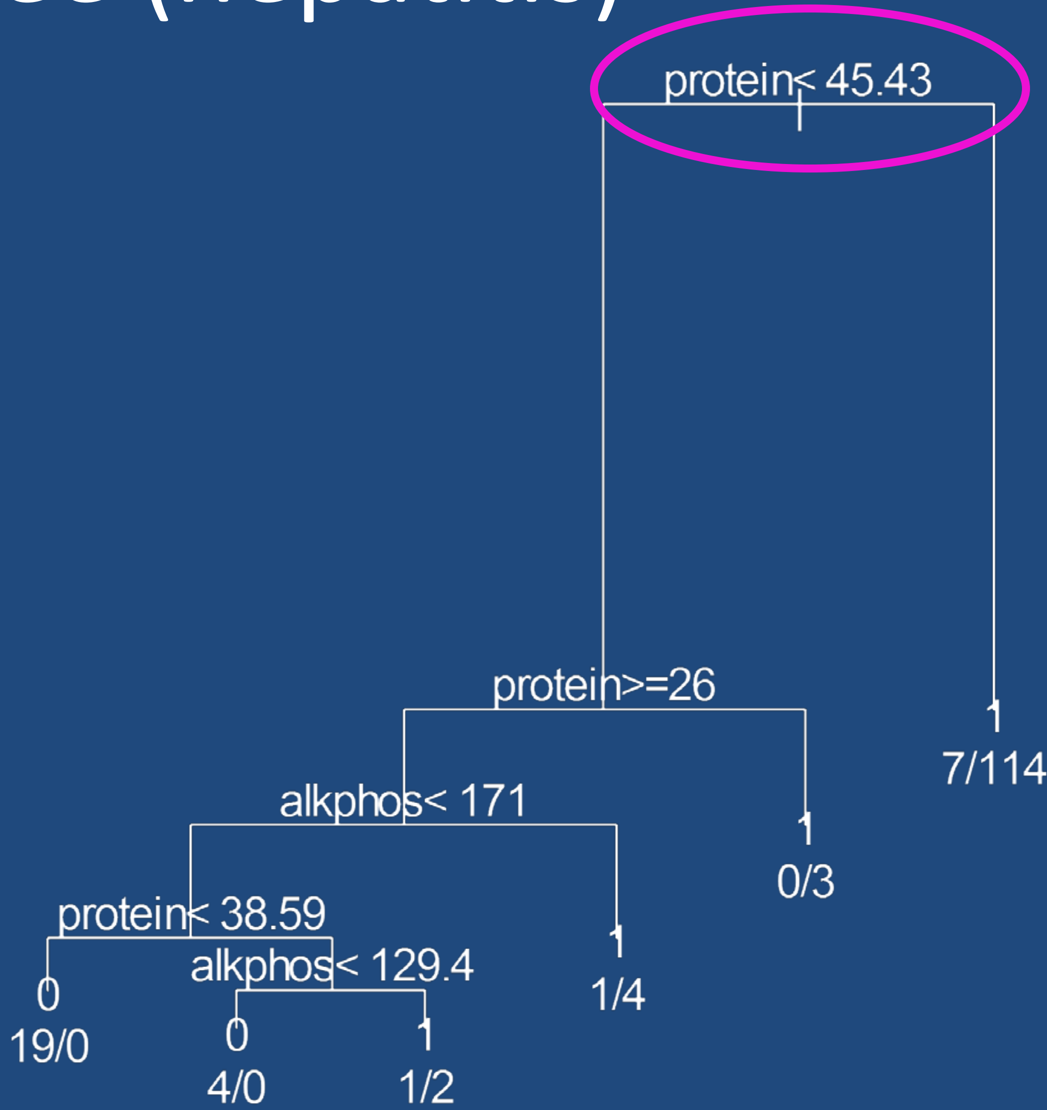


Regression tree (prostate cancer)

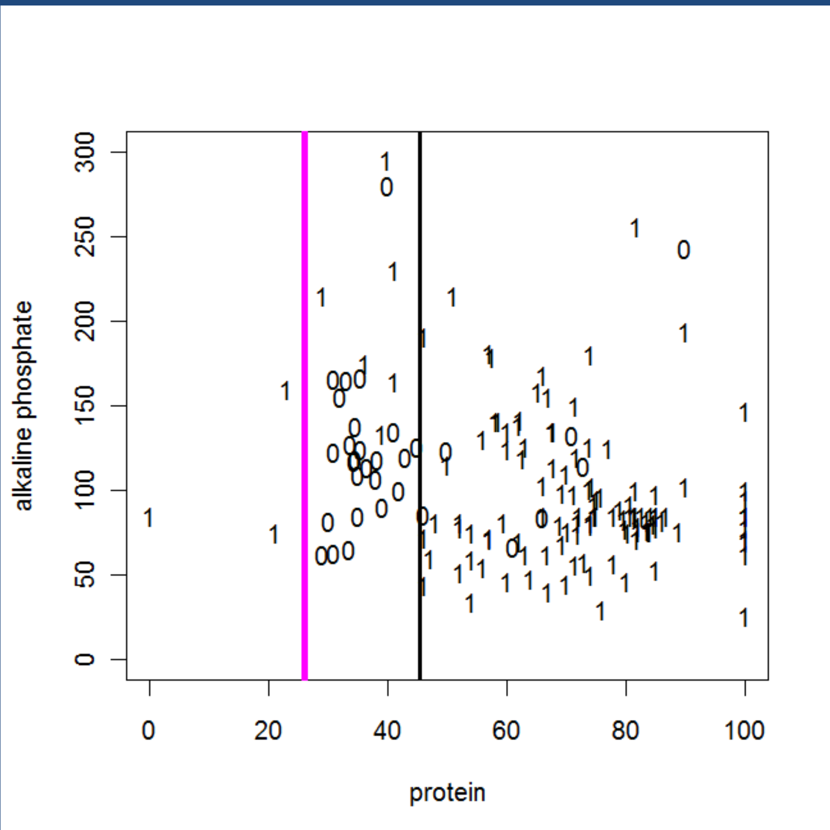


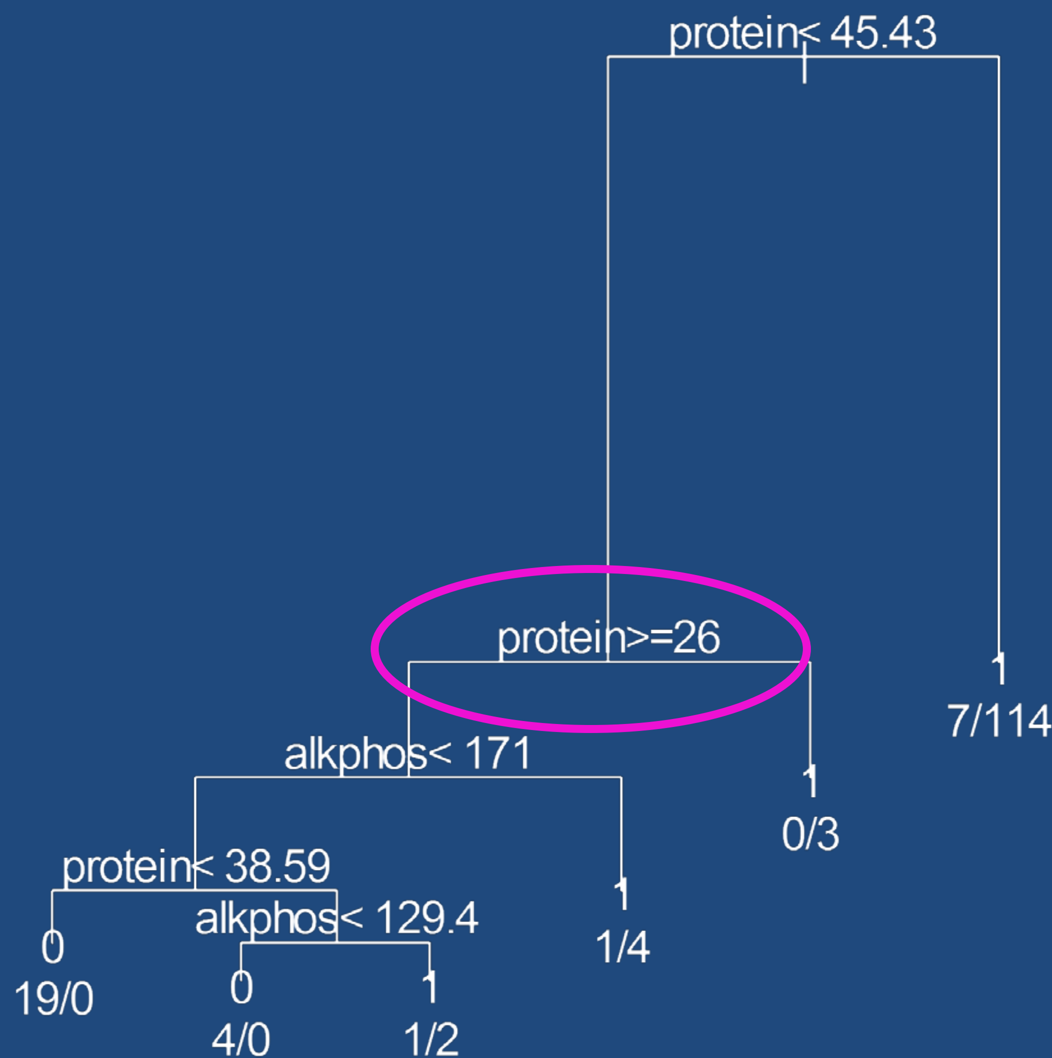
Classification tree (hepatitis)



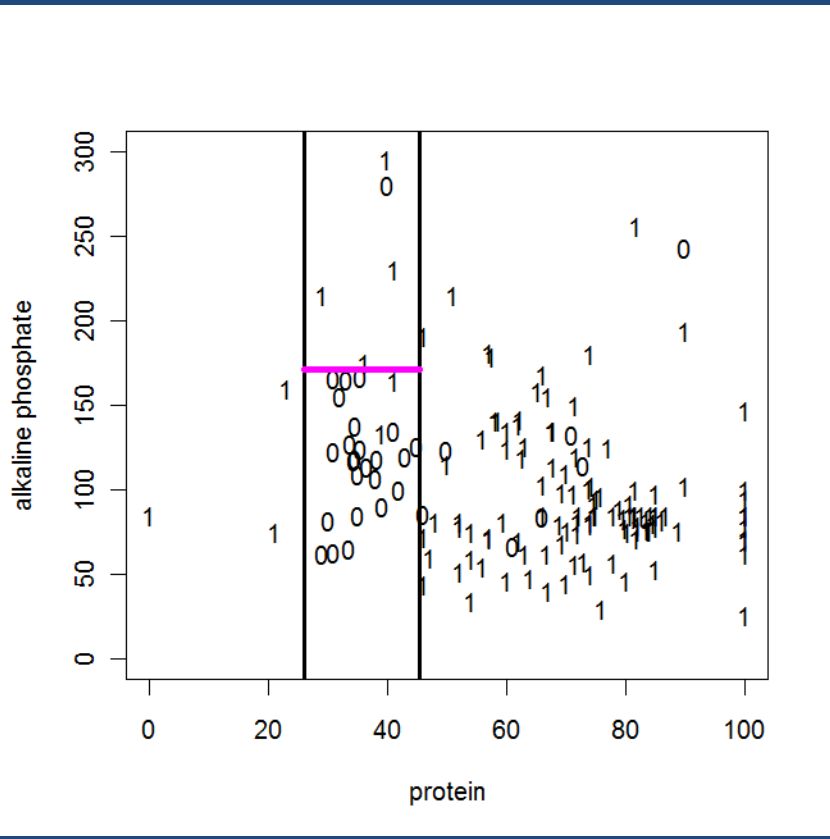


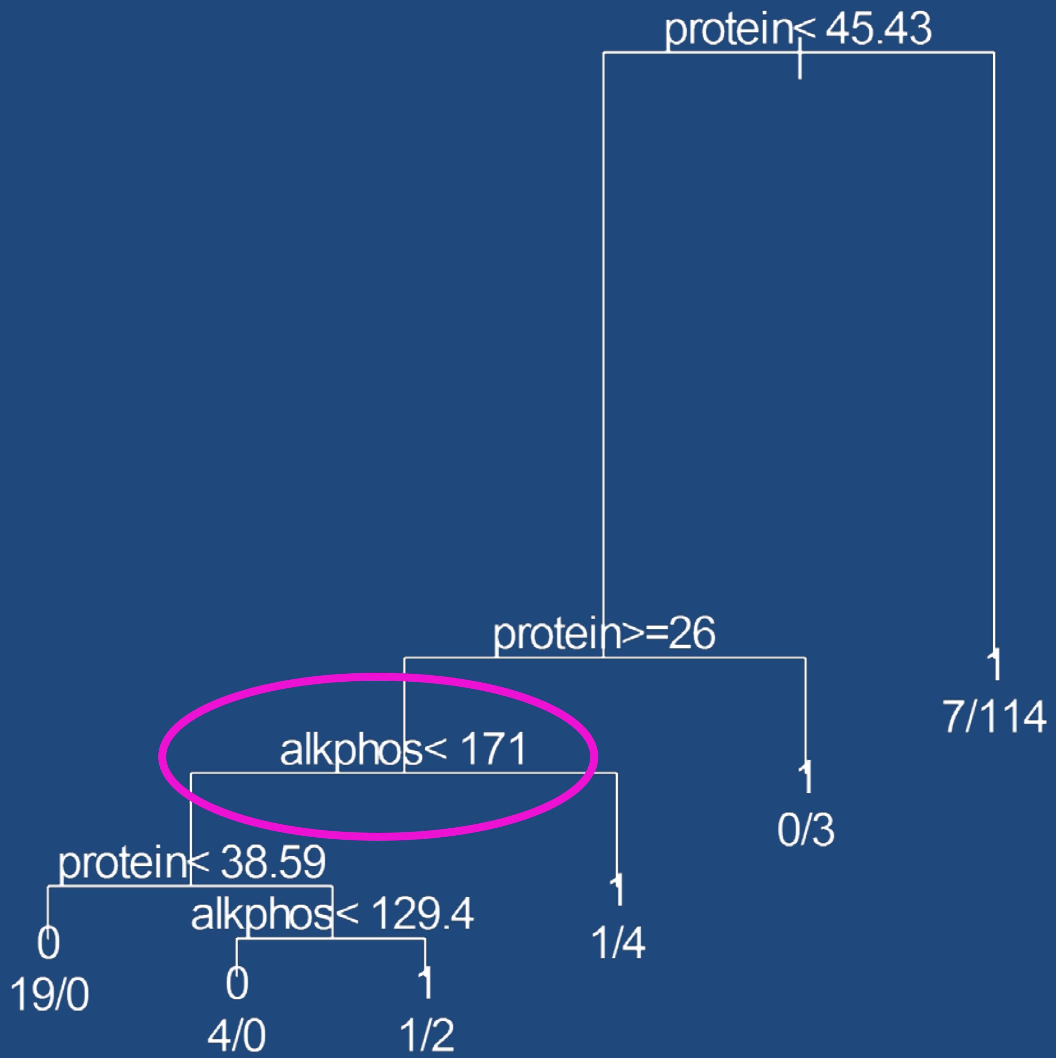
Classification tree (hepatitis)



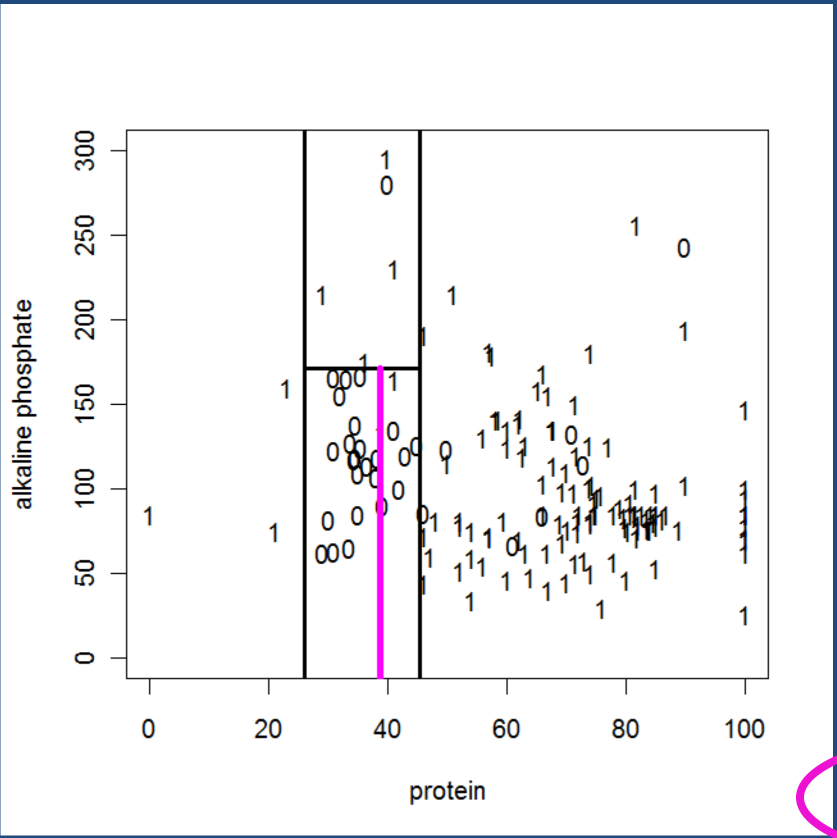


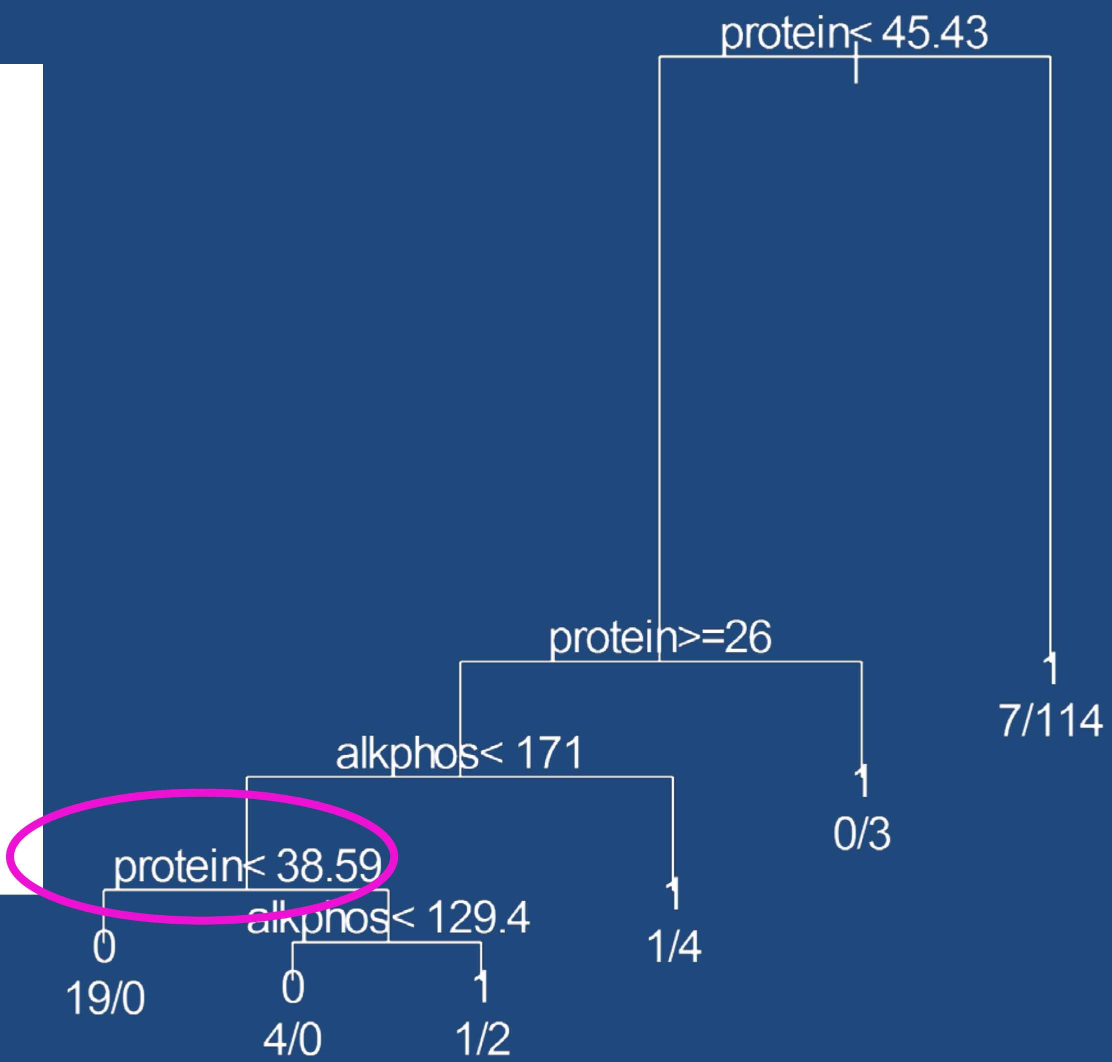
Classification tree (hepatitis)



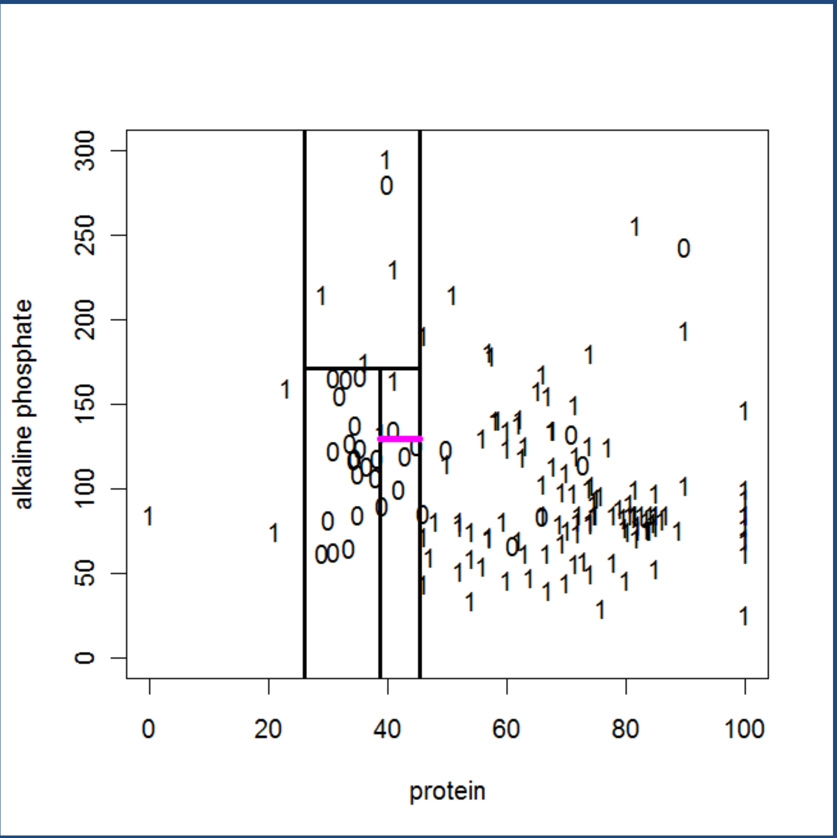


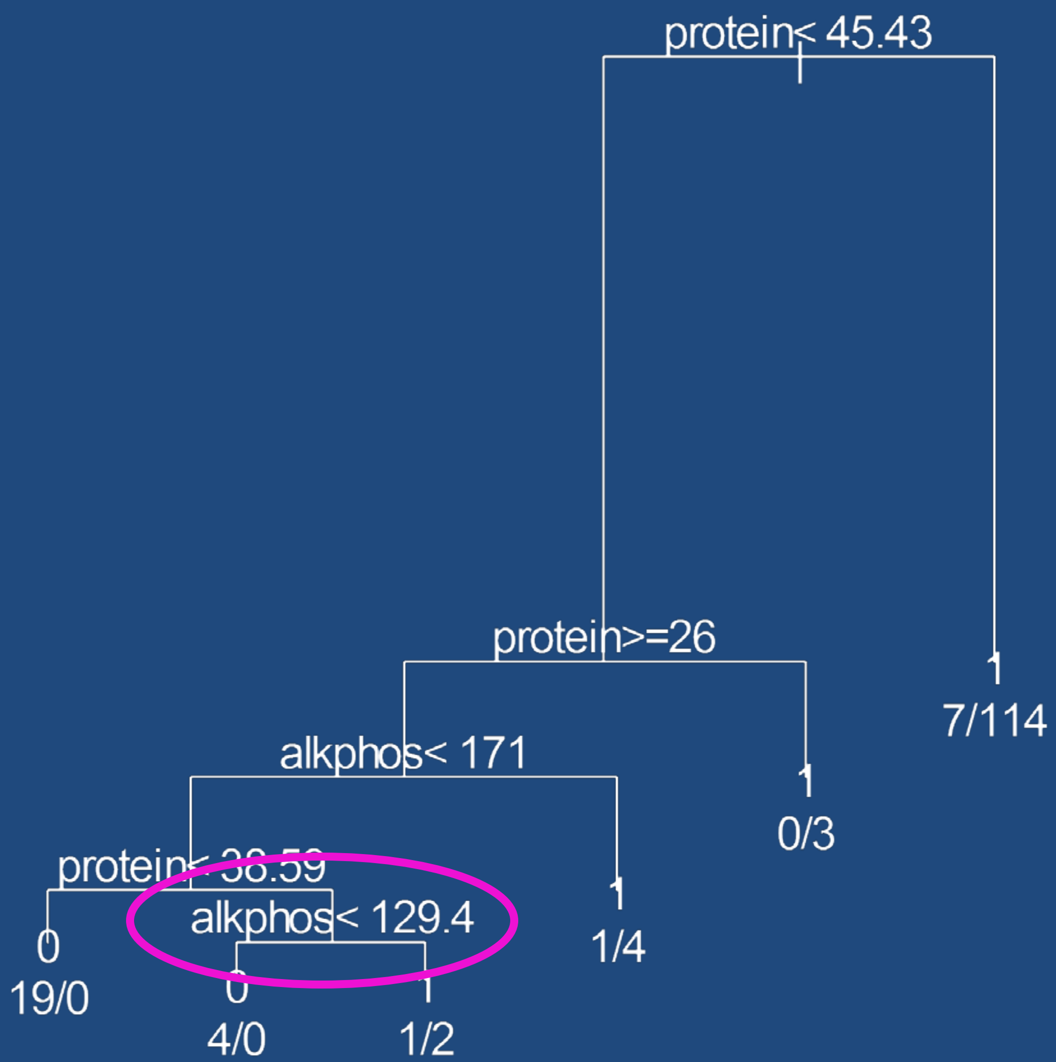
Classification tree (hepatitis)





Classification tree (hepatitis)





Pruning

- If the tree is too big, the lower branches are modeling noise in the data (overfitting)
- Grow the trees large and prune back unnecessary splits
- Pruning methods use some form of cross-validation
- May need to tune amount of pruning

Cavity Nesting Birds in the Uintahs

Red-naped sapsucker

Mountain chickadee

Northern flicker

Resubstitution – large tree

		Predicted Class 0	Predicted Class 1	Total
Actual Class	0	105	1	106
	1	0	107	107
	Total	105	108	213

Error rate = 1/213

Approximately 0.5%

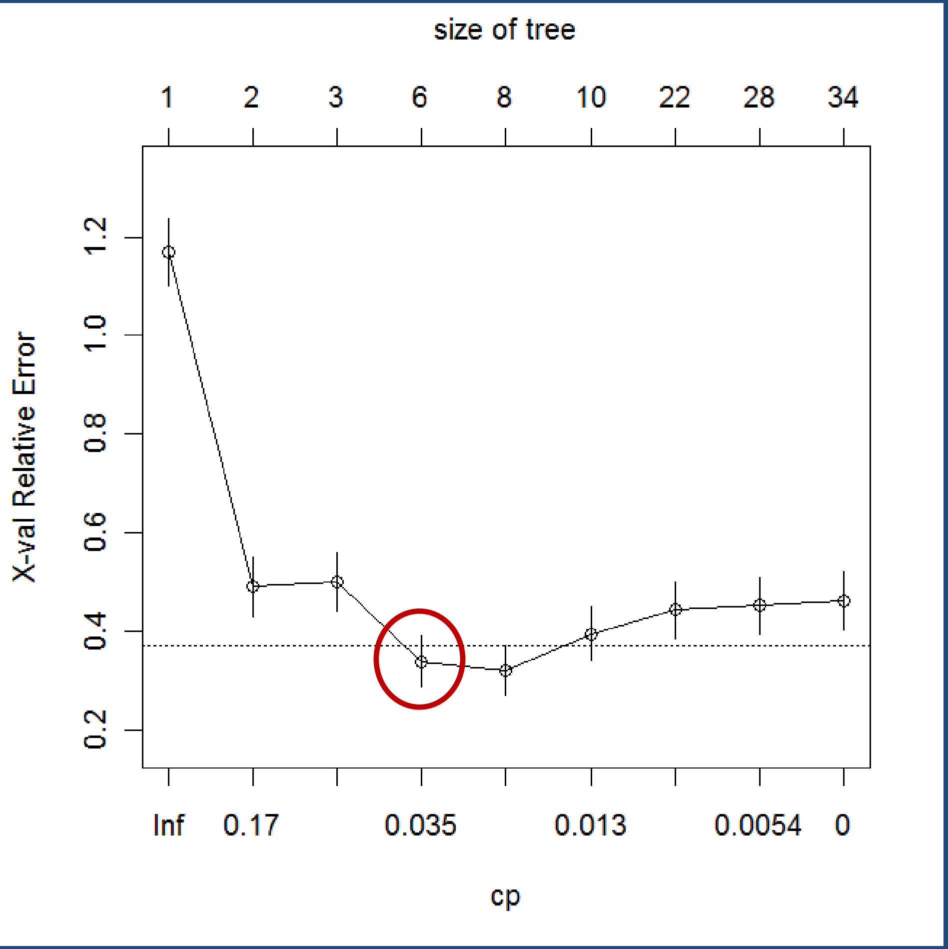
Cross-validation – large tree

		Predicted Class 0	Predicted Class 1	**Total**
Actual Class	0	83	23	106
	1	22	85	107
	Total	105	108	213

Error rate = 45/213

Approximately 21%

Cavity Nesting Birds in the Uintahs



Choose cp = .035

Resubstitution – pruned tree

Predicted Class

		0	1	Total
Actual Class	0	91	15	106
	1	14	93	107
	Total	105	108	213

Error rate = 29/213

Approximately 14%

Cross-validation – pruned tree

		Predicted Class 0	1	**Total**
Actual Class	0	86	20	106
	1	16	91	107
	Total	102	111	213

Error rate = 36/213

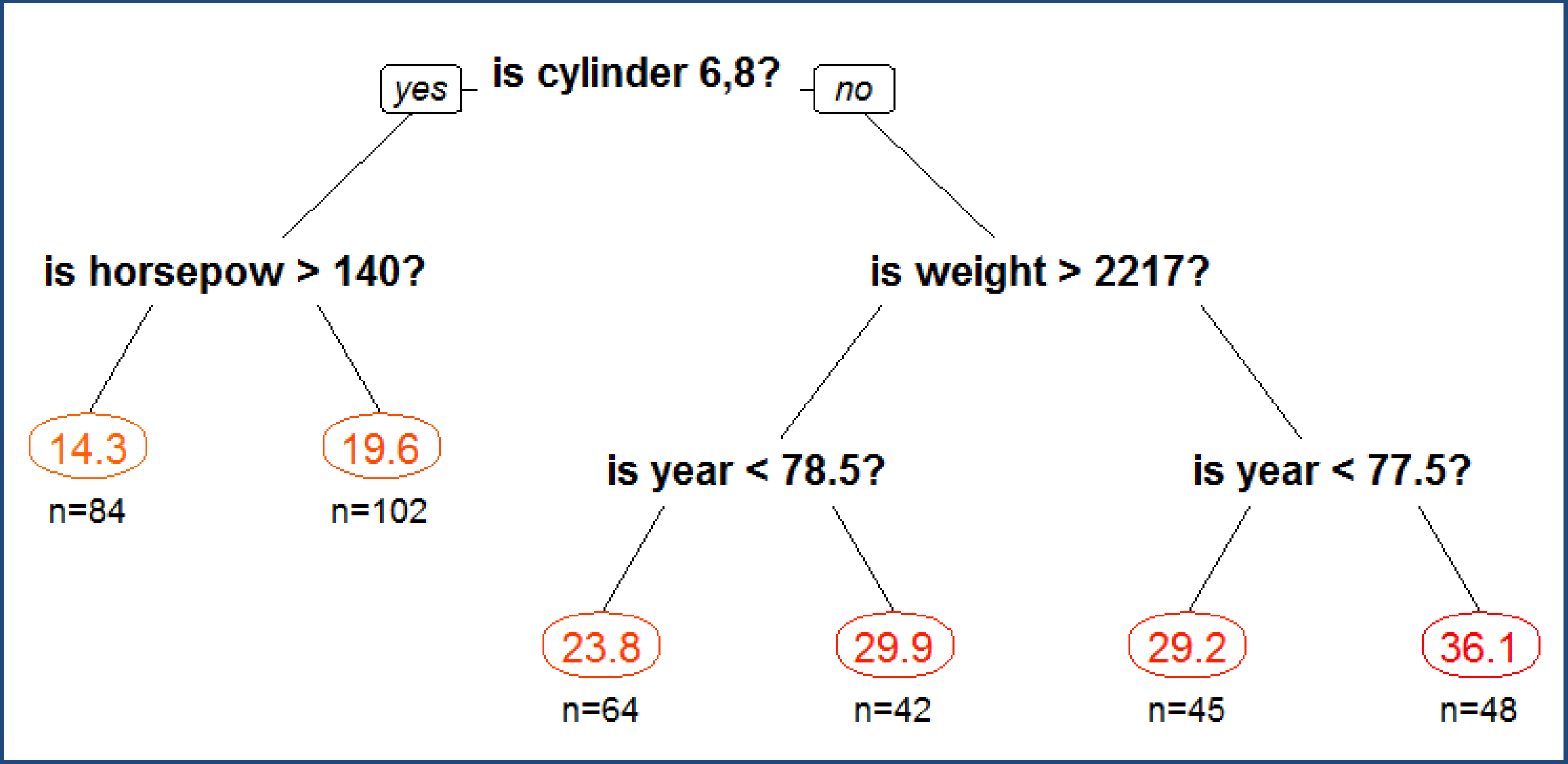
Approximately 17%

CART: Advantages over traditional statistical methods

- No formal distributional assumptions
- Can automatically fit highly non-linear interactions
- Automatic variable selection
- Handle missing values through surrogate variables
- Very easy to interpret if the tree is small
- The terminal nodes suggest a natural clustering

CART: Advantages over traditional statistical methods

- The picture can give valuable insights about which variables are important and where



CART: Advantages over other machine learning methods

- Same tool for regression and classification
- Handle categorical predictors naturally
- Quick to fit, even for large problems

CART: Disadvantages

- *Accuracy* – newer methods can have 30% lower error rates than CART
- *Instability* – if we change the data a little, the tree picture can change a lot

Random Forests!

Bagging

Breiman, Bagging Predictors, *Machine Learning*, 1996

Take a bootstrap sample from the data
Fit a classification or regression tree

Repeat

Combine by

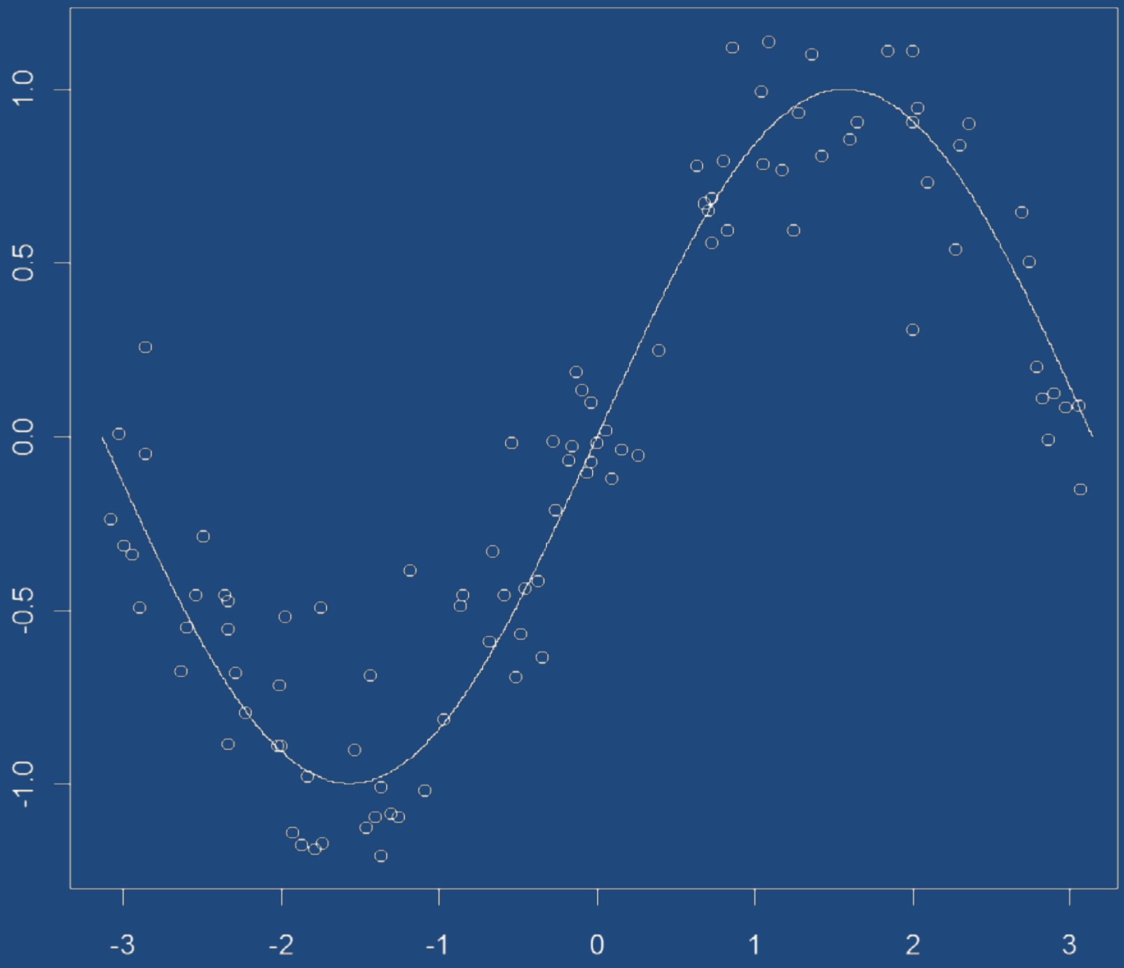
- voting (classification)
- averaging (regression)

Bagging CART

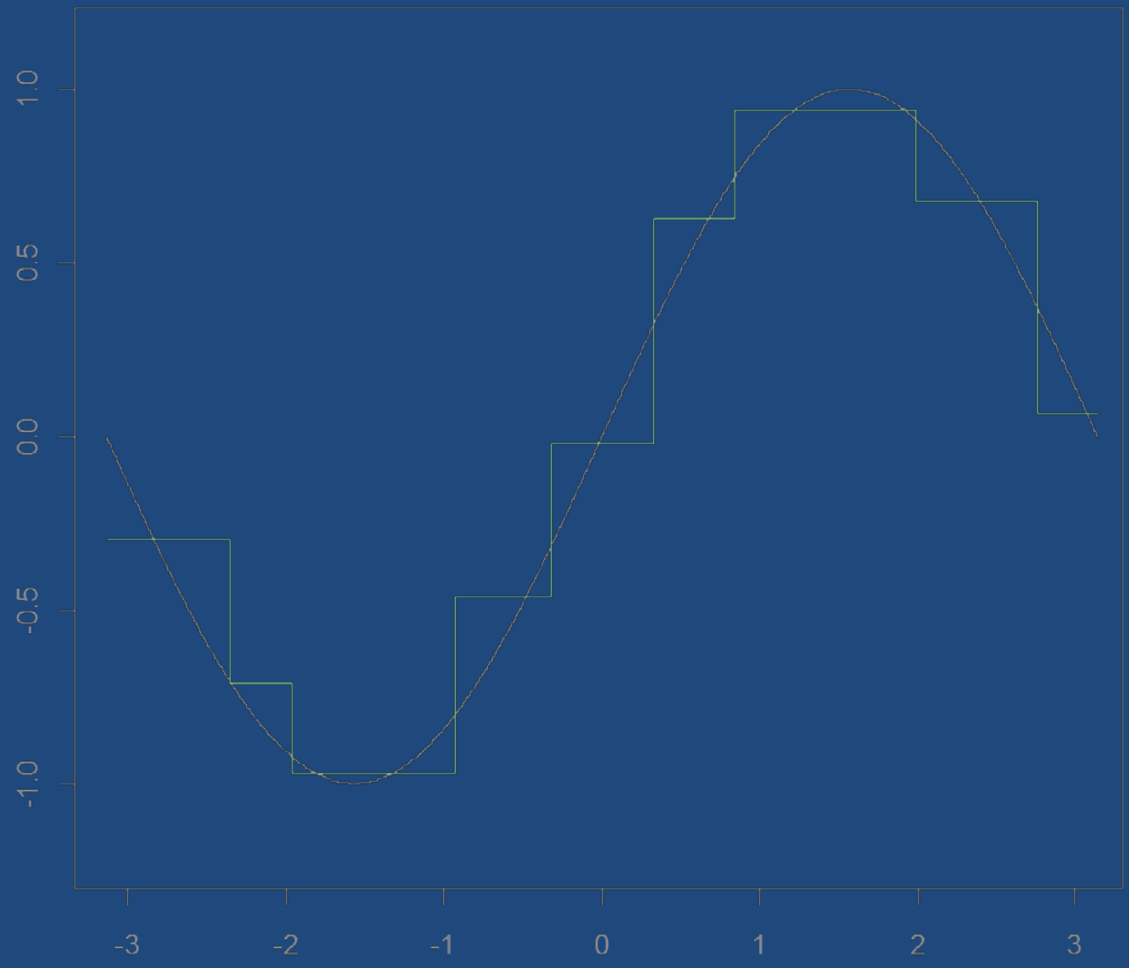
Dataset	Cases	Variables	Classes	CART	Bagged CART	Decrease %
Waveform	300	21	3	29.1	19.3	34
Breast cancer	699	9	2	5.9	3.7	37
Ionosphere	351	34	2	11.2	7.9	29
Diabetes	768	8	2	25.3	23.9	6
Glass	214	9	6	30.4	23.6	22

Leo Breiman (1996) “Bagging Predictors”, Machine Learning, 24, 123-140

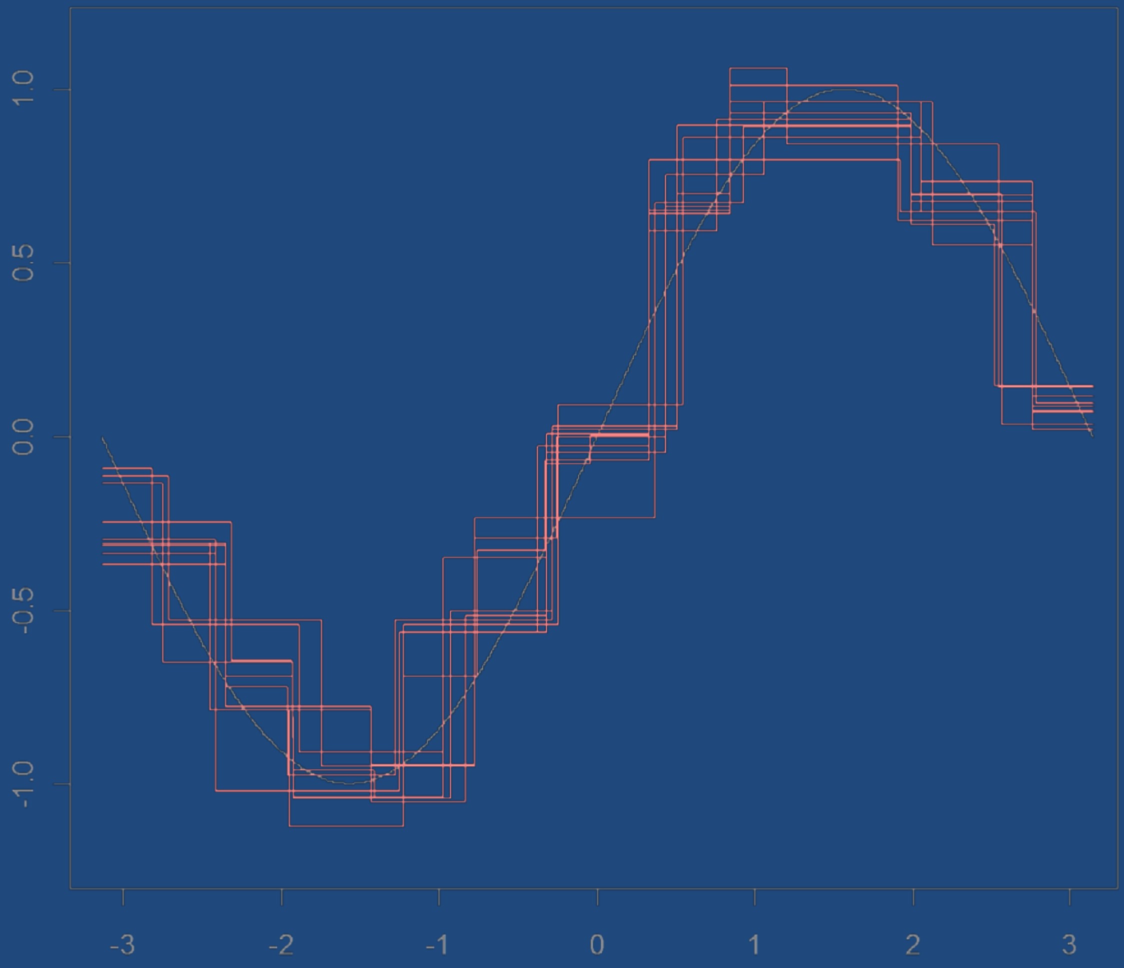
Data and Underlying Function



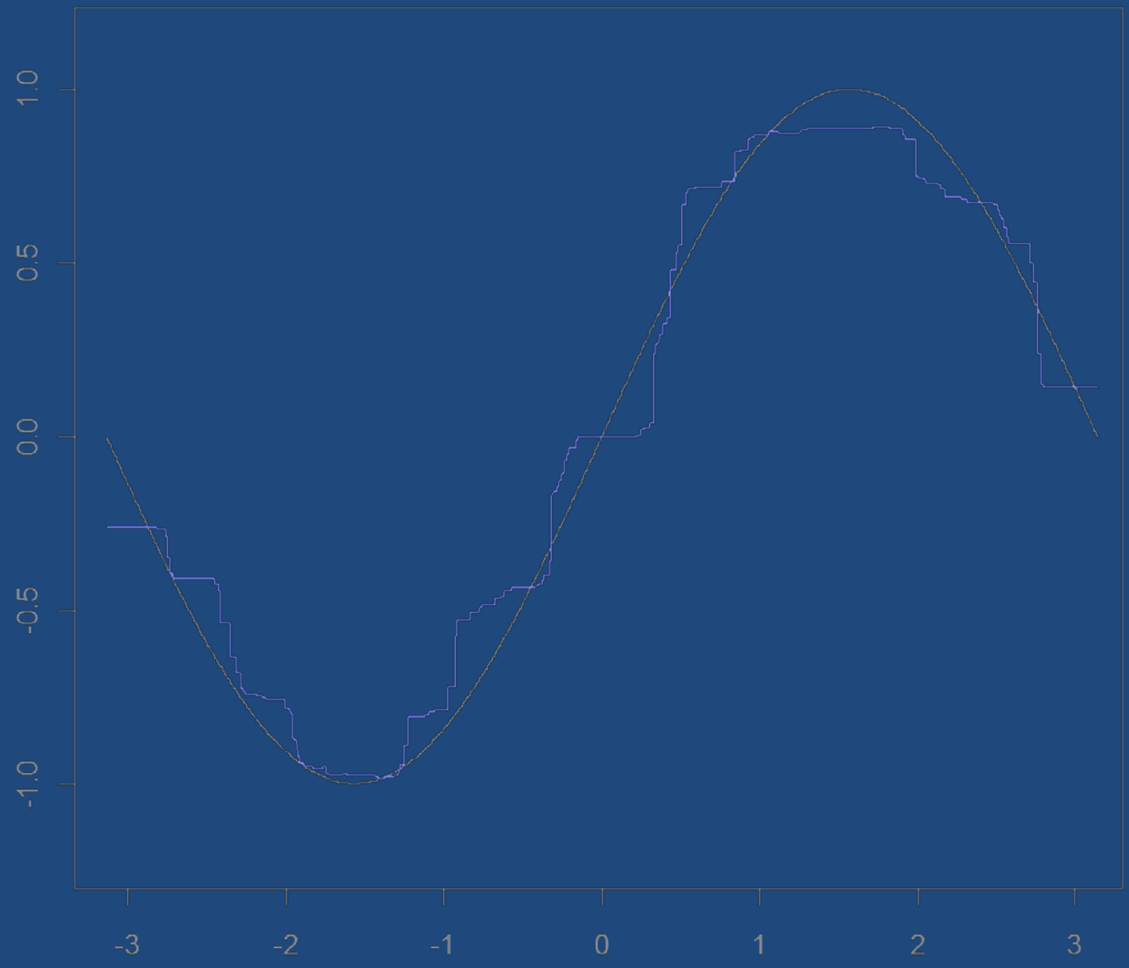
Single Regression Tree



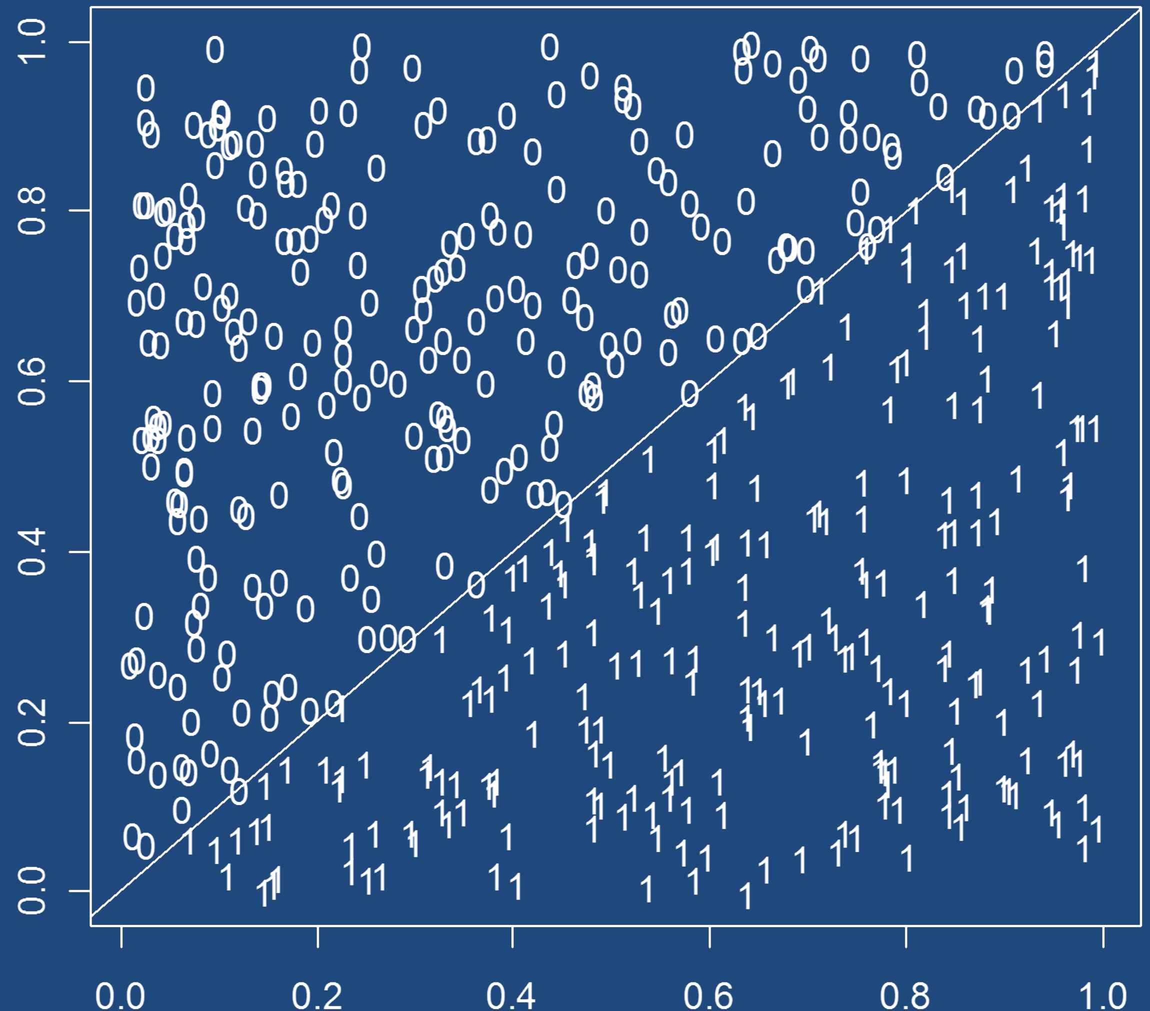
10 Regression Trees



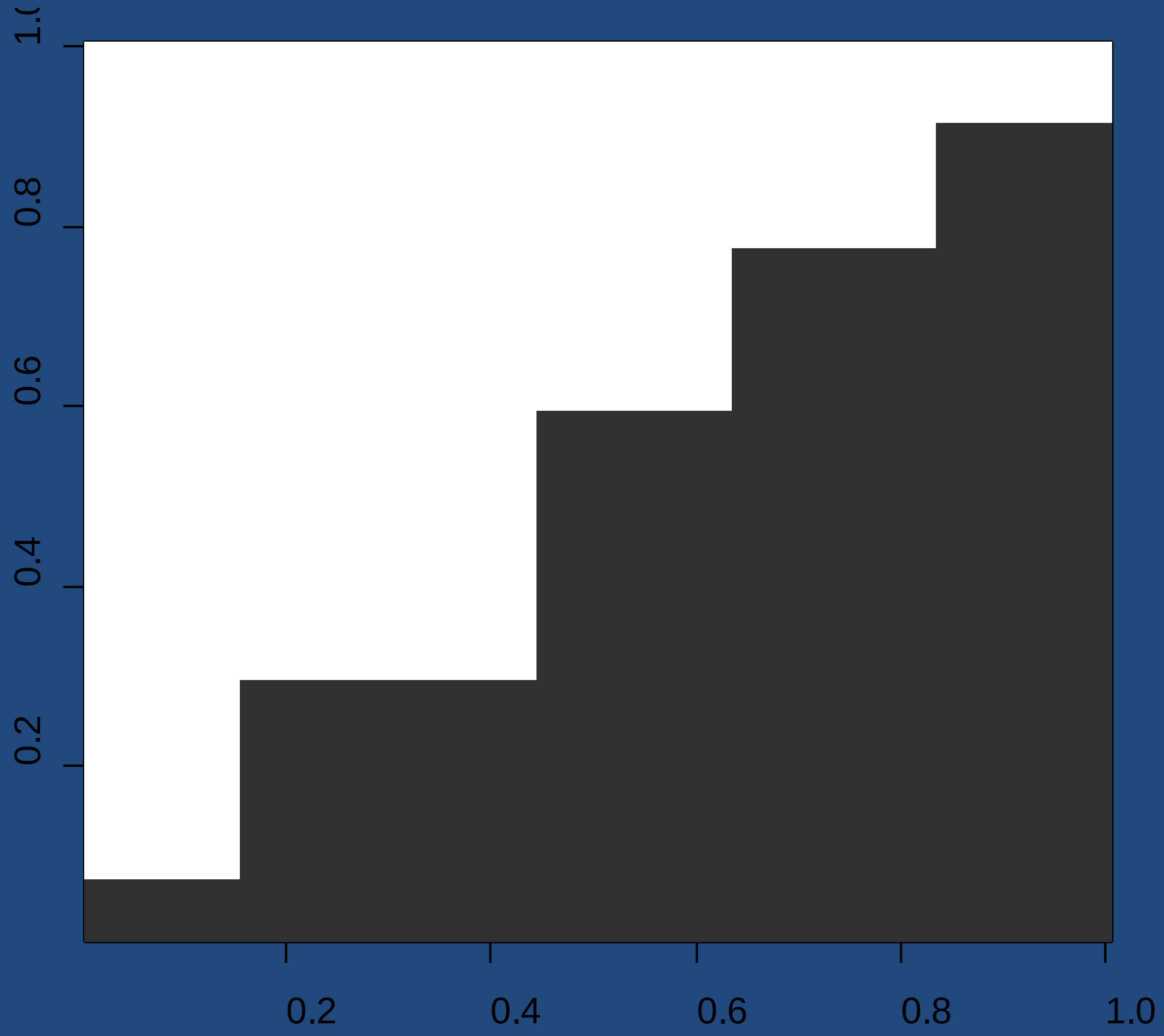
Average of 100 Regression Trees



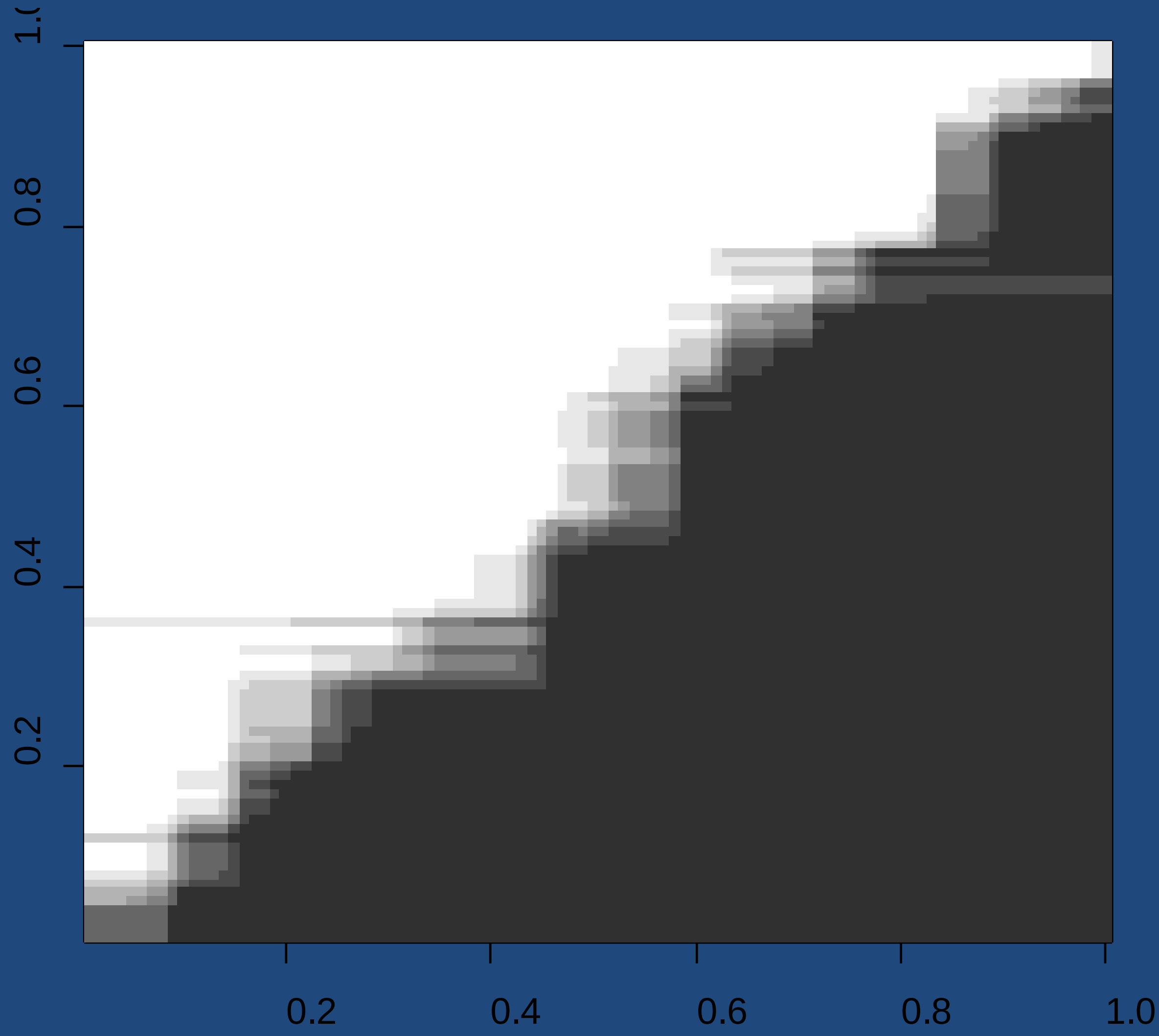
Hard problem for a single tree:



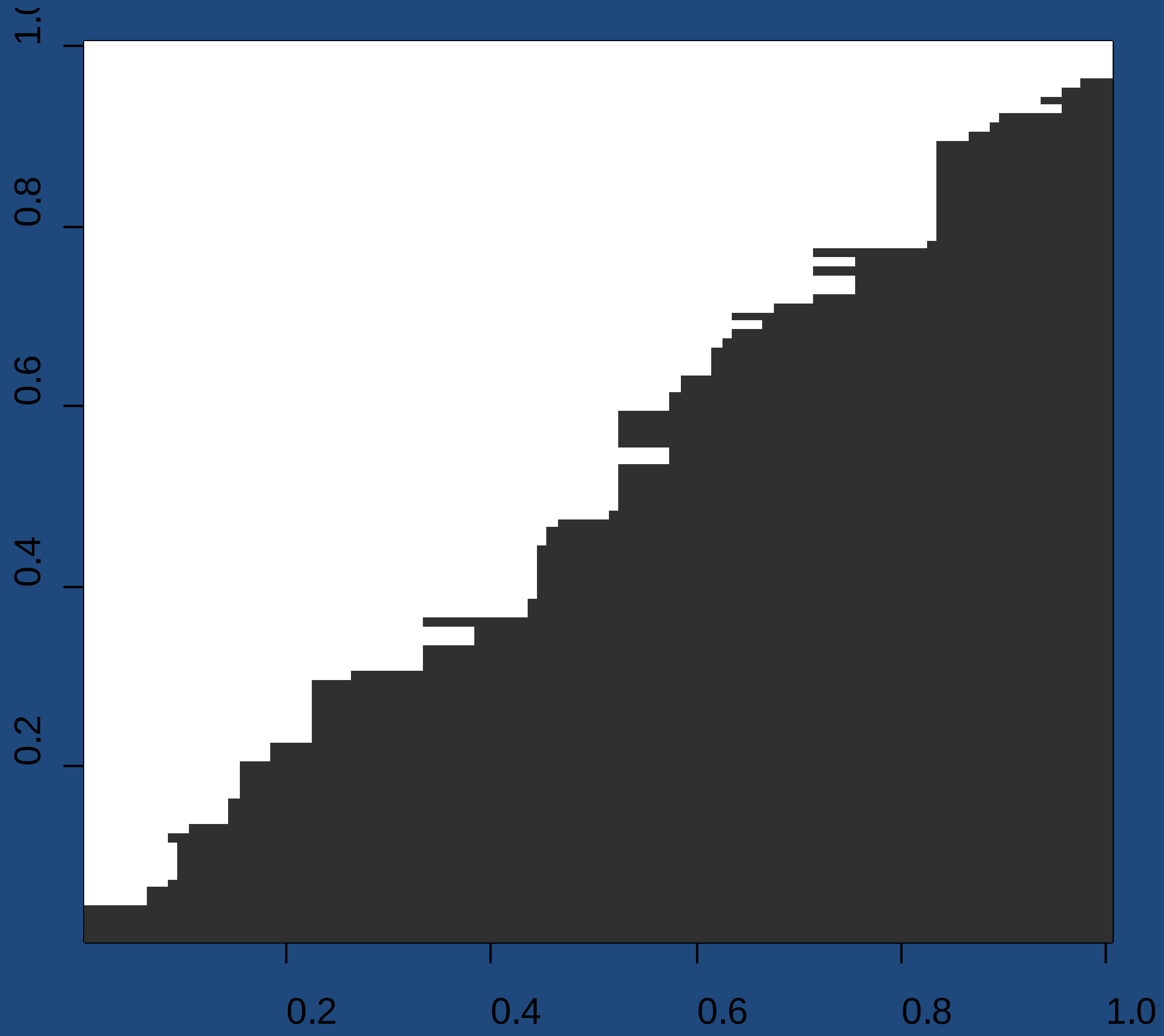
Single tree:



25 Averaged Trees:



25 Voted Trees:



Random Forests

Take a bootstrap sample from the data
Fit a classification or regression tree

Repeat

At each node:

1. Select ***m*** variables **at random** out of all ***M*** possible variables (independently at each node)
2. Find the best split on the selected ***m*** variables
3. Grow the trees big

Combine by

- voting (classification)
- averaging (regression)

Random Forests

Dataset	Cases	Variables	Classes	CART	Bagged CART	Random Forest
Waveform	300	21	3	29.1	19.3	17.2
Breast cancer	699	9	2	5.9	3.7	2.9
Ionosphere	351	34	2	11.2	7.9	7.1
Diabetes	768	8	2	25.3	23.9	24.2
Glass	214	9	6	30.4	23.6	20.6

Leo Breiman (2001) “Random Forests”, Machine Learning, 45, 5-32

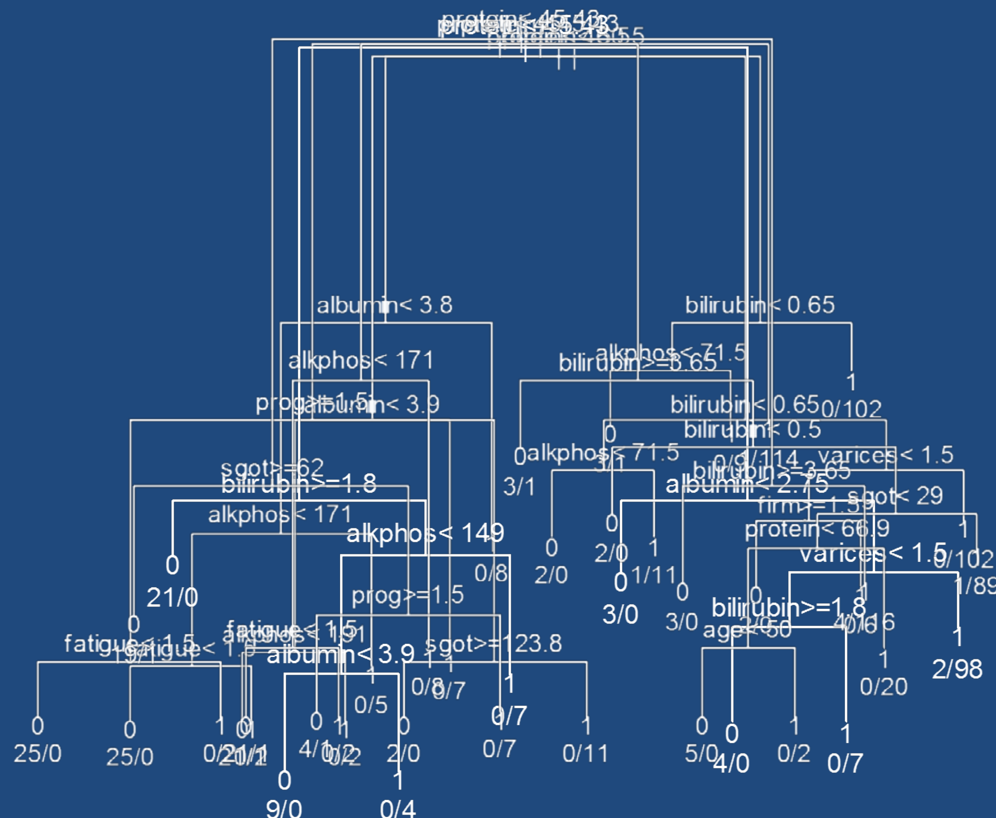
Random Forests

- Same idea for regression and classification YES!
- Handle categorical predictors naturally YES!
- Quick to fit, even for large problems YES!
- No formal distributional assumptions YES!
- Automatically fits highly non-linear interactions YES!
- Automatic variable selection YES! importance
- Handle missing values through proximities
- ~~Very easy to interpret if the tree is small~~ NO!
- ~~The terminal nodes suggest a natural clustering~~ NO!

Random Forests

~~The picture can give valuable insights into which variables are important and where~~

NO!



Random Forests

Improve on CART with respect to:

- *Accuracy* – Random Forests is competitive with the best known machine learning methods (but note the "no free lunch" theorem)
- *Instability* – if we change the data a little, the individual trees will change but the forest is more stable because it is a combination of many trees

The RF Predictor

- A case in the training data is *not* in the bootstrap sample for about one third of the trees (“oob”)
- Vote (or average) the predictions of *these trees* to give the RF predictor
- For new cases, vote (or average) *all* the trees to get the RF predictor

For example, suppose we fit 1000 trees, and a case is out-of-bag in 339 of them:

283 say “class 1”

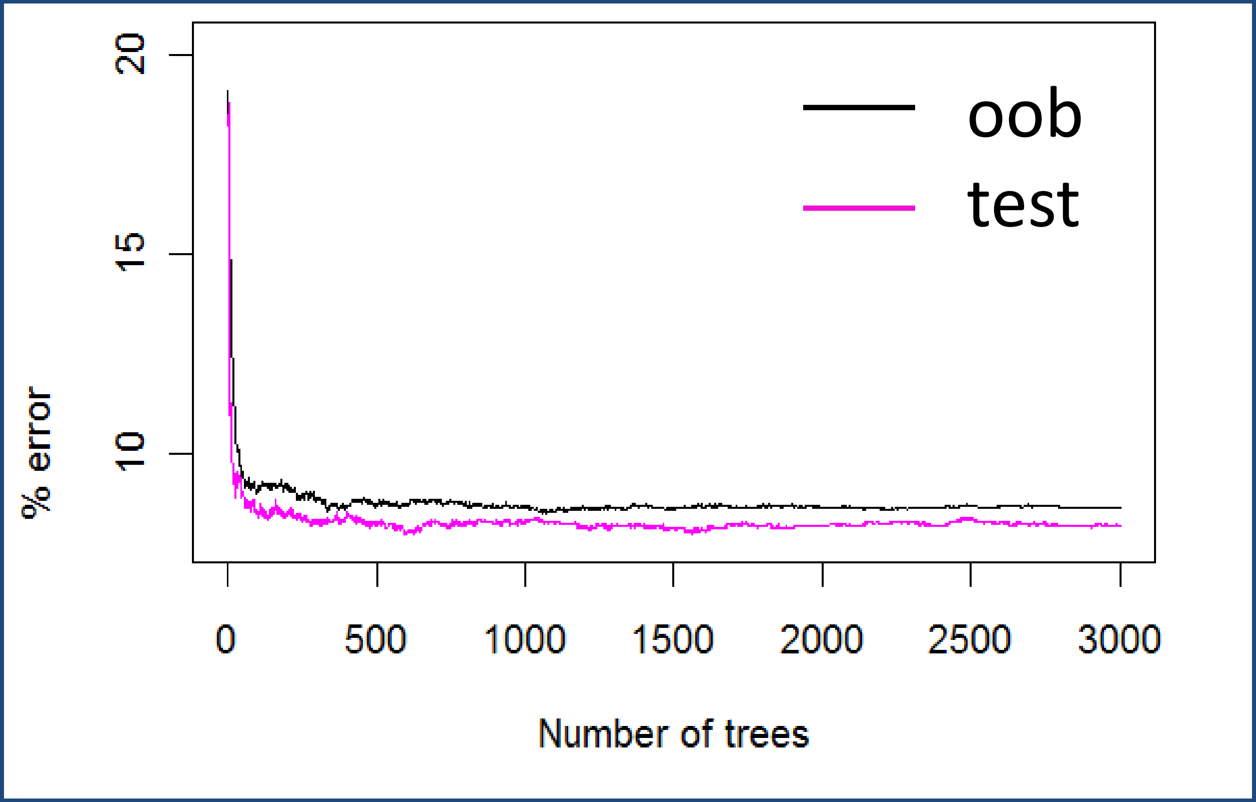
56 say “class 2”

The RF predictor is class 1

OOB Accuracy

- The oob accuracy is the accuracy of the RF predictor – it gives an estimate of test set accuracy (generalization error)
- The oob confusion matrix is the confusion matrix for the RF predictor (classification)

OOB accuracy



RF handles thousands of predictors

Ramón Díaz-Uriarte, Sara Alvarez de Andrés
Bioinformatics Unit, Spanish National Cancer Center
March, 2005 http://ligarto.org/rdiaz

Compared:

- SVM, linear kernel
- KNN/crossvalidation (Dudoit et al. JASA 2002)
- Shrunken Centroids (Tibshirani et al. PNAS 2002)
- Random forests

Given its performance, random forest and variable selection using random forest should probably become part of the standard tool-box of methods for the analysis of microarray data

Microarray Datasets

Data	M	N	# Classes
Leukemia	3051	38	2
Breast 2	4869	78	2
Breast 3	4869	96	3
NCI60	5244	61	8
Adenocar	9868	76	2
Brain	5597	42	5
Colon	2000	62	2
Lymphoma	4026	62	3
Prostate	6033	102	2
Srbct	2308	63	4

Microarray Error Rates

	SVM	KNN	DLDA	SC	RF	Rank
Leukemia	.014	.029	.020	.025	.051	**5**
Breast 2	.325	.337	.331	.324	.342	**5**
Breast 3	.380	.449	.370	.396	.351	**1**
NCI60	.256	.317	.286	.256	.252	**1**
Adenocar	.203	.174	.194	.177	.125	**1**
Brain	.138	.174	.183	.163	.154	**2**
Colon	.147	.152	.137	.123	.127	**2**
Lymphoma	.010	.008	.021	.028	.009	**2**
Prostate	.064	.100	.149	.088	.077	**2**
Srbct	.017	.023	.011	.012	.021	**4**
Mean	**.155**	**.176**	**.170**	**.159**	**.151**	

RF handles thousands of predictors

Add noise to some standard datasets and see how well Random Forests:

- predicts
- detects the important variables

RF error rates (%)

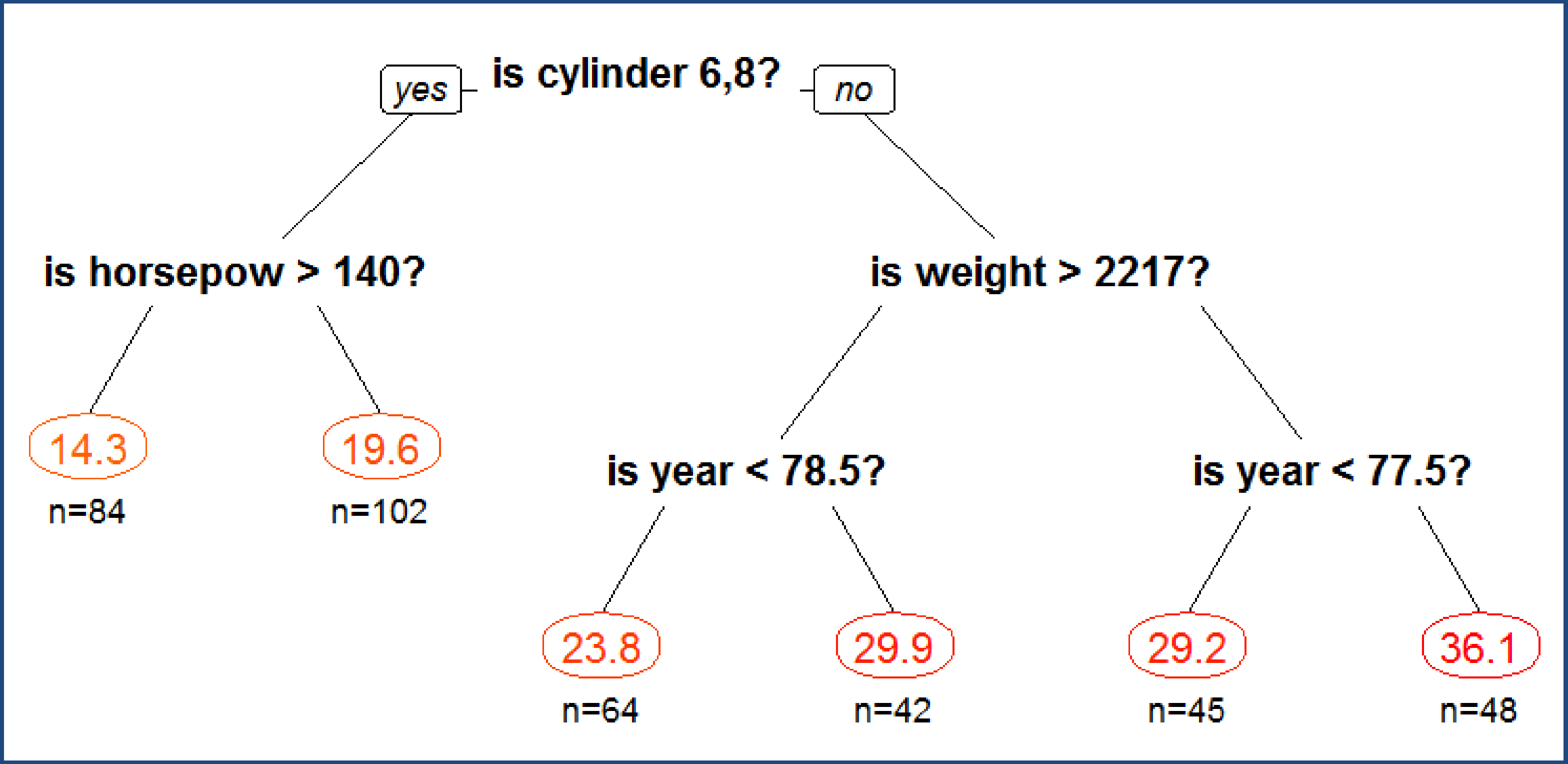
	No noise added	10 noise variables	100 noise variables
breast	3.1	2.9 (.94)	2.8 (0.91)
diabetes	23.5	23.8 (1.01)	25.8 (1.10)
ecoli	11.8	13.5 (1.14)	21.2 (1.80)
german	23.5	25.3 (1.07)	28.8 (1.22)
glass	20.4	25.9 (1.27)	37.0 (1.81)
image	1.9	2.1 (1.14)	4.1 (2.22)
iono	6.6	6.5 (0.99)	7.1 (1.07)
liver	25.7	31.0 (1.21)	40.8 (1.59)
sonar	15.2	17.1 (1.12)	21.3 (1.40)
soy	5.3	5.5 (1.06)	7.0 (1.33)
vehicle	25.5	25.0 (0.98)	28.7 (1.12)
votes	4.1	4.6 (1.12)	5.4 (1.33)
vowel	2.6	4.2 (1.59)	17.9 (6.77)

RF error rates (%)

	Number of noise variables				
	No noise added	10	100	1,000	10,000
breast	3.1	2.9	2.8	3.6	8.9
glass	20.4	25.9	37.0	51.4	61.7
votes	4.1	4.6	5.4	7.8	17.7

Local Variable Importance

In CART, variable importance is local:



Local Variable Importance

For each tree, look at the out-of-bag data:

- randomly permute the values of variable j
- pass these perturbed data down the tree

For case i and variable j find

error rate with variable j permuted	–	error rate with no permutation

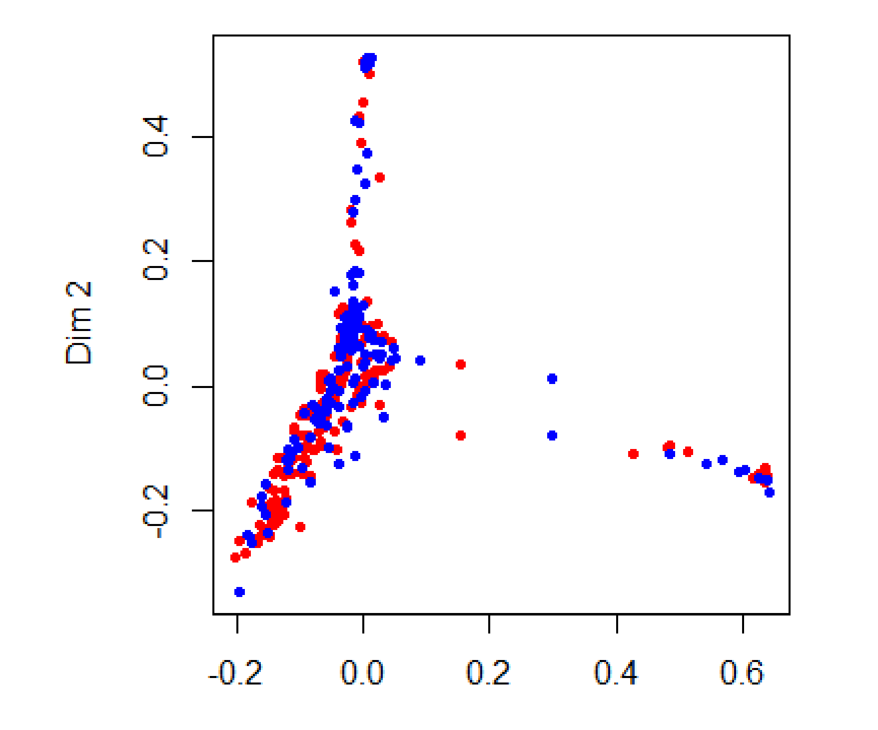
where the error rates are taken over all trees for which case i is out-of-bag

Local importance for a class 2 case

TREE	Original	Permute variable 1	…	Permute variable M
1	2	2	…	1
3	2	2	…	2
4	1	1	…	1
9	2	2	…	1
…	…	…	…	…
992	2	2	…	2
% Error	10%	11%	…	35%

Proximities

- Proximity of two cases is the proportion of the time that they end up in the same terminal node
- Multidimensional scaling or PCA can give a picture



Autism

Data courtesy of J.D.Odell and R. Torres, USU

154 subjects (308 chromosomes)

7 variables, all categorical (up to 30 categories)

2 classes:

- Normal, BLUE (69 subjects)
- Autistic, RED (85 subjects)

R demo

Random Forests Software

- Commercial version (academic discounts)
  www.salford-systems.com

- R package (Andy Liaw and Matthew Wiener)

References

Leo Breiman, Jerome Friedman, Richard Olshen, Charles Stone (1984) "Classification and Regression Trees" (Wadsworth).

Leo Breiman (1996) "Bagging Predictors" Machine Learning, 24, 123-140.

Leo Breiman (2001) "Random Forests" Machine Learning, 45, 5-32.

Trevor Hastie, Rob Tibshirani, Jerome Friedman (2009) "Statistical Learning" (Springer).