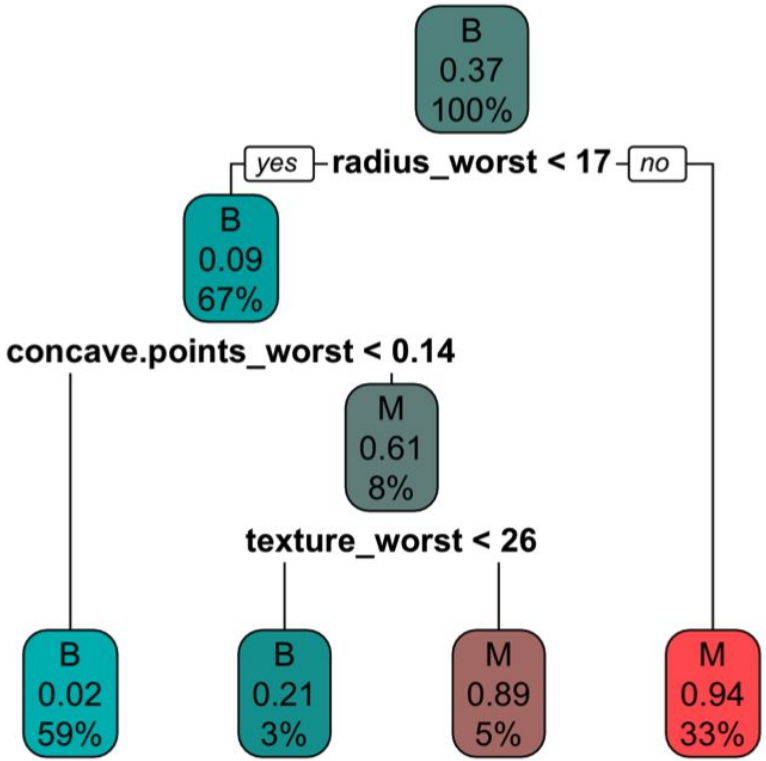
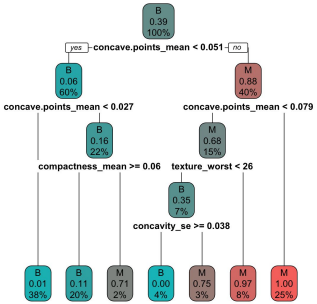
Chapter 13: Random Forests

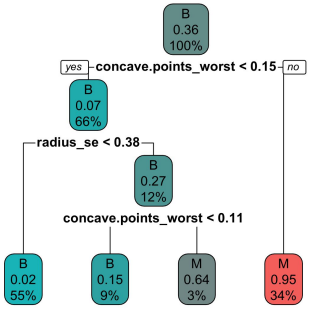
Modern Clinical Data Science
Chapter Guides
Bethany Percha, Instructor

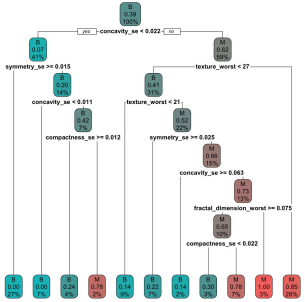
How to Use this Guide

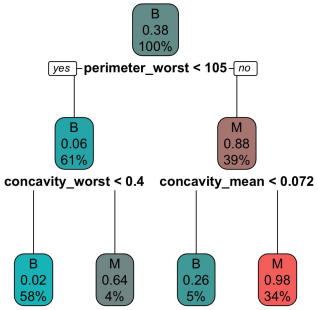
- Read the corresponding notes chapter first
- Try to answer the discussion questions on your own
- Listen to the chapter guide (should be 30 min, max) while following along in the notes

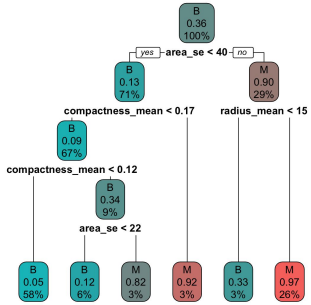

B
0.37
100%
yes
radius_worst < 17
no
B
0.09
67%
concave.points_worst < 0.14
M
0.61
8%
texture_worst < 26
B
0.02
59%
B
0.21
3%
M
0.89
5%
M
0.94
33%

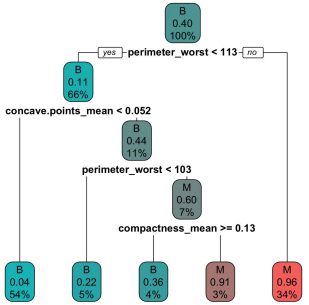

B 0.39 100%
yes concave.points_mean < 0.051 no
B 0.06 60%
M 0.88 40%
concave.points_mean < 0.027
concave.points_mean < 0.079
B 0.16 22%
M 0.68 15%
compactness_mean >= 0.06
texture_worst < 26
B 0.35 7%
concavity_se >= 0.038
B 0.01 38%
B 0.11 20%
M 0.71 2%
B 0.00 4%
M 0.75 3%
M 0.97 8%
M 1.00 25%

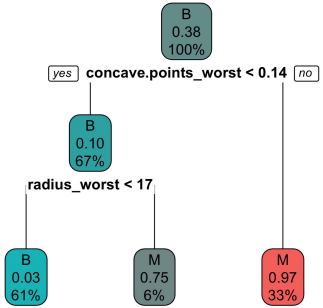

B 0.36 100%
yes concave.points_worst < 0.15 no
B 0.07 66%
radius_se < 0.38
B 0.27 12%
concave.points_worst < 0.11
B 0.02 55%
B 0.15 9%
M 0.64 3%
M 0.95 34%

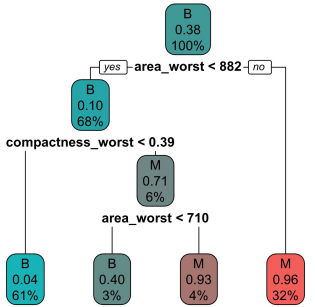

B 0.37 100%
yes concavity_se < 0.022 no
symmetry_se >= 0.015
texture_worst < 27
concavity_se < 0.011
texture_worst < 21
compactness_se >= 0.012
symmetry_se >= 0.025
concavity_se >= 0.063
fractal_dimension_worst >= 0.075
compactness_se < 0.022

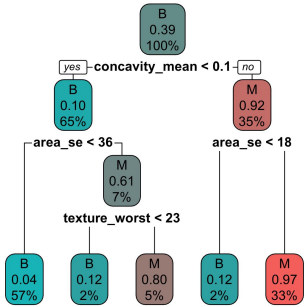

B 0.38 100%
yes perimeter_worst < 105 no
B 0.06 61%
M 0.88 39%
concavity_worst < 0.4
concavity_mean < 0.072
B 0.02 58%
M 0.64 4%
B 0.26 5%
M 0.98 34%

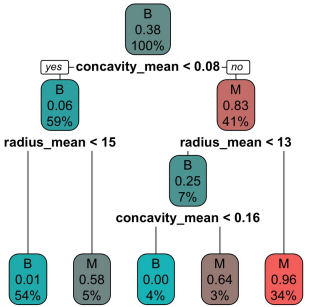

B 0.36 100%
yes area_se < 40 no
B 0.13 71%
M 0.90 29%
compactness_mean < 0.17
radius_mean < 15
B 0.09 67%
compactness_mean < 0.12
B 0.34 9%
area_se < 22
B 0.05 58%
B 0.12 6%
M 0.82 3%
M 0.92 3%
B 0.33 3%
M 0.97 26%

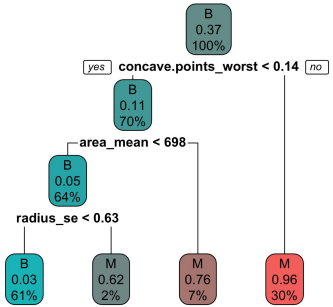

B 0.40 100%
yes perimeter_worst < 113 no
B 0.11 66%
concave.points_mean < 0.052
B 0.44 11%
perimeter_worst < 103
M 0.60 7%
compactness_mean >= 0.13
B 0.04 54%
B 0.22 5%
B 0.36 4%
M 0.91 3%
M 0.96 34%

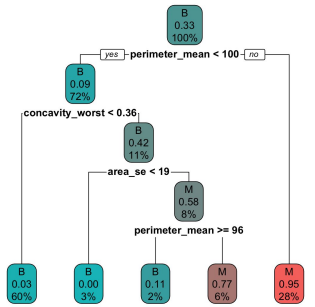

B 0.38 100%
yes concave.points_worst < 0.14 no
B 0.10 67%
radius_worst < 17
B 0.03 61%
M 0.75 6%
M 0.97 33%


B 0.38 100%
yes area_worst < 882 no
B 0.10 68%
compactness_worst < 0.39
M 0.71 6%
area_worst < 710
B 0.04 61%
B 0.40 3%
M 0.93 4%
M 0.96 32%


B 0.39 100%
yes concavity_mean < 0.1 no
B 0.10 65%
M 0.92 35%
area_se < 36
area_se < 18
M 0.61 7%
texture_worst < 23
B 0.04 57%
B 0.12 2%
M 0.80 5%
B 0.12 2%
M 0.97 33%


B 0.38 100%
yes concavity_mean < 0.08 no
B 0.06 59%
M 0.83 41%
radius_mean < 15
radius_mean < 13
B 0.25 7%
concavity_mean < 0.16
B 0.01 54%
M 0.58 5%
B 0.00 4%
M 0.64 3%
M 0.96 34%


B 0.37 100%
yes concave.points_worst < 0.14 no
B 0.11 70%
area_mean < 698
B 0.05 64%
radius_se < 0.63
B 0.03 61%
M 0.62 2%
M 0.76 7%
M 0.96 30%

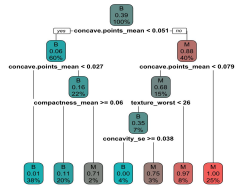

B 0.33 100%
yes perimeter_mean < 100 no
B 0.09 72%
concavity_worst < 0.36
B 0.42 11%
area_se < 19
M 0.58 8%
perimeter_mean >= 96
B 0.03 60%
B 0.00 3%
B 0.11 2%
M 0.77 6%
M 0.95 28%

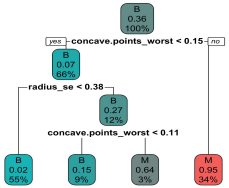
Assume we have a training dataset of N samples and P predictors. The basic strategy for building a random forest is as follows[1]:

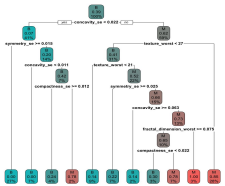
1. For $b = 1, \ldots, B$, where B is the desired number of trees:
   (a) Draw a bootstrap sample of size n from the training data.
   (b) Grow a tree, T_b, on the bootstrap sample by recursively repeating the following steps for each terminal node of the tree until the minimum node size, n_{min}, is reached:
      i. Select p predictors at random.
      ii. Pick the best predictor and split point.
      iii. Split the node into two child nodes.
2. Output the ensemble of trees, $T_1, \ldots, T_B$.

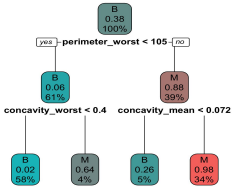
Question 13.8

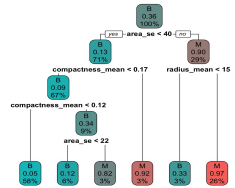
For an ensemble to be more accurate than any of its individual members, the learners comprising the ensemble must be *accurate* and *diverse*. Accuracy means that the learners must perform better than random on their designated task. Diversity means that the classifiers must make different errors on new data points. How do the parameters n and p impact diversity? How does the way the trees are trained ensure accuracy?

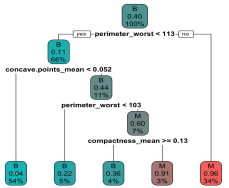


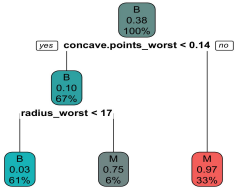


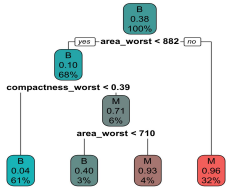


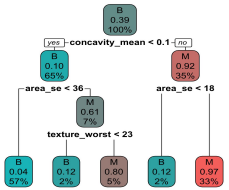


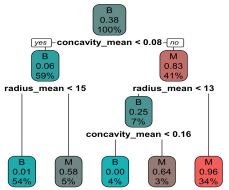


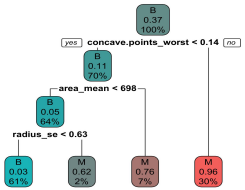


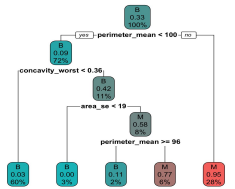












Advantages:

- Non-parametric
- Work for virtually any outcome
- Parallelizable
- Automatic feature selection

Question 13.1

If random forests are so great, why are linear, logistic, and Cox proportional hazards regression models still the standard approaches to predicting continuous, categorical, and survival outcomes in clinical research? What do you think are some of the main drawbacks of random forests and other ensemble methods?

Question 13.2

This algorithm leaves us with many choices. We must choose B, the number of trees; n, the size of each bootstrap sample; p, the number of predictors evaluated for each split; and n_{min}, the minimum node size. What impact does each parameter choice have on the properties of the forest, both in terms of the individual trees and the forest as a whole?

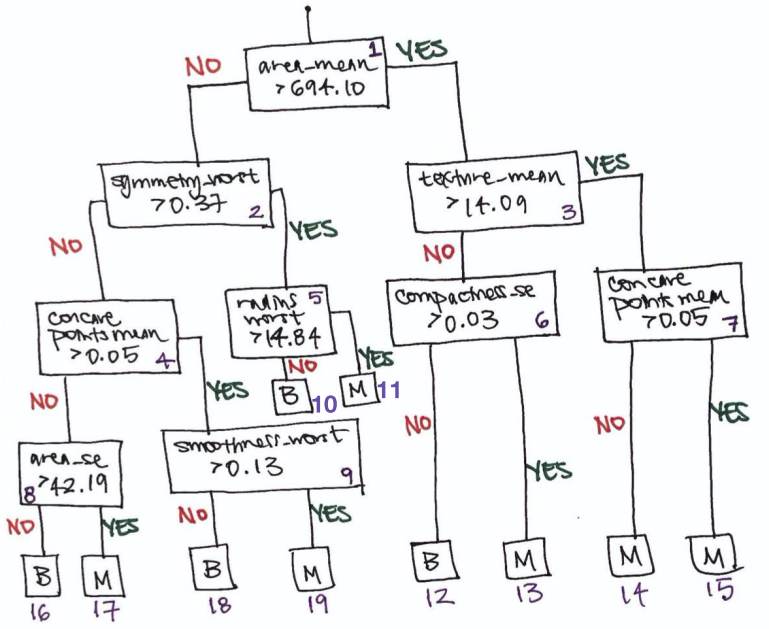
Question 13.3

What does the "best" predictor and split point refer to? How are these choices made for classification and regression models? Think back to our discussion in Chapter 7.

Question 13.4

Draw the two classification trees from the Wisconsin Breast Cancer Dataset random forest that are represented by these tables. Note: if a variable's value is less than the split point at a particular node, the training sample goes to the left.

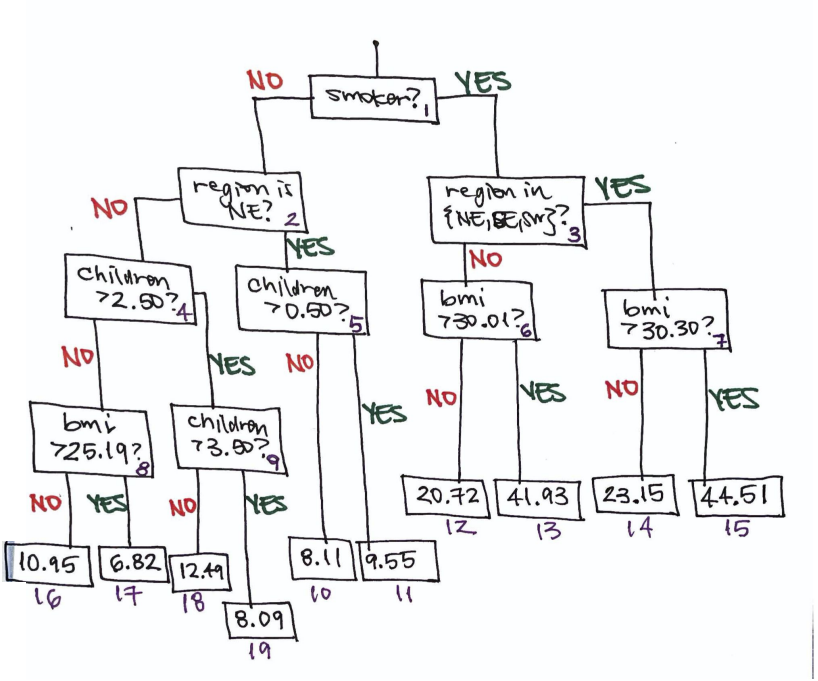
node_id	left_child	right_child	split_variable	split_point	prediction
1	2	3	area_mean	694.10	
2	4	5	symmetry_worst	0.37	
3	6	7	texture_mean	14.09	
4	8	9	concave.points_mean	0.05	
5	10	11	radius_worst	14.84	
6	12	13	compactness_se	0.03	
7	14	15	concave.points_mean	0.05	
8	16	17	area_se	42.19	
9	18	19	smoothness_worst	0.13	
10	0	0		0.00	B
11	0	0		0.00	M
12	0	0		0.00	B
13	0	0		0.00	M
14	0	0		0.00	M
15	0	0		0.00	M
16	0	0		0.00	B
17	0	0		0.00	M
18	0	0		0.00	B
19	0	0		0.00	M



Question 13.5

Draw the two regression trees from the Insurance Cost Dataset random forest that are represented by these tables. Note: if a variable's value is less than the split point at a particular node, the training sample goes to the left. For categorical variables, if the variable's value is one of the categories listed under "split point", the training sample goes to the right.

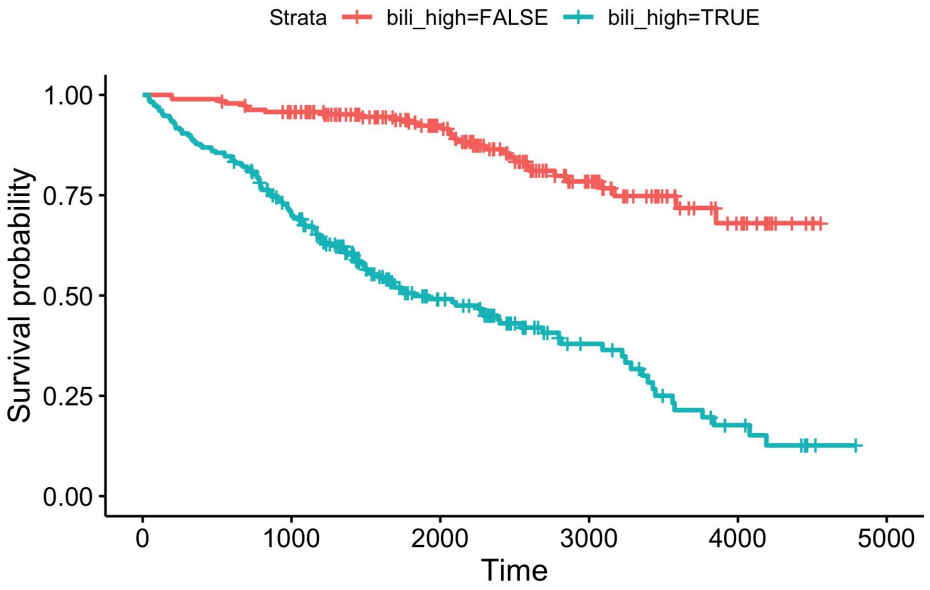
node_id	left_child	right_child	split_variable	split_point	prediction
1	2	3	smoker	yes	13.19
2	4	5	region	NE	8.65
3	6	7	region	NE, SE, SW	31.94
4	8	9	children	2.50	7.85
5	10	11	children	0.50	8.92
6	12	13	bmi	30.01	29.21
7	14	15	bmi	30.30	36.26
8	16	17	bmi	25.19	7.29
9	18	19	children	3.50	11.61
10	0	0		0.00	8.11
11	0	0		0.00	9.55
12	0	0		0.00	20.72
13	0	0		0.00	41.93
14	0	0		0.00	23.15
15	0	0		0.00	44.51
16	0	0		0.00	10.95
17	0	0		0.00	6.82
18	0	0		0.00	12.49
19	0	0		0.00	8.09

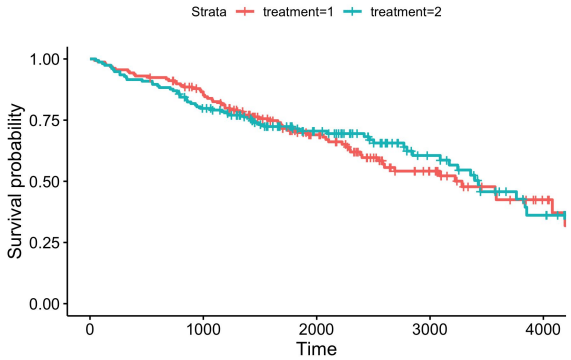


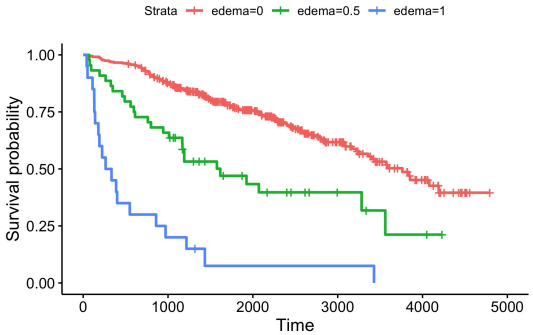
Question 13.6

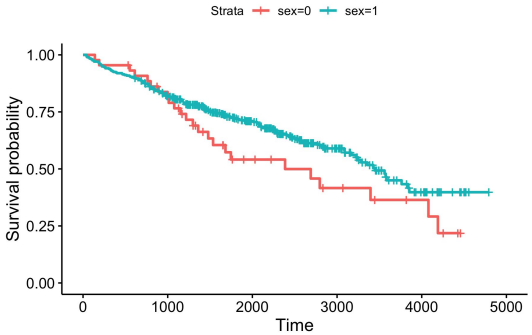
The following data come from a Mayo Clinic trial of the drug D-penicillamine for primary biliary cirrhosis (PBC) of the liver. The trial was conducted between 1974 and 1984. The data shown here are for 418 patients who completed the trial. The dataset comes from the `survival` package in R and contains information on 17 predictors, as well as the follow-up time and outcome (death or censoring) for each patient. Here are Kaplan-Meier curves (see Chapter 11) for four predictors, one of which (bilirubin) I manually binarized. Bilirubin is considered high if it is greater than 1.2 mg/dL.

Say you wanted to build a decision tree to predict survival in PBC. You would want to choose splits for which survival looks very *different* on either side of the split. This is analogous to choosing splits that increase the purity of the outcome (for classification) or reduce the variance of the outcome (regression). Speculate on how you might build such a tree. We will discuss the process of constructing **random survival forests** in much greater detail after we've seen a bit more survival analysis.









Deterministic vs. Random Splitting

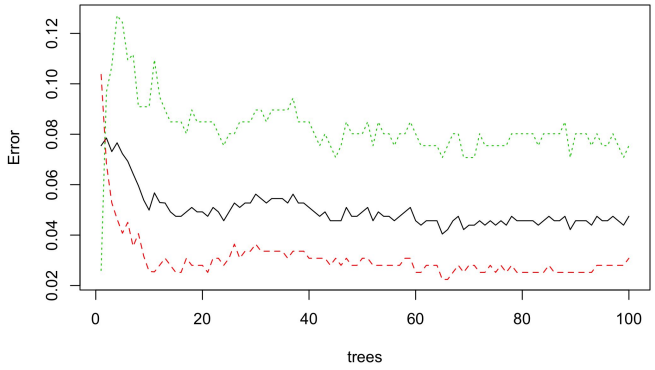
This (->) becomes an issue as the number of categories gets large.

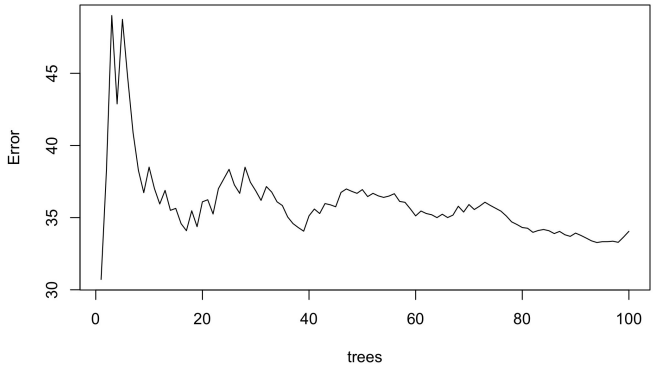
Group 1	Group 2
NE, SE, SW	NW
NE, SW, NW	SE
NE, SE, NW	SW
SE, SW, NW	NE
NE, SE	NW, SW
NE, SW	NW, SE
NE, NW	SE, SW

Question 13.7

A known issue with decision trees is their tendency to prefer to split on continuous predictors over discrete predictors. Why do you think this is? It can be avoided, in part, by using random splitting with a fixed number of possible splits.

Out-of-Bag (OOB) Error





Question 13.9

What does it mean that the green line is so much higher than the red line? What does this tell you about the relative rates of false positives and false negatives for this random forest?

Question 13.10

Here is the final **confusion matrix** for the Wisconsin Breast Cancer random forest.

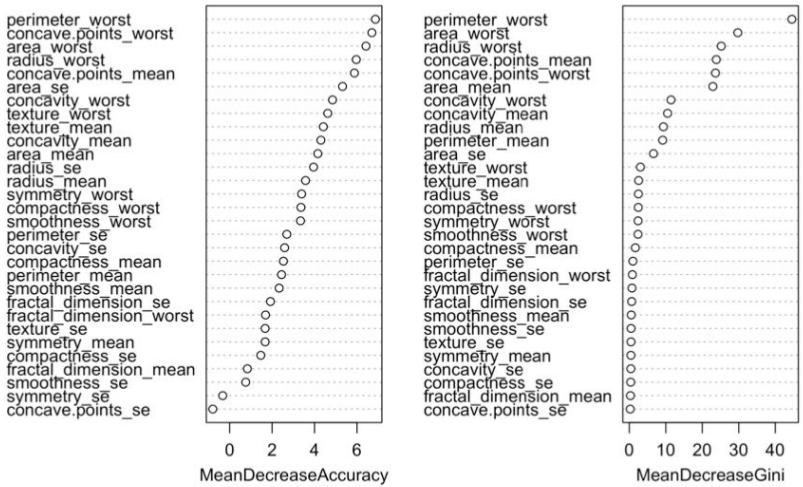
	B	M	class.error
B	346	11	0.03081232
M	16	196	0.07547170

It is probably more important to avoid false negatives (M tumors that are classified as B) than false positives. Speculate on ways in which you could force the forest to produce a lower rate of false negatives, even if it means increasing the number of false positives.

Variable Importance Measures

Question 13.11

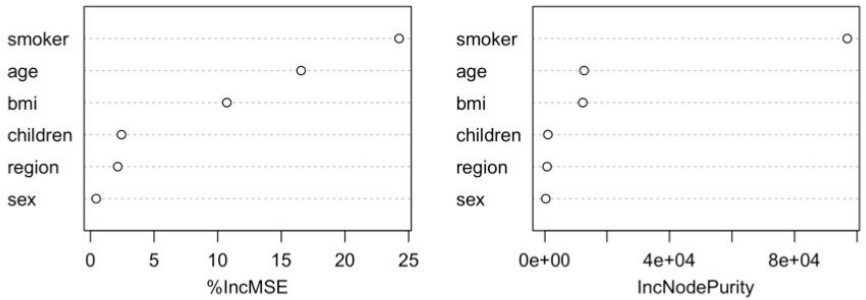
Here is a variable importance plot for the Wisconsin Breast Cancer classification forest. Which side is MDI and which is MDA? Which variables are most and least important?



Variable Importance Measures

Question 13.12

Here is a variable importance plot for the insurance cost dataset regression forest. Which side is MDI and which is MDA? Which variables are most and least important?



13.7 A Note on Software Packages

As we have seen in Chapter 7, there are many different ways to build and optimize decision trees. There are even more ways to build and optimize random forests. This chapter uses the `randomForest` R package by Andy Liaw, a faithful implementation of the original random forest implementation suggested by Breiman (2003), as well as `randomForestSRC`, a faster and more recent package by Ishwaran and Kogalur that provides a unified interface for random forest-based classification, regression, and survival analysis. The `scikit-learn` package in Python provides implementations of random forests for both classification and regression.