

Comprehension Check: Trees and Random Forests

Q1

1/1 point (graded)

Create a simple dataset where the outcome grows 0.75 units on average for every increase in a predictor, using this code:

```
library(rpart)
n <- 1000
sigma <- 0.25
set.seed(1) #set.seed(1, sample.kind = "Rounding") if using R 3.6 or later
x <- rnorm(n, 0, 1)
y <- 0.75 * x + rnorm(n, 0, sigma)
dat <- data.frame(x = x, y = y)
```

Which code correctly uses `rpart` to fit a regression tree and saves the result to `fit`?

- ☐ `fit <- rpart(y ~ .)`
- ☐ `fit <- rpart(y, ., data = dat)`
- ☐ `fit <- rpart(x ~ ., data = dat)`
- ☒ `fit <- rpart(y ~ ., data = dat)`



Submit

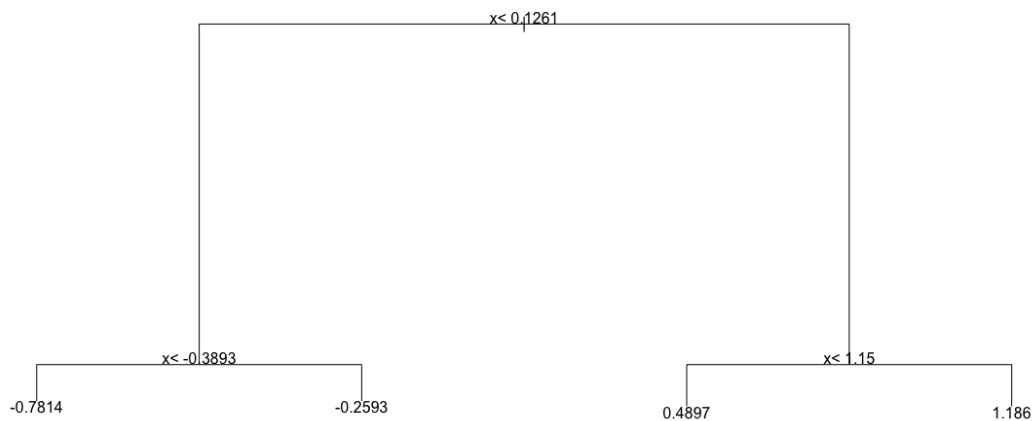
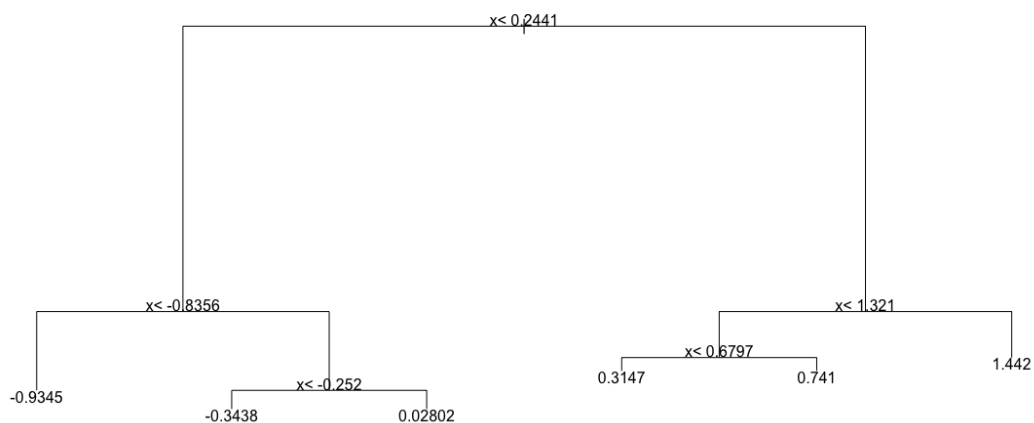
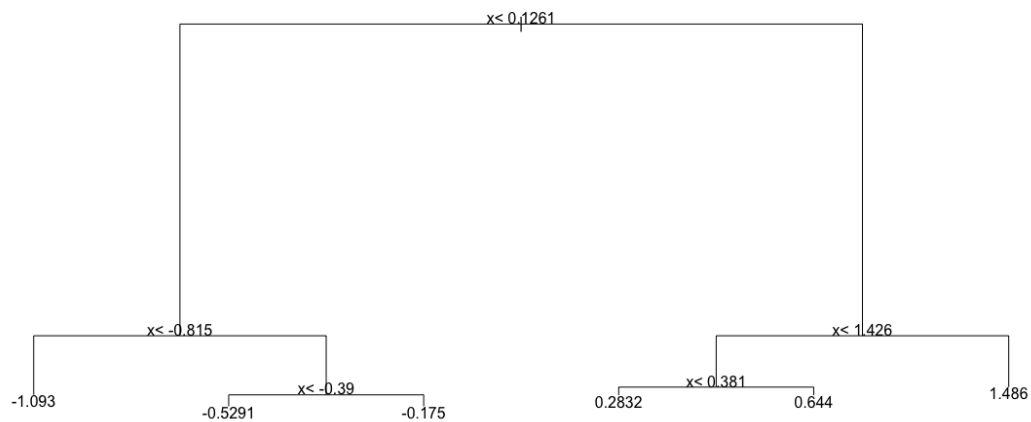
You have used 1 of 2 attempts

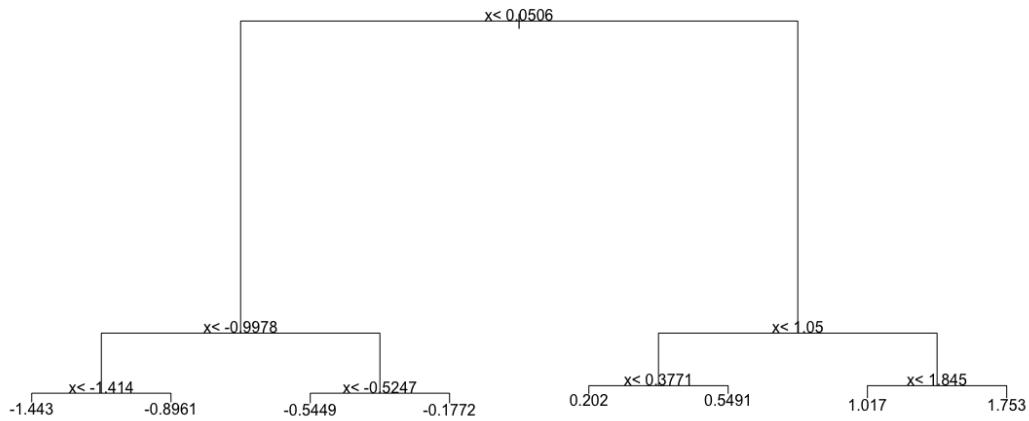
i Answers are displayed within the problem

Q2

1/1 point (graded)

Which of the following plots has the same tree shape obtained in Q1?





Explanation

The plot can be made using the following code:

```
plot(fit)
text(fit)
```

Submit

You have used 1 of 2 attempts

i Answers are displayed within the problem

Q3

1/1 point (graded)

Below is most of the code to make a scatter plot of `y` versus `x` along with the predicted values based on the fit.

```
dat %>%
  mutate(y_hat = predict(fit)) %>%
  ggplot() +
  geom_point(aes(x, y)) +
  #BLANK
```

Which line of code should be used to replace #BLANK in the code above?



`geom_step(aes(x, y_hat), col=2)`

☐ `geom_smooth(aes(y_hat, x), col=2)`

☐ `geom_quantile(aes(x, y_hat), col=2)`

☐ `geom_step(aes(y_hat, x), col=2)`



Submit

You have used 1 of 2 attempts

i Answers are displayed within the problem

Q4

1/1 point (graded)

Now run Random Forests instead of a regression tree using `randomForest` from the `__randomForest__` package, and remake the scatterplot with the prediction line. Part of the code is provided for you below.

```
library(randomForest)
fit <- #BLANK
dat %>%
  mutate(y_hat = predict(fit)) %>%
  ggplot() +
  geom_point(aes(x, y)) +
  geom_step(aes(x, y_hat), col = 2)
```

What code should replace #BLANK in the provided code?

☒ `randomForest(y ~ x, data = dat)`

☐ `randomForest(x ~ y, data = dat)`

☐ `randomForest(y ~ x, data = data)`

☐ `randomForest(x ~ y)`



Submit

You have used 1 of 2 attempts

Q5

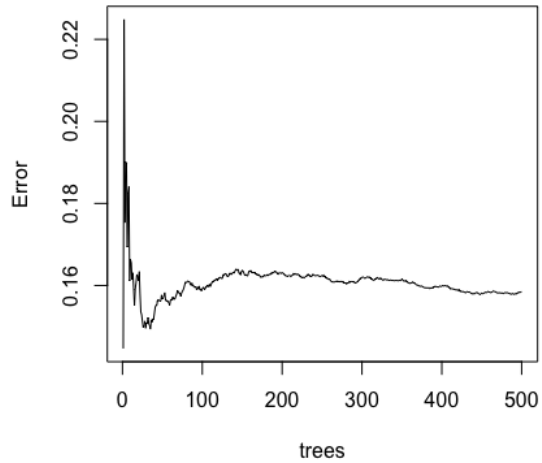
1/1 point (graded)

Use the `plot` function to see if the Random Forest from Q4 has converged or if we need more trees.

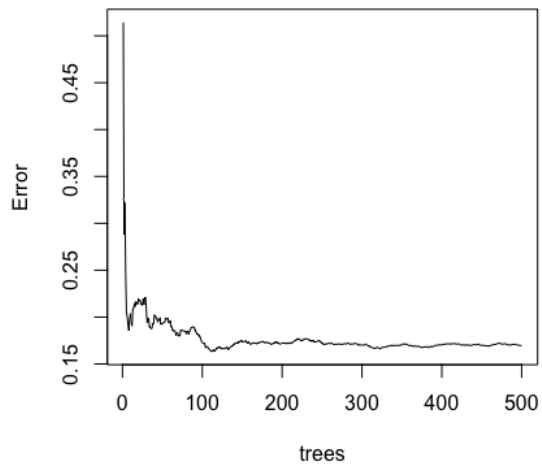
Which of these graphs is produced by plotting the random forest?



fit

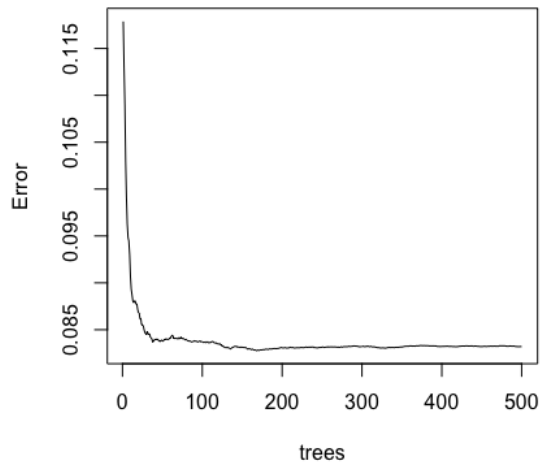


fit

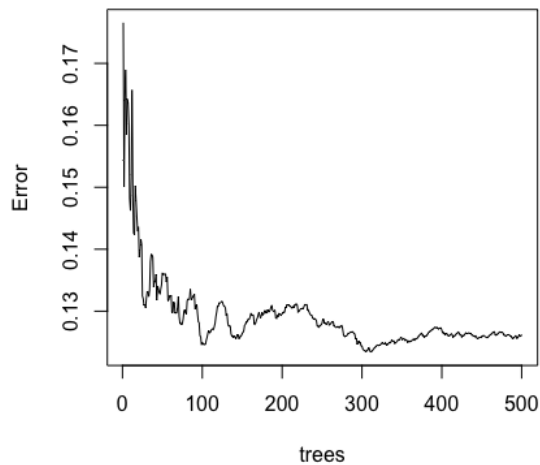




fit



fit



Explanation

`plot(fit)` will allow you to make the plot.

Submit

You have used 1 of 2 attempts

i Answers are displayed within the problem

Q6

1/1 point (graded)

It seems that the default values for the Random Forest result in an estimate that is too flexible (unsmooth). Re-run the Random Forest but this time with a node size of 50 and a maximum of 25 nodes. Remake the plot.

Part of the code is provided for you below.

```
library(randomForest)
fit <- #BLANK
dat %>%
  mutate(y_hat = predict(fit)) %>%
  ggplot() +
  geom_point(aes(x, y)) +
  geom_step(aes(x, y_hat), col = 2)
```

What code should replace #BLANK in the provided code?

☐ `randomForest(y ~ x, data = dat, nodesize = 25, maxnodes = 25)`

☐ `randomForest(y ~ x, data = dat, nodes = 50, max = 25)`

☐ `randomForest(x ~ y, data = dat, nodes = 50, max = 25)`

☒ `randomForest(y ~ x, data = dat, nodesize = 50, maxnodes = 25)`

☐ `randomForest(x ~ y, data = dat, nodesize = 50, maxnodes = 25)`



Explanation

We see that using `randomForest(y ~ x, data = dat, nodesize = 50, maxnodes = 25)` yields smoother results. We'll pick up with this exercise after we learn more about the `caret` package.

Submit

You have used 1 of 2 attempts

i Answers are displayed within the problem

Ask your questions or make your comments about Trees and Random Forests here! **Remember, one of the best ways to reinforce your own learning is by explaining something to someone else, so we encourage you to answer each other's questions (without giving away the answers, of course).**

Some reminders:

- Search the discussion board before posting to see if someone else has asked the same thing before asking a new question.
- Please be specific in the title and body of your post regarding which question you're asking about to facilitate answering your question.

- Posting snippets of code is okay, but posting full code solutions is not.
- If you do post snippets of code, please format it as code for readability. If you're not sure how to do this, there are instructions in a pinned post in the "general" discussion forum.

Discussion: Classification with More than Two Classes

[Show Discussion](#)

Topic: Section 5: Classification with more than two classes and the caret package
/ 5.1: Classification with More than Two Classes