Lab 11: Checking assumptions for linear regression

Name and Student ID

Today’s Date

* Due date: Friday, November 20th, 11:59 PM.

### Boston Data on Median Household Value and Distance to Employment Centers

We are examining a data set used to predict housing prices in the area around Boston (Harrison, D. and Rubinfeld, 1978). We wish to specifically examine the association of the measure of housing price (medv, median value of owner-occupied homes in the $1000s) and a measure of adjacency to employment (a weighted distance, roughly in miles). The data frame (called ‘Boston’) is contained in another package (MASS), which we will load below.

### NOTE: All of the code is to get you started on the lab. You do not need to  
### understand any functions below that you have not seen before.  
  
# Load the MASS library with the Boston data  
library(MASS)  
  
### NOTE: This package has a function `select()` that can be confused with  
### dplyr's select. To overcome this, we first import the data we need and then  
### detach the library before loading dplyr.  
  
# Load the data  
boston\_housing <- read.csv("Boston.csv")  
  
# List variables  
boston2 <- boston\_housing %>% dplyr::select(nox, dis, medv)  
  
# Variable definition - take a quick look at the variables in the data frame  
  
#help(Boston)  
detach(package:MASS)  
  
### Normally, when we are doing inference, we take a random sample from the   
### entire population so we can see how well we can make inference when we only   
### have a sample of 50 individuals (rows of data). \  
### If you have time after the lab, try taking a random sample   
### of 50 rows from the data, and see if your results change.

1. Perform a linear regression of medv versus nox using the Boston data and summarize the results. Be careful about which variable is explanatory and which is response!

# write your code here. Make a linear model, ~tidy~ the output, and report the slope  
  
p1 <- "Write the slope coefficient here rounded to 2 decimals"  
  
  
p1

## [1] "Write the slope coefficient here rounded to 2 decimals"

check\_problem1()

## [1] "Checkpoint 1 Error: Is p1 a number?"  
## [1] "Checkpoint 2 Error: Did you run the correct model? Check your response and explanatory variables."  
##   
## Problem 1  
## Checkpoints Passed: 0  
## Checkpoints Errored: 2  
## 0% passed  
## --------  
## Test: FAILED

1. Interpret the slope parameter for nox. Notice that the other columns are std.error, statistic, and p-value – these should remind you of things we’ve learned about in inference in Part III of the course. They correspond to the hypothesis test with the null hypothesis that the parameter is equal to 0. Thus, how would you interpret the p-value for nox?

[Write your answer here.]

1. Use glance() to look at the value for this model. Does nox explain a lot of the variation in median household value? Would you expect it to?

#Put your code here

[Write your answer here.]

1. Check the assumptions required for the simple linear model using the plots shown during lecture. Note that to make these plots you need to first fit the linear model and then use the augment() function from the broom package to store the residuals and fitted values into a new data frame.

The hardest plot to make is likely the boxplot because the data first needs to be reshaped. You can reshape the data with the gather() function that you see in the slides. Here is a helpful explanation for how gather() works: <https://twitter.com/WeAreRLadies/status/1059520693857996800>.

Basically, we need to gather the observed y values and the residuals by stacking them into one variable so that we can make two box plots side-by-side. Below, we include the gather() code for you since it is a bit tricky. You need to use the resulting data frame to make your box plots.

# Put your code here  
  
# augment your model  
  
# first plot  
plot1 <- "Your plot"  
  
# second plot  
plot2 <- "Your plot"  
  
# third plot  
plot3 <- "Your plot"  
  
# fourth plot (gather code included for you. It assumes your augmented data is  
# called `augmented\_1`, so you will likely need to update that to whatever your  
# augmented data is called.)  
  
#reshape <- augmented\_1 %>% dplyr::select(.resid, medv) %>%  
#gather(key = "type", value = "value", medv, .resid)  
  
plot4 <- "Your plot"  
  
plot1

## [1] "Your plot"

plot2

## [1] "Your plot"

plot3

## [1] "Your plot"

plot4

## [1] "Your plot"

#NO AG for this question

1. What do you think about the assumption plots? They appear to be a bit messier than the ones shown in class, but these reflect what we often see “in the real world”.

[Write your answer here.]

### Lab Conclusion (make sure to read this and understand it)\*\*

From this exercise, we can conclude that there is a negative association between nitrogen oxides and median household value. An increase of 1 part per 10 million (PPM) of nitrogen oxide is associated with a decrease in median household value of $33,900 (see help(Boston) to remind yourself of the units for nox and medv). Note that this “increase of 1 unit” is wider than what we see in the range of the scatterplot, so we should modify our interpretation to reflect a 0.1 unit increase in nox. In other words, an increase in Nitrogen Oxide of 0.1 PPM is associated with a decrease in median household value of $3,390. This is easier to visualize when you look at the scatterplot of the data and the line of best fit. Look at the increase from 0.5 to 0.6 on the x-axis and see how the model predicts a decrease in the household value from ~$25k to ~$22k.

1. Perform a linear regression of medv (median value of owner-occupied homes in $1000s) and dis (weighted mean of distances to five Boston employment centers) using the Boston data and summarize the results.

Assign the linear model to an object called p6.

Be careful about which variable is explanatory and which is response!

# write your code here.  
p6 <- "YOUR CODE HERE"  
  
p6

## [1] "YOUR CODE HERE"

check\_problem6()

## [1] "Checkpoint 1 Error: Did you create a linear model?"  
## [1] "Checkpoint 2 Error: Did you run the correct model? Check your response and explanatory variables."  
##   
## Problem 6  
## Checkpoints Passed: 0  
## Checkpoints Errored: 2  
## 0% passed  
## --------  
## Test: FAILED

1. Interpret the slope parameter and p-value from the table. What null and alternative hypotheses does this p-value refer to?

[Write your answer here.]

1. Derive a 95% CI for this slope parameter and assign the object p8 to a vector of the lower bound and upper bound of the interval. Round to AT LEAST one decimal place. In your opinion, would you expect the direction of this relationship to hold if the data were collected today?

#Put your code here  
p8 <- "YOUR ANSWER HERE"  
  
p8

## [1] "YOUR ANSWER HERE"

check\_problem8()

## [1] "Checkpoint 1 Error: Is p8 a vector of 2 values?"  
## [1] "Checkpoint 2 Error: The value of your lower bound is incorrect."  
## [1] "Checkpoint 3 Error: The value of your upper bound is incorrect."  
##   
## Problem 8  
## Checkpoints Passed: 0  
## Checkpoints Errored: 3  
## 0% passed  
## --------  
## Test: FAILED

[Write your answer here.]

1. Use a function to look at the r-squared value for this model. Round the r-squared value to 2 decimal places and assign this value to the object p9. Does dis explain a lot of the variance in median household value? Would you expect it to?

#use the function here  
  
p9 <- "put r-squared value here"  
  
p9

## [1] "put r-squared value here"

check\_problem9()

## [1] "Checkpoint 1 Error: Is p9 a number?"  
## [1] "Checkpoint 2 Error: Your r-squared is not correct."  
##   
## Problem 9  
## Checkpoints Passed: 0  
## Checkpoints Errored: 2  
## 0% passed  
## --------  
## Test: FAILED

[Write your answer here.]

1. Make a plot with the raw data points, the fitted line from the simple linear regression model (only containing medv and dis), and add a line with a slope of 0. You can have the horizontal line cross the y-axis at the average value of medv to vertically bisect the data points. Store your plot as the object p10.

## [1] "Checkpoint 1 Error: Is p10 a ggplot?"  
## [1] "Checkpoint 2 Error: Did you add a scatterplot, regression line, and horizontal line?"  
##   
## Problem 10  
## Checkpoints Passed: 0  
## Checkpoints Errored: 2  
## 0% passed  
## --------  
## Test: FAILED

1. Does your plot raise any concerns about the assumptions of the linear regression? What other plots might you create to explore the fit of the model? One helpful plot would compare the distribution of model residuals to a theoretical normal distribution. Assign the object p11 to the FIRST TWO LETTERS of the name of this plot.

p11 <- "YOUR ANSWER HERE"  
### OPTIONAL: CODE THE PLOT  
  
p11

## [1] "YOUR ANSWER HERE"

check\_problem11()

## [1] "Checkpoint 1 Error: Provide a character string of the first two letters of the plot type."  
##   
## Problem 11  
## Checkpoints Passed: 0  
## Checkpoints Errored: 1  
## 0% passed  
## --------  
## Test: FAILED

[Write your answer here.]

Regardless of your answer, we go forward using the model to make inferences about the points on the line.

### Pointwise Confidence Intervals and Multiple Testing

As you learned in lecture, there are two types of confidence intervals applicable to estimating a point on the plot which are related to whether one is predicting the population average among individuals with (**mean response**) or whether one is predicting the actual for a particular individual (**single observation**). For this assignment, we will concentrate on the confidence interval for the mean response. We do so because it is rare to use statistical models in public health as forecasting models (predicting an individual’s health in the future) and more common to use them to estimate population-level changes (how does the mean health change in a population as we change exposure). However, as precision medicine becomes more of a reality and the models accurately predict health (i.e., have high ’s), then statistical forecasting may become more common in our field.

1. Calculate four 95% confidence intervals for the mean response, one at each dis value: 2.5, 5.0, 7.5, and 10.0 miles. Store the lower bounds for each confidence interval, ROUNDED to two decimal places, in a vector called p12.

**Hint: Use the predict function, and be sure to specify interval = “confidence”**

OPTIONAL: If time allows, add the four CIs to a scatter plot of the data (along with the line of best fit).

#Put your code here  
### Helpful Data Frame:  
ci\_dataframe <-data.frame(dis = c(2.5, 5.0, 7.5, 10))  
p12 <- "YOUR CODE AND ANSWER HERE"  
  
p12

## [1] "YOUR CODE AND ANSWER HERE"

check\_problem12()

## [1] "Checkpoint 1 Error: Is p12 vector with four numbers?"  
## [1] "Checkpoint 2 Error: Is this the lower bound for the dis = 2.5 C.I.?"  
## [1] "Checkpoint 3 Error: Is this the lower bound for the dis = 5 C.I.?"  
## [1] "Checkpoint 4 Error: Is this the lower bound for the dis = 7.5 C.I.?"  
## [1] "Checkpoint 5 Error: Is this the lower bound for the dis = 10 C.I.?"  
##   
## Problem 12  
## Checkpoints Passed: 0  
## Checkpoints Errored: 5  
## 0% passed  
## --------  
## Test: FAILED

1. Interpret the pointwise 95% confidence interval of the median house price when distance = 10.

[Write your answer here.]

1. Do the CI’s differ in length for different values of dis? Why or why not?

[Write your answer here.]

### Check your score

Click on the middle icon on the top right of this code chunk (with the downwards gray arrow and green bar) to run all your code in order. Then, run this chunk to check your score.

# Just run this chunk.  
total\_score()

## Test Points\_Possible Type  
## Problem 1 FAILED 1 autograded  
## Problem 2 NOT YET GRADED 1 free-response  
## Problem 3 NOT YET GRADED 1 free-response  
## Problem 4 NOT YET GRADED 1 free-response  
## Problem 5 NOT YET GRADED 1 free-response  
## Problem 6 FAILED 1 autograded  
## Problem 7 NOT YET GRADED 1 free-response  
## Problem 8 FAILED 1 autograded  
## Problem 9 FAILED 1 autograded  
## Problem 10 FAILED 1 autograded  
## Problem 11 FAILED 1 autograded  
## Problem 12 FAILED 1 autograded  
## Problem 13 NOT YET GRADED 1 free-response  
## Problem 14 NOT YET GRADED 1 free-response

### Submission

For assignments in this class, you’ll be submitting using the **Terminal** tab in the pane below. In order for the submission to work properly, make sure that:

1. Any image files you add that are needed to knit the file are in the src folder and file paths are specified accordingly.
2. You **have not changed the file name** of the assignment.
3. The file is saved (the file name in the tab should be **black**, not red with an asterisk).
4. The file knits properly.

Once you have checked these items, you can proceed to submit your assignment.

1. Click on the **Terminal** tab in the pane below.
2. Copy-paste the following line of code into the terminal and press enter.

cd; cd ph142-fa20/lab/lab11; python3 turn\_in.py

1. Follow the prompts to enter your Gradescope username and password. When entering your password, you won’t see anything come up on the screen–don’t worry! This is just for security purposes–just keep typing and hit enter.
2. If the submission is successful, you should see “Submission successful!” appear as output.
3. If the submission fails, try to diagnose the issue using the error messages–if you have problems, post on Piazza.

The late policy will be strictly enforced, **no matter the reason**, including submission issues, so be sure to submit early enough to have time to diagnose issues if problems arise.