Lab 11: Checking assumptions for linear regression

Felicia Liu

11/18/2020

- 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
```

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!

```
boston_housing_lm = lm(formula = medv ~ nox, data = boston_housing)
tidy(boston_housing_lm)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)     41.3      1.81      22.8 9.87e-80
## 2 nox            -33.9      3.20     -10.6 7.07e-24
```

```
p1 <- -33.92


p1
```

```
## [1] -33.92
```

```
check_problem1()
```

```
## [1] "Checkpoint 1 Passed: Correct! You set p1 to a numeric value."
## [1] "Checkpoint 2 Passed: Correct! You ran the linear model with the correct explanatory and respons
##
## Problem 1
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

2. 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`?

For each 1 part per 10 million (PPM) increase of nitrogen oxide, the median value of the housing price decreases by 33.92 thousand dollars. Because the p-value of 'nox' is 7.065042e-24, this means that there is evidence against the null that the parameter is equal to 0, in favor of the alternative hypothesis.

3. Use `glance()` to look at the r^2 value for this model. Does `nox` explain a lot of the variation in median household value? Would you expect it to?

```
glance(boston_housing_lm)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.183         0.181  8.32      113. 7.07e-24     1 -1789. 3584. 3597.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

The r^2 value for this model is 0.182603. 'nox' does not explain a lot of the variation in median household value because the sigma 8.32 is considered relatively high. This means that the model is not a good fitting model, since sigma is the regression standard error. I would not expect it to because the value people likely do not measure nitrogen oxide levels to determine the value of a home.

4. 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.

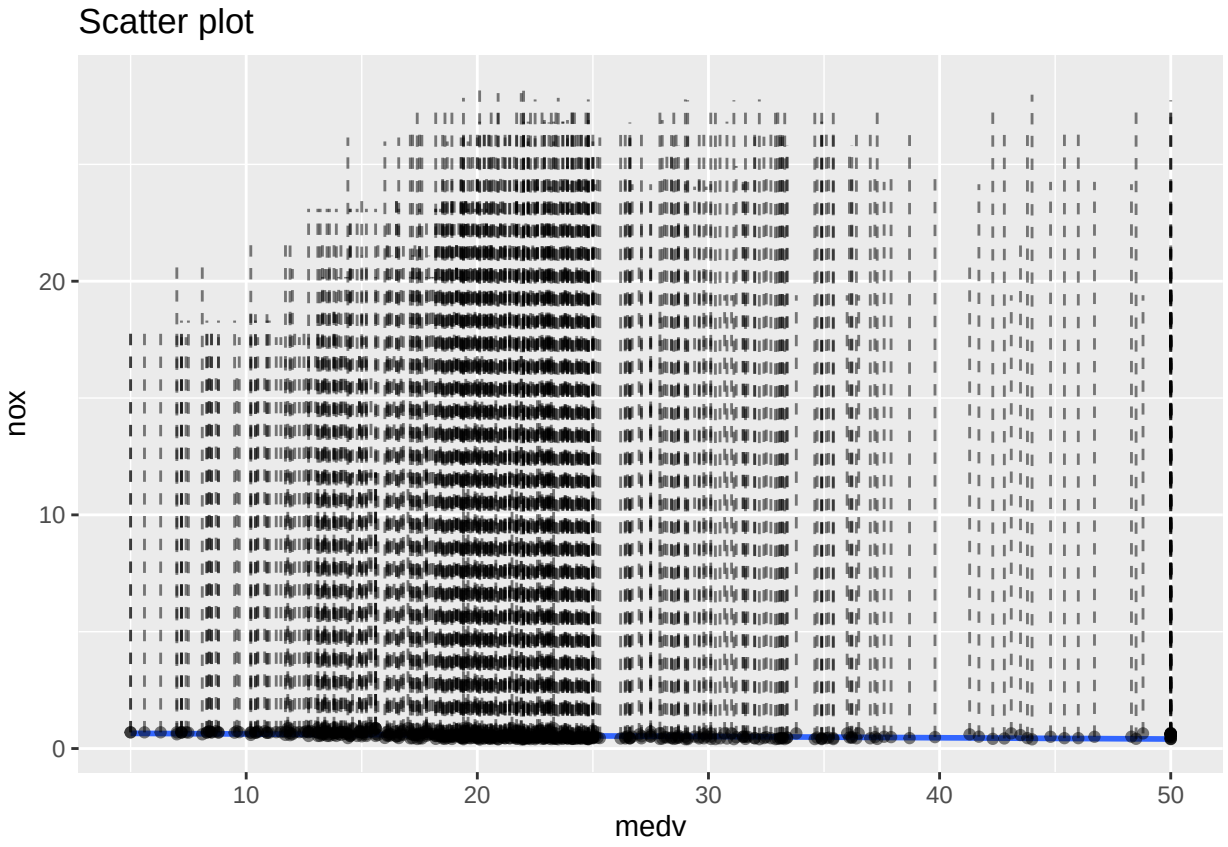
```
boston_housing_augment <- augment(boston_housing_lm)
boston_housing_augment %>% select(medv, nox, .fitted, .resid) %>% head()
```

```
## # A tibble: 6 x 4
##    medv   nox .fitted .resid
##   <dbl> <dbl>   <dbl>  <dbl>
## 1  24   0.538    23.1  0.901
## 2  21.6 0.469    25.4 -3.84
## 3  34.7 0.469    25.4  9.26
## 4  33.4 0.458    25.8  7.59
## 5  36.2 0.458    25.8 10.4
## 6  28.7 0.458    25.8  2.89
```

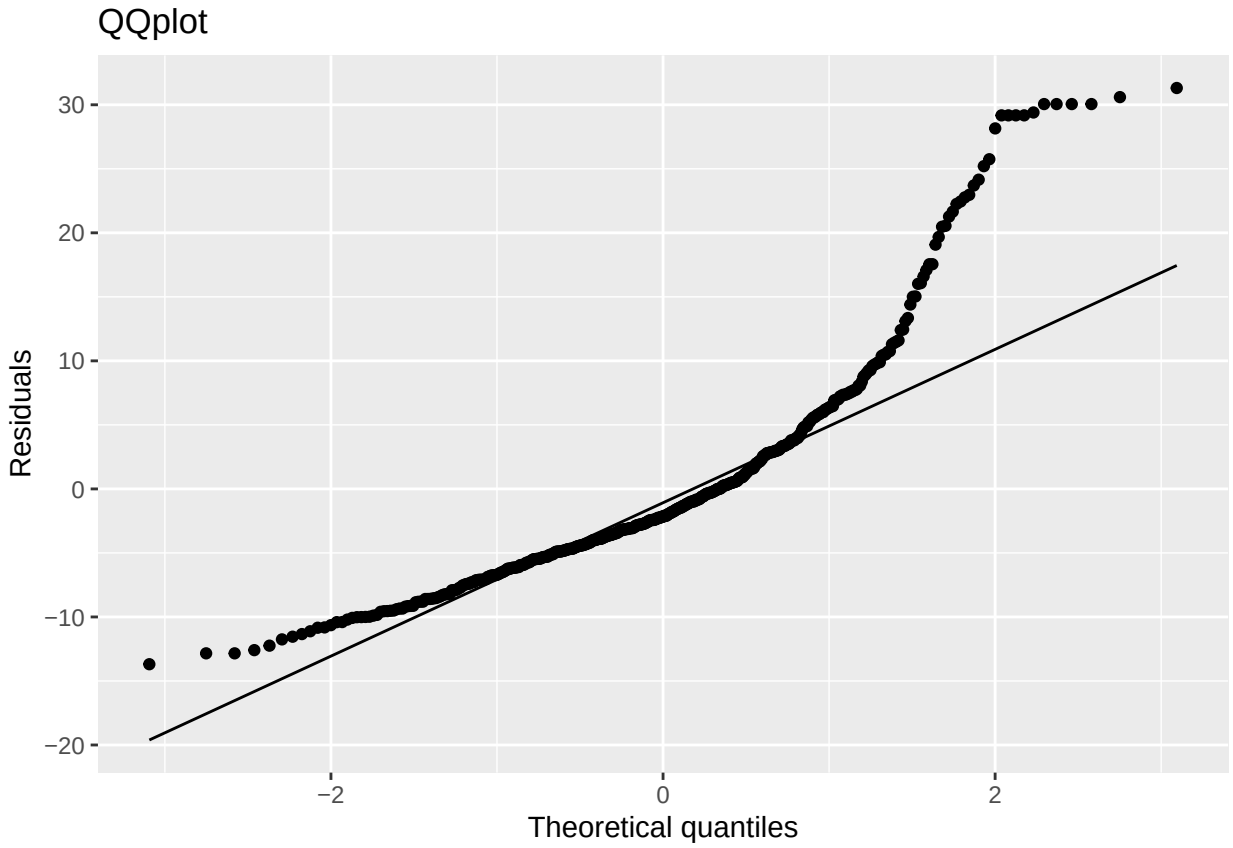
```
# augment your model

# first plot
plot1 <- ggplot(boston_housing_augment, aes(medv, nox)) +
  geom_smooth(method = "lm", se = F) +
  geom_point(alpha = 0.5) +
  geom_segment(aes(xend = medv, yend = .fitted), lty = 2, alpha = 0.5) +
  labs (title = "Scatter plot")

# second plot
plot2 <- ggplot(boston_housing_augment, aes(sample = .resid)) +
  geom_qq() + geom_qq_line() +
  labs(y = "Residuals", x = "Theoretical quantiles", title = "QQplot")

# third plot
plot3 <- ggplot(boston_housing_augment, aes(y = .resid, x = .fitted)) +
  geom_point() + geom_hline(aes(yintercept = 0)) +
  labs(y = "Residuals", x = "Fitted values", title = "Fitted vs Residuals")

# 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 <- boston_housing_augment %>% dplyr::select(.resid, medv) %>%
gather(key = "type", value = "value", medv, .resid)

plot4 <- ggplot(reshape, aes(y = value)) +
  geom_boxplot(aes(fill = type)) +
  labs(title = "Amount explained")

plot1
```

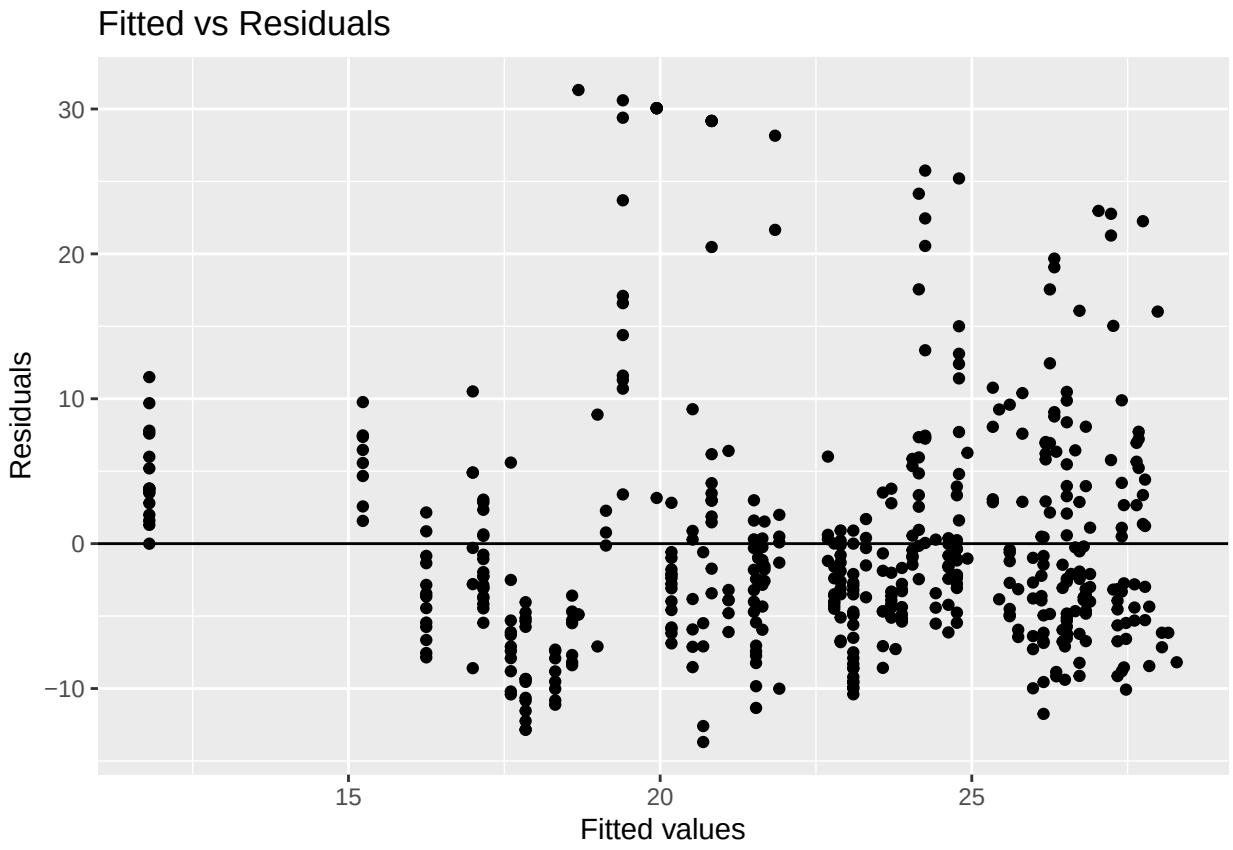
```
## `geom_smooth()` using formula 'y ~ x'
```



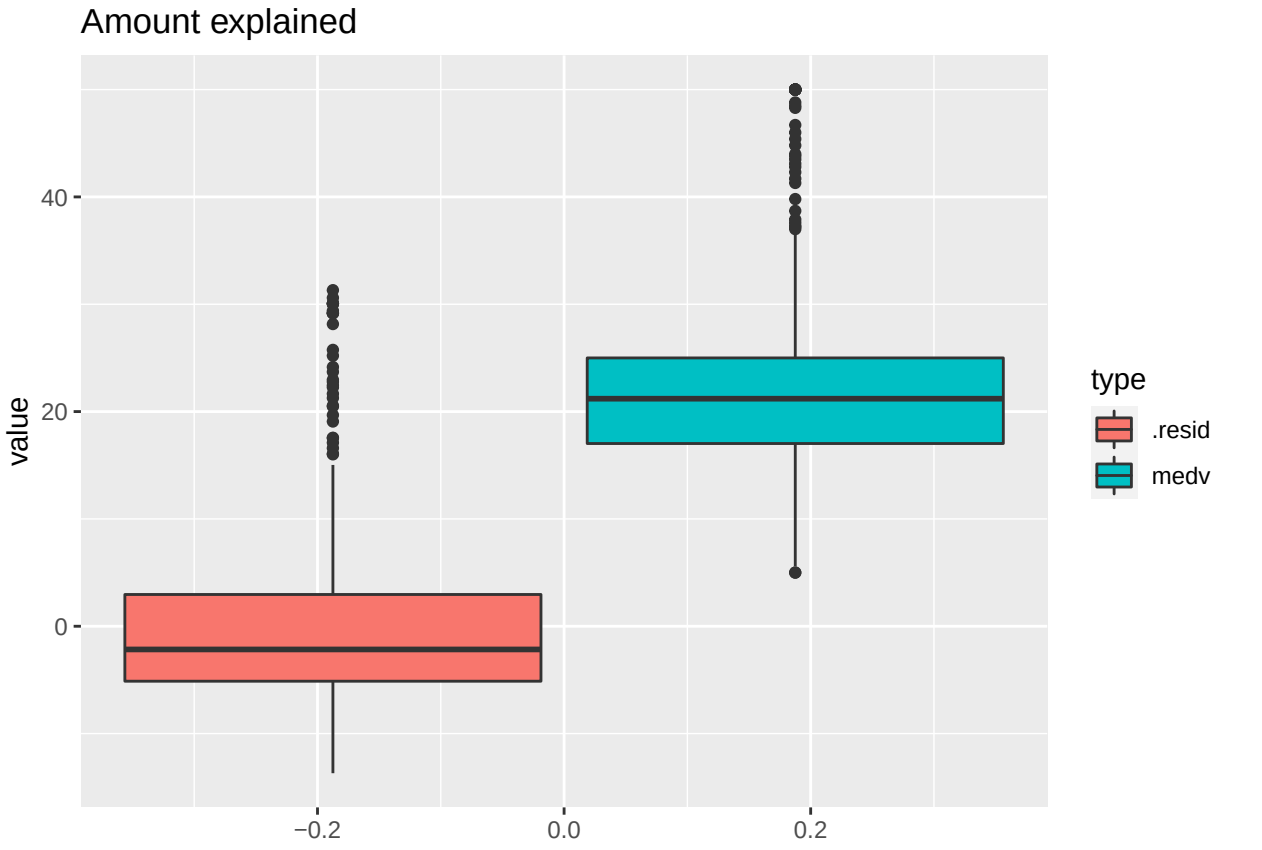

```
plot2
```




```
plot3
```


Fitted vs Residuals
Residuals
30
20
10
0
−10
15
20
25
Fitted values

```
plot4
```




```
#NO AG for this question
```

5. 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".

The scatter plot does not show a fitted regression line and most residuals are big. The QQ plot shows a slight curvature, which suggests that the data is not normally distributed. The residuals vs fitted plot shows that there is a random scatter, which is a good sign. In the amount explained graph, the model fits the data well because the variation in residuals is much smaller than the variation in the y variable to begin with. Only the residual vs fitted plot provides evidence for the assumption that a linear fit is the most appropriate one.

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.

6. 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 <- lm(formula = medv ~ dis, data = boston_housing)
tidy(p6)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    18.4      0.817     22.5  4.01e-78
## 2 dis             1.09     0.188      5.79 1.21e- 8
```

```
p6
```

```
##
## Call:
## lm(formula = medv ~ dis, data = boston_housing)
##
## Coefficients:
## (Intercept)          dis
##      18.390        1.092
```

```
check_problem6()
```

```
## [1] "Checkpoint 1 Passed: Correct! You created a linear model."
## [1] "Checkpoint 2 Passed: Correct! You used the correct explanatory and response variables in your m
##
## Problem 6
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

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

The slope parameter is 1.092, which means that for each mile increase in distance from the employment center, the median value of the house increases by 1.092 thousand dollars. The p-value of 1.206612e-08 is extremely small, so we would reject the null hypothesis in favor of the alternative hypothesis. The null hypothesis refers to there is no relationship between the x and y variables (the slope is 0) and the alternative hypothesis refers to there is a relationship between the x and y variables (the slope is not 0).

8. 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?

```
t_star = qt(p = 0.975, df = 504)
p8 <- c(1.091613 - t_star * 0.1883784, 1.091613 + t_star * 0.1883784)

p8
```

```
## [1] 0.7215093 1.4617167
```

```
check_problem8()
```

```
## [1] "Checkpoint 1 Passed: Correct! You have a vector of 2 values."
## [1] "Checkpoint 2 Passed: Correct! Your lower bound is correct."
## [1] "Checkpoint 3 Passed: Correct! Your upper bound is correct."
##
## Problem 8
## Checkpoints Passed: 3
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

I would not expect this relationship to hold if the data were collected today because there are a lot more factors other than distance to employment centers that people would consider when purchasing a home. For example, the surrounding neighborhood (suburban vs urban), noise, distance to parks, etc. could all play a part in someone's decision which could then increase the median value of the household. Because there are likely more factors today, the relationship is probably not the same anymore.

9. 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?

```
glance(p6)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1    0.0625        0.0606  8.91      33.6 1.21e-8     1 -1824. 3654. 3667.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
p9 <- 0.06

p9
```

```
## [1] 0.06
```

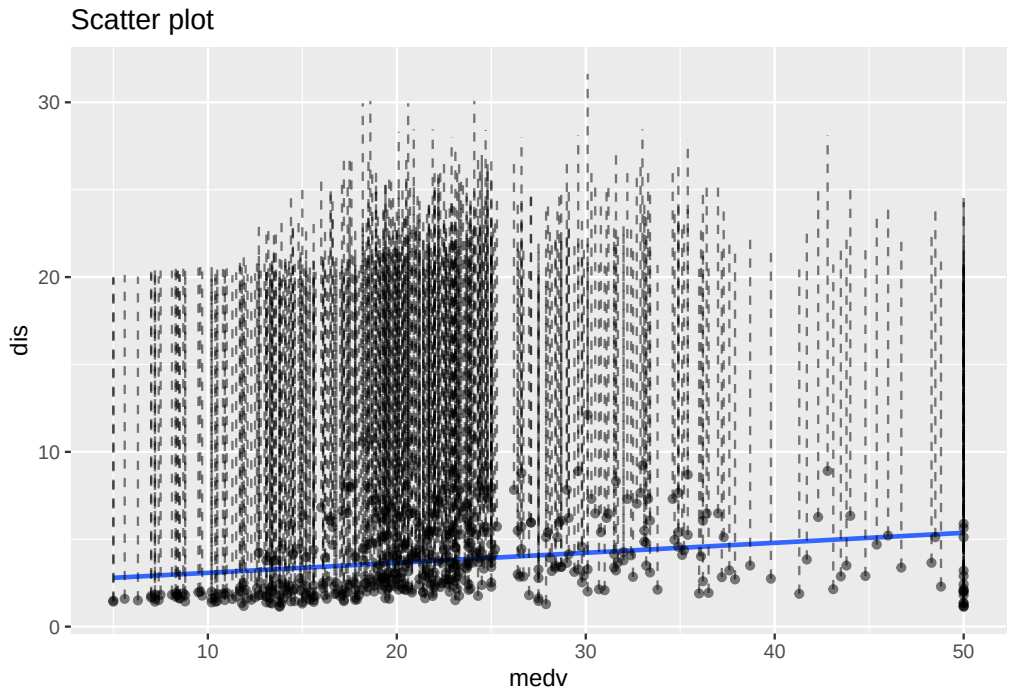
```
check_problem9()
```

```
## [1] "Checkpoint 1 Passed: Correct! Your r-squared is a number."
## [1] "Checkpoint 2 Passed: Correct! You have the correct r-squared."
##
## Problem 9
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

'dis' does not explain a lot of the variance in median household value because the sigma (regression standard error) is relatively large, and a good fitting model will have a low regression standard error.

10. 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`.

```
## `geom_smooth()` using formula 'y ~ x'
```




```
## [1] "Checkpoint 1 Passed: Correct! You created a ggplot of the data."
## [1] "Checkpoint 2 Passed: Correct! You added a scatterplot, regression line, and horizontal line."
##
## Problem 10
## Checkpoints Passed: 2
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

11. 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 <- " QQ"
### OPTIONAL: CODE THE PLOT

p11
```

```
## [1] " QQ"
```

```
check_problem11()
```

```
## [1] "Checkpoint 1 Passed: A qq plot is correct!"
##
## Problem 11
## Checkpoints Passed: 1
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

My scatter plot suggests that the residuals are large and are not represented well by the fitted line. Most residuals are also positive.

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 $X = x$ (**mean response**) or whether one is predicting the actual Y 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 R^2's), then statistical forecasting may become more common in our field.

12. 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))
confidence_interval <- predict(p6, ci_dataframe, interval = "confidence")
confidence_interval
```

```
##        fit      lwr      upr
## 1 21.11912 20.20485 22.03339
## 2 23.84815 22.95092 24.74539
## 3 26.57719 25.00035 28.15402
## 4 29.30622 26.88135 31.73108
```

```
p12 <- c(20.2, 22.95, 25.00, 26.88)

p12
```

```
## [1] 20.20 22.95 25.00 26.88
```

```
check_problem12()
```

```
## [1] "Checkpoint 1 Passed: Correct! p12 is a vector with four numbers."
## [1] "Checkpoint 2 Passed: Correct! This is the lower bound for the dis = 2.5 C.I."
## [1] "Checkpoint 3 Passed: Correct! This is the lower bound for the dis = 5 C.I."
## [1] "Checkpoint 4 Passed: Correct! This is the lower bound for the dis = 7.5 C.I."
## [1] "Checkpoint 5 Passed: Correct! This is the lower bound for the dis = 10 C.I."
##
## Problem 12
## Checkpoints Passed: 5
## Checkpoints Errored: 0
## 100% passed
## --------
## Test: PASSED
```

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

The 95% confidence interval of the median house price when distance = 10 is from 26.88135 to 31.73108. This means that when the house is 10 miles from the employment center, the 95% of the average median house prices are between 26.88 and 31.73 thousand dollars.

14. Do the CI's differ in length for different values of `dis`? Why or why not?

The CI's do differ in length for different values of 'dis' because depending on the distance from the employment centers, each house has a different median house price. The average of these medians would not have the same distribution and confidence interval since the data points are unique.

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          PASSED               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          PASSED               1    autograded
## Problem 7  NOT YET GRADED               1 free-response
## Problem 8          PASSED               1    autograded
## Problem 9          PASSED               1    autograded
## Problem 10         PASSED               1    autograded
## Problem 11         PASSED               1    autograded
## Problem 12         PASSED               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

3. 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.
4. If the submission is successful, you should see "Submission successful!" appear as output.
5. 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.