Problem Set 10

Your name and student ID

Today's date

Instructions

- Solutions will be released on Friday, November 17th.
- This semester, problem sets are for practice only and will not be turned in for marks.

Helpful hints:

- Every function you need to use was taught during lecture! So you may need to revisit the lecture code to help you along by opening the relevant files on Datahub. Alternatively, you may wish to view the code in the condensed PDFs posted on the course website. Good luck!
- Knit your file early and often to minimize knitting errors! If you copy and paste code for the slides, you are bound to get an error that is hard to diagnose. Typing out the code is the way to smooth knitting! We recommend knitting your file each time after you write a few sentences/add a new code chunk, so you can detect the source of knitting errors more easily. This will save you and the GSIs from frustration!
- To avoid code running off the page, have a look at your knitted PDF and ensure all the code fits in the file. If it doesn't look right, go back to your .Rmd file and add spaces (new lines) using the return or enter key so that the code runs onto the next line.

Section 1: Voting during the 1992 election

Let's consider some historical data on voting patterns across US counties. This code loads in the dataframe called `counties`:

```
load("data/A10_counties.sav")
```

These data are from the 1992 election and looks at the percent of votes cast (in each county) for the `democrat` (Bill Clinton), `republican` (George Bush), and independent presidential nominees (Ross `Perot`).

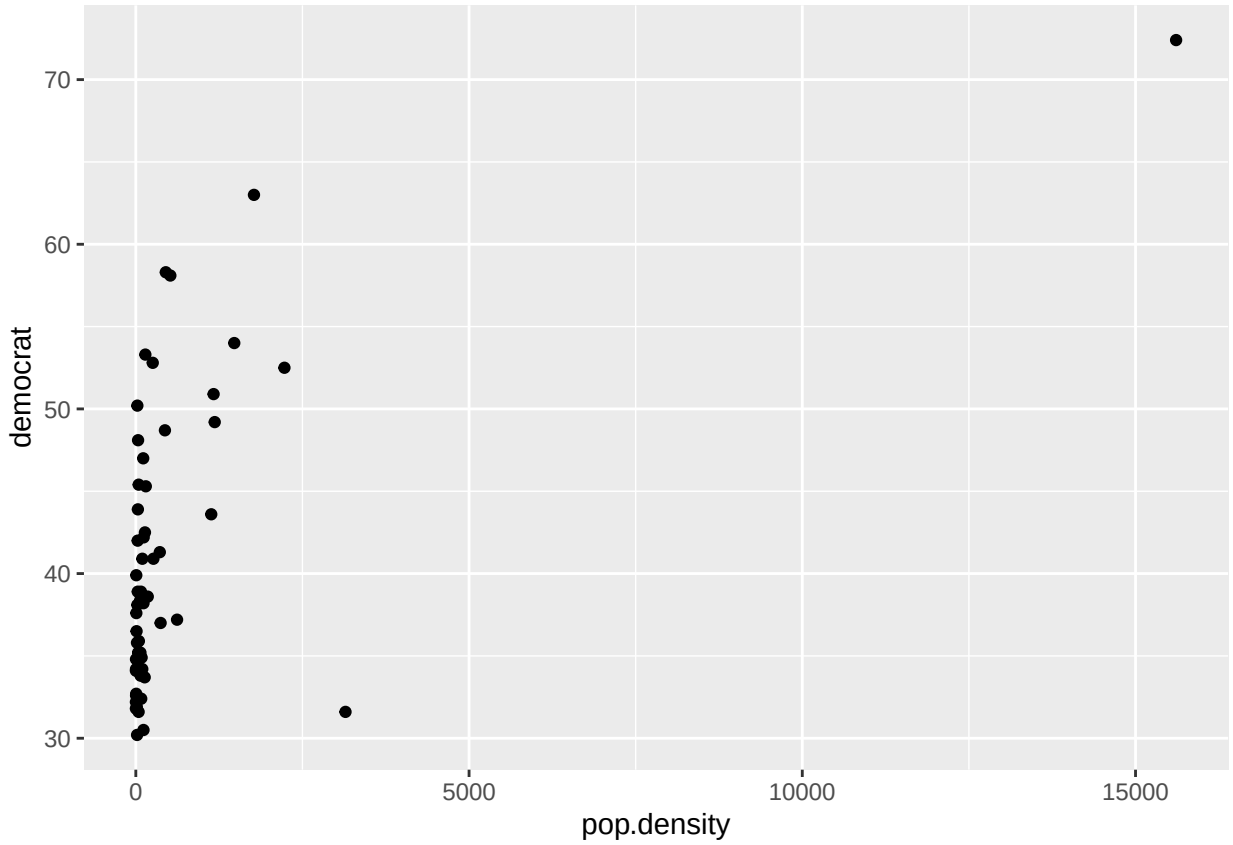
Ideally, if you were interested in voting patterns, you might look at the relationship between individual characteristics and whether each individual voted Democrat or Republican. However, data like that is often hard to come by. The `counties` data provide data on 3141 counties. Use `View()` to examine these data briefly and read the labels corresponding to the variables. Note that Alaska is not included and that two other counties with populations = 0 have also been excluded.

As discussed in class, we have the entire population (not just a sample), so strictly speaking we don't need to perform statistical inference. However, we might pretend this is a sample so that we can apply the techniques of inference and gain competence creating and interpreting a linear model.

1. Make a subset of the original `counties` dataset called `counties_CA` that only includes California counties. Then use this new dataset to plot the relationship between the percent of votes cast for the Democratic candidate (`democrat`) and the population density of the county (`pop.density`).

```
counties_CA <- counties %>% filter(state == "CA")

p1 <- ggplot(counties_CA, aes(x = pop.density, y = democrat)) +
  geom_point()
p1
```




```
. = ottr::check("tests/p1.R")
```

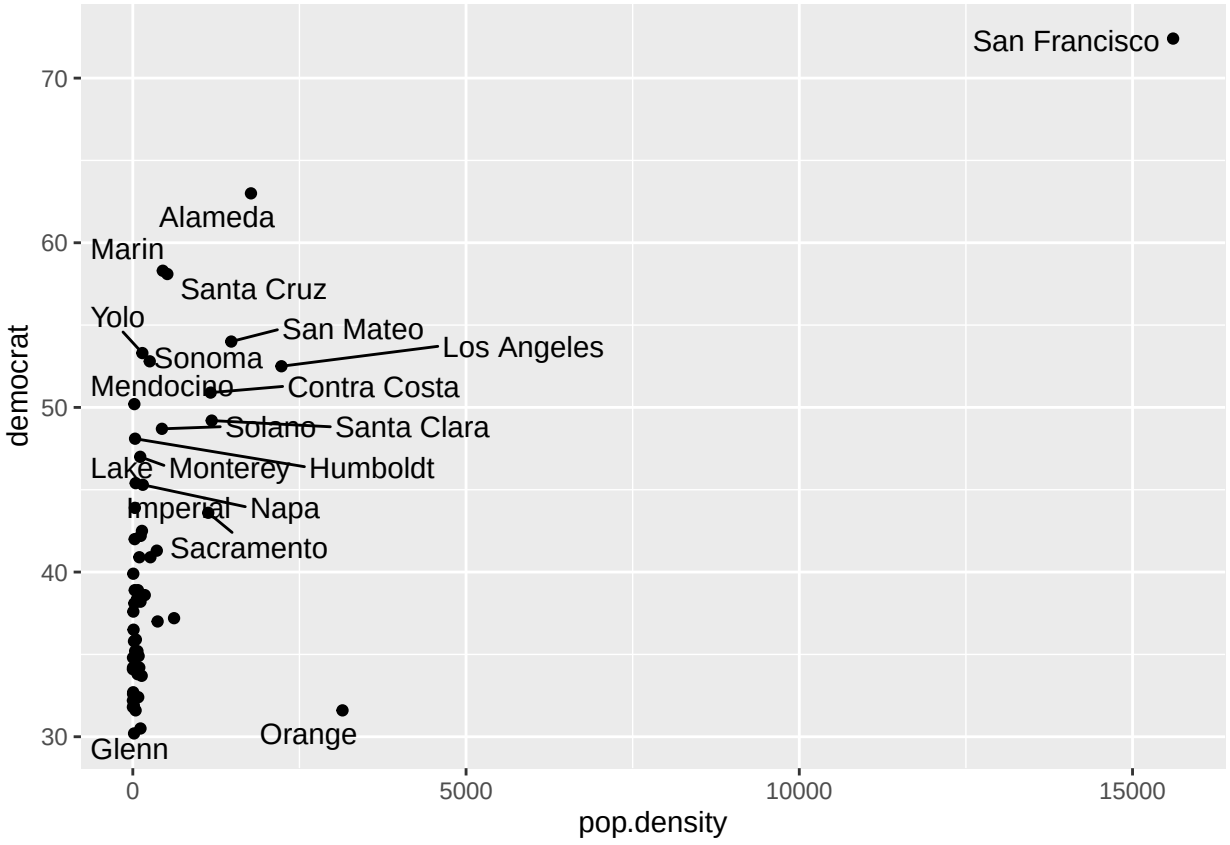
```
##
## All tests passed!
```

2. The above plot is difficult to interpret. The distribution of population density is skewed right, with a few counties having much higher densities than the majority of counties. To see which counties these are, we will use `geom_text_repel` from the library `ggrepel`. This will label the points with the county names. The template for using this function is: `geom_text_repel(aes(label = your_labeling_var))`. The labeling variable should be the county names. Create the same plot as in question 1 with this added `geom_text_repel()` component.

```
p2 <- ggplot(counties_CA, aes(x = pop.density, y = democrat)) +
  geom_point() +
  geom_text_repel(aes(label = county))

p2
```

```
## Warning: ggrepel: 38 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```




```
. = ottr::check("tests/p2.R")
```

```
##
## All tests passed!
```

The current issue with these data is that San Francisco has a much higher population density than other counties, and that generally there is a large right skew in the distribution of the population density variable.

If we tried to fit a linear model to these data, it would not fit well because the relationship between population density and the response variable is not linear. However, this is the perfect situation to try transforming the x-variable.

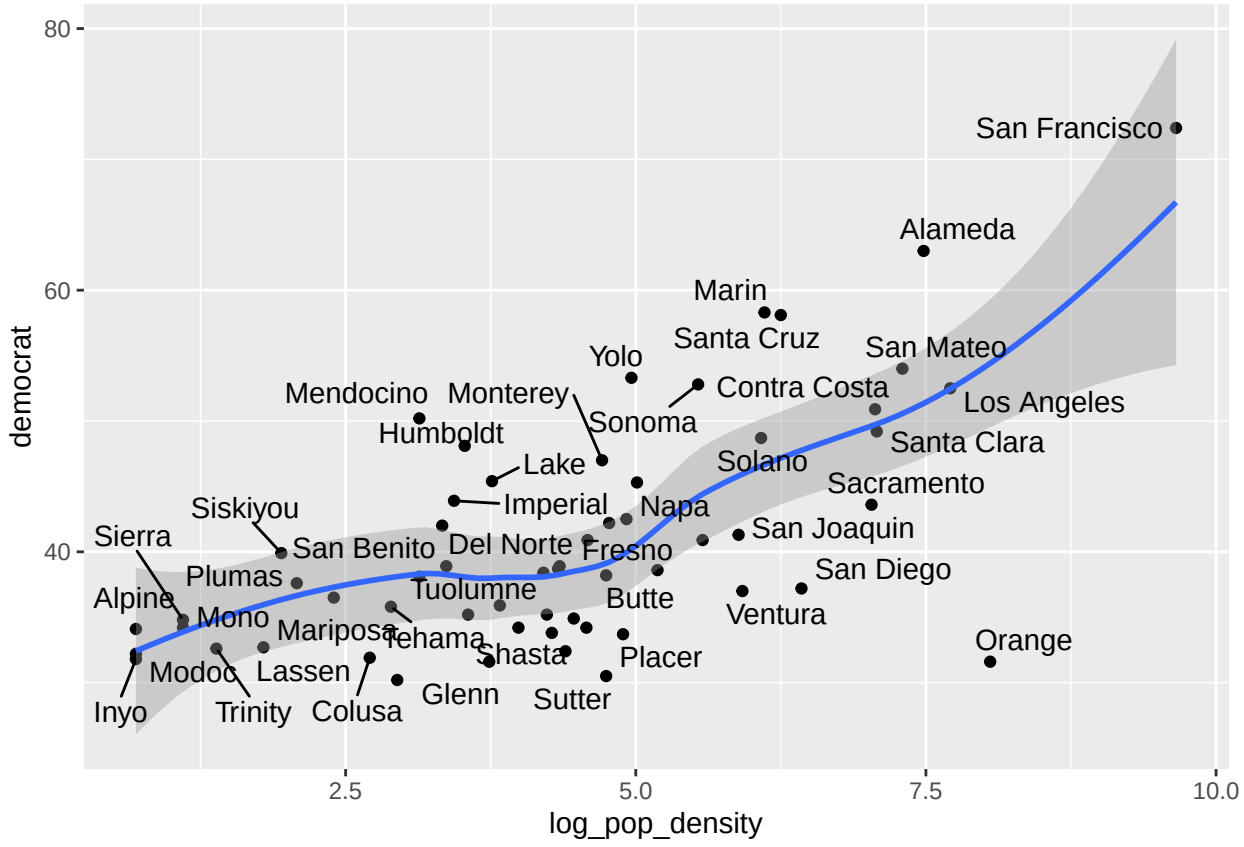
3. Add a new variable to the `counties_CA` dataset called `log_pop_density` that is the log of the population density variable. Remake the plot above using this new variable, add a smoothed fitted line, and assign this to `p3`.

```
counties_CA <- counties_CA  %>%
  mutate(log_pop_density = log(pop.density))

p3 <- ggplot(counties_CA, aes(x = log_pop_density, y = democrat)) +
  geom_point() +
  geom_smooth() +
  geom_text_repel(aes(label = county))
p3
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

```
## Warning: ggrepel: 15 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```




```
. = ottr::check("tests/p3.R")
```

```
##
## All tests passed!
```

4. Describe the relationship between the (logged) population density and the response variable in terms of the strength, form, direction, and outliers. Calculate the correlation coefficient and assign this value to p4.

```
p4 <- counties_CA %>% summarise(cor = cor(democrat, log_pop_density)) %>% pull(cor)
p4
```

```
## [1] 0.6381187
```

Strength: 0.6381. The correlation between the variables is 64%, indicating a moderate positive association.

Form: Roughly linear, or slightly curved.

Direction: There is a positive association between logged population density and the % of votes cast for the democratic candidate.

Outliers: No large outliers, though SF and Orange County are a bit further out from the rest of the points.

```
. = ottr::check("tests/p4.R")
```

```
##
## All tests passed!
```

5. Run a linear regression model of the percent votes cast for the democratic candidate as a function of the logged population density. Use the `tidy()` function to show the slope and intercept estimates. Interpret the relationship between the logged population density and the response variable. Use another function from the `broom` package show the r-squared value. Assign this value to `p5` and interpret its meaning in the context of the problem.

```
lm_CA <- lm(formula = democrat ~ log_pop_density, data = counties_CA)
tidy(lm_CA)
```

```
## # A tibble: 2 x 5
##   term            estimate std.error statistic  p.value
##   <chr>              <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)        28.4      2.22      12.8  3.10e-18
## 2 log_pop_density     2.88     0.464      6.20 7.12e- 8
```

```
p5 <- glance(lm_CA) %>% pull(r.squared)
p5
```

```
## [1] 0.4071955
```

A one unit change in the logged population density (where population density was the 1992 population per square-mile) is associated with a 2.88 percentage point increase in the percent of votes cast for the democratic candidate.

The r-squared is 0.41, implying that 41% of the variation in percentage votes casts is explained by the logged population density.

```
. = ottr::check("tests/p5.R")
```

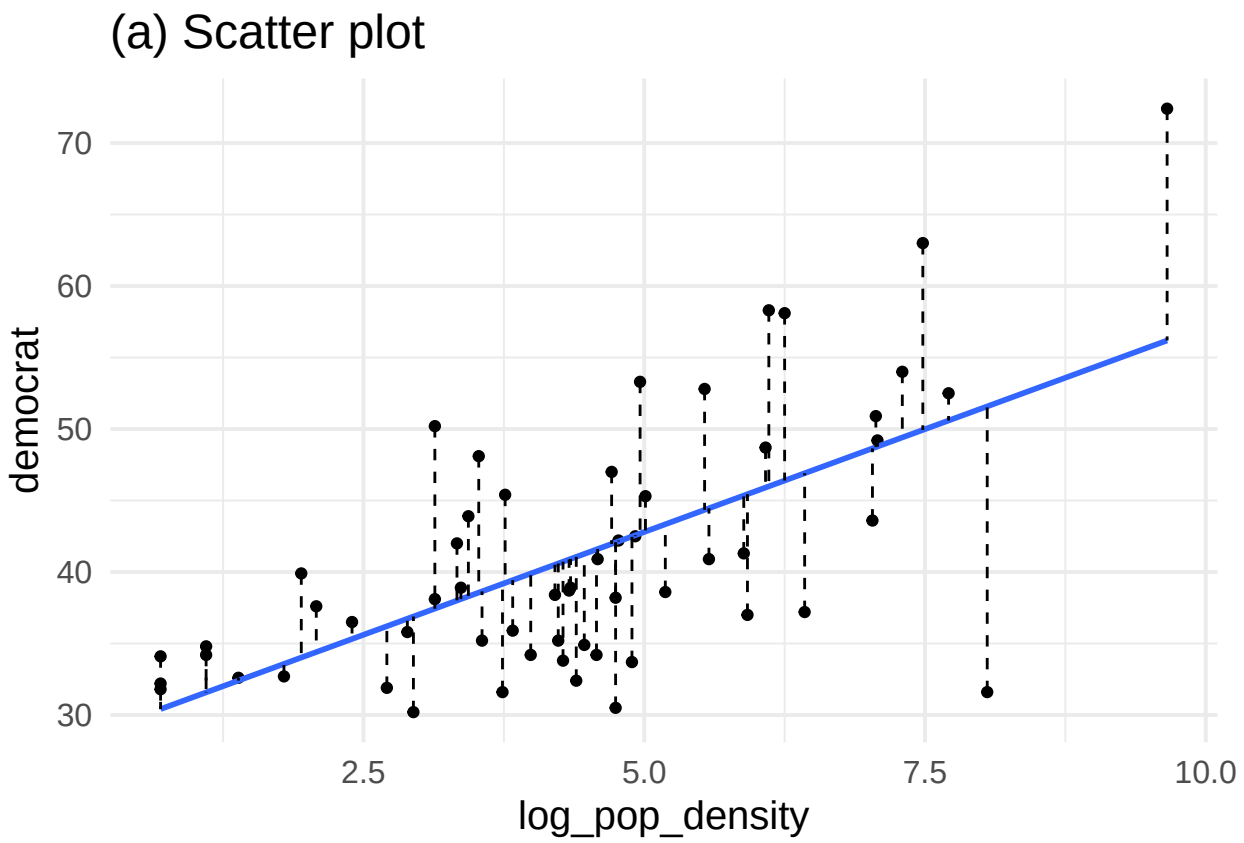
```
##
## All tests passed!
```

6. Fit the augmented model and assign this to `CA_augment`. Then create the 4 plots used to check the assumptions for using linear regression.

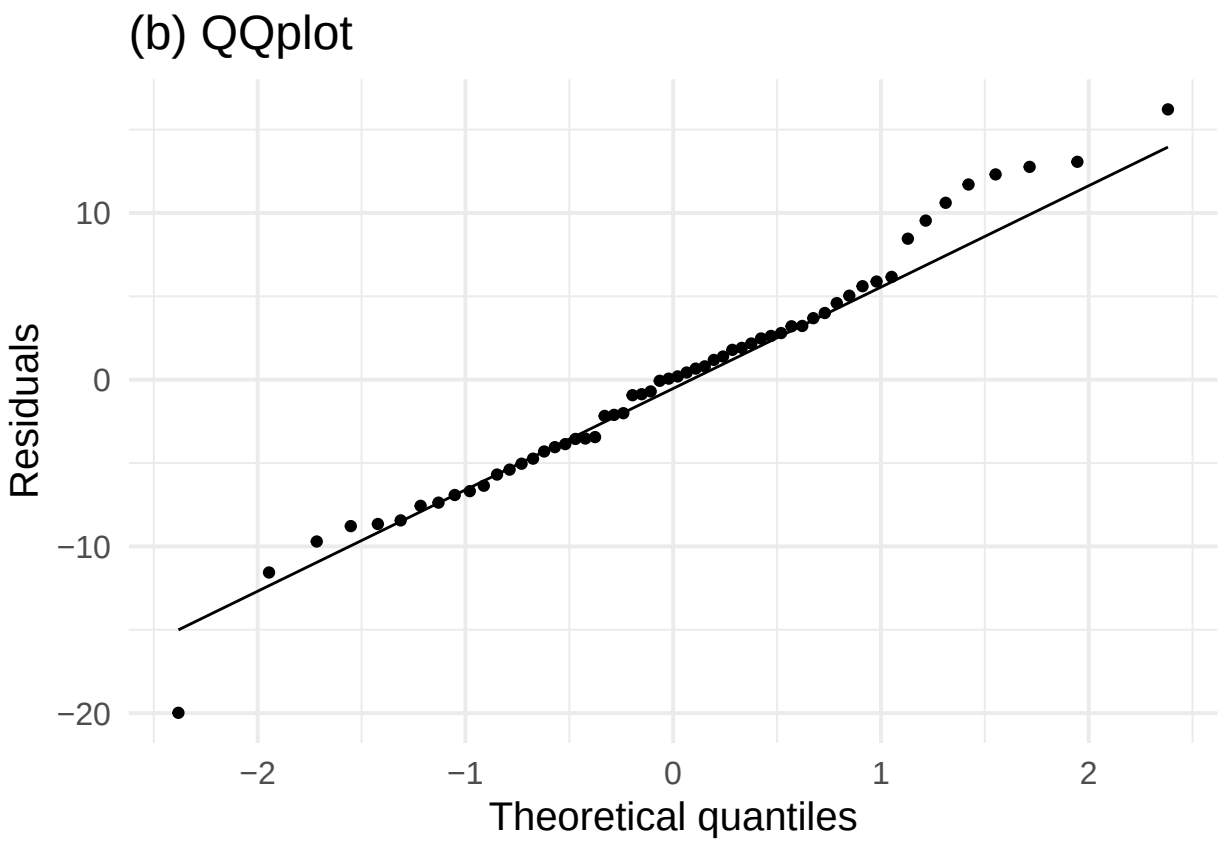
```
CA_augment <- augment(lm_CA)

# scatter plot
plot1 <- ggplot(CA_augment, aes(y = democrat, x = log_pop_density)) +
  geom_point() +
  geom_smooth(method = "lm", se = F) +
  geom_segment(aes(xend = log_pop_density, yend = .fitted), lty = 2) +
  theme_minimal(base_size = 15) +
  labs(title = "(a) Scatter plot")
plot1
```

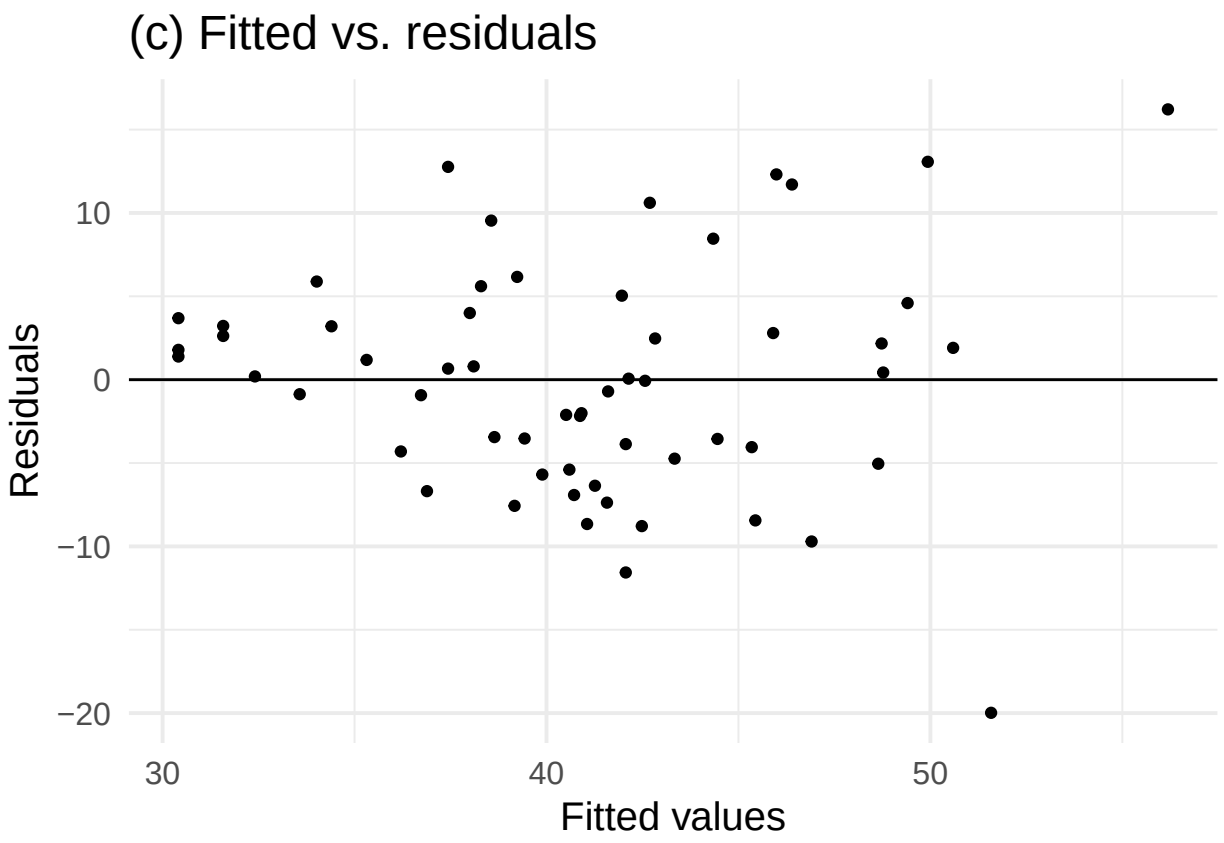
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
# QQ plot
plot2 <- ggplot(CA_augment, aes(sample = .resid)) +
  geom_qq() +
  geom_qq_line() +
  theme_minimal(base_size = 15) +
  labs(y = "Residuals", x = "Theoretical quantiles", title = "(b) QQplot")
plot2
```

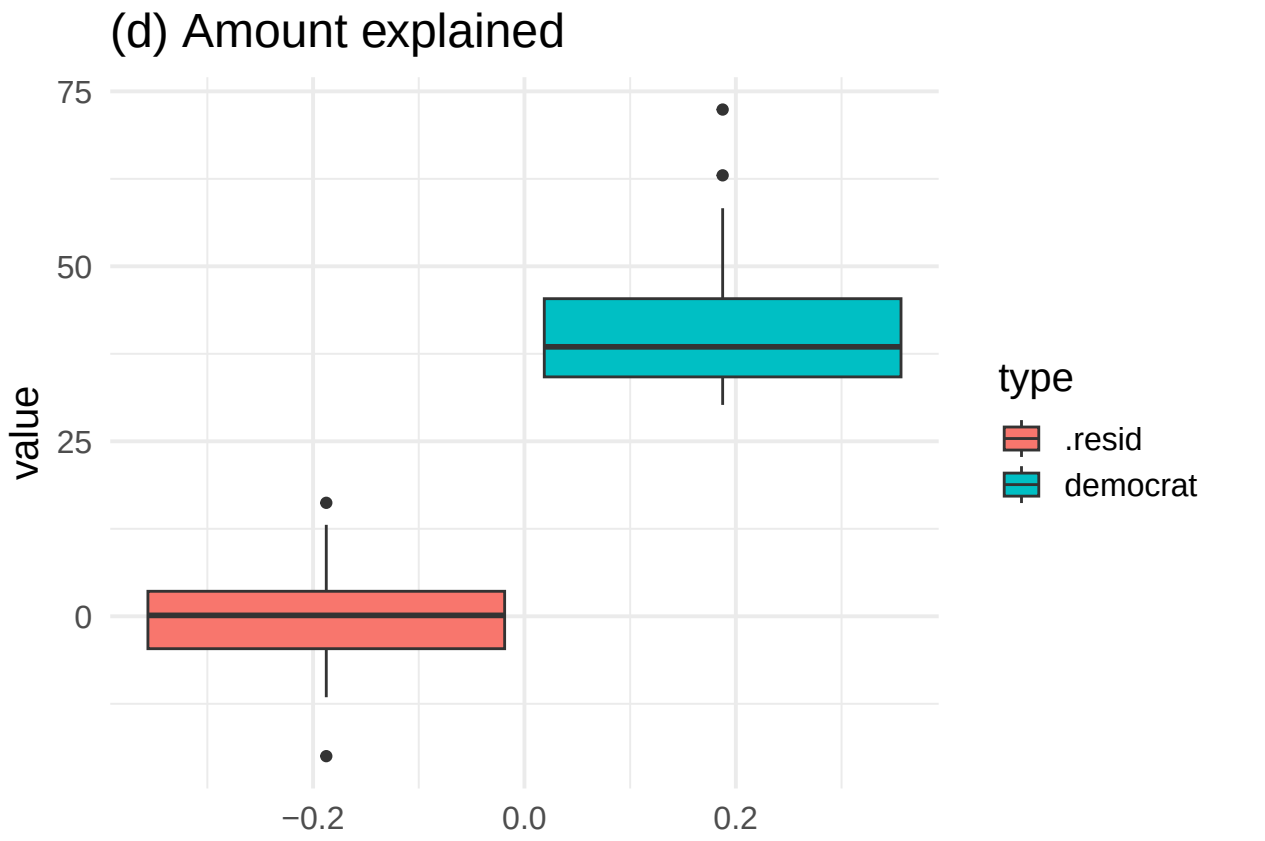



```
## Fitted vs. residuals
plot3 <- ggplot(CA_augment, aes(y = .resid, x = .fitted)) +
  geom_point() +
  theme_minimal(base_size = 15) +
  geom_hline(aes(yintercept = 0)) +
  labs(y = "Residuals", x = "Fitted values", title = "(c) Fitted vs. residuals")
plot3
```




```
## Amount explained
bolt_gather <- CA_augment %>% select(democrat, .resid) %>%
  gather(key = "type", value = "value", democrat, .resid)

plot4 <- ggplot(bolt_gather, aes(y = value)) +
  geom_boxplot(aes(fill = type)) +
  theme_minimal(base_size = 15) +
  labs(title = "(d) Amount explained")
plot4
```




```
. = ottr::check("tests/p6.R")
```

```
##
## All tests passed!
```

7. Comment on each of the plots and conclude whether any assumptions appear to be violated. Don't forget to comment on the one assumption that cannot be investigated using plots!

- There is a violation of the assumption that the standard deviation of the response variable is identical for all values of the explanatory variable. We see here that as the log of the population density increases, the residuals become larger.
- The relationship between X and Y is approximately linear, though there is a lot of variation around the line of best fit. (This points to the fact that though population density is predictive of the response variable, there are other factors that are not included in our model that have further predictive power).
- The QQ plot looks okay. (May be interpreted as a problem with the largest residuals).
- We can't check the assumption that the points are independent using this plot. That corresponds to the counties being independent of one another. This model treats them as independent units. [More sophisticated models can take the spatial relationships between the counties into account (to account for the fact that counties closer to each other may be more similar)].

Section 2: Abstract interpretation

Read the following abstract and answer the questions that follow. J Asthma. 2018 Oct 11:1-12. doi: 10.1080/02770903.2018.1508471. [Epub ahead of print] Impact of scenario based training on asthma first aid knowledge and skills in school staff: an open label, three-arm, parallel-group repeated measures study. Luckie K1, Saini B1, Soo YYB1,2, Kritikos V1,3, Charles Collins JB1, Jane Moles R1.

OBJECTIVE: To test the hypothesis that scenario-based skills training is more effective than knowledge training alone in improving the asthma first aid (AFA) skills of school personnel. Education developed specifically for non-primary caregivers such as school staff is vital to minimize the risk of mortality associated with asthma.

METHODS: Schools were allocated to one of three arms to compare AFA knowledge and AFA skills. Arm 1 underwent conventional asthma training, arm 2 underwent scenario-based training and arm 3 had a combination of the two. Conventional asthma training involved a didactic oral presentation. The scenario-based skills training required the participant to describe and demonstrate how they would manage a child having a severe exacerbation of asthma using equipment provided. Follow-up occurred at 3 weeks post baseline and again between 3-7 months after the first training/education visit.

RESULTS: Nineteen primary schools (204 participants) were recruited. One-way ANOVA and Bonferroni Post-Hoc Tests showed there was a significant difference in AFA skills scores between the study arms who underwent scenario-based training; arms 2 and 3 (91.5% and 91.1%) and arm 1 who underwent conventional asthma training (77.3%) ($p<0.001$). AFA knowledge improved significantly in all study arms with no differences between study arms. Improvements seen in both AFA knowledge and AFA skills were maintained over time.

CONCLUSIONS: Scenario-based training was superior to conventional didactic asthma training for AFA skills acquisition and overall competency in the administration of AFA and should be included in future asthma training programs.

8. Two methods of hypothesis testing (types of tests) are mentioned in the abstract. What is the null hypothesis for each of these tests?

H_0: There is no difference between scenario-based skills training and knowledge training alone in improving the asthma first aid (AFA) skills of school personnel.

H_0: There is no difference in the improvement of asthma first aid skills of school personnel after receiving training scenario-based skills training and a combination of didactic and skills training.

9. There are two outcomes of interest in this study. For which *outcome* would you conclude that there is a significant difference between the training groups?

There is a significant difference in outcome between arm1 and arms2 and 3 of the training groups.

10. If you were a school administrator why might you choose the arm 3 training?

Arm 3 showed a significant improvement over just receiving didactic training but it includes some components that are more traditional to other teaching scenarios. The teaching program would not need to be completely overhauled but rather just include skills-based training.

11. List one question you might want to ask about the methods, sample or results that would help you interpret the findings of this study.

Possible questions could be: 1. What, if any, training had these recruits already experienced? 2. Were the participants randomized to a study arm? 3. What was the difference between arm 2 and arm 3? 4. How was improvement defined in the context of this study?

12. What is another test that could have been considered for these study data?

Two sample t test could have been considered to see if there was a difference in average improvement between arm 2 and arm 3.

Section 3: ANOVA and Tukey's HSD

For this question we will use the data from the NHANES survey.

```
## Rows: 2503 Columns: 40
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (27): agegroup, gender, military, born, citizen, drinkscat, bmicat, sys1...
## dbl (13): ridageyr, drinks, bmxwt, bmxht, bmxbmi, bpxpls, bpxsy1, bpxsy2, bp...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
## # A tibble: 6 x 40
##   ridageyr agegroup gender military   born  citizen drinks drinkscat bmxwt bmxht
##      <dbl> <chr>    <chr>  <chr>      <chr> <chr>    <dbl> <chr>     <dbl> <dbl>
## 1       72 65+      Male   History o~ Born~ US cit~      0 0          88.9  175.
## 2       73 65+      Female No         Born~ US cit~      0 0          52    162.
## 3       61 50-64    Female No         Born~ US cit~      2 11-Jan     93.4  162.
## 4       26 20-34    Female No         Born~ US cit~    209 96-364     47.1  152.
## 5       33 20-34    Female No         No    US cit~     NA <NA>       56.8  158
## 6       32 20-34    Male   No         No    No         300 96-364     79.7  166.
## # i 30 more variables: bmxbmi <dbl>, bmicat <chr>, bpxpls <dbl>, bpxsy1 <dbl>,
## #   bpxsy2 <dbl>, sys1d <chr>, sys2d <chr>, bpxdi1 <dbl>, bpxdi2 <dbl>,
## #   dias1d <chr>, dias2d <chr>, bpcat <chr>, chest <chr>, fs1 <chr>, fs2 <chr>,
## #   fs3 <chr>, lbdhdd <dbl>, hdlcat <chr>, highhdl <chr>, hi <chr>,
## #   asthma <chr>, vwa <chr>, vra <chr>, va <chr>, aspirin <chr>, sleep <dbl>,
## #   is <chr>, hs <chr>, lbdldl <dbl>, highldl <chr>
```

13. Use `dplyr` functions to create a dataframe with the mean (`mean`) and standard deviations (`sd`) for blood lipid level, `lbdldl`, by blood pressure group, `bpcat`.

```
p13 <- nhanes %>% group_by(bpcat) %>% summarize(mean = mean(lbdldl), sd = sd(lbdldl))
p13
```

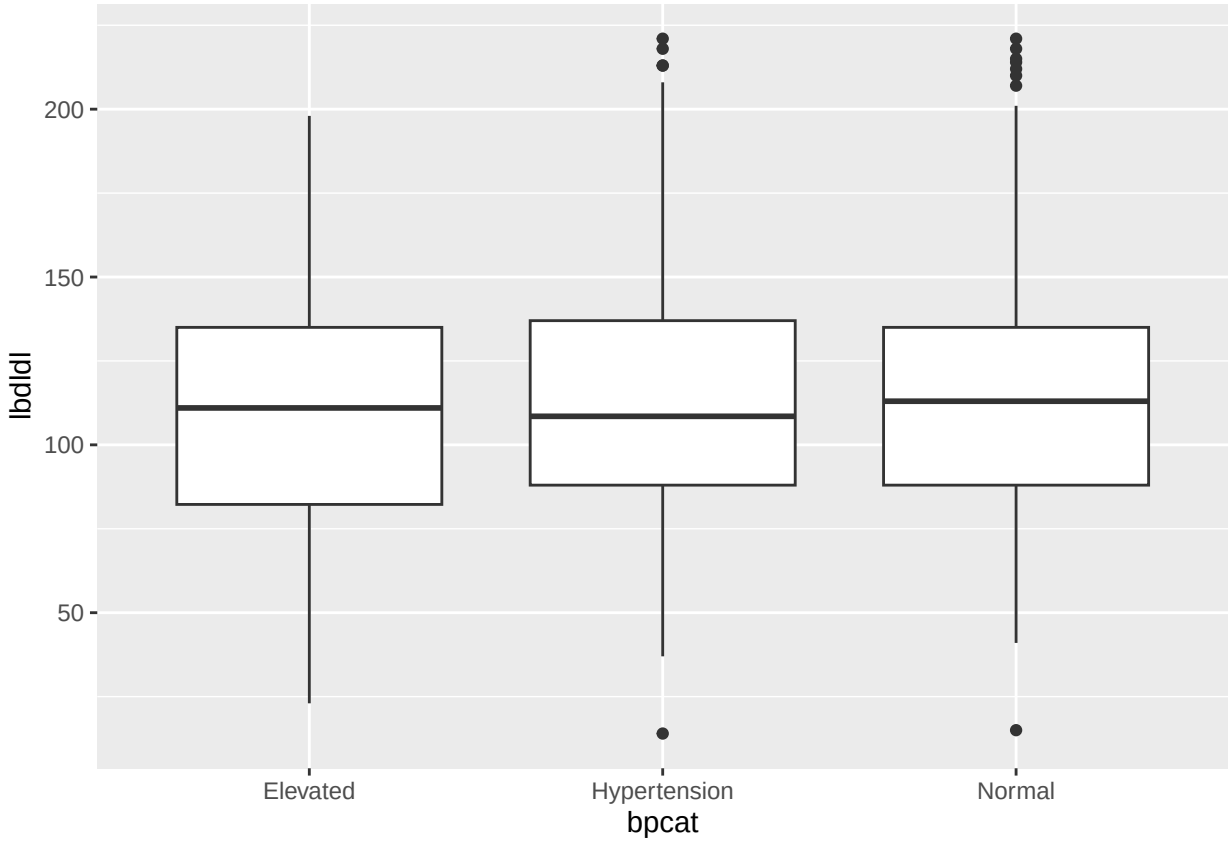
```
## # A tibble: 3 x 3
##   bpcat         mean    sd
##   <chr>        <dbl> <dbl>
## 1 Elevated      111.  35.0
## 2 Hypertension  113.  35.1
## 3 Normal        113.  35.1
```

```
. = ottr::check("tests/p13.R")
```

```
##
## All tests passed!
```

14. Create a boxplot that helps you to visualize the blood pressure and blood lipid level data.

```
p14 <- ggplot(nhanes, aes(x = bpcat, y = lbdldl)) +
        geom_boxplot()
p14
```




```
. = ottr::check("tests/p14.R")
```

```
##
## All tests passed!
```

15. What are the null and alternative hypotheses for testing the difference in blood lipid level by blood pressure category?

H_0 = The average blood lipid level is equal across blood pressure groups.

H_A = The average blood lipid level is different for at least one of the three blood pressure groups.

16. Conduct an ANOVA test with Tukey's HSD for these data and assign it to an object called `tukey`. Then make a dataframe using the `tidy()` function on `tukey` to the object `p16`.

```
tukey <- TukeyHSD(aov(lbdldl~bpcat, data = nhanes), conf.level = 0.95)
p16 <- tidy(tukey)
p16
```

```
## # A tibble: 3 x 7
##   term  contrast              null.value estimate conf.low conf.high adj.p.value
##   <chr> <chr>                      <dbl>    <dbl>    <dbl>     <dbl>       <dbl>
## 1 bpcat Hypertension-Elevated          0    2.06     -4.66      8.77       0.752
## 2 bpcat Normal-Elevated                0    2.35     -4.56      9.26       0.705
## 3 bpcat Normal-Hypertension            0    0.288    -5.03      5.61       0.991
```

```
. = ottr::check("tests/p16.R")
```

```
##
## All tests passed!
```

17. What do you conclude from your analysis?

Since all of the confidence intervals contain 0 and none of the p-values are significant at an alpha = 0.05 level, we fail to reject the null hypothesis that the average blood lipid level is equal across blood pressure groups.

Section 3: Non-parametric tests

You are testing the change in test scores following an intensive tutoring session. You have the following data from a small group of students each student is tested before and after the tutoring session. Each row represents one student.

Time 1	Time 2
65	77
87	100
77	75
90	89
70	80
84	81
92	91
83	96
85	84
91	89
68	88
72	100
81	81

```
# This code makes a dataframe of the table you see above
test_scores <- tribble(
  ~time1, ~time2,
  65, 77,
  87, 100,
  77, 75,
  90, 89,
  70, 80,
  84, 81,
  92, 91,
  83, 96,
  85, 84,
  91, 89,
  68, 88,
  72, 100,
  81, 81)
```

18. Calculate the appropriate non-parametric test for these data by hand. Assign your p-value (rounded to 4 decimal places) to the object `p18`.

```
test_scores <- test_scores %>% mutate(diff = time1 - time2)
n <- nrow(test_scores) - 1 # one obs was dropped as the difference was 0
t <- 21
mu <- (n*(n + 1))/4

sigma <- sqrt((n*(n+1)*(2*n+1))/24)

z.stat <- (t - mu)/sigma

p18 <- round(2*pnorm(z.stat),4)
p18
```

```
## [1] 0.1579
```

```
. = ottr::check("tests/p18.R")
```

```
##
## All tests passed!
```

19. Check your work using a function in R. Assign the p-value from your output to the object p19.

```
wilcox.test(test_scores %>% pull(time1),test_scores %>% pull(time2), paired=T, correct=FALSE)
```

```
## Warning in wilcox.test.default(test_scores %>% pull(time1), test_scores %>% :
## cannot compute exact p-value with ties
```

```
## Warning in wilcox.test.default(test_scores %>% pull(time1), test_scores %>% :
## cannot compute exact p-value with zeroes
```

```
##
##  Wilcoxon signed rank test
##
## data:  test_scores %>% pull(time1) and test_scores %>% pull(time2)
## V = 21, p-value = 0.157
## alternative hypothesis: true location shift is not equal to 0
```

```
p19 <- 0.157
p19
```

```
## [1] 0.157
```

```
# YOUR CODE HERE
```

```
. = ottr::check("tests/p19.R")
```

```
##
## All tests passed!
```