STA130H1S – Winter 2024

Week 10 Problem Set

N. Moon & J. Speagle

Instructions

How do I hand in my solutions and how do I check my work

You will submit your solutions (.Rmd and .pdf) on MarkUs at the following link: https://markus4.teach.cs.toronto.edu/2024-01/courses/1 Submissions are due at 5pm on Thursdays; see Quercus page for the specific deadline for each problem set.

Usually when you do an assignment, you don't find out whether your answers are correct until *after* the deadline, when you get your grade back. However, using MarkUs, you can submit your work before the deadline and run tests to check your solutions!

Note: Some parts of some questions may not be covered by tests in MarkUs, but you're still responsible for reviewing the posted solutions and make sure you understand them. Some of these parts will also be graded in some weeks.

Some of the optional questions have tests in MarkUs, but you won't be penalized for not completing these (or failing the tests for these parts) when we grade your work after the submission deadline. The tests for these parts are provided for your guidance.

What to do if a test fails on MarkUs

- Take a deep breath! Your work won't really be graded until the deadline, so start early to make sure you have lots of time to resolve issues before the deadline.
- Read the message to get hints about what the problem is. For example "variable X not present" means that you may have a typo in the name of the variable we're looking for - re-read the question carefully and make sure you're following the instructions.
- Search on Piazza to see if other classmates have encountered a similar error (and if not, consider posting a screenshot of the error message).
- Come to TA or instructor office hours with your issue.

Part 1

Question 1

In a 1965 article, George Moore predicted the number of transistors on processors would double every year. He projected that level of growth would continue for at least another decade. A decade later, in 1975, he revised the forecast to doubling every two years. This is now commonly known as Moore's law.

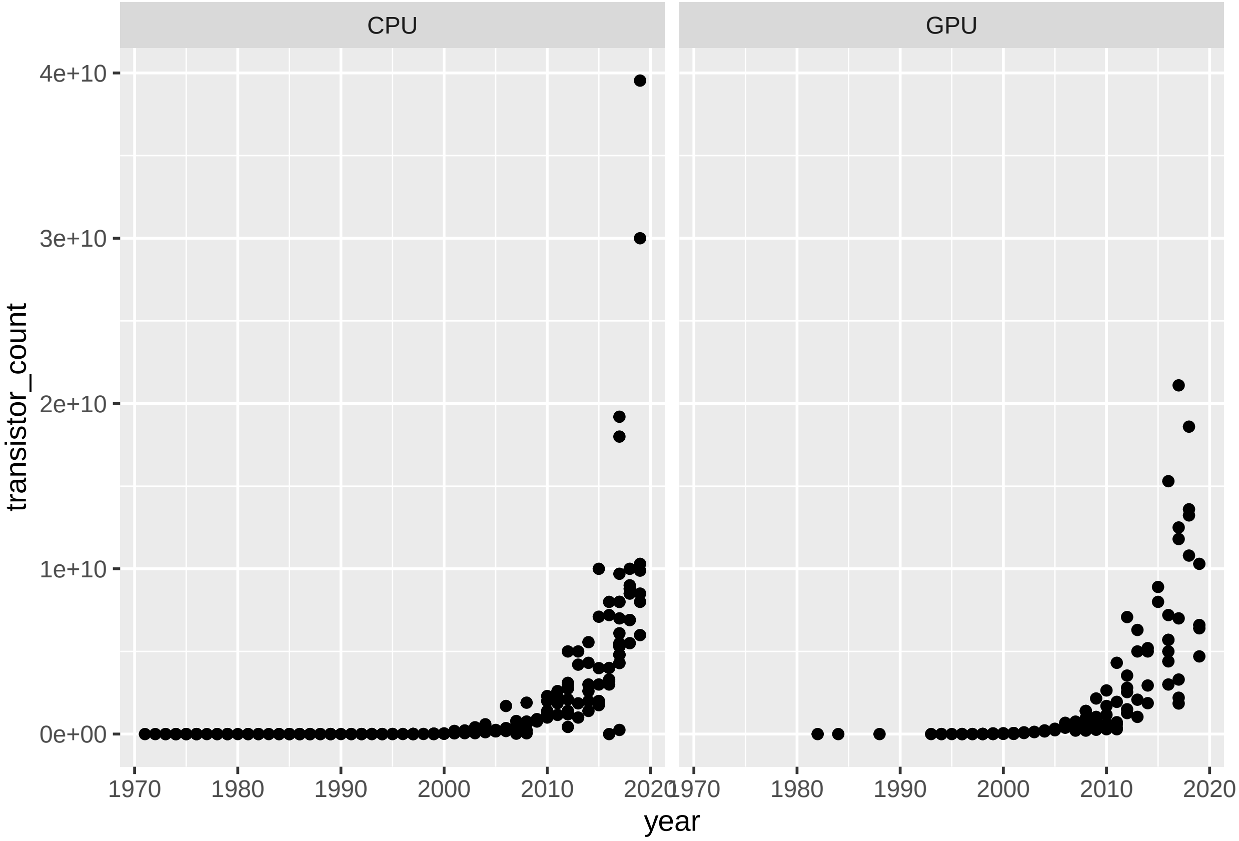
The `processors.csv` dataset contains a sample of data on processors, scraped from Wikipedia about the number of transistors in central processing units (CPUs) and general processing units (GPUs). It also shows the name of the processor and the year it was introduced.

```
processors <- read_csv("processors.csv")
glimpse(processors)
```

```
## Rows: 299
## Columns: 4
## $ processor       <chr> "Intel 4004 (4-bit, 16-pin)", "Intel 8008 (8-bit, 18-~
## $ year            <dbl> 1971, 1972, 1973, 1973, 1974, 1974, 1974, 1974, 1975,~
## $ transistor_count <dbl> 2250, 3500, 2500, 11000, 3000, 4100, 6000, 8000, 4528~
## $ unit_type       <chr> "CPU", "CPU", "CPU", "CPU", "CPU", "CPU", "CPU", "CPU~
```

(a) Create an appropriate plot for the number of transistors per year, faceted by `unit_type`. In other words, create two scatterplots, one for each value of the `unit_type` variable.

```
processors %>% ggplot(aes(x=year, y=transistor_count)) + geom_point() + facet_wrap(~unit_type)
```



(b) Do you believe it is appropriate to fit a straight line through this plot as it is displayed? Replace NULL by your answer ("Yes" or "No") and write a sentence explaining your answer.

```
Q1b_is_straight_line_model_appropriate <- "No"
```

(c) Add a new variable called `log_transistors` to the dataset. You can use `mutate()` and the `log()` function.

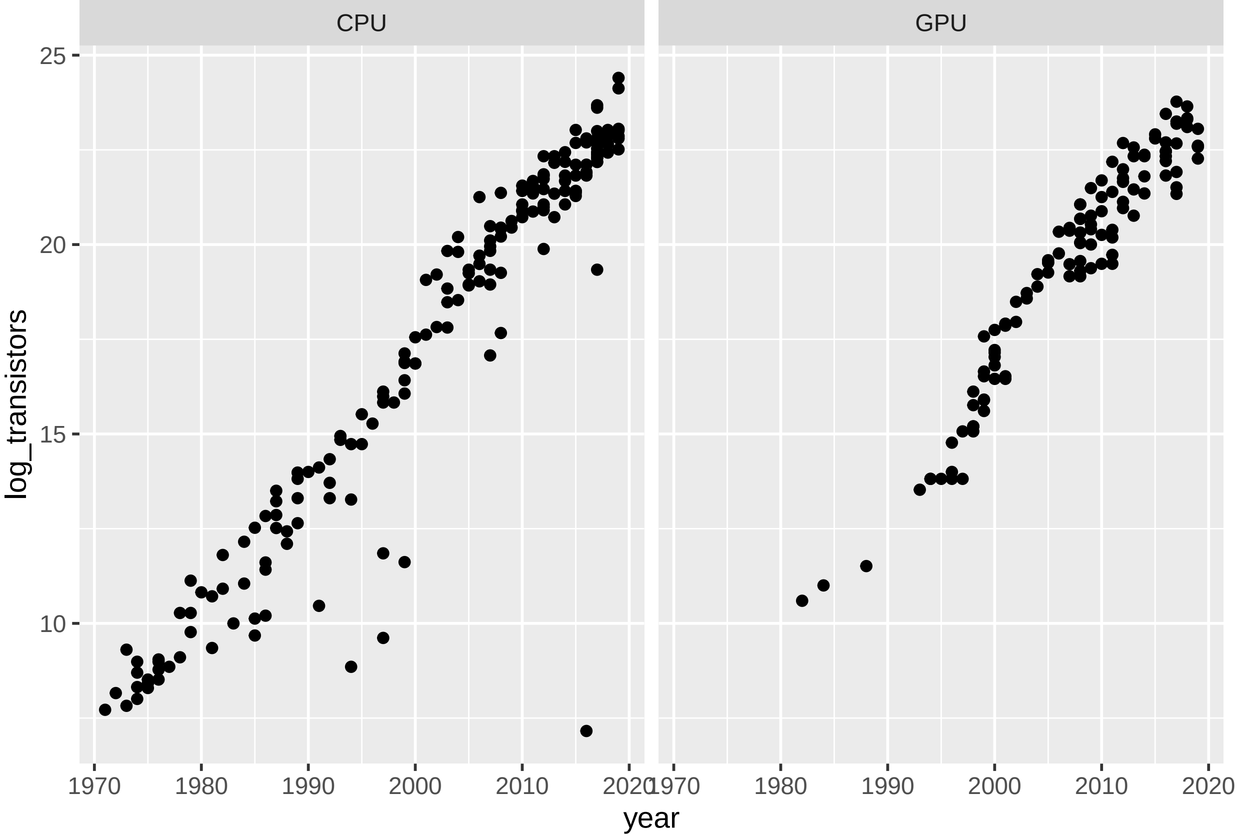
```
# Replace NA by the correct expression in the code below
Q1c_updated_processors_tibble <- processors %>%
  mutate(log_transistors = log(transistor_count))

processors <- Q1c_updated_processors_tibble # DO NOT CHANGE THIS LINE
```

(d) Plot the association between `log_transistors` and `year`, faceted by `unit_type` and use `geom_smooth(se=FALSE, method="lm")` to add a line of best fit to both plots. Describe this association in each plot.

Note: You will learn more about transforming variables in future courses and are not required to be able to explain why we've done this here. You can just treat `log_transistors` as we have other variables in class and refer to it as "the log number of transistors".

```
processors %>% ggplot(aes(x=year, y=log_transistors)) + geom_point() + facet_wrap(~unit_type)
```



(e) Before calculating anything, do you think the correlation is stronger between log transistor count and year for GPUs or CPUs? Justify your answer.

I think the correlation in CPU count is much better. The graph in GPU count is slightly wavy and fans out in the end. The graph in the CPU count, on the other hand, pretty constrained outside of some data points.

(f) Calculate the correlation between `log_transistors` and `year` for CPUs and GPUs. You may find `group_by()`, `summarise()` and `cor()` to be helpful functions. Replace NULL by your answers (you can read them off the summary table), rounded to 2 decimal places.

```
processors %>% group_by(unit_type) %>% summarize(r = cor(year, log_transistors))
```

```
## # A tibble: 2 x 2
##   unit_type     r
##   <chr>     <dbl>
## 1 CPU       0.948
## 2 GPU       0.966
```

```
# Your answers should be simple numbers (you can type the numbers, rounded
# to 2 decimals)
Q1f_CPU_correlation <- 0.95
Q1f_CPU_correlation <- 0.97
```

(g) Write down a simple linear regression model to predict log number of transistors in a processor based on the year it was introduced. Be sure to explain each term in the model.

$$\log(y) = 0.3357x - 654.4899$$

Hint: If you copy math equations from another software into your .Rmd document, you'll get errors when trying to knit. Instead, you should type your math equations directly in your .Rmd document. Here are some tips and examples for doing this:

1. In a .Rmd document, math equations and symbols must be typed between dollar symbols ($).
2. If you want your equation/symbol to appear in the middle of a sentence, use only one dollar sign before and one dollar sign after. For example, we can typeset beta-hat-0 in .Rmd as $\hat{\beta}_0$.
3. If you want your equation to appear on a line on its own, type it on a separate line and put two dollar signs at the begining and the end. For example,

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1$$

4. A few other useful symbols you may need in this question are epsilon (ϵ), "not equal" ($\neq$), and superscripts (e.g. i^{th}).

(h) State the null and alternative hypotheses you would use assess whether year is a useful predictor of the log number of transistors in this linear regression model.

H_0: R^2 <= 0.5 H_a: R^2 > 0.5

(i) Restrict your data to CPUs and use R to fit the linear model that corresponds with your line of best fit above. Report the fitted equation of the line. Interpret the regression coefficients in the context of this data AND make a conclusion about the hypotheses you defined above.

```
summary(lm(log_transistors ~ year, processors %>% filter(unit_type == "CPU")))$r.square
```

```
## [1] 0.8989549
```

(j) Briefly explain why or why not the interpretation of the intercept is helpful for understanding Moore's Law.

The interpretation isn't very helpful since we are using a log model.

(k) Extract the R^2 for your model, and write one sentence interpreting it in context.

0.8989549

Question 2 (Adapted from Exercise 7.18 in Dietz, Barr, Cetinkaya-Rundel, "OpenIntro Statistics", Second Edition) [Optional]

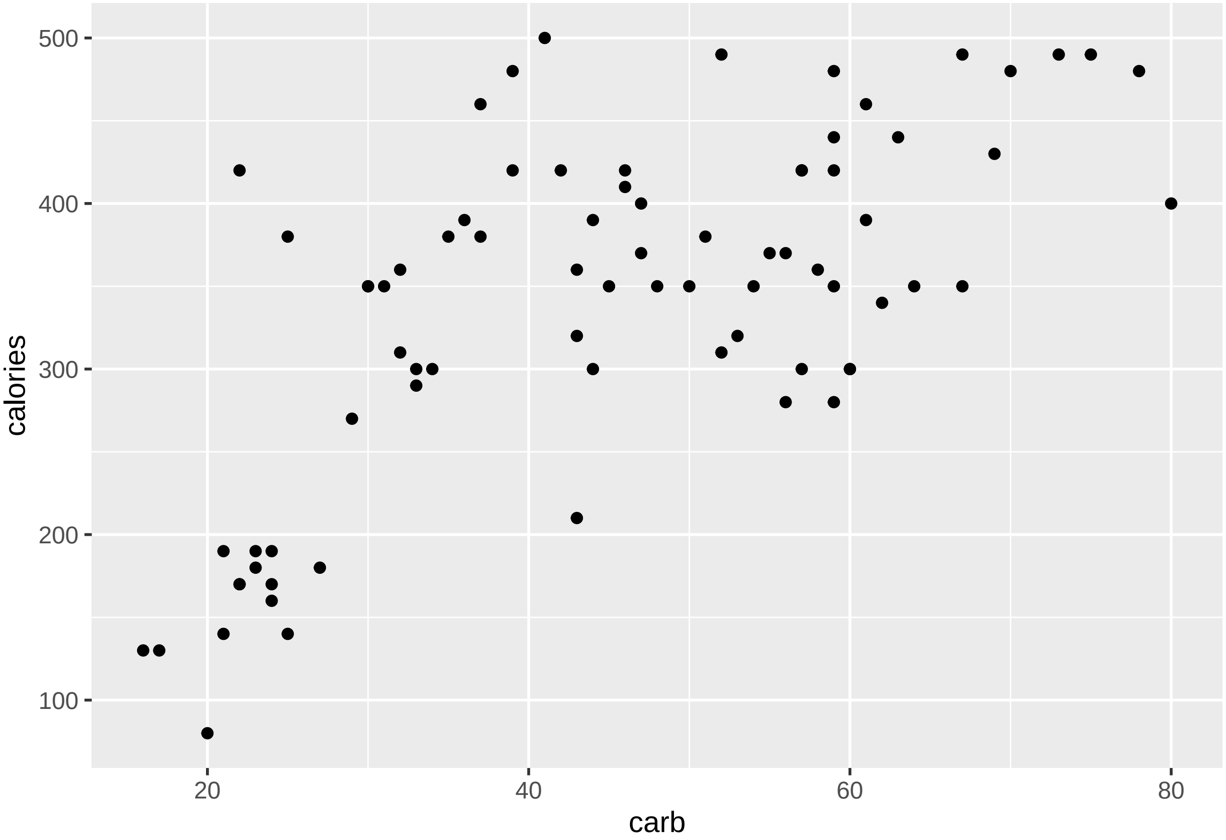
The `starbucks.csv` dataset contains data on calories and carbohydrates (in grams) in Starbucks food menu items.

```
starbucksdata<-read_csv("starbucks.csv")
glimpse(starbucksdata)
```

```
## Rows: 77
## Columns: 7
## $ item     <chr> "8-Grain Roll", "Apple Bran Muffin", "Apple Fritter", "Banana~
## $ calories <dbl> 350, 350, 420, 490, 130, 370, 460, 370, 310, 420, 380, 320, 3~
## $ fat      <dbl> 8, 9, 20, 19, 6, 14, 22, 14, 18, 25, 17, 12, 17, 21, 5, 18, 1~
## $ carb     <dbl> 67, 64, 59, 75, 17, 47, 61, 55, 32, 39, 51, 53, 34, 57, 52, 7~
## $ fiber    <dbl> 5, 7, 0, 4, 0, 5, 2, 0, 0, 0, 2, 3, 2, 2, 3, 3, 2, 3, 0, 2, 0~
## $ protein  <dbl> 10, 6, 5, 7, 0, 6, 7, 6, 5, 7, 4, 6, 5, 5, 12, 7, 8, 6, 0, 10~
## $ type     <chr> "bakery", "bakery", "bakery", "bakery", "bakery", "bakery", "~
```

(a) Produce a plot that shows the association between carbohydrates and calories in Starbucks menu items. Describe this association.

```
starbucksdata %>% ggplot(aes(x=carb, y=calories)) + geom_point()
```



(b) Before calculating anything, estimate the correlation coefficient between carbohydrates and calorie content in Starbucks menu items based on the plot you produced in (a). Justify your answer.

The correlation coefficient is most likely very low.

(c) Calculate the correlation between carbohydrate and calorie content of Starbucks menu items. How does this compare to your estimate in part (b)?

```
cor(starbucksdata$carb, starbucksdata$calories)
```

```
## [1] 0.674999
```

(d) Write down a simple linear regression model to predict calories based on carbohydrate content of Starbucks menu items. Be sure to explain each term in the model.

(e) State the null hypothesis and alternative hypothesis you would use to assess whether the slope of the linear association between these two variables is different from 0.

(f) Use R to fit the regression model in (d) to these data. Report the fitted regression line. Interpret the regression coefficients in the context of this study AND make a conclusion about the hypotheses you defined above.

(g) Add the estimated linear regression line that you calculated in (f) to the plot you generated in (a). Compute the coefficient of determination, R^2. How well does the linear regression line seem to capture the relationship between `carb` and `calories`? Justify your answer.

(h) Based on the Starbucks data, create a new dataset called `starbucks_lunch` which only contains food items which are of one of two types: "sandwich" and "bistro box". Create a boxplot comparing the distribution of calories for these two types of items.

(i) Fit a linear regression model to test whether there is a difference in mean calories for items of type "bistro box" and items of type "sandwich". Write a sentence summarizing your conclusion.