Assignment 9

Housekeeping

```
library(modelr)
library(ggplot2)
library(tidyverse)
library(data.table)
library(purrr)
```

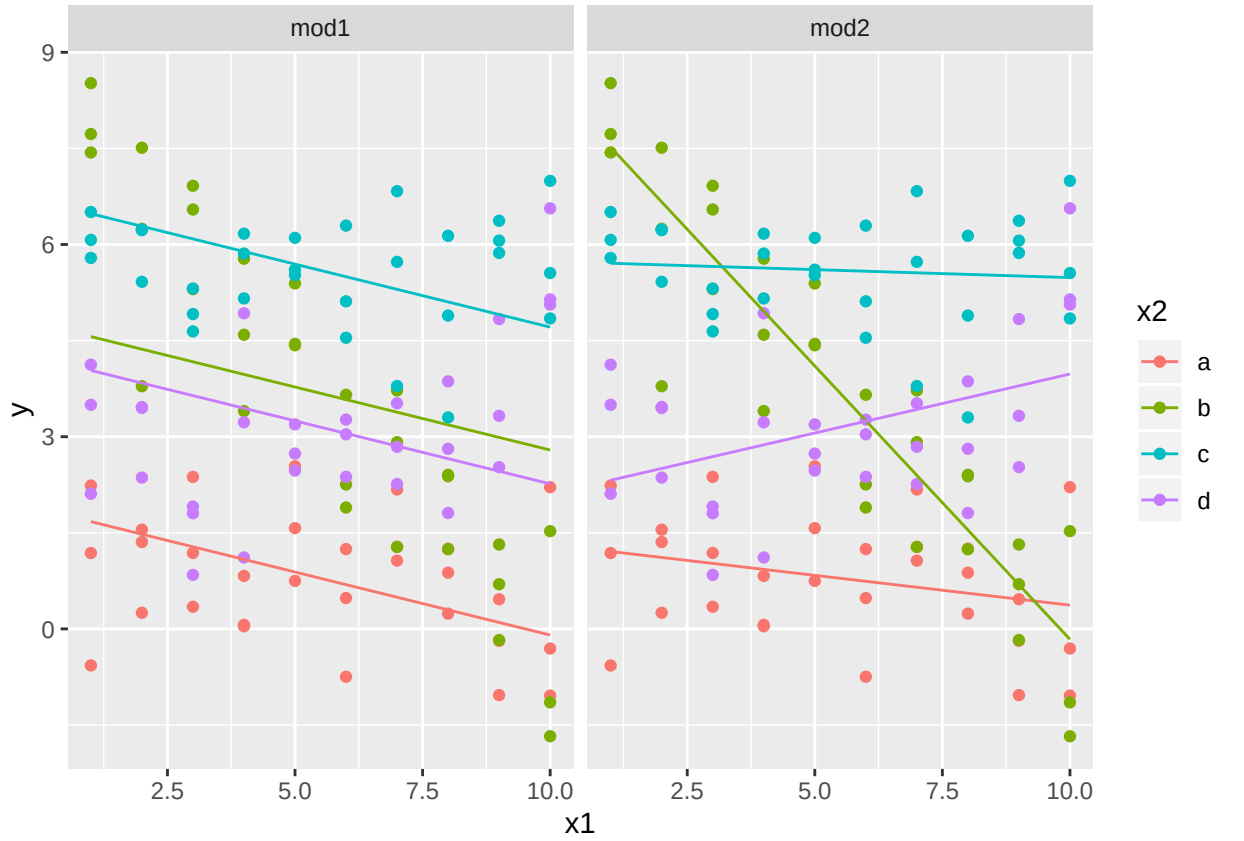
Exercise 1:

Map each coefficient from mod1 and mod2 to a feature of the plot with two facets. For instance, what is x1 in summaryd(mod2)? Where could you read it off (roughly) from the graph? Etc for x1:x2b and so on. If you get stuck, do ask for specific questions on Discourse. Correct answers for any parameter look like this: x1 is the [slope/intercept/difference between slopes/intercepts of] for ... Since it is [positive/negative] this means that ... is [larger/smaller] than ...

```
mod1 <- lm(y ~ x1 + x2, data = sim3)
mod2 <- lm(y ~ x1 * x2, data = sim3)



predicted_data_mod1 <- sim3 %>% add_predictions(mod1) %>% add_residuals(mod1) %>% mutate(model = "mod1")
predicted_data_mod2 <- sim3 %>% add_predictions(mod2) %>% add_residuals(mod2) %>% mutate(model = "mod2")
predicted_data_combined <- rbind(predicted_data_mod1, predicted_data_mod2)
#
predicted_data_combined %>% ggplot(aes(x1, y, colour = x2)) +
  geom_point() +
  geom_line(data = predicted_data_combined, aes(y = pred)) +
  facet_wrap(~model)
```




```
mod1$coefficients
```

```
## (Intercept)          x1         x2b         x2c         x2d
##   1.8716659  -0.1967378   2.8878108   4.8057359   2.3595867
```

```
mod2$coefficients
```

```
## (Intercept)          x1         x2b         x2c         x2d      x1:x2b
##  1.30124266 -0.09302444  7.06937991  4.43089525  0.83455115 -0.76028528
##      x1:x2c      x1:x2d
##  0.06815284  0.27727920
```

Ans: Among the coefficient values from **mod1**; Intercept is the y-intercept of the line x2a and x1 is the slope of all of the 4 lines. x2b is the advantage/premium of each value in the line x2b over each value i x2a - when x2b is to added to intercept it equals the y-intercept of the line x2b (~4.75). x2c is the advantage/premium of each value in the line x2c over each value i x2a - when x2c is to added to intercept it equals the y-intercept of the line x2c (~6.67). x2d is the advantage/premium of each value in the line x2d over each value i x2a - when x2d is to added to intercept it equals the y-intercept of the line x2d (~4.23).

Among the coefficents from **mod2**; Intercept is the y-intercept of the line x2a and x1 is the slope of this line x2a. x2b is the advantage/premium of each value in the line x2b over each value i x2a - when x2b is to added to intercept it equals the y-intercept of the line x2b (~8.37). x1:x2b is the difference in the slopes of the lines x2a and x2b, when x1 is added to x1:x2b it forms the slope of the line x2b (~ -0.85) x2c is the advantage/premium of each value in the line x2c over each value i x2a - when x2c is to added to intercept it equals the y-intercept of the line x2c (~5.73). x1:x2c is the difference in the slopes of the lines x2a and x2c, when x1 is added to x1:x2c it forms the slope of the line x2c (~ -0.03) x2d is the advantage/premium of each value in the line x2d over each value i x2a - when x2d is to added to intercept it equals the y-intercept of the line x2d (~2.23). x1:x2d is the difference in the slopes of the lines x2a and x2d, when x1 is added to x1:x2d it

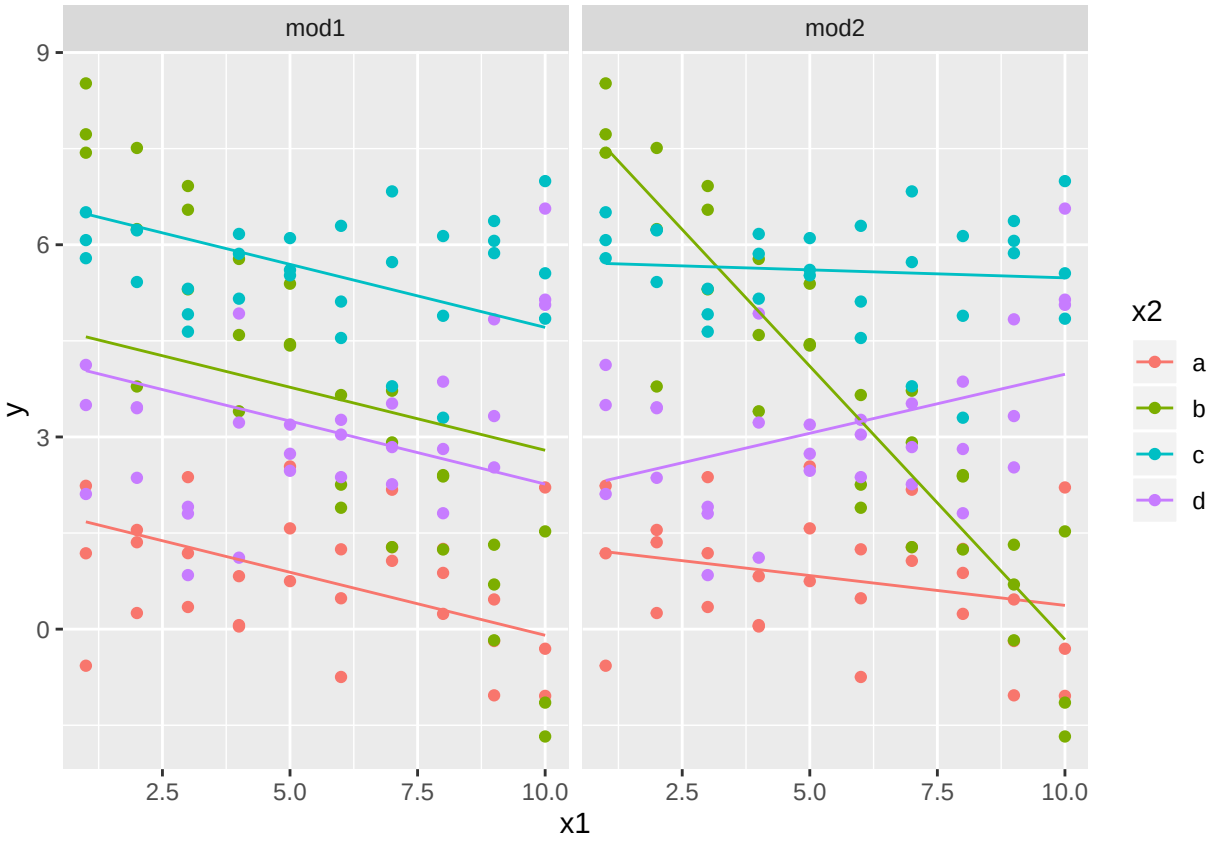
forms the slope of the line x2d (~ 0.18)

Exercise 2:

Do the faceting with gather_predictions and if needed with data_grid. Look at chapter 23 for help.

```
grid <- sim3 %>%
  data_grid(x1, x2) %>%
  gather_predictions(mod1, mod2)

ggplot(sim3, aes(x1, y, colour = x2)) +
  geom_point() +
  geom_line(data = grid, aes(y = pred)) +
  facet_wrap(~ model)
```



Exercise 3: Read/Skim 21.2, 21.3, and 21.4 so you are aware of some issues.

Pick a short example from the notes that you feel you want to understand better and use some other use case to illustrate it (using the Vienna data, or diamonds, or the same but in a different way.)

Sol:I did Excercise 21.3.5.3 for this question: Write a function that prints the mean of each numeric column in a data frame, along with its name. For example, show_mean(iris) would print.

```
files <- dir("data/", pattern = "\\.csv$", full.names = TRUE)

read_files <- function(file){
  df <- read_csv(file)
}
```

```r
read_files_df <- lapply(files, read_files)

merged_files_df <- rbindlist(read_files_df)


show_mean <- function(df) {
output <- vector("double", ncol(df))

for(i in seq_along(df)){
  if(is_numeric(df[[i]])){
    output[[i]] <- mean(df[[i]])
  }
}

for(i in seq_along(output)){
  if(!is.na(output[[i]])){
      cat(paste0(names(df)[i], ": ", round(mean(df[[i]]), 2)), fill = TRUE)
    }
  }
}

show_mean(mtcars)
```

```
## mpg: 20.09
## cyl: 6.19
## disp: 230.72
## hp: 146.69
## drat: 3.6
## wt: 3.22
## qsec: 17.85
## vs: 0.44
## am: 0.41
## gear: 3.69
## carb: 2.81
```