Pairs Plots

September 10, 2018

One of my favorite data sets contains a bunch of information about movies, including how the movie scores on the Bechdel test. A movie passes the Bechdel test if it satisfies 3 rules:

1. it has at least two women;
2. the women talk to each other; and
3. they talk to each other about something or someone other than a man.

The full data set contains the following variables:

- `year` is the year the movie was released
- `title` is the title of the movie
- `bechdel_test` is a version of the results of the Bechdel test with 5 categories, according to users of the website www.bechdeltest.com: "nowomen" means there are not at least two women in the movie; "notalk" means there are at least two women in the movie, but they don't talk to each other; "men" means there are at least two women in the movie, but they only talk to each other about men; "dubious" means there was some disagreement among users of bechdeltest.com about whether or not the movie passed the test; and "ok" means that the movie passes the Bechdel test.
- `bechdel_test_binary` is a the results of the Bechdel test with 2 categories, according to the users of the website www.bechdeltest.com: "PASS" means that the movie passed the test (i.e., its value for `bechdel_test` is "ok"); "FAIL" means it did not pass the test (i.e., its value for `bechdel_test` is something other than "ok")
- `budget` is the movie's approximate production budget, in the dollars of the year the movie was made.
- `domgross` is the movie's domestic gross earnings (i.e., total earnings from the U.S.), in the dollars of the year the movie was made.
- `intgross` is the movie's combined domestic and international gross earnings (i.e., total earnings both in the U.S. and internationally), in the dollars of the year the movie was made.
- `budget_2013` is the same as `budget` but in inflation-adjusted 2013 dollars.
- `domgross_2013` is the same as `domgross`, but in inflation-adjusted 2013 dollars.
- `intgross_2013` is the same as `intgross`, but in inflation-adjusted 2013 dollars.
- `imdb_rating` is the average rating for the movie by users of the website www.imdb.com, on a scale of 0 to 10 (higher ratings are better)
- `num_imdb_ratings` is the number of distinct users of www.imdb.com who have rated the movie.
- `mpaa_rating` is the MPAA rating for the movie, like PG or R.
- `run_time_min` is the length of the movie in minutes.

First, let's load the data in, filter to include only MPAA ratings categories with a reasonable number of movies in them, and set categorical variables to factors. If anything in the following R code isn't clear to you, you should ask about it.

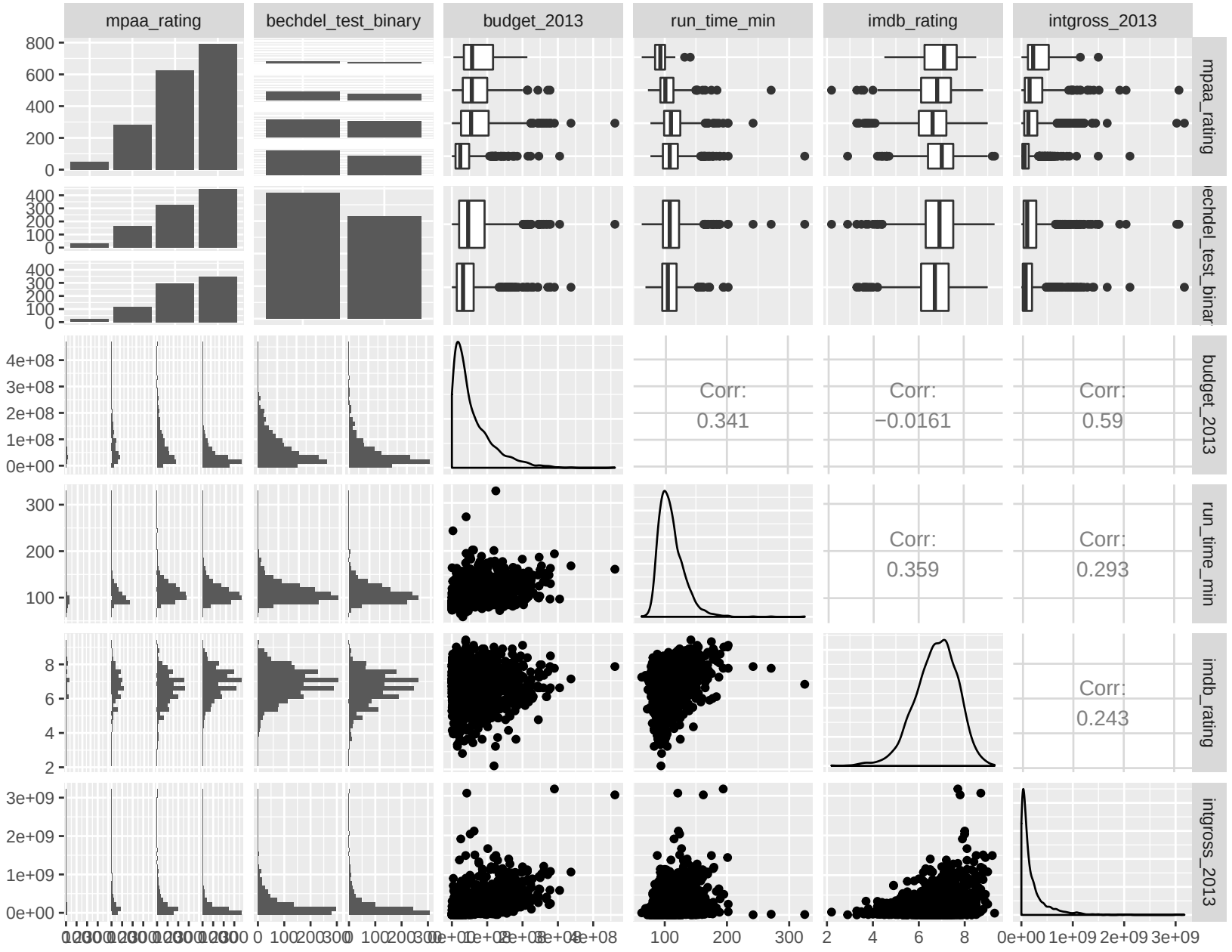
```
library(readr)
library(dplyr)

movies <- read_csv("http://www.evanlray.com/data/bechdel/bechdel.csv") %>%
  filter(mpaa_rating %in% c("G", "PG", "PG-13", "R")) %>%
  mutate(
    bechdel_test = factor(bechdel_test, levels = c("nowomen", "notalk", "men", "dubious", "ok"), ordered = TRUE),
    bechdel_test_binary = factor(bechdel_test_binary, levels = c("FAIL", "PASS"), ordered = TRUE),
    mpaa_rating = factor(mpaa_rating, levels = c("G", "PG", "PG-13", "R"), ordered = TRUE)
  )
```

Over the course of the next few days, we will try to build some models for a movie's international gross earnings in inflation-adjusted 2013 dollars (`intgross_2013`) based on the following 5 explanatory variables:

1. `budget_2013`
2. `run_time_min`
3. `imdb_rating`
4. `mpaa_rating`
5. `bechdel_test_binary`

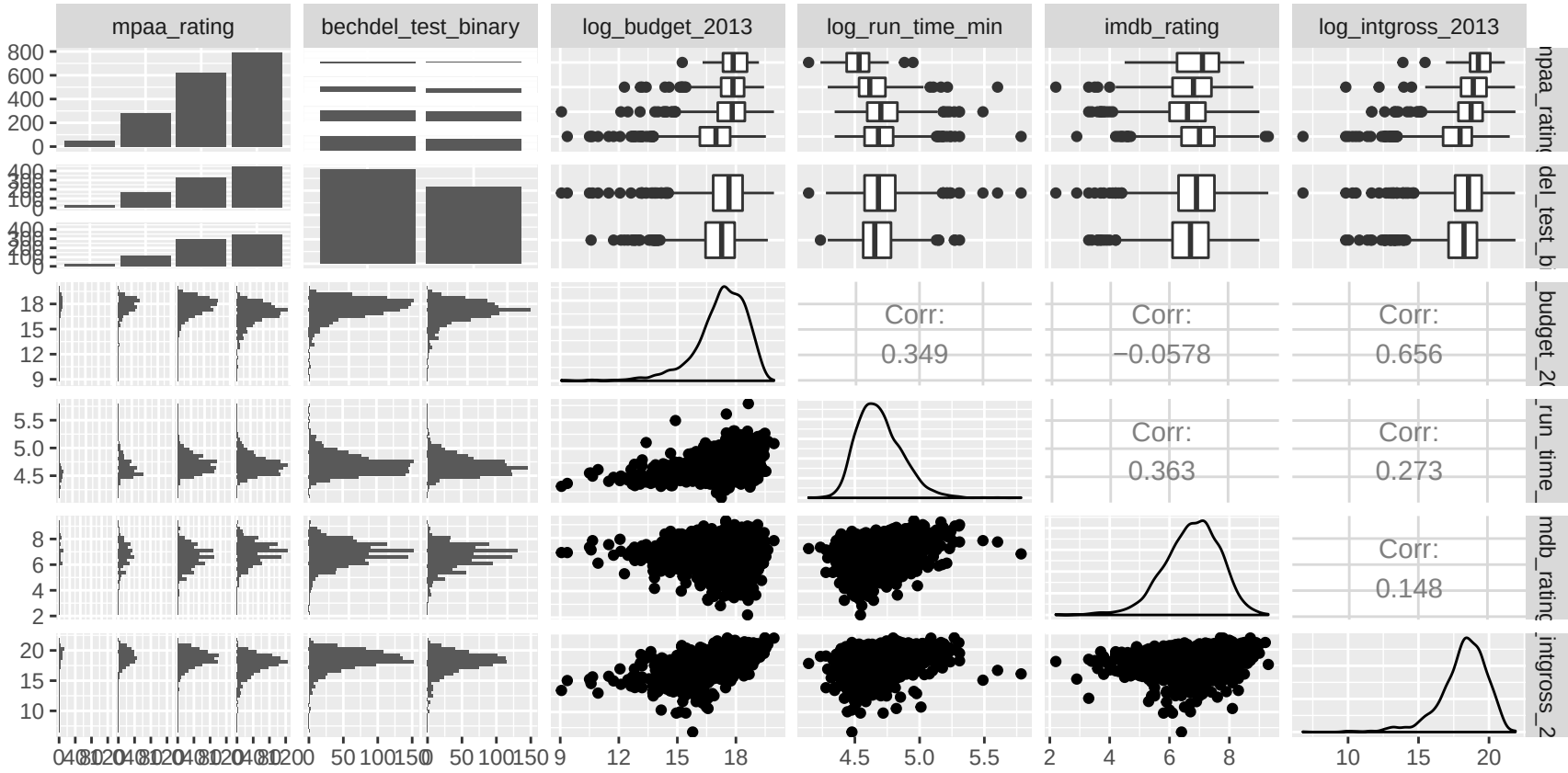
The first thing to do is always to make some plots. When we're thinking about multiple regression type problems with a reasonably small number of variables, the go-to plot is a pairs plot.

```
library(ggplot2) # general plotting functionality
library(GGally) # includes the ggpairs function, pairs plots via ggplot2

# I find it helpful to put the response variable last in this list, and to put quantitative variables next to each other.
vars_to_use <- c("mpaa_rating", "bechdel_test_binary", "budget_2013", "run_time_min", "imdb_rating", "intgross_2013")
ggpairs(movies[, vars_to_use])
```


mpaa_rating
bechdel_test_binary
budget_2013
run_time_min
imdb_rating
intgross_2013
Corr:
0.341
Corr:
−0.0161
Corr:
0.59
Corr:
0.359
Corr:
0.293
Corr:
0.243

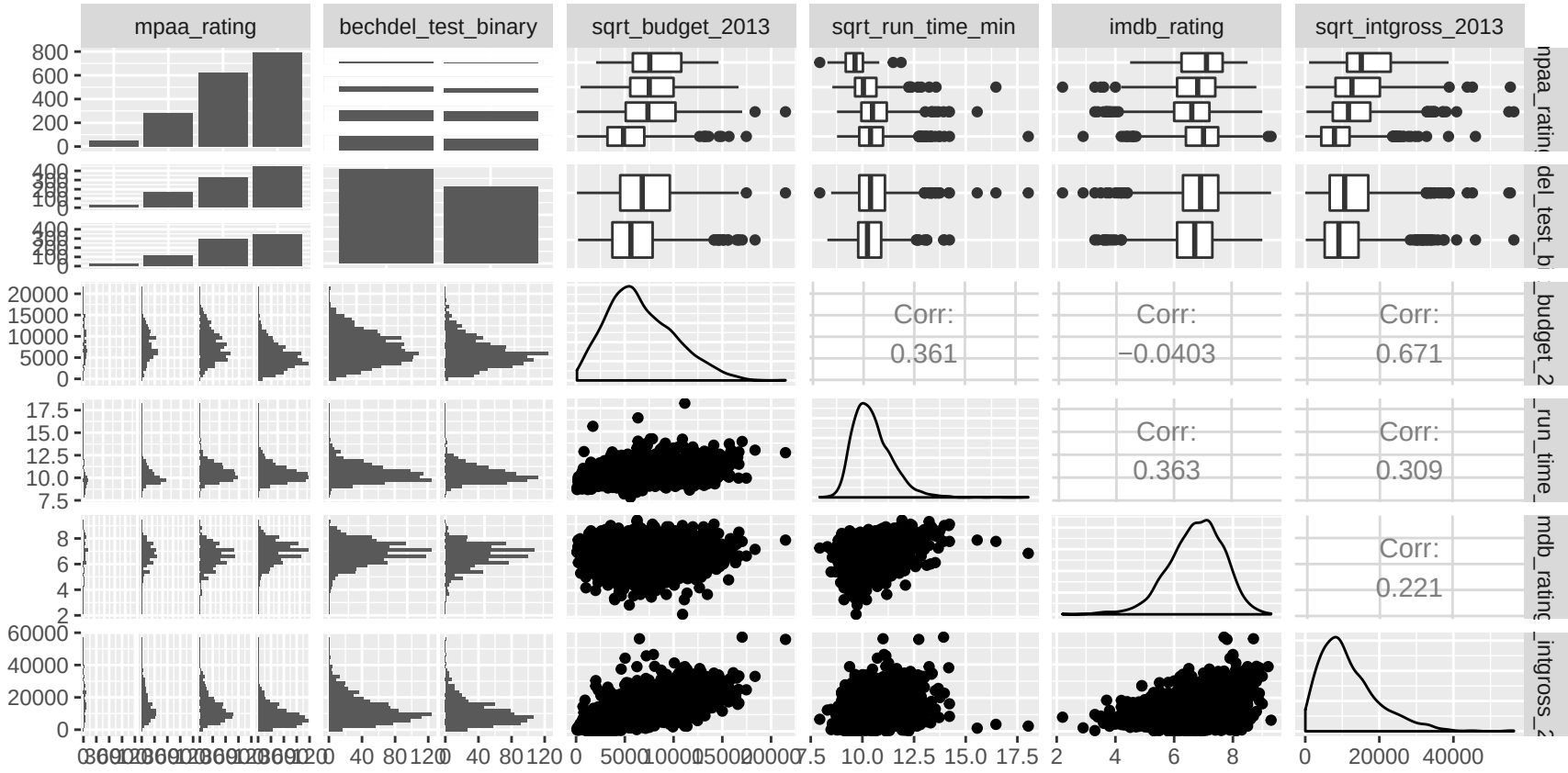
We will need to try some transformations if we want to fit a linear model with normally distributed residuals. The most common option in cases like this is log, followed by square root. Let's try a log transformation first:

```
movies <- movies %>%
  mutate(
    log_intgross_2013 = log(intgross_2013),
    log_budget_2013 = log(budget_2013),
    log_run_time_min = log(run_time_min)
  )

vars_to_use <- c("mpaa_rating", "bechdel_test_binary", "log_budget_2013", "log_run_time_min", "imdb_rating", "log_intgross_2013")
ggpairs(movies[, vars_to_use])
```



Too far. How about square root?

```
movies <- movies %>%
  mutate(
    sqrt_intgross_2013 = sqrt(intgross_2013),
    sqrt_budget_2013 = sqrt(budget_2013),
    sqrt_run_time_min = sqrt(run_time_min)
  )

vars_to_use <- c("mpaa_rating", "bechdel_test_binary", "sqrt_budget_2013", "sqrt_run_time_min", "imdb_rating", "sqrt_intgross_2013")
ggpairs(movies[, vars_to_use])
```



Not far enough. We'll investigate more options for transformations another day.