Multiple Regression with Quantitative Explanatory Variables

September 12, 2018

Recall that we are thinking about a data set with several variables recorded about 1753 movies. Over the next week or so, we will explore building multiple regression 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`

Today, we'll look at just a few models based on the three quantitative explanatory variables, `budget_2013`, `run_time_min`, and `imdb_rating`.

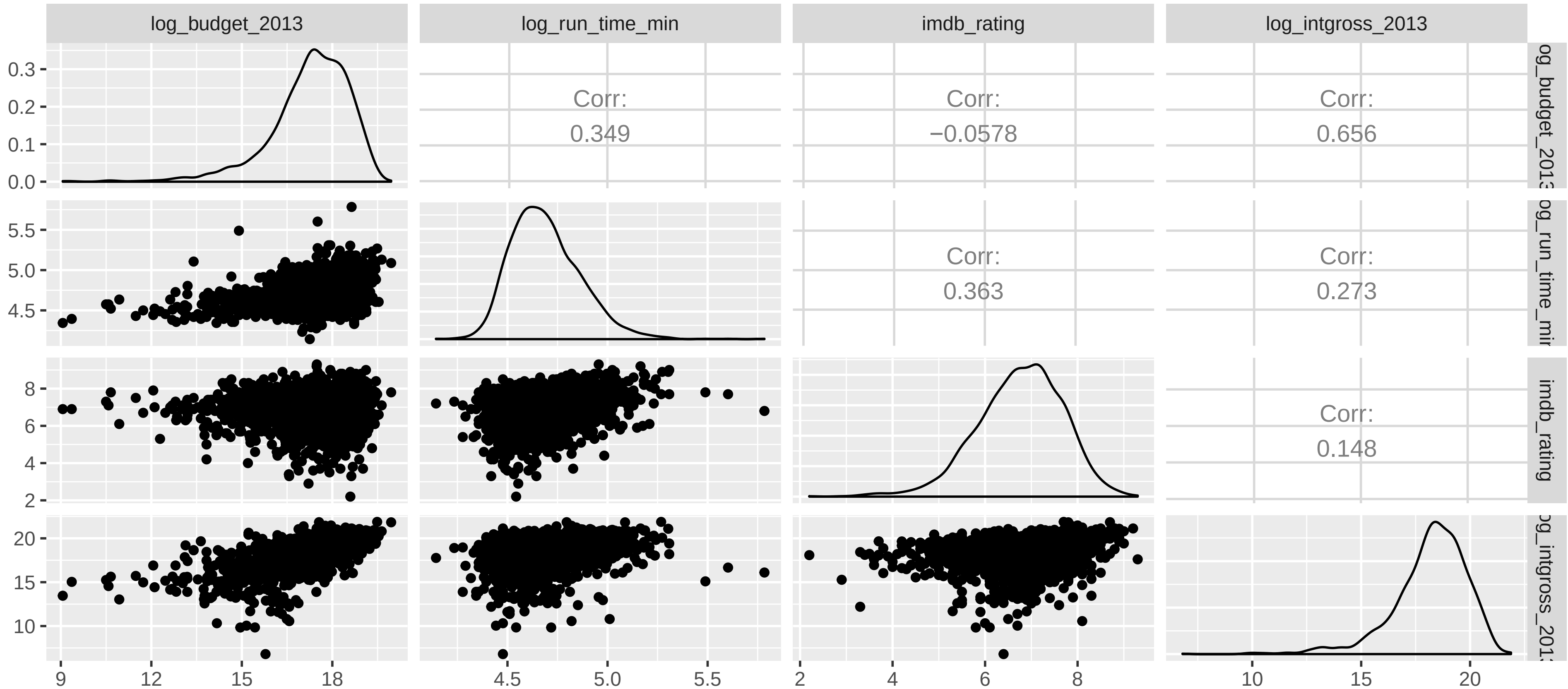
First, let's load the data in, filter to include only MPAA ratings categories with a reasonable number of movies in them, and set categorical variables to factors. If anything in the following R code isn't clear to you, you should ask about it.

```
library(readr)
library(dplyr)
library(ggplot2) # general plotting functionality
library(GGally) # includes the ggpairs function, pairs plots via ggplot2
library(gridExtra) # for grid.arrange, which arranges the plots next to each other

options(na.action = na.exclude)


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)
  )
```

Last class we looked at a pairs plot and decided some transformations were needed to make a linear model feasible. Neither a log transformation nor a square root transformation were great, but a log transformation will be good enough for our explorations today (again, we'll return to a more complete discussion of transformations later).

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

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



For today, let's fit a few quick linear models based on the log transformation.

There are 3 main things to get out of this:

1. Coefficient estimates, interpretations, and hypothesis test results for a variable depend on what other variables are in the model
2. Multiple linear regression can be used to fit surfaces other than planes (i.e., curved surfaces). We'll see how to do this with either:
    a. higher-degree terms in one of the explanatory variables
    b. interactions between two explanatory variables
3. t tests (about one coefficient's value) vs. F tests (simultaneous test about the values of multiple coefficients)

Concept 1: Coefficient estimates, interpretations, and hypothesis test results for a variable depend on what other variables are in the model!

Fit 1: Explanatory variable is run time

```
fit_run_time <- lm(log_intgross_2013 ~ log_run_time_min, data = movies)
summary(fit_run_time)
```

```
##
## Call:
## lm(formula = log_intgross_2013 ~ log_run_time_min, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.8147  -0.7438   0.2203   1.0939   3.5112
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        5.6707     1.0667   5.316  1.2e-07 ***
## log_run_time_min   2.6679     0.2272  11.742  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.633 on 1717 degrees of freedom
##   (34 observations deleted due to missingness)
## Multiple R-squared:  0.07433,    Adjusted R-squared:  0.07379
## F-statistic: 137.9 on 1 and 1717 DF,  p-value: < 2.2e-16
```

- Does a movie's (log) run time explain a statistically significant amount of variation in a movie's earnings? Conduct a hypothesis test.

- Interpret the coefficient estimate for (log) run time.

Fit 2: Explanatory variables are budget, imdb rating, and run time

```r
fit_all_x <- lm(log_intgross_2013 ~ log_budget_2013 + imdb_rating + log_run_time_min, data = movies)
summary(fit_all_x)
```

```
##
## Call:
## lm(formula = log_intgross_2013 ~ log_budget_2013 + imdb_rating +
##     log_run_time_min, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.0298  -0.5455   0.1202   0.7109   4.7260
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       2.29181    0.81810   2.801  0.00515 **
## log_budget_2013   0.87806    0.02464  35.636  < 2e-16 ***
## imdb_rating       0.36915    0.03527  10.467  < 2e-16 ***
## log_run_time_min -0.37597    0.20104  -1.870  0.06164 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.237 on 1715 degrees of freedom
##   (34 observations deleted due to missingness)
## Multiple R-squared:  0.4697, Adjusted R-squared:  0.4688
## F-statistic: 506.4 on 3 and 1715 DF,  p-value: < 2.2e-16
```

- Does a movie's (log) run time explain a statistically significant amount of variation in a movie's earnings after accounting for the linear associations between log budget, imdb rating, and log earnings? Conduct a hypothesis test.

- Interpret the coefficient estimate for (log) run time

- Compare what we learned about the value of run time for predicting earnings from fits 1 and 2. Are the findings consistent?

Concept 2: Surfaces other than planes!

First, let's verify that a "standard" multiple linear regression fit gives us a plane. Then we'll see two specific examples of fitting something other than a plane.

Fit 3: Explanatory variables are budget and IMDB rating - fitting a plane

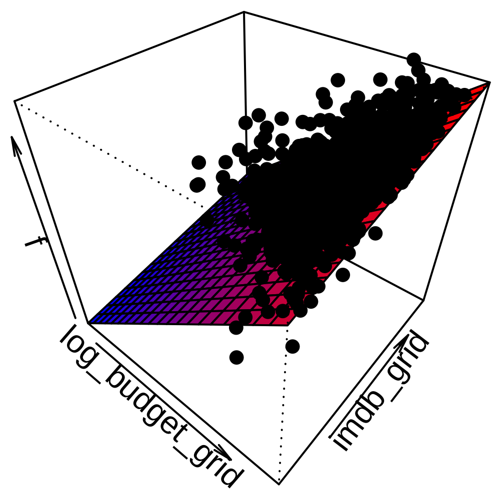
```
fit_budget_imdb <- lm(log_intgross_2013 ~ log_budget_2013 + imdb_rating, data = movies)
summary(fit_budget_imdb)
```

```
##
## Call:
## lm(formula = log_intgross_2013 ~ log_budget_2013 + imdb_rating,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.9917 -0.5506  0.1280  0.7075  4.7186
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      1.01674    0.45803    2.22   0.0266 *
## log_budget_2013  0.85844    0.02253   38.10   <2e-16 ***
## imdb_rating      0.34780    0.03207   10.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.238 on 1743 degrees of freedom
##   (7 observations deleted due to missingness)
## Multiple R-squared:  0.4664, Adjusted R-squared:  0.4658
## F-statistic: 761.7 on 2 and 1743 DF,  p-value: < 2.2e-16
```

- What's the equation of the estimated surface?

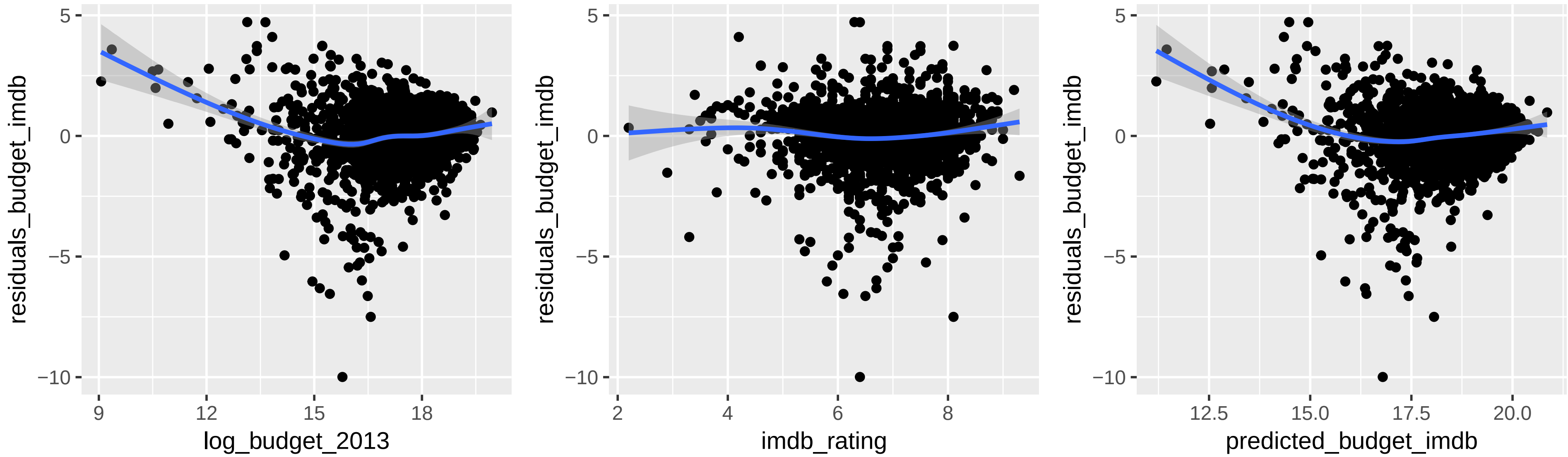
Here's a plot of our estimated surface and the data points (I will never ask you to make a plot like this):

```
log_budget_grid <- seq(from = min(movies$log_budget_2013), to = max(movies$log_budget_2013), length = 21)
imdb_grid <- seq(from = min(movies$imdb_rating), to = max(movies$imdb_rating), length = 21)

f = outer(log_budget_grid, imdb_grid, function(log_budget, imdb) {
  predict(fit_budget_imdb, data.frame(log_budget_2013 = log_budget, imdb_rating = imdb))
})

nrz <- nrow(f)
ncz <- ncol(f)
# Create a function interpolating colors in the range of specified colors
jet.colors <- colorRampPalette( c("blue", "red") )
# Generate the desired number of colors from this palette
nbcol <- 100
color <- jet.colors(nbcol)
# Compute the z-value at the facet centres
ffacet <- f[-1, -1] + f[-1, -ncz] + f[-nrz, -1] + f[-nrz, -ncz]
# Recode facet z-values into color indices
facetcol <- cut(ffacet, nbcol)

res <- persp(log_budget_grid, imdb_grid, f, col = color[facetcol], theta = 40, phi = 40)
points(trans3d(movies$log_budget_2013, movies$imdb_rating, movies$log_intgross_2013, pmat = res), col = 1, pch = 16)
```



OK… let's look at residual diagnostic plots for this model.

```r
movies <- movies %>%
  mutate(
    residuals_budget_imdb = residuals(fit_budget_imdb),
    predicted_budget_imdb = predict(fit_budget_imdb)
  )

p1 <- ggplot(data = movies, mapping = aes(x = log_budget_2013, y = residuals_budget_imdb)) +
  geom_point() +
  geom_smooth()

p2 <- ggplot(data = movies, mapping = aes(x = imdb_rating, y = residuals_budget_imdb)) +
  geom_point() +
  geom_smooth()

p3 <- ggplot(data = movies, mapping = aes(x = predicted_budget_imdb, y = residuals_budget_imdb)) +
  geom_point() +
  geom_smooth()

grid.arrange(p1, p2, p3, nrow = 1)
```



The residual diagnostic plots give an appearance of a lack of linearity in the effect of `log_budget_2013` on `log_intgross_2013`. This is driven by a small number of movies with low budgets. It's not clear to me whether our sample size in that region is really large enough to give strong/reliable evidence about a non-linear effect. However, let's consider a fit with a quadratic term in `log_budget_2013`.

Concept 2 (a): Adding polynomial terms in one variable

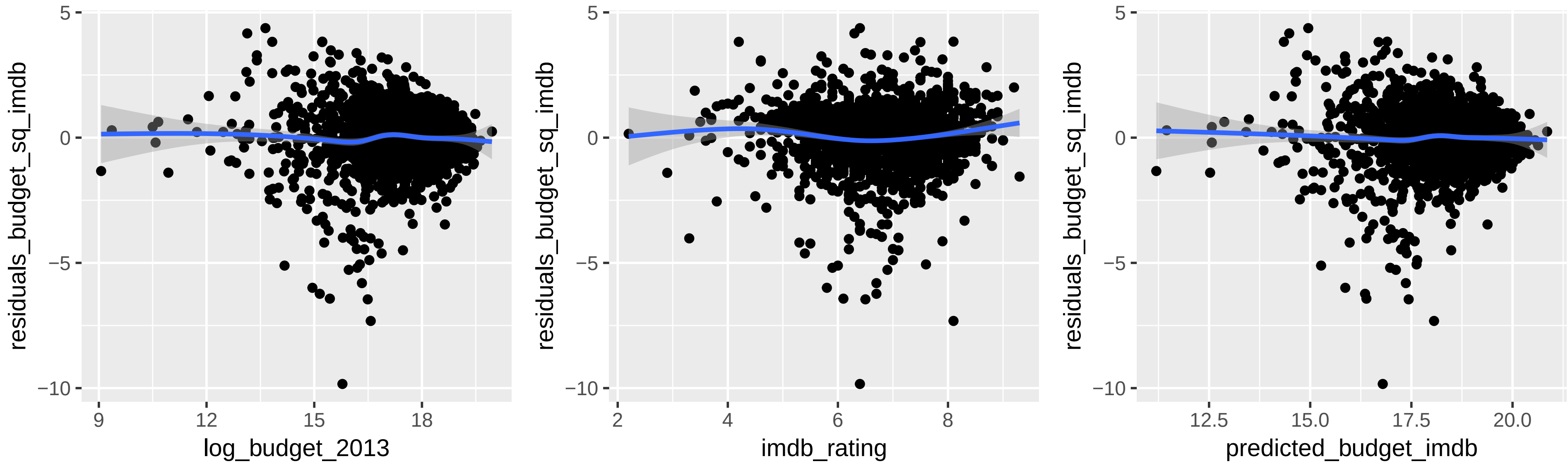
Fit 4: Explanatory variables are budget (quadratic effect) and IMDB rating (linear effect)

```
fit_budget_sq_imdb <- lm(log_intgross_2013 ~ log_budget_2013 + I(log_budget_2013^2) + imdb_rating, data = movies)
summary(fit_budget_sq_imdb)
```

```
##
## Call:
## lm(formula = log_intgross_2013 ~ log_budget_2013 + I(log_budget_2013^2) +
##     imdb_rating, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.8337 -0.5342  0.0967  0.6739  4.3710
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          19.834596   2.330134   8.512  < 2e-16 ***
## log_budget_2013      -1.462479   0.282857  -5.170 2.61e-07 ***
## I(log_budget_2013^2)  0.070942   0.008619   8.230 3.61e-16 ***
## imdb_rating           0.345210   0.031479  10.966  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.215 on 1742 degrees of freedom
##   (7 observations deleted due to missingness)
## Multiple R-squared:  0.4864, Adjusted R-squared:  0.4855
## F-statistic: 549.8 on 3 and 1742 DF,  p-value: < 2.2e-16
```

```
movies <- movies %>%
  mutate(
    residuals_budget_sq_imdb = residuals(fit_budget_sq_imdb),
    predicted_budget_sq_imdb = predict(fit_budget_sq_imdb)
  )

p1 <- ggplot(data = movies, mapping = aes(x = log_budget_2013, y = residuals_budget_sq_imdb)) +
  geom_point() +
  geom_smooth()

p2 <- ggplot(data = movies, mapping = aes(x = imdb_rating, y = residuals_budget_sq_imdb)) +
  geom_point() +
  geom_smooth()
```

```r
p3 <- ggplot(data = movies, mapping = aes(x = predicted_budget_imdb, y = residuals_budget_sq_imdb)) +
  geom_point() +
  geom_smooth()

grid.arrange(p1, p2, p3, nrow = 1)
```



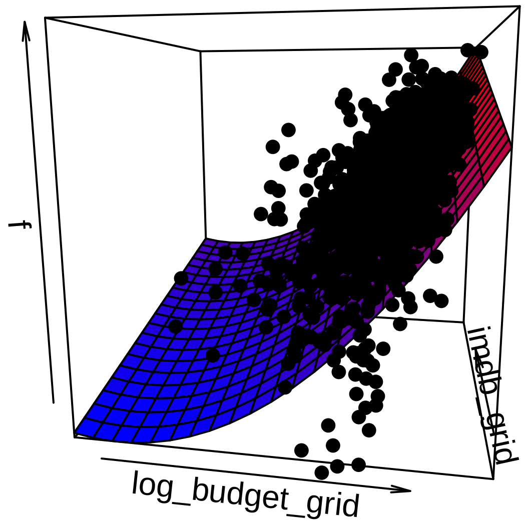
- What is the equation of the estimated surface?

The coefficient for `I(log_budget_2013^2)` describes the curvature of the fitted surface along the `log_budget_2013` axis.

Here is a view of the resulting fitted surface.

```r
log_budget_grid <- seq(from = min(movies$log_budget_2013), to = max(movies$log_budget_2013), length = 21)
imdb_grid <- seq(from = min(movies$imdb_rating), to = max(movies$imdb_rating), length = 21)

f = outer(log_budget_grid, imdb_grid, function(log_budget, imdb) {
  predict(fit_budget_sq_imdb, data.frame(log_budget_2013 = log_budget, imdb_rating = imdb))
})

nrz <- nrow(f)
ncz <- ncol(f)
# Create a function interpolating colors in the range of specified colors
```

```r
jet.colors <- colorRampPalette( c("blue", "red") )
# Generate the desired number of colors from this palette
nbcol <- 100
color <- jet.colors(nbcol)
# Compute the z-value at the facet centres
ffacet <- f[-1, -1] + f[-1, -ncz] + f[-nrz, -1] + f[-nrz, -ncz]
# Recode facet z-values into color indices
facetcol <- cut(ffacet, nbcol)

res <- persp(log_budget_grid, imdb_grid, f, col = color[facetcol], theta = 10, phi = 10)
points(trans3d(movies$log_budget_2013, movies$imdb_rating, movies$log_intgross_2013, pmat = res), pch = 16)
```



Concept 2 (b): Adding interactions between two explanatory variables

Fit 5: Interaction between budget and run time

```r
fit_budget_runtime_interaction <- lm(log_intgross_2013 ~ log_budget_2013 * run_time_min, data = movies)
summary(fit_budget_runtime_interaction)
```

```
##
## Call:
## lm(formula = log_intgross_2013 ~ log_budget_2013 * run_time_min,
##     data = movies)
##
## Residuals:
```

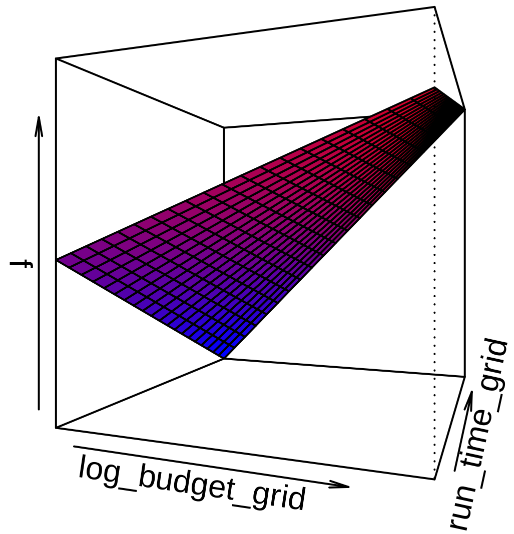
```
##      Min       1Q   Median       3Q      Max
## -10.1790  -0.5817   0.1228   0.7087   4.6511
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 11.168716   2.139903   5.219 2.01e-07 ***
## log_budget_2013              0.396240   0.120917   3.277 0.001070 **
## run_time_min                -0.071341   0.020590  -3.465 0.000544 ***
## log_budget_2013:run_time_min  0.004201   0.001154   3.642 0.000279 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.271 on 1715 degrees of freedom
##   (34 observations deleted due to missingness)
## Multiple R-squared:  0.4397, Adjusted R-squared:  0.4387
## F-statistic: 448.5 on 3 and 1715 DF,  p-value: < 2.2e-16
```

- What is the equation of the estimated surface?

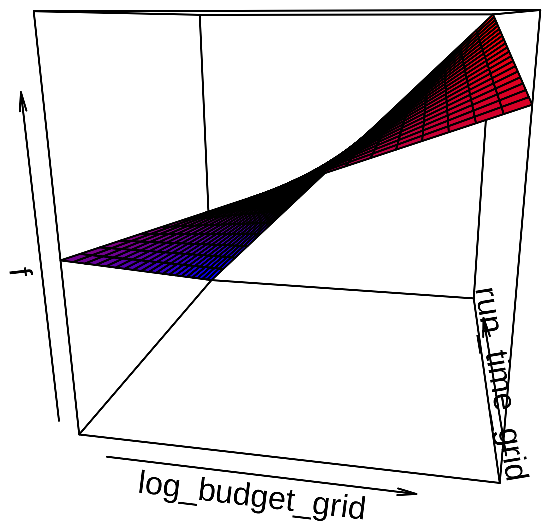
- What is the interpretation of the coefficient for the interaction?

```r
log_budget_grid <- seq(from = min(movies$log_budget_2013), to = max(movies$log_budget_2013), length = 21)
run_time_grid <- seq(from = min(movies$run_time_min, na.rm = TRUE), to = max(movies$run_time_min, na.rm = TRUE), length = 21)

f = outer(log_budget_grid, run_time_grid, function(log_budget, run_time) {
  predict(fit_budget_runtime_interaction, data.frame(log_budget_2013 = log_budget, run_time_min = run_time))
})

nrz <- nrow(f)
ncz <- ncol(f)
# Create a function interpolating colors in the range of specified colors
jet.colors <- colorRampPalette( c("blue", "red") )
# Generate the desired number of colors from this palette
nbcol <- 100
color <- jet.colors(nbcol)
# Compute the z-value at the facet centres
```

```
ffacet <- f[-1, -1] + f[-1, -ncz] + f[-nrz, -1] + f[-nrz, -ncz]
# Recode facet z-values into color indices
facetcol <- cut(ffacet, nbcol)

res <- persp(log_budget_grid, run_time_grid, f, col = color[facetcol], theta = 20, phi = 0)
```




```
res <- persp(log_budget_grid, run_time_grid, f, col = color[facetcol], theta = 10, phi = 15)
```



Concept 3: t tests (one coefficient) vs. F tests (multiple coefficients simultaneously); groups of individually non-significant terms can be jointly significant

Fit 6: Interaction between budget and imdb rating

```
fit_budget_imdb_interaction <- lm(log_intgross_2013 ~ log_budget_2013 * imdb_rating, data = movies)
summary(fit_budget_imdb_interaction)
```

```
##
## Call:
## lm(formula = log_intgross_2013 ~ log_budget_2013 * imdb_rating,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.9812 -0.5522  0.1329  0.7132  4.7505
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 -0.53579    3.36245  -0.159    0.873
## log_budget_2013              0.94755    0.19252   4.922 9.39e-07 ***
## imdb_rating                  0.57402    0.48643   1.180    0.238
## log_budget_2013:imdb_rating -0.01299    0.02787  -0.466    0.641
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.238 on 1742 degrees of freedom
##   (7 observations deleted due to missingness)
## Multiple R-squared:  0.4665, Adjusted R-squared:  0.4655
## F-statistic: 507.7 on 3 and 1742 DF,  p-value: < 2.2e-16
```

```
fit_budget_only <- lm(log_intgross_2013 ~ log_budget_2013, data = movies)
anova(fit_budget_imdb_interaction, fit_budget_only)
```

```
## Analysis of Variance Table
##
## Model 1: log_intgross_2013 ~ log_budget_2013 * imdb_rating
## Model 2: log_intgross_2013 ~ log_budget_2013
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   1742 2669.6
## 2   1744 2850.0 -2   -180.44 58.872 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Conduct a hypothesis test of the claim that after accounting for the linear effect of `log_budget_2013` and the interaction effect between `log_budget_2013` and `imdb_rating`, there is no linear association between `imdb_rating` and `log_intgross_2013`.

- Conduct a hypothesis test of the claim that after accounting for the linear effect of `log_budget_2013` and the linear effect of `imdb_rating`, there is no interaction effect of `log_budget_2013` and `imdb_rating`, there is no linear association between `imdb_rating` and `log_intgross_2013`.

- Conduct a hypothesis test of the claim that after accounting for the linear effect of `log_budget_2013`, there is no linear effect of `imdb_rating` and no interaction effect between `imdb_rating` and `log_budget_2013`.

- Conduct a hypothesis test of the claim that there is no linear effect of either `log_budget_2013` or `imb_rating`, and there is also no interaction effect between `imdb_rating` and `log_budget_2013`.