

Regression

Part 3

Prof. Bisbee

Seoul National University

Slides Updated: 2024-07-11

Agenda

1. Recap of Movie Analysis
2. Multiple Regression
3. Categorical Predictors

Recap of Movie Analysis

```
require(tidyverse)

mv <-
read_rds('https://github.com/jbisbee1/ISP_Data_Science_2024/raw/main/da
```

- **Theory**: the more a movie costs, the more it should make
 - If not, Hollywood would go out of business!
- X : budget
- Y : gross

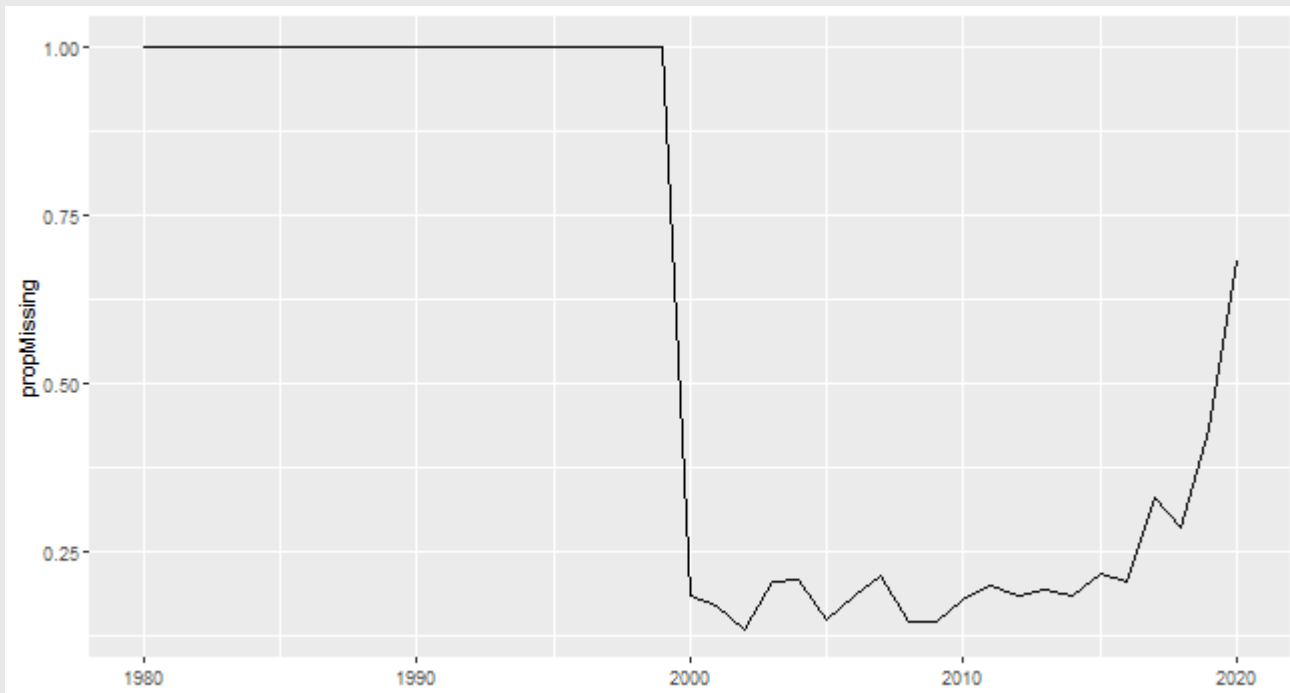
Step 1: Look

```
summary(mv %>% select(gross,budget))
```

```
##      gross      budget
##  Min.   :7.140e+02   Min.    :    5172
## 1st Qu.:1.121e+07   1st Qu.: 16865322
## Median :5.178e+07   Median : 37212044
## Mean   :1.402e+08   Mean    : 57420173
## 3rd Qu.:1.562e+08   3rd Qu.: 77844746
## Max.   :3.553e+09   Max.    :387367903
## NA's   :3668       NA's    :4482
```

Step 1: Look

```
mv %>%  
  mutate(missing = ifelse(is.na(gross) | is.na(budget),1,0)) %>%  
  group_by(year) %>%  
  summarise(propMissing = mean(missing)) %>%  
  ggplot(aes(x = year,y = propMissing)) +  
  geom_line()
```



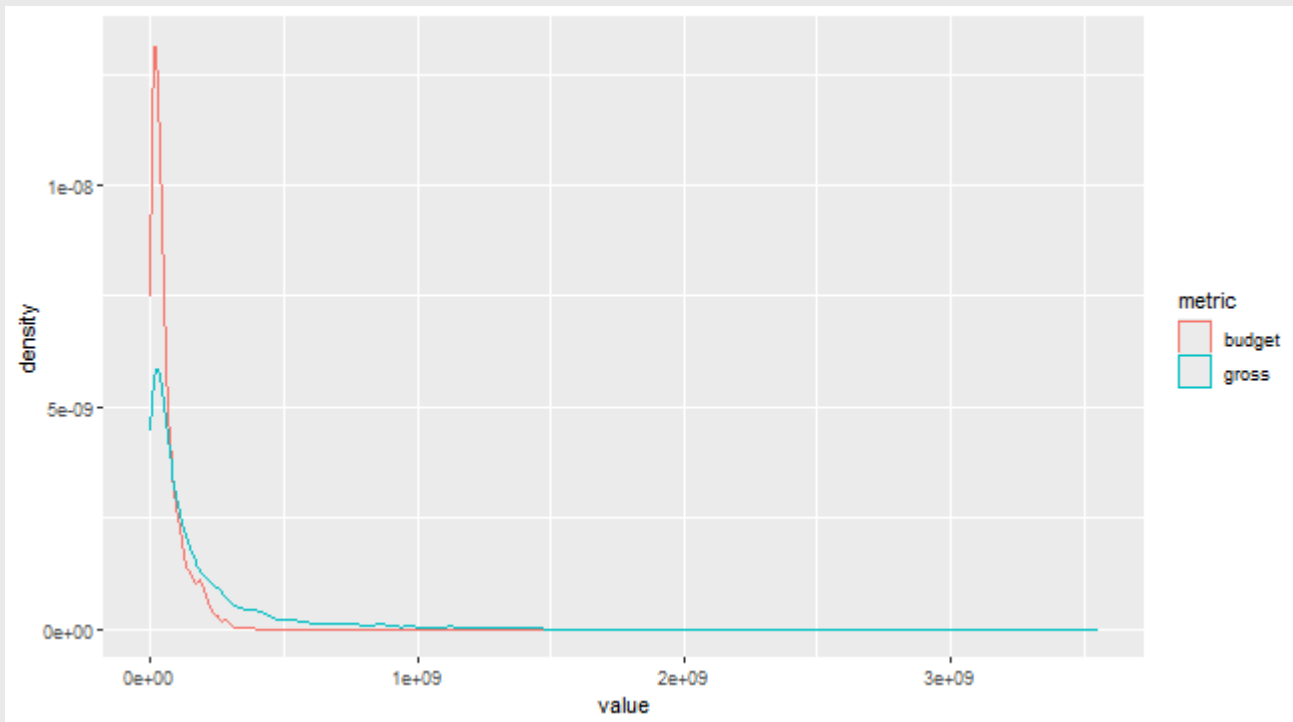
Some quick wrangling

```
mv <- mv %>%  
  drop_na(gross,budget)  
  
mv %>%  
  select(gross,budget) %>%  
  glimpse()
```

```
## Rows: 3,179  
## Columns: 2  
## $ gross  <dbl> 73677478, 53278578, 723586629, 11490339, 62...  
## $ budget <dbl> 93289619, 10883789, 160147179, 6996721, 139...
```

Step 2: Univariate Viz

```
mv %>%  
  select(title,gross,budget) %>%  
  gather(metric,value,-title) %>%  
  ggplot(aes(x = value,color = metric)) +  
  geom_density()
```



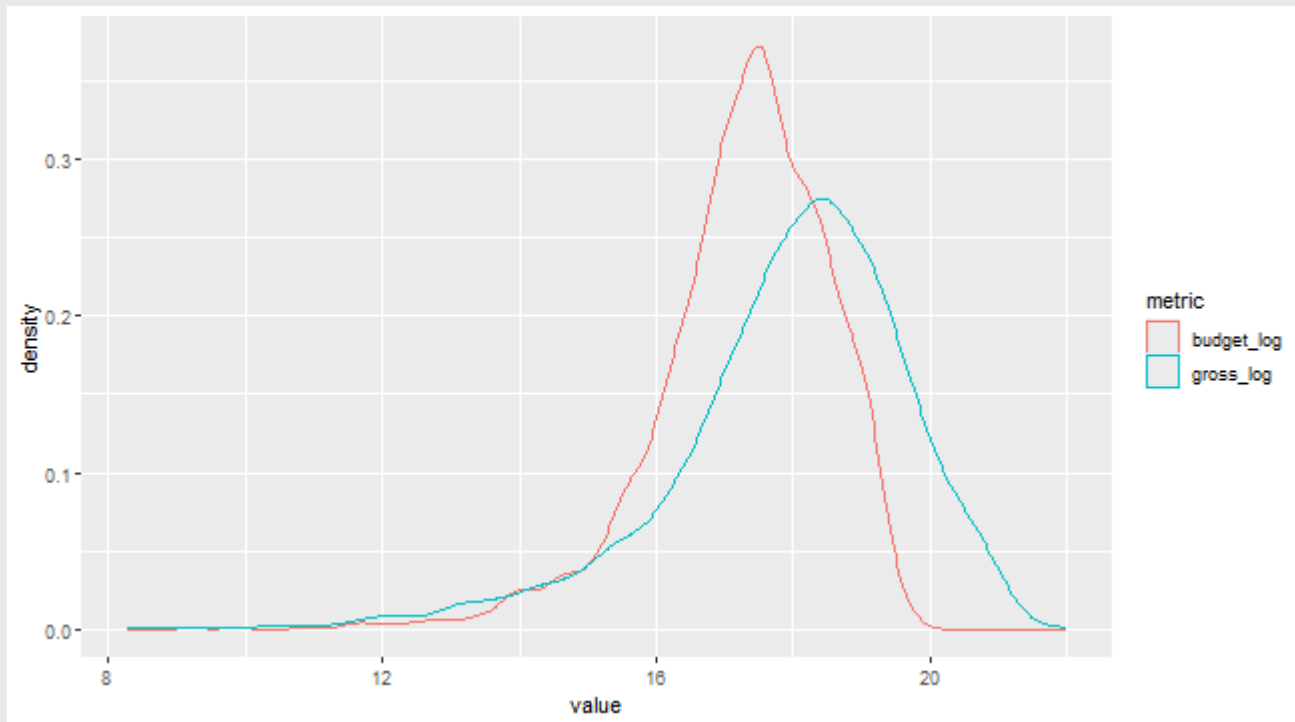
More Wrangling?

- Univariate visualization highlights significant **skew** in both measures
 - Most movies don't cost a lot and don't make a lot
 - But there are a few blockbusters that pull the density way out
- Let's **wrangle** two new variables that take the log of these skewed measures
 - Logging transforms skewed measures to more "normal" measures

```
mv <- mv %>%  
  mutate(gross_log = log(gross),  
         budget_log = log(budget))
```


Step 2: Univariate Viz

```
mv %>%  
  select(title,gross_log,budget_log) %>%  
  gather(metric,value,-title) %>%  
  ggplot(aes(x = value,color = metric)) +  
  geom_density()
```



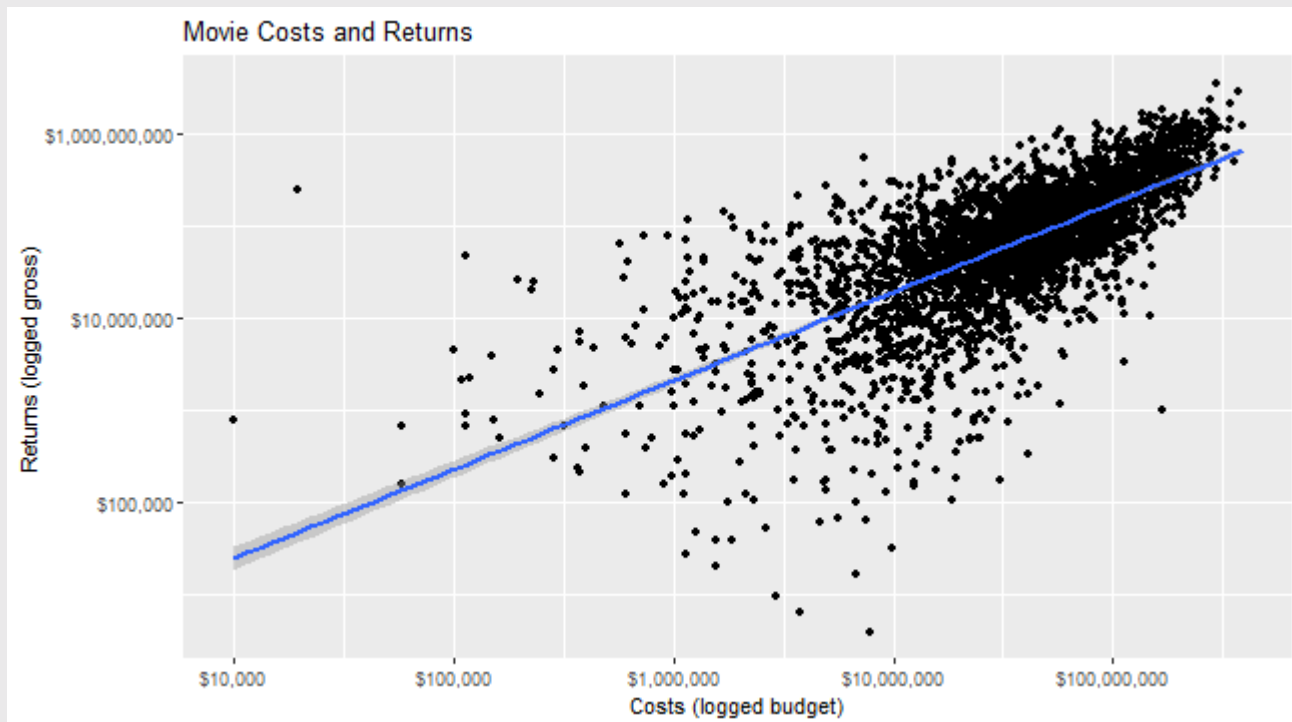
Step 3: Multivariate Viz

```
pClean <- mv %>%  
  ggplot(aes(x = budget,y = gross)) +  
  geom_point() +  
  scale_x_log10(labels = scales::dollar) +  
  scale_y_log10(labels = scales::dollar) +  
  labs(title = "Movie Costs and Returns",  
        x = "Costs (logged budget)",  
        y = "Returns (logged gross)")
```

Step 3: Multivariate Viz

```
pClean + geom_smooth(method = 'lm')
```

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



Step 4: Regression!

```
require(broom) # Make the output nicer Looking
```

```
## Loading required package: broom
```

```
m <- lm(gross_log ~ budget_log, data = mv)
tidy(m)
```

```
## # A tibble: 2 × 5
##   term          estimate std.error statistic  p.value
##   <chr>         <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    1.26      0.310      4.07 0.0000473
## 2 budget_log     0.964     0.0179     54.0 0
```

Step 5.1: Univariate Viz of Errors

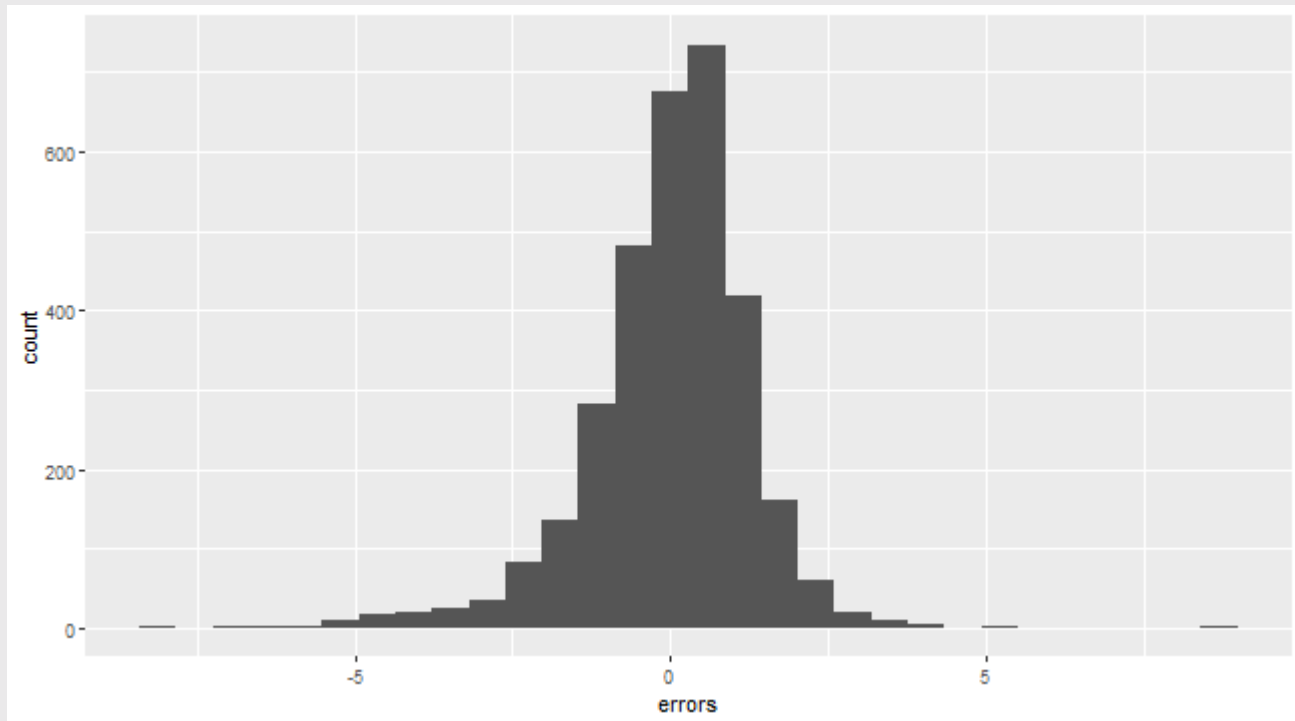
- Errors $\varepsilon = Y - \hat{Y}$
 - In R, can also get them via `resid()` function

```
mv %>%  
  mutate(errors_manual = gross_log - predict(m),  
         errors_resid = resid(m))
```

```
## # A tibble: 3,179 × 24  
##   title    rating genre  year released  score  votes director  
##   <chr>   <chr>  <chr> <dbl> <chr>    <dbl> <dbl> <chr>  
## 1 Almost... R      Adve... 2000 Septemb... 7.9 2.6 e5 Cameron...  
## 2 Americ... R      Come... 2000 April 1... 7.6 5.14e5 Mary Ha...  
## 3 Gladia... R      Acti... 2000 May 5, ... 8.5 1.4 e6 Ridley ...  
## 4 Requie... Unrat... Drama 2000 Decembe... 8.3 7.86e5 Darren ...  
## 5 Memento R      Myst... 2000 May 25,... 8.4 1.20e6 Christo...  
## 6 Cast A... PG-13  Adve... 2000 Decembe... 7.8 5.42e5 Robert ...  
## 7 Scary ... R      Come... 2000 July 7,... 6.2 2.38e5 Keenen ...  
## 8 The Pe... PG-13  Acti... 2000 June 30... 6.4 1.6 e5 Wolfgan...  
## 9 Coyote... PG-13  Come... 2000 August ... 5.7 1.08e5 David M...  
## 10 X-Men PG-13  Acti... 2000 July 14... 7.4 5.82e5 Bryan S...  
## # i 3 169 more rows
```

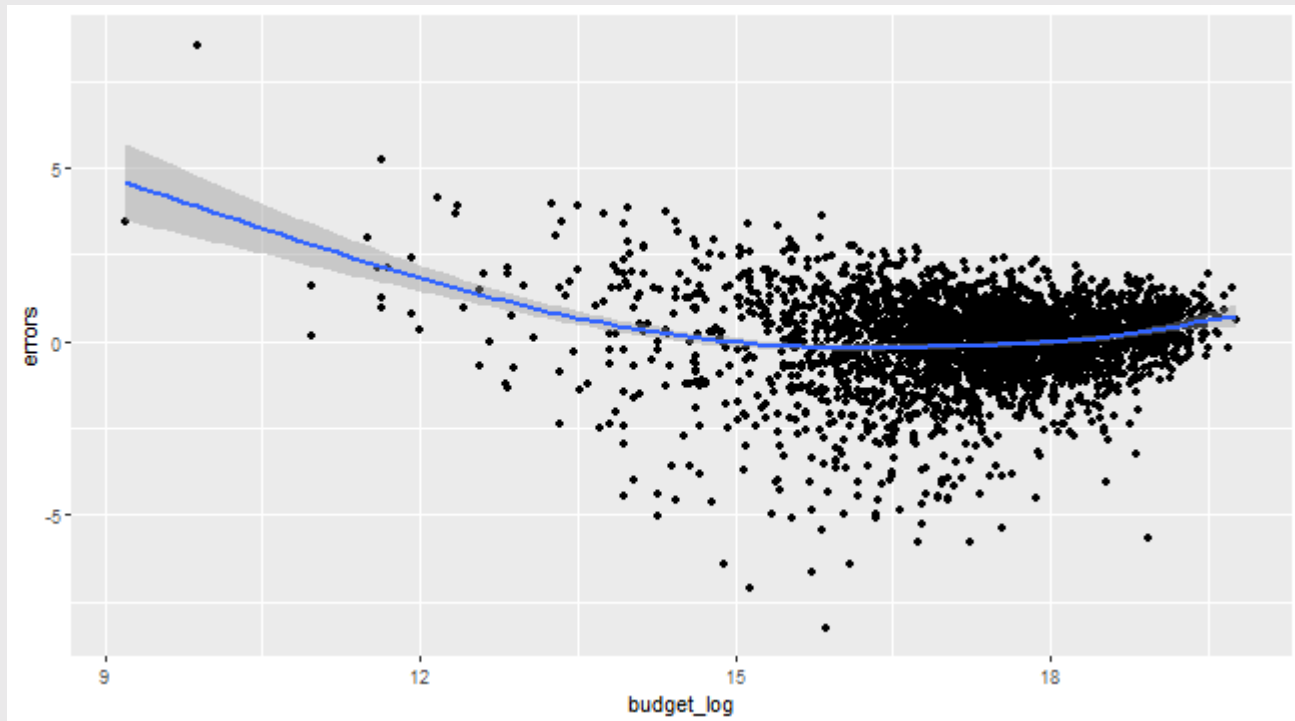
Step 5.1: Univariate Viz of Errors

```
mv %>%  
  ggplot(aes(x = errors)) +  
  geom_histogram()
```



Step 5.2: Multivariate Viz of Errors

```
mv %>%  
  ggplot(aes(x = budget_log, y = errors)) +  
  geom_point() +  
  geom_smooth()
```



Step 5.3: Cross Validated RMSE

```
set.seed(123)
rmseBudget <- NULL
for(i in 1:100) {
  train <- mv %>%
    sample_n(size = round(nrow(mv)*.5),replace = F)
  test <- mv %>% anti_join(train)

  mTrain <- lm(gross_log ~ budget_log,train)

  test$preds <- predict(mTrain,newdata = test)

  rmse <- sqrt(mean((test$gross_log - test$preds)^2,na.rm=T))
  rmseBudget <- c(rmseBudget,rmse)
}

mean(rmseBudget)
```

```
## [1] 1.279899
```


Thinking like a scientist

- Our previous model predicted **gross** as a function of **budget**
- **Theoretically**, is this sensible?
 1. Bigger budgets → famous actors → mass appeal → more tickets
 2. Bigger budgets → advertising money → mass appeal → more tickets
- But what if the movie is just...not good?

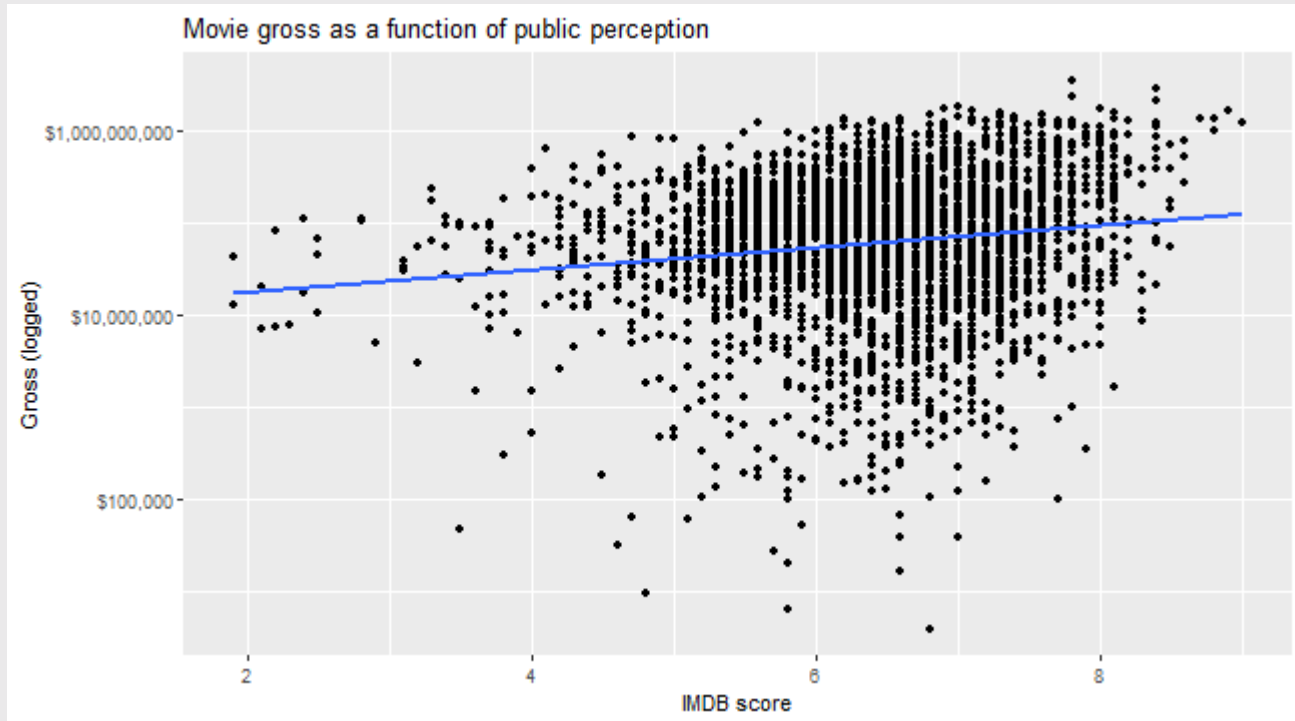
Alternative Theory

- Good movies make more money
 - Theory: good movies → recommendations → more tickets
- Predict gross with IMDB rating (score)

```
pIMDB <- mv %>%  
  ggplot(aes(x = score,y = gross)) +  
  geom_point() +  
  labs(title = "Movie gross as a function of public perception",  
        x = "IMDB score",  
        y = "Gross (logged)") +  
  scale_y_log10(label = scales::dollar) +  
  geom_smooth(method = 'lm',se = F)
```

Alternative Model

pIMDB



Evaluating the Model

- Let's go straight to RMSE
 - We can have R calculate errors for us with `residuals()` command

```
m2 <- lm(gross_log ~ score,mv)
error <- residuals(m2)
(rmseScore <- sqrt(mean(error^2)))
```

```
## [1] 1.753146
```

- Even worse!

Multivariate Regression

- Recall that we can **model** our outcome with multiple **predictors**

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \cdots + \varepsilon$$

- How much better can we predict **gross** with **BOTH** **budget** and **score**?

```
m3 <- lm(gross_log ~ budget_log + score,mv)
error <- residuals(m3)
(rmseBudgScore <- sqrt(mean(error^2)))
```

```
## [1] 1.248817
```

Comparing Models

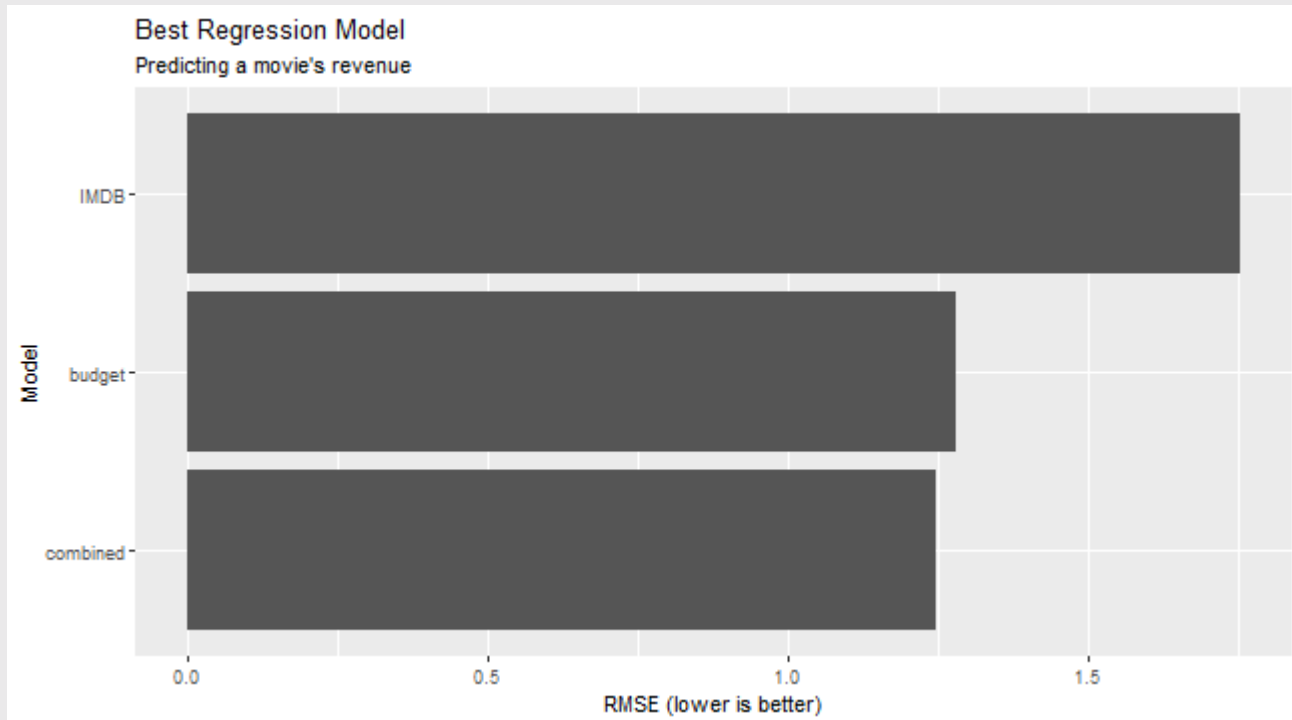
- Which model best predicts movie revenues?

```
p <- data.frame(budget = mean(rmseBudget),
               IMDB = rmseScore,
               combined = rmseBudgScore) %>%
  gather(model,rmse) %>%
  ggplot(aes(x = reorder(model,rmse),y = rmse)) +
  geom_bar(stat = 'identity') +
  labs(title = "Best Regression Model",
       subtitle = "Predicting a movie's revenue",
       y = "RMSE (lower is better)",
       x = "Model") +
  coord_flip()
```

Comparing Models

- Which model best predicts movie revenues?

p



Why RMSE?

- Want to understand how good / bad our model is
- Can use it to compare models

Why RMSE?

- Do we improve our model with `score`?

```
set.seed(123)
bsRes <- NULL
for(i in 1:100) {
  train <- mv %>%
    sample_n(size = round(nrow(mv)*.8),replace = F)
  test <- mv %>% anti_join(train)

  mB <- lm(gross_log ~ budget_log,train)
  mS <- lm(gross_log ~ score,train)
  mC <- lm(gross_log ~ budget_log + score,train)

  bsRes <- test %>%
    mutate(pB = predict(mB,newdata = test),
           pS = predict(mS,newdata = test),
           pC = predict(mC,newdata = test)) %>%
    summarise(Budget = sqrt(mean((gross_log - pB)^2,na.rm=T)),
              Score = sqrt(mean((gross_log - pS)^2,na.rm=T)),
              Combined = sqrt(mean((gross_log - pC)^2,na.rm=T))) %>%
    bind_rows(bsRes)
}
```

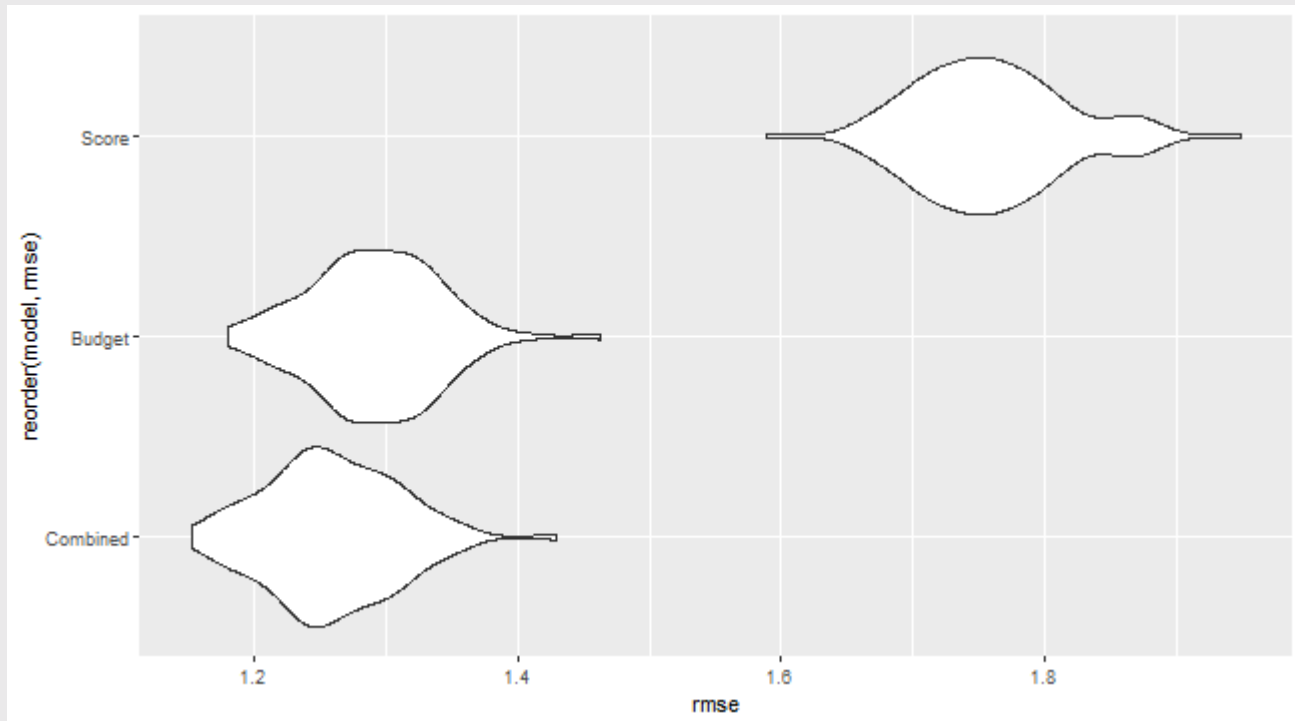
Why RMSE?

```
bsRes %>%  
  summarise_all(mean, na.rm=T)
```

```
## # A tibble: 1 × 3  
##   Budget Score Combined  
##   <dbl> <dbl>    <dbl>  
## 1   1.29   1.76    1.26
```

Visualizing

```
bsRes %>%  
  gather(model,rmse) %>%  
  ggplot(aes(x = rmse,y = reorder(model,rmse))) +  
  geom_violin()
```



Categorical Data

- Thus far, only using continuous variables
- But we can do regression with categorical data too!
- The Bechdel Test: 3 questions of a movie
 1. Does it have two women in it?
 2. Who talk to each other?
 3. About something other than a man?

```
mv %>%  
  count(bechdel_score)
```

```
## # A tibble: 5 × 2  
##   bechdel_score      n  
##           <dbl> <int>  
## 1             0    141  
## 2             1    526  
## 3             2    206  
## 4             3   1185  
## 5            NA   1121
```

Research Question

- Do movies that pass the Bechdel Test make more money?
 - **Theory:** Women are ~50% of the population. Movies that pass the test are more appealing to women.
 - **Hypothesis:** Movies that pass the test make more money.
- **Wrangling:** Let's turn the `bechdel_score` variable into a binary

```
mv <- mv %>%  
  mutate(bechdel_bin = ifelse(bechdel_score == 3, "Pass", "Fail"))
```

Regression

- We can add the binary factor to our regression

```
tidy(lm(gross_log ~ bechdel_bin, mv))
```

```
## # A tibble: 2 × 5
##   term                estimate std.error statistic p.value
##   <chr>              <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)        18.3      0.0563    325.      0
## 2 bechdel_binPass   -0.160    0.0742    -2.15    0.0316
```

Regression

- Coefficient is negative
- What is the interpretation?
 - Movies that pass make less money...
 - ...than what?
 - Movies that fail the Bechdel Test
- Categorical variables are **always interpreted in relation to the hold-out category!**

Regression

- Movies that fail the test make more money!?
- **REMEMBER:** Correlation \neq causation
 - What might explain this pattern?
 - Budgets in a sexist Hollywood!
 - Movies that fail the test get larger budgets
 - Budgets are positively associated with gross
- So we want to "control" for budget by adding it to our regression

```
mBechCtrl <- lm(gross_log ~ budget_log + bechdel_bin,mv)
```


Regression

```
tidy(mBechCtrl)
```

```
## # A tibble: 3 × 5
##   term                estimate std.error statistic    p.value
##   <chr>              <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)         2.12      0.353      6.01 2.25e- 9
## 2 budget_log          0.921     0.0199     46.2 3.57e-320
## 3 bechdel_binPass     0.188     0.0525      3.58 3.56e- 4
```

Regression

- Our hypothesis is supported!
- What about non-binary categorical variables?

```
mv %>%  
  count(rating)
```

```
## # A tibble: 9 × 2  
##   rating      n  
##   <chr>    <int>  
## 1 G         54  
## 2 NC-17      6  
## 3 Not Rated  34  
## 4 PG        434  
## 5 PG-13     1249  
## 6 R        1388  
## 7 TV-MA      2  
## 8 Unrated     7  
## 9 <NA>        5
```

Categorical

- Let's first remove rarely-occurring ratings

```
mvAnalysis <- mv %>%  
  filter(!rating %in% c('Approved', 'TV-14', 'TV-MA', 'TV-PG', 'X'))
```

Categorical

```
tidy(lm(gross_log ~ rating,mvAnalysis))
```

```
## # A tibble: 7 × 5
##   term                estimate std.error statistic  p.value
##   <chr>              <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)        19.2      0.218     88.1      0
## 2 ratingNC-17        -2.45     0.688     -3.56 3.81e- 4
## 3 ratingNot Rated    -4.43     0.350    -12.7 7.62e-36
## 4 ratingPG           -0.391    0.231     -1.69 9.08e- 2
## 5 ratingPG-13        -0.763    0.222     -3.43 6.05e- 4
## 6 ratingR            -1.91     0.222     -8.62 1.06e-17
## 7 ratingUnrated      -4.66     0.643     -7.25 5.38e-13
```

Categorical

- Everything makes less money than the hold-out category!
 - "G"-rated movies are powered by children
- What if we wanted to compare to a different reference category?

```
mvAnalysis <- mvAnalysis %>%  
  mutate(rating = factor(rating,  
                          levels = c('R', 'PG-13', 'PG', 'G', 'Not  
Rated')))  
mRating2 <- lm(gross_log ~ rating, mvAnalysis)
```

Categorical

```
tidy(mRating2)
```

```
## # A tibble: 5 × 5
##   term                estimate std.error statistic  p.value
##   <chr>              <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)        17.3      0.0430    402.      0
## 2 ratingPG-13         1.15     0.0624     18.4  5.81e-72
## 3 ratingPG            1.52     0.0880     17.3  4.66e-64
## 4 ratingG             1.91     0.222      8.61  1.09e-17
## 5 ratingNot Rated    -2.52     0.278    -9.07  2.04e-19
```

Cross Validation

```
set.seed(123)
rmseRes_rating <- NULL
for(i in 1:100) {
  train <- mvAnalysis %>%
    group_by(rating) %>%
    sample_n(size = round(n()*.8),replace = F)
  test <- mvAnalysis %>% anti_join(train)

  m <- lm(gross_log ~ rating,train)
  rmseRes_rating <- test %>%
    mutate(preds = predict(m,newdata = test)) %>%
    summarise(rmse = sqrt(mean((gross_log - preds)^2,na.rm=T))) %>%
    bind_rows(rmseRes_rating)
}
rmseRes_rating %>%
  summarise(rmse = mean(rmse))
```

```
## # A tibble: 1 × 1
##   rmse
##   <dbl>
## 1  1.60
```

BREAK

Practice

- How to interpret a regression table:
 1. Check if any of the variables are logged
 2. Check if the X variable is categorical
 3. (Coming soon) Check if the Y variable is binary

Practice: 1, 2, 3 = "No"

- If 1, 2 & 3 are "No", then
 - α is the value of Y when X is zero
 - β is the amount Y increases when X increases by 1

Practice: 1, 2, 3 = "No"

- RQ: What is the relationship between head shots and eliminations?

```
fn <-  
read_rds('https://github.com/jbisbee1/ISP_Data_Science_2024/raw/main/data')  
tidy(lm(eliminations ~ head_shots,fn))
```

```
## # A tibble: 2 × 5  
##   term          estimate std.error statistic    p.value  
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>  
## 1 (Intercept)    2.00     0.0768    26.0 2.31e-113  
## 2 head_shots    0.109    0.0102    10.7 2.71e- 25
```

Practice: 1, 2, 3 = "No"

- **NB:** Pay attention to the units of X and Y !
- RQ: What is the relationship between accuracy and head shots?

```
tidy(lm(head_shots ~ accuracy, fn))
```

```
## # A tibble: 2 × 5
##   term          estimate std.error statistic  p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)      2.61      0.390      6.68 3.94e-11
## 2 accuracy         8.52      1.32      6.43 1.99e-10
```

Practice: Logged = "Yes"

- Then need to remember THESE rules:

1. $\log(Y) \sim X$: 1 unit change in $X \rightarrow (\exp(\beta) - 1) * 100$ % change in Y

2. $Y \sim \log(X)$: 1% increase in $X \rightarrow \beta/100$ unit change in Y

3. $\log(Y) \sim \log(X)$: 1% increase in $X \rightarrow \beta$ % change in Y

Practice: Logged = "Yes"

1. $\log(Y) \sim X$: 1 unit change in $X \rightarrow (\exp(\beta) - 1) * 100$ % change in Y

- RQ: What is the relationship between movie gross and IMDB score?

```
tidy(lm(gross_log ~ score, mv))
```

```
## # A tibble: 2 × 5
##   term          estimate std.error statistic  p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)    16.1      0.211     76.5      0
## 2 score          0.279    0.0325     8.58 1.49e-17
```

Practice: Logged = "Yes"

1. $\log(Y) \sim X$: 1 unit change in $X \rightarrow (\exp(\beta) - 1) * 100$ % change in Y

2. $Y \sim \log(X)$: 1% increase in $X \rightarrow \beta/100$ unit change in Y

RQ: What is the relationship between IMDB score and budget?

```
tidy(lm(score ~ budget_log, mv))
```

```
## # A tibble: 2 × 5
##   term          estimate std.error statistic  p.value
##   <chr>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)    6.61      0.231     28.6 6.21e-160
## 2 budget_log   -0.0109    0.0133    -0.817 4.14e- 1
```

Practice: Logged = "Yes"

1. $\log(Y) \sim X$: 1 unit change in $X \rightarrow (\exp(\beta) - 1) * 100$ % change in Y
2. $Y \sim \log(X)$: 1% increase in $X \rightarrow \beta/100$ unit change in Y
3. $\log(Y) \sim \log(X)$: 1% increase in $X \rightarrow \beta$ % change in Y

RQ: What is the relationship between gross and budget?

```
tidy(lm(gross_log ~ budget_log, mv))
```

```
## # A tibble: 2 × 5
##   term          estimate std.error statistic    p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)    1.26      0.310      4.07 0.0000473
## 2 budget_log    0.964     0.0179    54.0  0
```


Practice: Categorical $X = \text{"Yes"}$

- RQ: What is the relationship between a movie's PG rating and it's IMBD rating?

```
tidy(lm(score ~ rating,mv))
```

```
## # A tibble: 8 × 5
##   term                estimate std.error statistic p.value
##   <chr>                <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)          6.37      0.129     49.2      0
## 2 ratingNC-17          0.431     0.409      1.06     0.291
## 3 ratingNot Rated      0.481     0.208      2.31     0.0207
## 4 ratingPG            -0.0775    0.137     -0.565    0.572
## 5 ratingPG-13         -0.0540    0.132     -0.409    0.683
## 6 ratingR              0.168     0.132      1.27     0.204
## 7 ratingTV-MA          0.481     0.684      0.704    0.482
## 8 ratingUnrated        0.546     0.382      1.43     0.153
```

Practice: Categorical $X = \text{"Yes"}$

- RQ: What is the relationship between mental state and accuracy (Fortnite data)?

```
tidy(lm(accuracy ~ mental_state,fn))
```

```
## # A tibble: 2 × 5
##   term                estimate std.error statistic  p.value
##   <chr>              <dbl>    <dbl>    <dbl>    <dbl>
## 1 (Intercept)        0.245    0.00640    38.3 4.19e-195
## 2 mental_statesober  0.0301   0.00889     3.38 7.47e- 4
```

Aside: scientific notation

- What is $7.47\text{e-}4$?
 - Decimal, then 4 zeros, then the number
 - 0.0000747
- What is $1.39\text{e-}1$?
 - 0.0139

Practice: Categorical $Y = \text{"Yes"}$

- Next class!