What makes a Hollywood movie profitable?

Jimmy Ng

October 29, 2018

Data Preparation

```
> # load packages
> library(plyr)
> library(lubridate)
> library(tidyverse)
>
> # load file from github
> # source from tidytuesday - they have a new dataset coming out every Tuesday
> github <- "https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data"
> date <- "2018-10-23"
> csv <- "movie_profit.csv"
> myfile <- paste(github, date, csv, sep = "/")
> df <- readr::read_csv(myfile)
>
> ####################
> ### have a look ###
> ####################
> # head, dim, str
> head(df)
## # A tibble: 6 x 9
##      X1 release_date movie production_budg~ domestic_gross worldwide_gross
##   <int> <chr>        <chr>            <dbl>          <dbl>           <dbl>
## 1     1 6/22/2007    Evan~        175000000      100289690       174131329
## 2     2 7/28/1995    Wate~        175000000       88246220       264246220
## 3     3 5/12/2017    King~        175000000       39175066       139950708
## 4     4 12/25/2013   47 R~        175000000       38362475       151716815
## 5     5 6/22/2018    Jura~        170000000      416769345      1304866322
## 6     6 8/1/2014     Guar~        170000000      333172112       771051335
## # ... with 3 more variables: distributor <chr>, mpaa_rating <chr>,
## #   genre <chr>
> dim(df)  # [1] 3401    9
## [1] 3401    9
> str(df)
## Classes 'tbl_df', 'tbl' and 'data.frame':    3401 obs. of  9 variables:
##  $ X1               : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ release_date     : chr  "6/22/2007" "7/28/1995" "5/12/2017" "12/25/2013" ...
##  $ movie            : chr  "Evan Almighty" "Waterworld" "King Arthur: Legend of the Sword" "47 Ronin
##  $ production_budget: num  1.75e+08 1.75e+08 1.75e+08 1.75e+08 1.70e+08 1.70e+08 1.70e+08 1.70e+08 1
##  $ domestic_gross   : num  1.00e+08 8.82e+07 3.92e+07 3.84e+07 4.17e+08 ...
##  $ worldwide_gross  : num  1.74e+08 2.64e+08 1.40e+08 1.52e+08 1.30e+09 ...
##  $ distributor      : chr  "Universal" "Universal" "Warner Bros." "Universal" ...
##  $ mpaa_rating      : chr  "PG" "PG-13" "PG-13" "PG-13" ...
##  $ genre            : chr  "Comedy" "Action" "Adventure" "Action" ...
##  - attr(*, "spec")=List of 2
##   ..$ cols   :List of 9
```

```
##   .. ..$ X1               : list()
##   .. .. ..- attr(*, "class")= chr  "collector_integer" "collector"
##   .. ..$ release_date     : list()
##   .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
##   .. ..$ movie            : list()
##   .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
##   .. ..$ production_budget: list()
##   .. .. ..- attr(*, "class")= chr  "collector_double" "collector"
##   .. ..$ domestic_gross   : list()
##   .. .. ..- attr(*, "class")= chr  "collector_double" "collector"
##   .. ..$ worldwide_gross  : list()
##   .. .. ..- attr(*, "class")= chr  "collector_double" "collector"
##   .. ..$ distributor      : list()
##   .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
##   .. ..$ mpaa_rating      : list()
##   .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
##   .. ..$ genre            : list()
##   .. .. ..- attr(*, "class")= chr  "collector_character" "collector"
##   ..$ default: list()
##   .. ..- attr(*, "class")= chr  "collector_guess" "collector"
##   ..- attr(*, "class")= chr "col_spec"
>
> #############
> # clean-up #
> #############
> # check duplication: is there any movie that's duplicated in this data set?
> plyr::count(df, "movie") %>%
+         filter(freq >1)
##            movie freq
## 1 Tau ming chong    2
>
> # what is this movie?
> df %>%
+         filter(movie == "Tau ming chong") %>%
+         print
## # A tibble: 2 x 9
##      X1 release_date movie production_budg~ domestic_gross worldwide_gross
##   <int> <chr>        <chr>            <dbl>          <dbl>           <dbl>
## 1  2974 4/2/2010     Tau ~          4000000         129078        38899792
## 2  2975 4/2/2010     Tau ~          4000000         129078        38899792
## # ... with 3 more variables: distributor <chr>, mpaa_rating <chr>,
## #   genre <chr>
> # id == 2974, 2975
>
> # let's remove either one of these identical rows
> df <- df %>%
+         filter(X1 != 2974)
>
> # is there any movie that has 0 or even negative domestic/worldwide gross?
> df %>%
+         filter(domestic_gross <=0 | worldwide_gross <= 0)
## # A tibble: 66 x 9
##       X1 release_date movie production_budg~ domestic_gross worldwide_gross
```

```
##    <int> <chr>        <chr>            <dbl>          <dbl>           <dbl>
##  1    31 12/21/2018   Aqua~        160000000              0               0
##  2   229 3/15/2019    Wond~        100000000              0               0
##  3  1031 11/11/2016   USS ~         40000000              0         1641255
##  4  1089 4/14/2017    Quee~         36000000              0         1578543
##  5  1184 3/13/2015    The ~         35000000              0           11106
##  6  1360 12/14/2007   Good~         30000000              0         2717302
##  7  1446 3/17/2015    Acci~         26000000              0          135436
##  8  1567 7/8/2011     Iron~         25000000              0         5297411
##  9  1826 3/31/2004    The ~         20000000              0         5918742
## 10  1827 8/29/2014    Dweg~         20000000              0               0
## # ... with 56 more rows, and 3 more variables: distributor <chr>,
## #   mpaa_rating <chr>, genre <chr>
> # there are 66 of these in this data set (as of Oct 29)
> # some of them have not been released yet, like the Aquaman!
>
> # let's not remove them but create a flag for each of these variables
> df$domestic_flag <- ifelse(df$domestic_gross <=0, 0, 1); sum(df$domestic_flag)
## [1] 3334
> df$worldwide_flag <- ifelse(df$worldwide_gross <=0, 0, 1); sum(df$worldwide_flag)
## [1] 3364
>
> # let's rename the X1 column and rename it as a movie id column
> names(df)[1] <- "movie_id"
>
> # change "date": turn the release_date column as date data type; add release day, month, year
> df <- df %>%
+         mutate(release_date = lubridate::mdy(release_date),
+                release_day = lubridate::wday(release_date,
+                                              week_start = getOption("lubridate.week.start", 1)),
+                release_month = lubridate::month(release_date),
+                release_year = lubridate::year(release_date))
>
> # rescale the production_budget, domestic_gross & worldwide_gross (by dividing 1 million)
> # so that they are easier to read and visualize
> df <- df %>%
+         mutate(production_budget = production_budget / 1000000,
+                domestic_gross = domestic_gross / 1000000,
+                worldwide_gross = worldwide_gross / 1000000)
>
> # change mpaa_rating & genre data type from character to factor
> df <- df %>%
+         mutate(mpaa_rating = factor(mpaa_rating,
+                                     levels = c("G", "PG", "PG-13", "R")),
+                genre = as.factor(genre))
>
> # complete.cases - remove all NA's
> dfComplete <- df[complete.cases(df), ]
>
> # let's look at the clean data set
> head(dfComplete)
## # A tibble: 6 x 14
##   movie_id release_date movie production_budg~ domestic_gross
```

```
##      <int> <date>       <chr>            <dbl>          <dbl>
## 1        1 2007-06-22   Evan~              175          100.
## 2        2 1995-07-28   Wate~              175           88.2
## 3        3 2017-05-12   King~              175           39.2
## 4        4 2013-12-25   47 R~              175           38.4
## 5        5 2018-06-22   Jura~              170          417.
## 6        6 2014-08-01   Guar~              170          333.
## # ... with 9 more variables: worldwide_gross <dbl>, distributor <chr>,
## #   mpaa_rating <fct>, genre <fct>, domestic_flag <dbl>,
## #   worldwide_flag <dbl>, release_day <dbl>, release_month <dbl>,
## #   release_year <dbl>
> dim(dfComplete)  # [1] 3230   14
## [1] 3230   14
> str(dfComplete)
## Classes 'tbl_df', 'tbl' and 'data.frame':    3230 obs. of  14 variables:
##  $ movie_id         : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ release_date     : Date, format: "2007-06-22" "1995-07-28" ...
##  $ movie            : chr  "Evan Almighty" "Waterworld" "King Arthur: Legend of the Sword" "47 Ronin
##  $ production_budget: num  175 175 175 175 170 170 170 170 170 170 ...
##  $ domestic_gross   : num  100.3 88.2 39.2 38.4 416.8 ...
##  $ worldwide_gross  : num  174 264 140 152 1305 ...
##  $ distributor      : chr  "Universal" "Universal" "Warner Bros." "Universal" ...
##  $ mpaa_rating      : Factor w/ 4 levels "G","PG","PG-13",..: 2 3 3 3 3 3 3 3 3 1 ...
##  $ genre            : Factor w/ 5 levels "Action","Adventure",..: 3 1 2 1 1 1 1 1 2 2 ...
##  $ domestic_flag    : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ worldwide_flag   : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ release_day      : num  5 5 5 3 5 5 5 5 5 3 ...
##  $ release_month    : num  6 7 5 12 6 8 5 4 7 11 ...
##  $ release_year     : num  2007 1995 2017 2013 2018 ...
```

Research question

This is a movie data set that provides various categorical and numerical variables about any major Hollywood released movie since 1936. Our main goal is to answer, "what makes a Hollywood movie profitable?" For example, do genre, mpaa rating, production budget and/or release month(s) contribute to a successful/useful multiple-regression formula in terms of predicting a movie worldwide revenue gross? Box office revenue and production budget are reported in USD and scaled to current monetary value.

Cases

After removing a single duplicated case, we are left with 3400 cases from the original data set; however, we need to filter out movies that are not yet released as of this moment (such as the Aquaman won't be released until December 14, 2018). In addition, we need to remove cases that have missing values. Finally, we are left with 3202 cases.

Data collection

The data set (csv) is cloned from github, and it is complied from a social science project "tidytuesday" - a weekly social data project in R. As the name described, "tidytuesday" would post a data set on github every Tuesday. This movie data set is from Oct 23, 2018 and the original data is come from numbers.com - a movie industry data website that tracks box office revenue in a systematic, algorithmic way.

Type of study

This is an observational study with no interference of the box office.

Data Source

Please see the links,

https://github.com/rfordatascience/tidytuesday/tree/master/data/2018-10-23 https://thomasmock.netlify.com/post/tidytuesday-a-weekly-social-data-project-in-r/ https://www.the-numbers.com/research-analysis

Dependent Variable

The "worldwide_gross" would be the dependent variable. It is quantitative. However, in the final project, we will likely transform and come up with a different dependent variable, e.g. "worldwide_gross / production_budget" in order to better capture the ROI of a movie.

Independent Variable

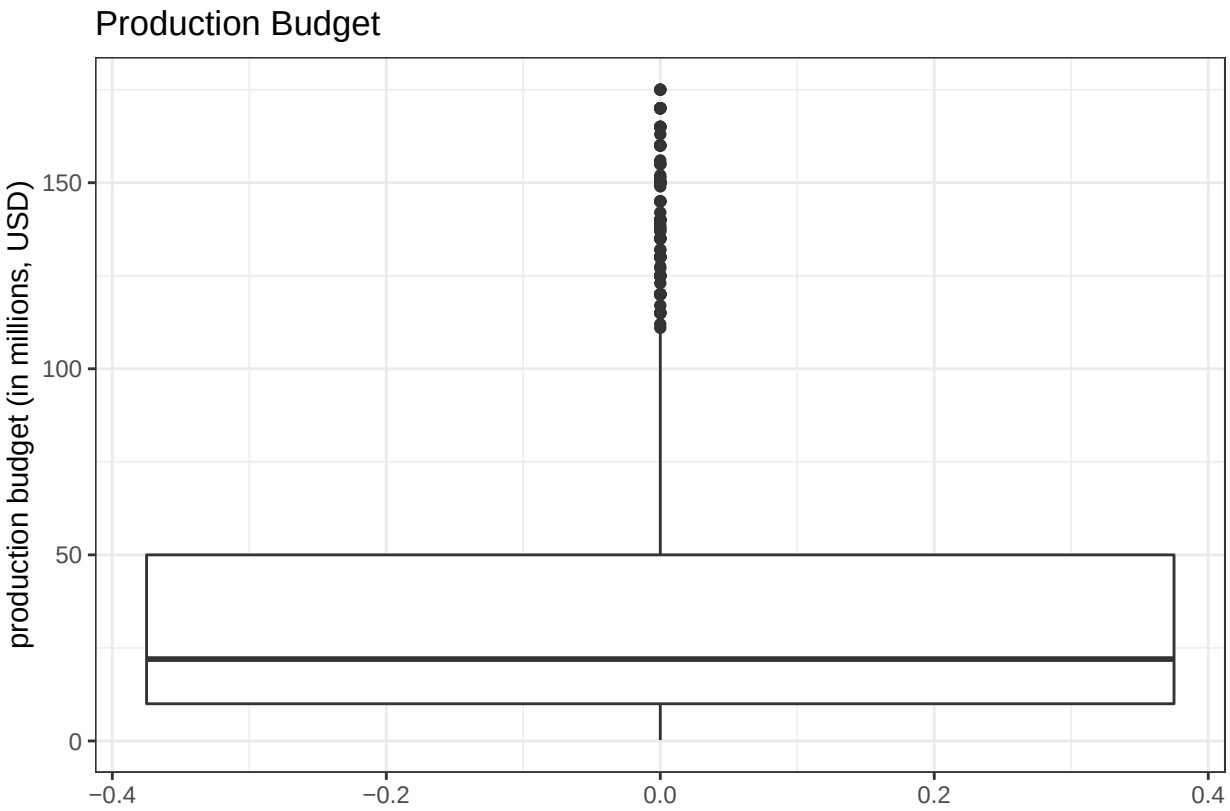
We have the following from the data set, such as

- "production_budget" (quantitative)
- "mpaa_rating" (qualitative)
- "genre" (qualitative)
- "release_month" (qualitative - we should treat it as a seasonal factor)

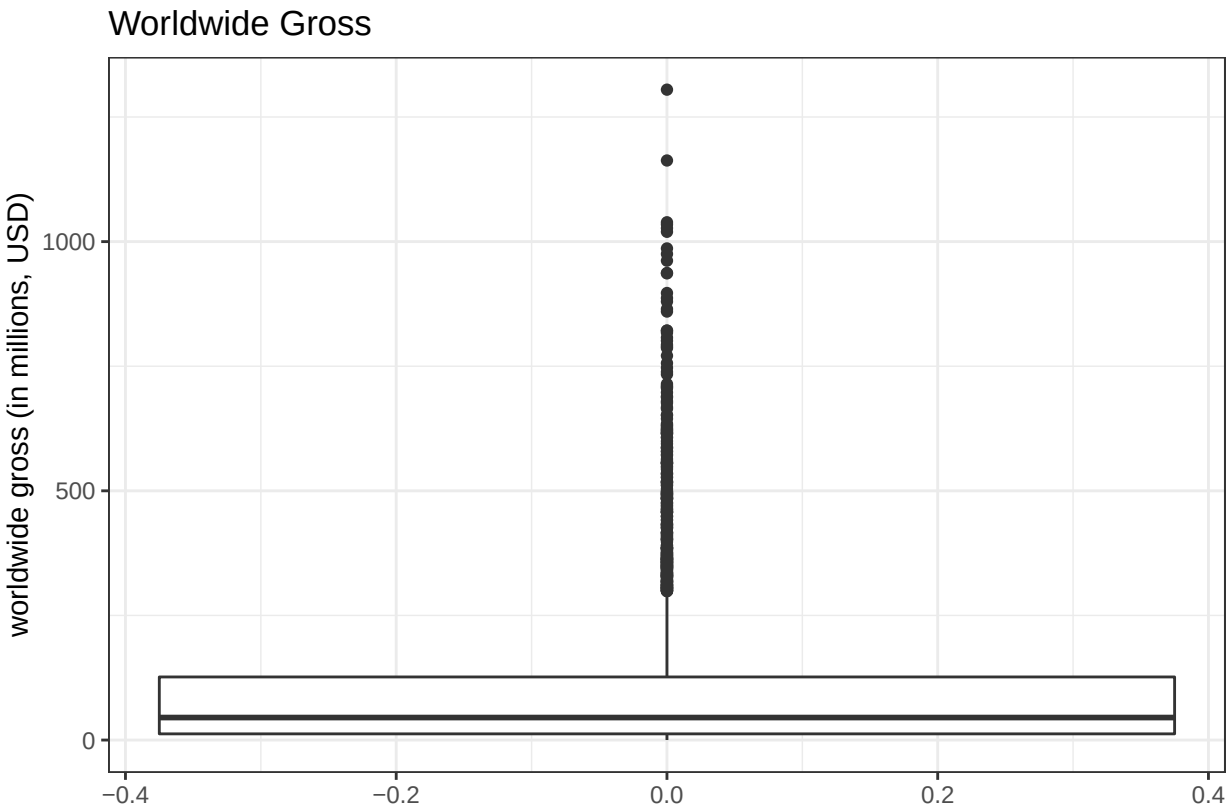
Relevant summary statistics

Below is some summary statistics and visualization for some of the variables discussed above.

```
>
> ##############################################################################################
> # production_budget
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         select(production_budget) %>%
+         summary
##  production_budget
##  Min.   :  0.25
##  1st Qu.: 10.00
##  Median : 22.00
##  Mean   : 34.77
##  3rd Qu.: 50.00
##  Max.   :175.00
>
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         ggplot(data = ., aes(y = production_budget)) +
+         geom_boxplot() +
+         ggtitle("Production Budget") +
+         theme_bw() +
+         labs(x = "", y = "production budget (in millions, USD)")
```

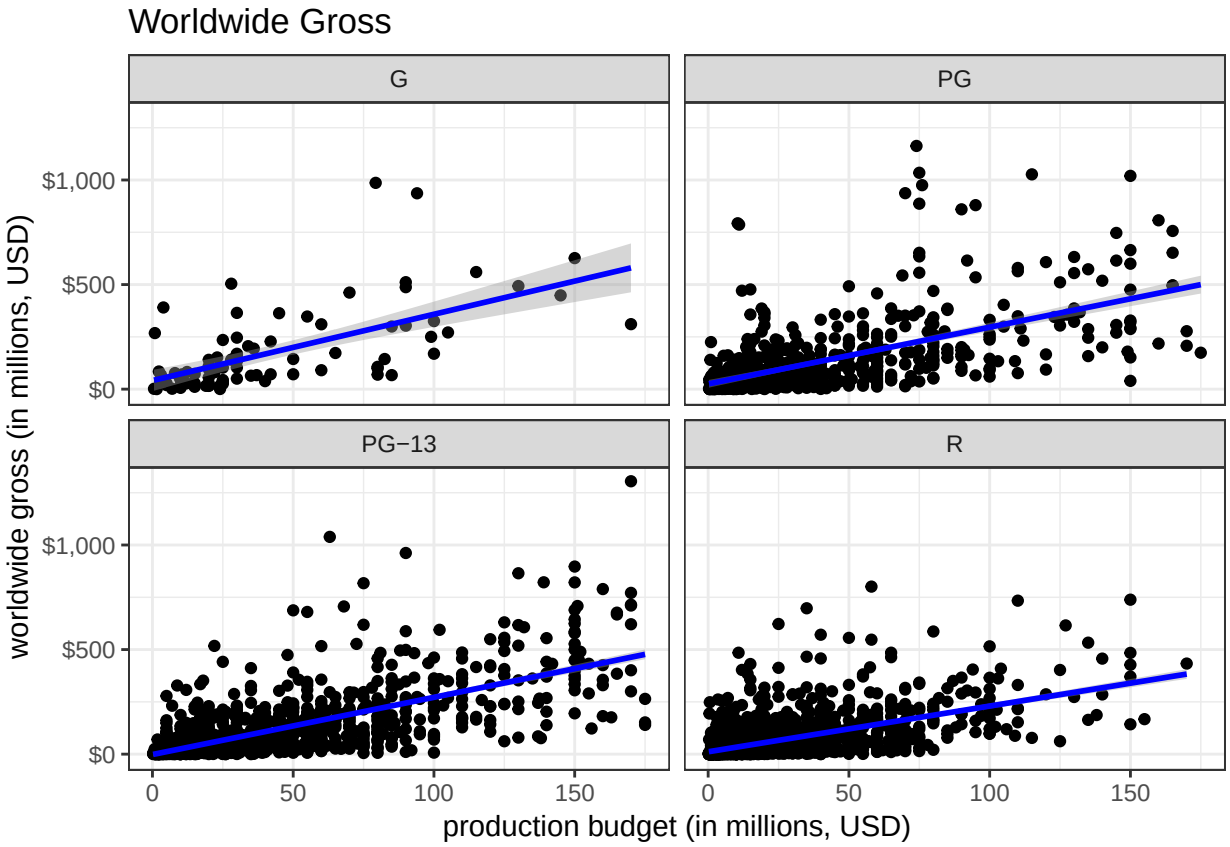



```
>
> # worldwide_gross
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         select(worldwide_gross) %>%
+         summary
##  worldwide_gross
##  Min.   :   0.0004
##  1st Qu.:  12.4073
##  Median :  45.1106
##  Mean   :  99.2777
##  3rd Qu.: 126.5470
##  Max.   :1304.8663
>
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         ggplot(data = ., aes(y = worldwide_gross)) +
+         geom_boxplot() +
+         ggtitle("Worldwide Gross") +
+         theme_bw() +
+         labs(x = "", y = "worldwide gross (in millions, USD)")
```



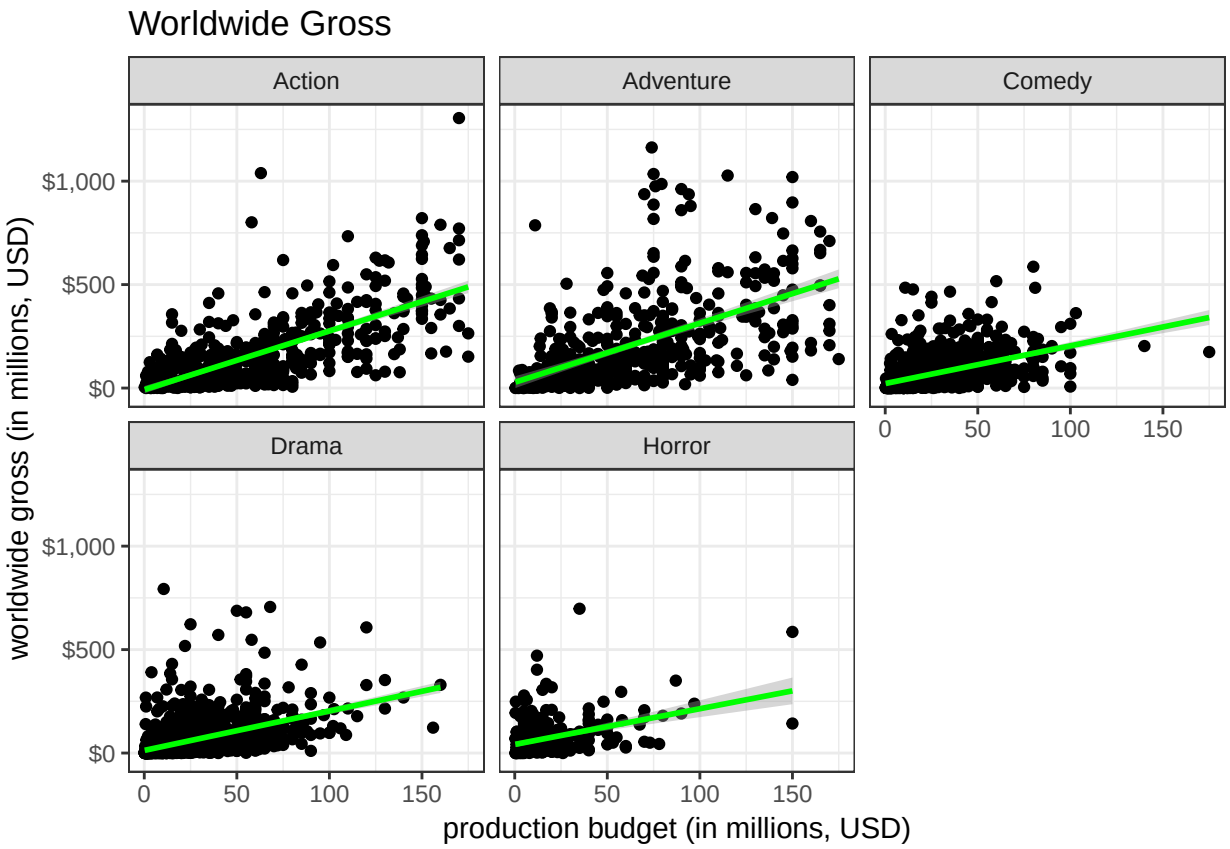

```
>
> ##############################################################################################
> # table for categorical variables
> # it would be interesting to create a 2 x 2 contingency table
> # or an array, in order to look at the data from a multi-dimensional perspective
> # alternatively, we can do an ANOVA test on any of these with worldwide_gross
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         select(mpaa_rating) %>%
+         ftable
## x    G   PG PG-13    R
##
##     84  560  1082 1476
>
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         select(genre) %>%
+         ftable
## x Action Adventure Comedy Drama Horror
##
##      532       465    769  1172    264
>
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         select(release_month) %>%
+         ftable
```

```
## x   1   2   3   4   5   6   7   8   9  10  11  12
##
##   181 217 267 230 211 286 258 276 282 328 305 361
>
> ##############################################################################################
> # in general, it seems reasonable to believe that there's a positive correlation
> # between production budget and worldwide gross
>
> # PG, PG-13 generate the most revenue
> # Action, Adventure are the most profitable movie genre
> # summer (May, June & July) is always the best time to roll out blockbuster movies!
>
> # production_budget x worldwide_gross by mpaa_rating
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         ggplot(data = ., aes(x = production_budget, y = worldwide_gross)) +
+         geom_point() +
+         scale_y_continuous(labels = scales::dollar) +
+         geom_smooth(method = "lm", col = "blue") +
+         facet_wrap(~mpaa_rating) +
+         ggtitle("Worldwide Gross") +
+         labs(x = "production budget (in millions, USD)", y = "worldwide gross (in millions, USD)") +
+         theme_bw()
```

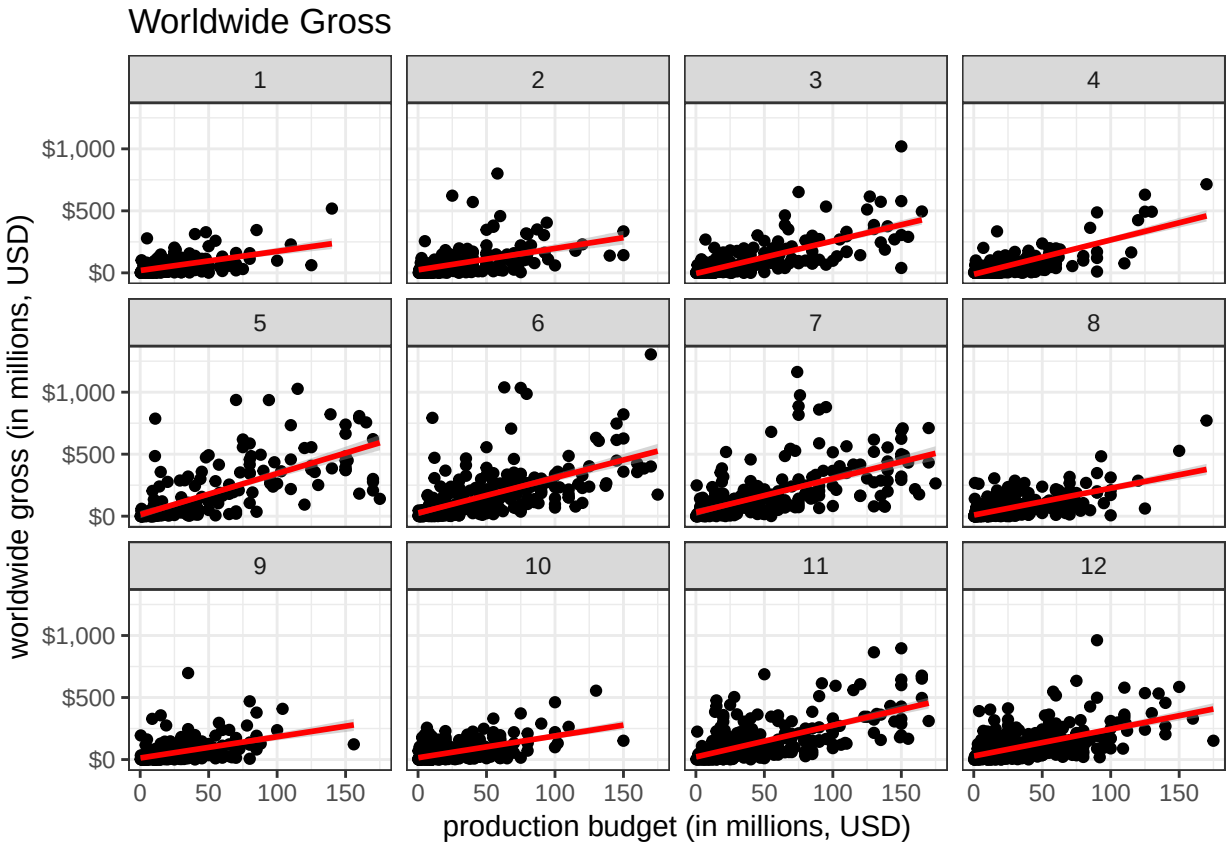



```
>
> # production_budget x worldwide_gross by genre
```

```
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         ggplot(data = ., aes(x = production_budget, y = worldwide_gross)) +
+         geom_point() +
+         scale_y_continuous(labels = scales::dollar) +
+         geom_smooth(method = "lm", col = "green") +
+         facet_wrap(~genre) +
+         ggtitle("Worldwide Gross") +
+         labs(x = "production budget (in millions, USD)", y = "worldwide gross (in millions, USD)") +
+         theme_bw()
```




```
>
> # production_budget x worldwide_gross by release_month
> dfComplete %>%
+         filter(worldwide_flag == 1) %>%
+         ggplot(data = ., aes(x = production_budget, y = worldwide_gross)) +
+         geom_point() +
+         scale_y_continuous(labels = scales::dollar) +
+         geom_smooth(method = "lm", col = "red") +
+         facet_wrap(~release_month) +
+         ggtitle("Worldwide Gross") +
+         labs(x = "production budget (in millions, USD)", y = "worldwide gross (in millions, USD)") +
+         theme_bw()
```


Worldwide Gross
1
2
3
4
5
6
7
8
9
10
11
12
$1,000
$500
$0
0
50
100
150
worldwide gross (in millions, USD)
production budget (in millions, USD)