Analysis of Lead Gender and Box Office

by Max Woolf (@minimaxir)

```r
#options(warn = -1)

# IMPORTANT: This assumes that all packages in "Rstart.R" are installed,
# and the fonts "Source Sans Pro" and "Open Sans Condensed Bold" are installed
# via extrafont. If ggplot2 charts fail to render, you may need to change/remove the theme call.

source("Rstart.R")
library(outliers)

sessionInfo()
```

```
Attaching package: ‘’dplyr

The following objects are masked from ‘package:’stats:

    filter, lag

The following objects are masked from ‘package:’base:

    intersect, setdiff, setequal, union

Registering fonts with R

Attaching package: ‘’scales

The following objects are masked from ‘package:’readr:

    col_factor, col_numeric






R version 3.2.3 (2015-12-10)
Platform: x86_64-apple-darwin13.4.0 (64-bit)
Running under: OS X 10.11.4 (El Capitan)

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] grid      stats     graphics  grDevices utils     datasets  methods
[8] base

other attached packages:
```

```
[1] outliers_0.14      stringr_1.0.0      digest_0.6.8       RColorBrewer_1.1-2
[5] scales_0.3.0       extrafont_0.17     ggplot2_2.0.0      dplyr_0.4.3
[9] readr_0.1.1

loaded via a namespace (and not attached):
 [1] Rcpp_0.12.1      Rttf2pt1_1.3.3   magrittr_1.5     munsell_0.4.2
 [5] uuid_0.1-2       colorspace_1.2-6 R6_2.1.1         plyr_1.8.3
 [9] tools_3.2.3      parallel_3.2.3   gtable_0.1.2     DBI_0.3.1
[13] extrafontdb_1.0  assertthat_0.1   IRdisplay_0.3    repr_0.4
[17] base64enc_0.1-3  IRkernel_0.5     evaluate_0.8     rzmq_0.7.7
[21] stringi_0.5-5    jsonlite_0.9.19
```

Process the Data

Take the movies data, load in R friendly format, and combine with Rotten Tomatoes data.

```
df <- read_delim("~/Downloads/omdb0316/omdbMovies.txt", "\t",
    col_types="iccicccccccidi_c_____")
df_tomatoes <- read_delim("~/Downloads/omdb0316/tomatoes.txt",  "\t",
    col_types="i_diiiicidi_cc_c")
df <- df %>% left_join(df_tomatoes, by="ID")
rm(df_tomatoes)
```

```
|================================================================================| 100%  435
    MB
```

```
parseBoxOffice <- function(x) {
    unit <- 0
    if (is.na(x) | x=="") {return (NA)}
    if (substr(x, nchar(x), nchar(x)) == "k") {unit <- 10^3}
    else {unit <- 10^6}

    number <- as.numeric(substr(x,2,nchar(x)-1))

    return(number * unit)
}

df <- df %>% mutate(BoxOffice = as.numeric(sapply(BoxOffice, parseBoxOffice)))
```

```
df_dup <- df %>% select(Title, Year) %>% mutate(Title = gsub("The ", "", Title))
dup <- duplicated(df_dup)   # find entry indices which are duplicates
rm(df_dup)   # remove temp dataframe

df <- df %>% filter(!dup)   # keep entries which are *not* dups
```

Inflation

```
inflation <- read_csv("http://research.stlouisfed.org/fred2/data/CPIAUCSL.csv") %>%
                    group_by(Year = as.integer(substr(DATE, 1, 4))) %>%
                    summarize(Avg_Value = mean(VALUE)) %>%   # average across all months
                    mutate(Adjust = tail(Avg_Value, 1) / Avg_Value)   # normalize by
                        most-recent year
```

```
df <- df %>% inner_join(inflation) %>% mutate(AdjBoxOffice = floor(BoxOffice * Adjust))
```

```
Joining by: "Year"
```

Select only data we need now.

```
df <- df %>% filter(Year >= 2000, AdjBoxOffice >= 10^7, Cast != '') %>%
        select(imdbID, Title, Year, Cast, Meter, Metacritic, AdjBoxOffice) %>%
        arrange(desc(AdjBoxOffice))

#write.csv(df, "test.csv", row.names=F)
print(nrow(df))
```

```
[1] 2048
```

Determine Gender of Lead

```
# Helper function to get first actor given a string of actors
getLeadActor <- function(actors) {
    return(unlist(strsplit(actors,", "))[1])
}

# Unicode issues during testing, so use string w/ unicode as a test case
print(getLeadActor("Will Smith, Robert De Niro, Renée Zellweger, Jack Black"))
```

```
[1] "Will Smith"
```

```
df$LeadActor <- as.character(lapply(enc2utf8(df$Cast), getLeadActor))

print(head(df %>% select(Title, LeadActor)))
print(head(df %>% filter(imdbID=="tt0307453") %>% select(Title, LeadActor)))
```

```
Source: local data frame [6 x 2]

                                       Title         LeadActor
                                       (chr)             (chr)
1 Star Wars: Episode VII - The Force Awakens     Harrison Ford
2                                     Avatar   Sam Worthington
3                             Jurassic World       Chris Pratt
4                               The Avengers Robert Downey Jr.
5                            The Dark Knight    Christian Bale
6                                    Shrek 2        Mike Myers
Source: local data frame [1 x 2]

       Title  LeadActor
       (chr)      (chr)
1 Shark Tale Will Smith
```

Attempt #1: Merge known gender data of actors. (via list from Matt Daniels)

```
actor_gender <- read_csv("actor_list.csv") %>% select(LeadActor = name, Gender = gender)

print(head(actor_gender))
```

```
Source: local data frame [6 x 2]

          LeadActor Gender
              (chr)  (chr)
1       Keir Dullea      m
2     Gary Lockwood      m
3 William Sylvester      m
4    Daniel Richter      m
5  Leonard Rossiter      m
6   Margaret Tyzack      f
```

```
df <- df %>% left_join(actor_gender)

print(head(df %>% select(Title, LeadActor, Gender)))
```

```
Joining by: "LeadActor"


Source: local data frame [6 x 3]

                                       Title         LeadActor Gender
                                       (chr)             (chr)  (chr)
1 Star Wars: Episode VII - The Force Awakens     Harrison Ford      m
2                                     Avatar   Sam Worthington     NA
3                             Jurassic World       Chris Pratt     NA
4                               The Avengers Robert Downey Jr.     NA
5                            The Dark Knight    Christian Bale     NA
6                                    Shrek 2        Mike Myers     NA
```

Attempt #2: Determine gender from most-likely guess from first name. (Using male and female lists from Carnegie-Mellon University)

```
male_names <- unlist(read_delim("male_names.txt", "\n", skip = 6, col_names=F))
female_names <- unlist(read_delim("female_names.txt", "\n", skip = 6, col_names=F))

print(head(male_names))
print(head(female_names))
```

```
     X11      X12      X13      X14      X15      X16
 "Aamir"  "Aaron"  "Abbey"  "Abbie"  "Abbot" "Abbott"
      X11       X12       X13       X14       X15       X16
"Abagael" "Abagail"    "Abbe"   "Abbey"    "Abbi"   "Abbie"
```

```
getGenderFromFullName <- function (full_name) {
    first_name <- unlist(strsplit(full_name, " "))[1]
    gender <- ifelse(first_name %in% male_names, "m",
                     ifelse(first_name %in% female_names, "f", "[EDIT ME]"))
    return (gender)
}

print(getGenderFromFullName("Sam Worthington"))
print(getGenderFromFullName("Kristen Wiig"))
```

```
[1] "m"
[1] "f"
```

```r
gender_guess <- as.character(lapply(as.character(df$LeadActor), getGenderFromFullName))

# if a known gender from IMDB is present, use that; else, use the gender guess
df$Gender <- ifelse(is.na(df$Gender), gender_guess, df$Gender)

print(head(df %>% select(Title, LeadActor, Gender)))
print(tail(df %>% select(Title, LeadActor, Gender)))
```

```
Source: local data frame [6 x 3]

                                       Title         LeadActor Gender
                                       (chr)             (chr)  (chr)
1 Star Wars: Episode VII - The Force Awakens     Harrison Ford      m
2                                     Avatar   Sam Worthington      m
3                             Jurassic World       Chris Pratt      m
4                               The Avengers Robert Downey Jr.      m
5                            The Dark Knight    Christian Bale      m
6                                    Shrek 2        Mike Myers      m
Source: local data frame [6 x 3]

                         Title              LeadActor    Gender
                         (chr)                  (chr)     (chr)
1                      The Man      Samuel L. Jackson         m
2           Aliens of the Deep Anatoly M. Sagalevitch         m
3                 A Single Man            Colin Firth         m
4                      Pollock              Ed Harris         m
5             Connie and Carla           Nia Vardalos [EDIT ME]
6 I Don't Know How She Does It   Sarah Jessica Parker         f
```

Attempt #3: Manually edit edge cases in a GUI (not shown)

```r
write.csv(df, "movie_gender_intermediate.csv", row.names=F)
```

Begin the Analysis

Reload the updated dataset (a few rows were removed due to being dupes)

```r
df <- read_csv("movie_gender_fixed.csv")

print(head(df %>% select(Title, LeadActor, Gender)))
print(nrow(df))
```

```
Source: local data frame [6 x 3]

                                       Title         LeadActor Gender
                                       (chr)             (chr)  (chr)
1 Star Wars: Episode VII - The Force Awakens      Daisy Ridley      f
2                                     Avatar   Sam Worthington      m
3                             Jurassic World       Chris Pratt      m
4                               The Avengers Robert Downey Jr.      m
5                            The Dark Knight    Christian Bale      m
6                                    Shrek 2        Mike Myers      m
[1] 2020
```

Can we remove any points (e.g. Star Wars) as outliers? (tests via R Explorations)

```
AdjRevenue <- unlist(head(df %>% select(AdjBoxOffice)))

dixon.test(AdjRevenue, opposite=F)
grubbs.test(AdjRevenue, opposite=F)
chisq.out.test(AdjRevenue, variance=var(AdjRevenue), opposite=F)
```

```
    Dixon test for outliers

data:  AdjRevenue
Q.AdjBoxOffice1 = 0.23695, p-value = 0.8916
alternative hypothesis: highest value 934381231 is an outlier








    Grubbs test for one outlier

data:  AdjRevenue
G.AdjBoxOffice1 = 1.52510, U = 0.44175, p-value = 0.2634
alternative hypothesis: highest value 934381231 is an outlier








    chi-squared test for outlier

data:  AdjRevenue
X-squared.AdjBoxOffice1 = 2.326, p-value = 0.1272
alternative hypothesis: highest value 934381231 is an outlier
```

No outlier detection test supports it.

Plot Box Office Revenues

```
df_summary <- df %>%
                group_by(Gender) %>%
                summarize(count = n(),
                          perc = n()/nrow(df),
                          mean = mean(AdjBoxOffice),
                          median = median(AdjBoxOffice))

color_m <- "#2980b9"
color_f <- "#27ae60"

print(df_summary)
```

```
Source: local data frame [2 x 5]

  Gender count      perc     mean   median
   (chr) (int)     (dbl)    (dbl)    (int)
1      f   467 0.2311881 65586882 44144648
2      m  1553 0.7688119 79786060 49841069
```

Plot Male and Female distributions separately, since medians are too close.

```
df_summary_m <- df_summary %>% filter(Gender=="m")

plot <- ggplot(df %>% filter(Gender=="m"), aes(x=AdjBoxOffice)) +
            geom_histogram(fill=color_m, bins=50, alpha=0.75) +
            fte_theme() +
            scale_x_log10(limits=c(10^7, 10^9), breaks=10^c(7:9), labels=c("$10M", "$100M",
                "$1B")) +
            geom_vline(xintercept=df_summary_m$mean, color="#1a1a1a") +
            geom_vline(xintercept=df_summary_m$median, color="#7f8c8d") +
            annotate(geom="text", label = "Mean:\n$79.8M", x=df_summary_f$mean+7*10^7, y=80,
                color="#1a1a1a", family="Source Sans Pro Bold", hjust=1, size=2) +
            annotate(geom="text", label = "Median:\n$49.8M", x=df_summary_m$median-0.5*10^7,
                y=80, color="#7f8c8d", family="Source Sans Pro Bold", hjust=1, size=2) +
            labs(title="Distribution of Box Office Revenues of Blockbusters w/ Male Lead",
                x="Domestic Box Office Revenue (2016 Dollars)", y="# of Movies")

max_save(plot, "movie-gender-1", "IMDb + Rotten Tomatoes via OMDb")

df_summary_f <- df_summary %>% filter(Gender=="f")

plot <- ggplot(df %>% filter(Gender=="f"), aes(x=AdjBoxOffice)) +
            geom_histogram(fill=color_f, bins=50, alpha=0.75) +
            fte_theme() +
            scale_x_log10(limits=c(10^7, 10^9), breaks=10^c(7:9), labels=c("$10M", "$100M",
                "$1B")) +
            geom_vline(xintercept=df_summary_f$mean, color="#1a1a1a") +
            geom_vline(xintercept=df_summary_f$median, color="#7f8c8d") +
            annotate(geom="text", label = "Mean:\n$65.6M", x=df_summary_f$mean+5*10^7, y=35,
                color="#1a1a1a", family="Source Sans Pro Bold", hjust=1, size=2) +
            annotate(geom="text", label = "Median:\n$44.1M", x=df_summary_f$median-0.5*10^7,
                y=35, color="#7f8c8d", family="Source Sans Pro Bold", hjust=1, size=2) +
            labs(title="Distribution of Box Office Revenues of Blockbusters w/ Female Lead",
                x="Domestic Box Office Revenue (2016 Dollars)", y="# of Movies")

max_save(plot, "movie-gender-2", "IMDb + Rotten Tomatoes via OMDb")
```

```
Error in vapply(x, is.null, logical(1)): object 'df_summary_f' not found



Warning message:
: Removed 2 rows containing missing values (geom_bar).
```

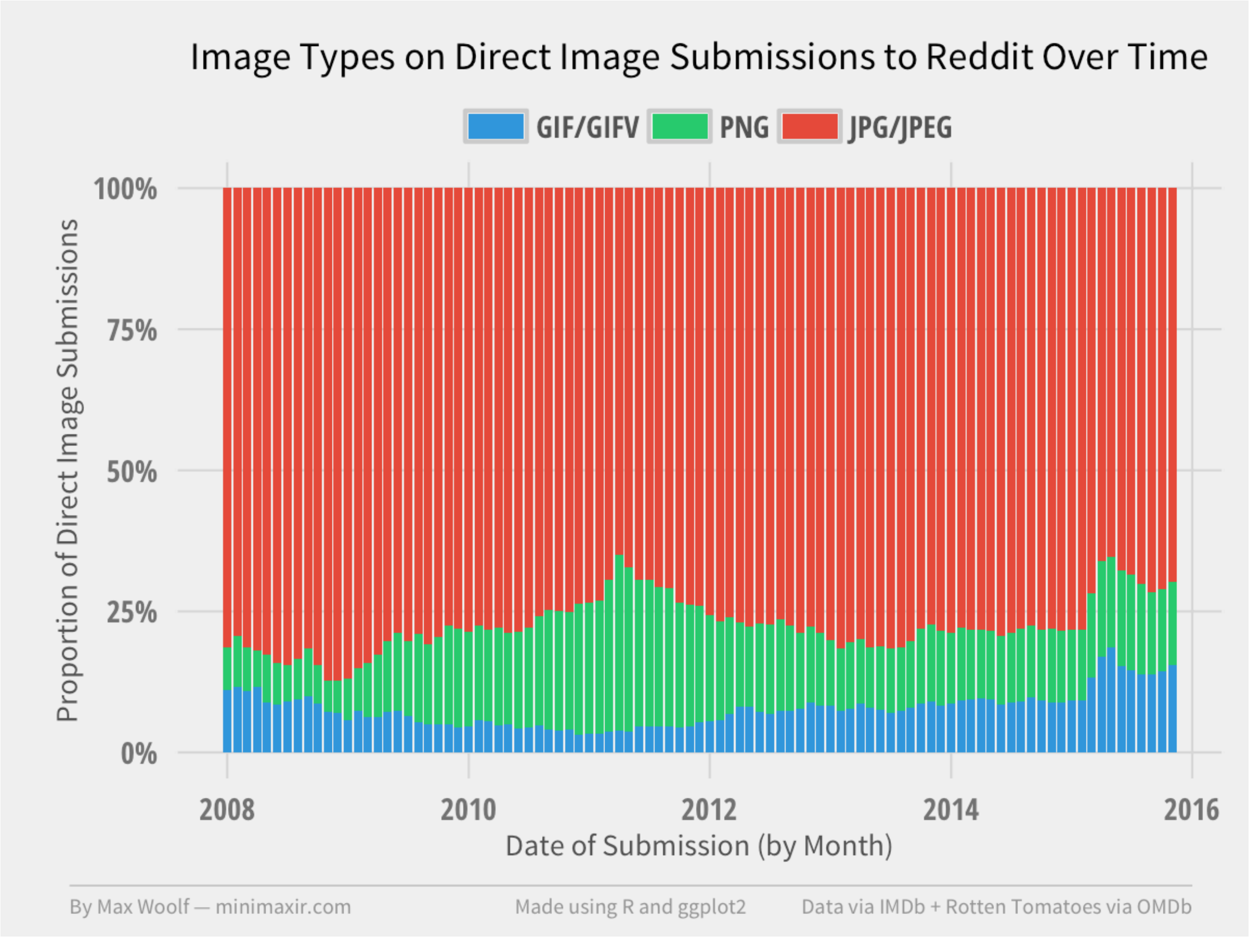
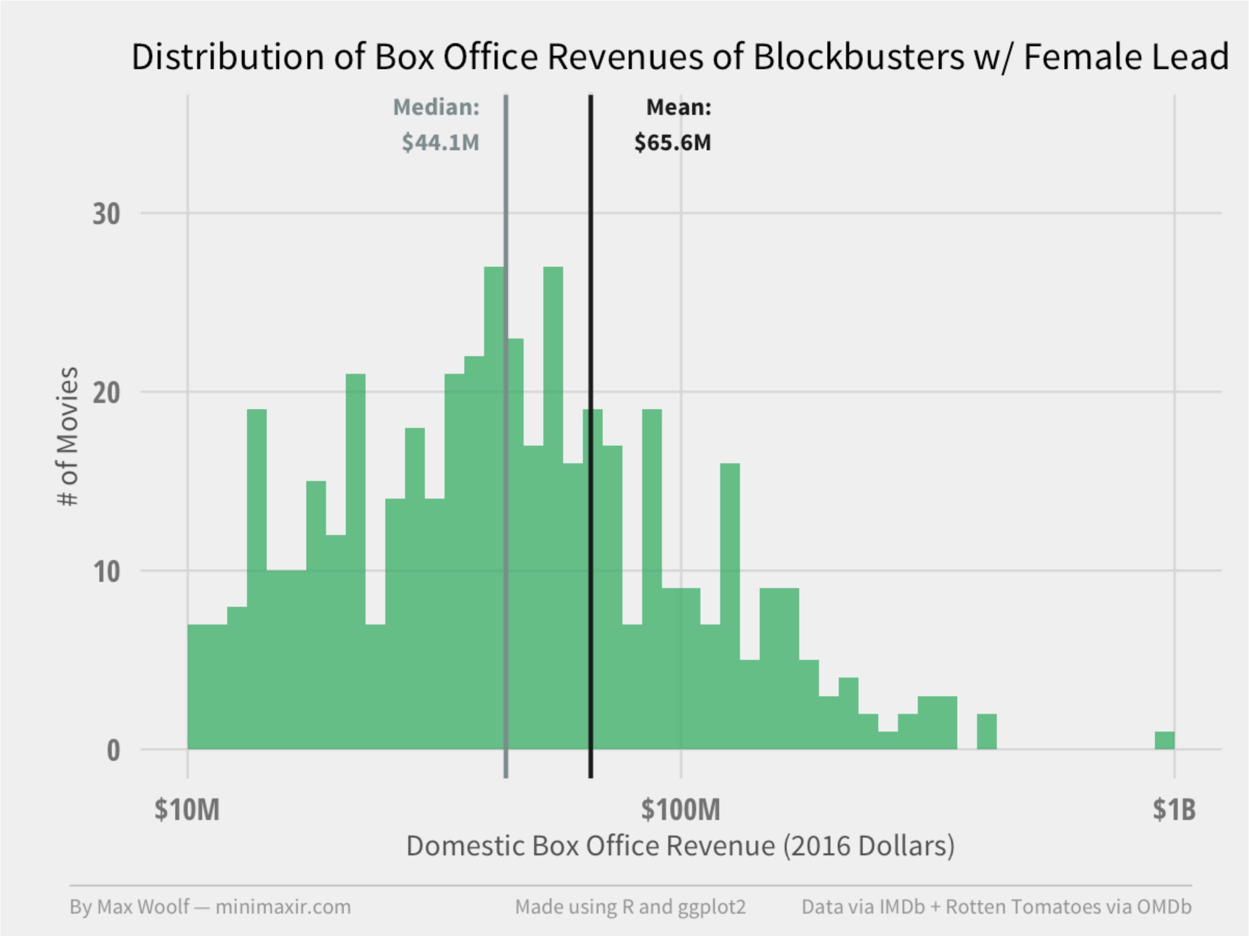


Image Types on Direct Image Submissions to Reddit Over Time
GIF/GIFV
PNG
JPG/JPEG
100%
75%
50%
25%
0%
Proportion of Direct Image Submissions
2008
2010
2012
2014
2016
Date of Submission (by Month)
By Max Woolf — minimaxir.com
Made using R and ggplot2
Data via IMDb + Rotten Tomatoes via OMDb



Plot kernel density distributions on each other.

```
plot <- ggplot(df, aes(x=AdjBoxOffice, fill=Gender)) +
            geom_density(alpha=0.75) +
            fte_theme() +
            scale_x_log10(limits=c(10^7, 10^9), breaks=10^c(7:9), labels=c("$10M", "$100M",
                "$1B")) +
            theme(legend.title = element_blank(), legend.position="top",
                legend.direction="horizontal", legend.key.width=unit(0.5, "cm"),
                legend.key.height=unit(0.25, "cm"), legend.margin=unit(0,"cm"),
                axis.title.y=element_blank(), axis.text.y=element_blank()) +
            scale_fill_manual(labels=c("Female Lead", "Male Lead"),
                values=c(color_f,color_m)) +
            labs(title="Density Distribution of B.O. Revenues of Blockbusters by Lead
                Gender", x="Domestic Box Office Revenue (2016 Dollars)")

max_save(plot, "movie-gender-3", "IMDb + Rotten Tomatoes via OMDb")
```

Do they come from a different distribution statistically speaking?

```
ks_test <- ks.test(
            unlist(df %>% filter(Gender=="m") %>% select(AdjBoxOffice)),
            unlist(df %>% filter(Gender=="f") %>% select(AdjBoxOffice)))

print(ks_test)
```

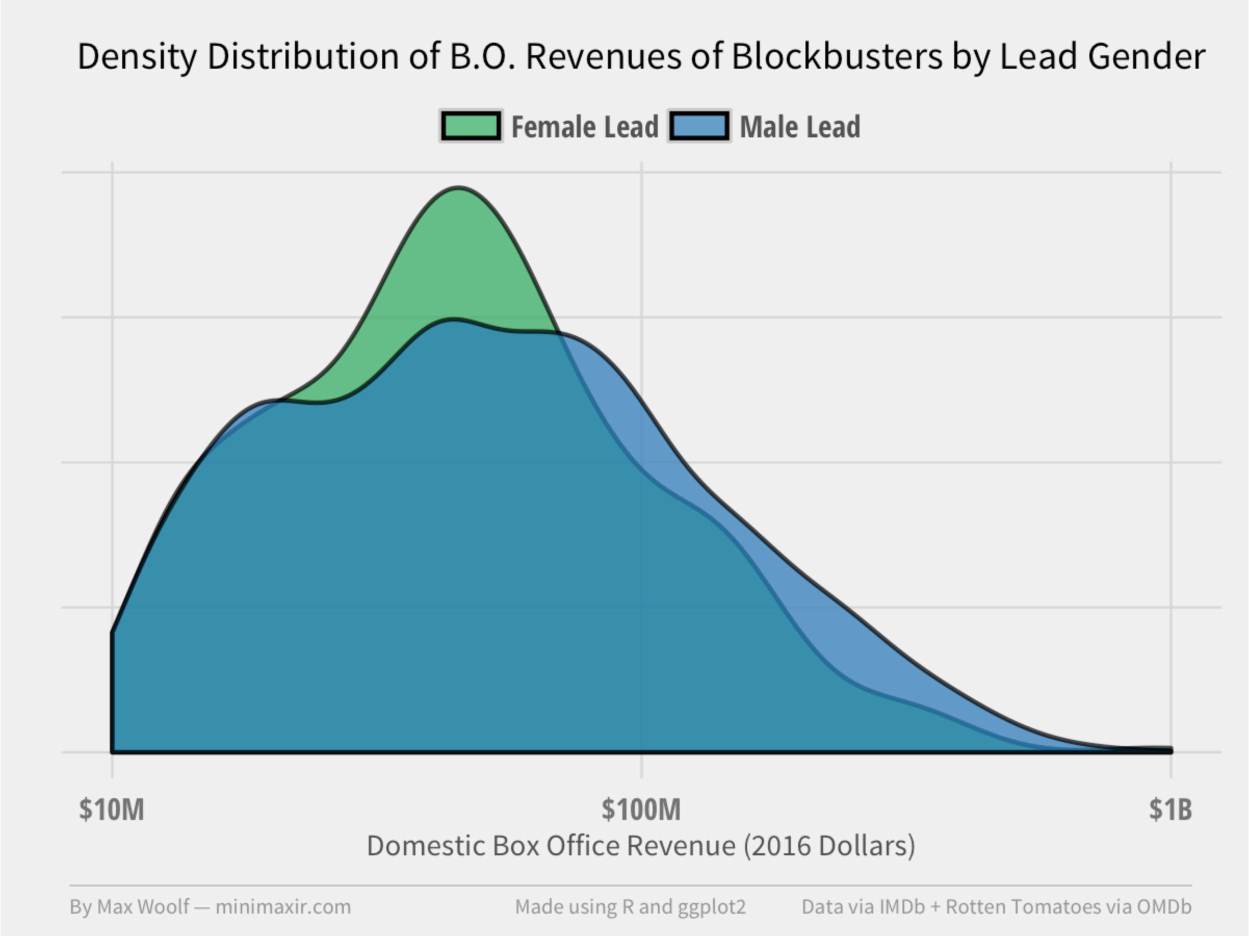

Density Distribution of B.O. Revenues of Blockbusters by Lead Gender
🟩 Female Lead
🟦 Male Lead
$10M
$100M
$1B
Domestic Box Office Revenue (2016 Dollars)
By Max Woolf — minimaxir.com
Made using R and ggplot2
Data via IMDb + Rotten Tomatoes via OMDb

Figure 1:

```

## check if log-scaling changes the result

ks_test <- ks.test(
            log10(unlist(df %>% filter(Gender=="m") %>% select(AdjBoxOffice))),
            log10(unlist(df %>% filter(Gender=="f") %>% select(AdjBoxOffice))))

print(ks_test)
```

```
Warning message:
In ks.test(unlist(df %>% filter(Gender == "m") %>% select(AdjBoxOffice)), : p-value will be approximate in the presence of ties


    Two-sample Kolmogorov-Smirnov test

data:  unlist(df %>% filter(Gender == "m") %>% select(AdjBoxOffice)) and unlist(df %>% filter(Gender == "f") %>% select(AdjBoxOffice))
D = 0.10585, p-value = 0.0006411
alternative hypothesis: two-sided



Warning message:
In ks.test(log10(unlist(df %>% filter(Gender == "m") %>% select(AdjBoxOffice))), : p-value will be approximate in the presence of ties


    Two-sample Kolmogorov-Smirnov test

data:  log10(unlist(df %>% filter(Gender == "m") %>% select(AdjBoxOffice))) and log10(unlist(df %>% filter(Gender == "f") %>% select(AdjBoxOffice)))
D = 0.10585, p-value = 0.0006411
alternative hypothesis: two-sided
```

The distributtion is different! Are the differences in means statistically significant?

```
wilcox_test <- wilcox.test(
            unlist(df %>% filter(Gender=="m") %>% select(AdjBoxOffice)),
            unlist(df %>% filter(Gender=="f") %>% select(AdjBoxOffice)),
            alternative = "g")

print(wilcox_test)

## check if log-scaling changes the result

wilcox_test <- wilcox.test(
            log10(unlist(df %>% filter(Gender=="m") %>% select(AdjBoxOffice))),
            log10(unlist(df %>% filter(Gender=="f") %>% select(AdjBoxOffice))),
            alternative = "g")

print(wilcox_test)
```

```
    Wilcoxon rank sum test with continuity correction

```

```
data:  unlist(df %>% filter(Gender == "m") %>% select(AdjBoxOffice)) and unlist(df %>% filter(Gender == "f") %>% select(AdjBoxOffice))
W = 390070, p-value = 0.006514
alternative hypothesis: true location shift is greater than 0


    Wilcoxon rank sum test with continuity correction

data:  log10(unlist(df %>% filter(Gender == "m") %>% select(AdjBoxOffice))) and log10(unlist(df %>% filter(Gender == "f") %>% select(AdjBoxOffice)))
W = 390070, p-value = 0.006514
alternative hypothesis: true location shift is greater than 0
```

Plot Rotten Tomatoes Meter

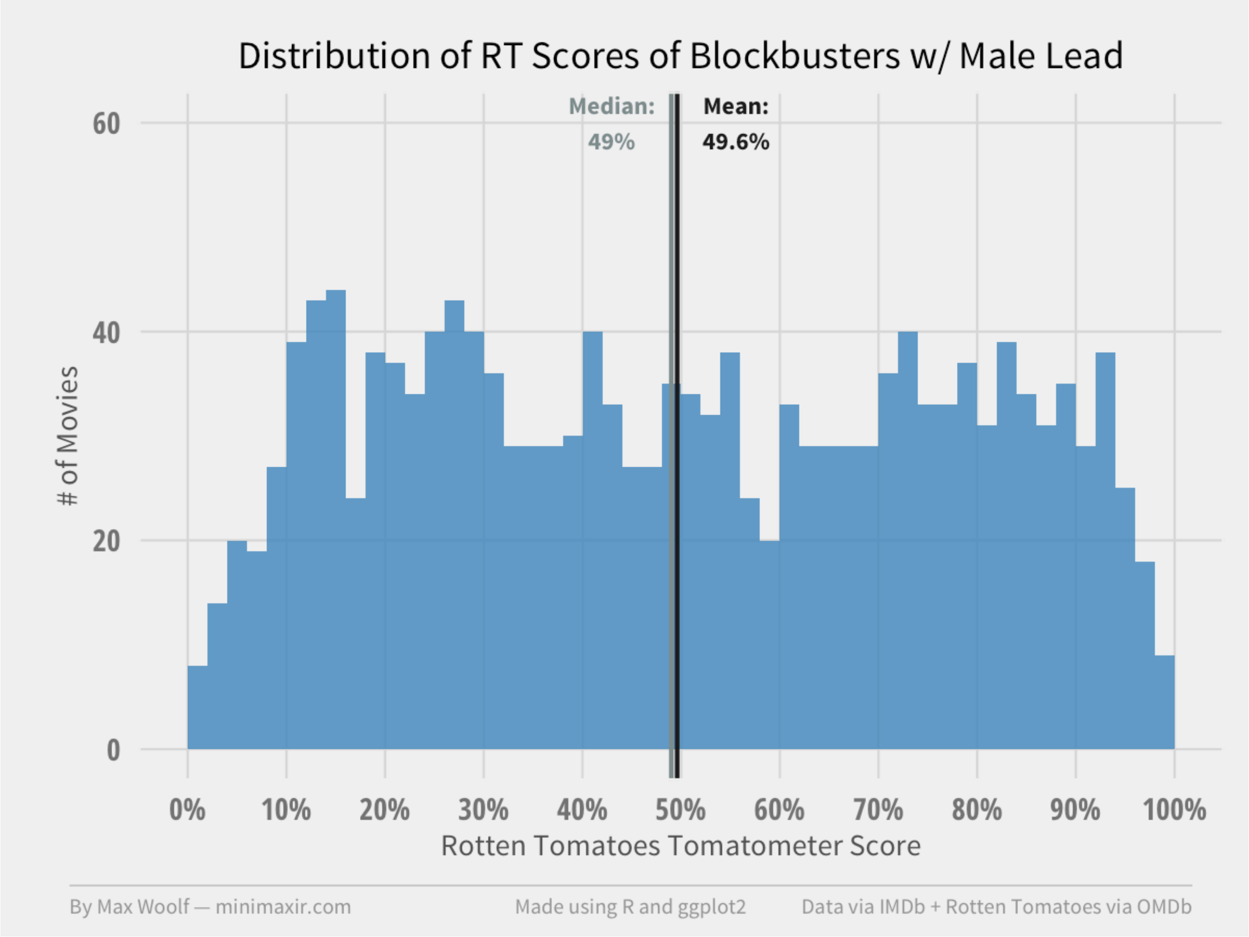
Can reuse most of the code, unfortunately have to violate DRY for ad-hoc fixes.

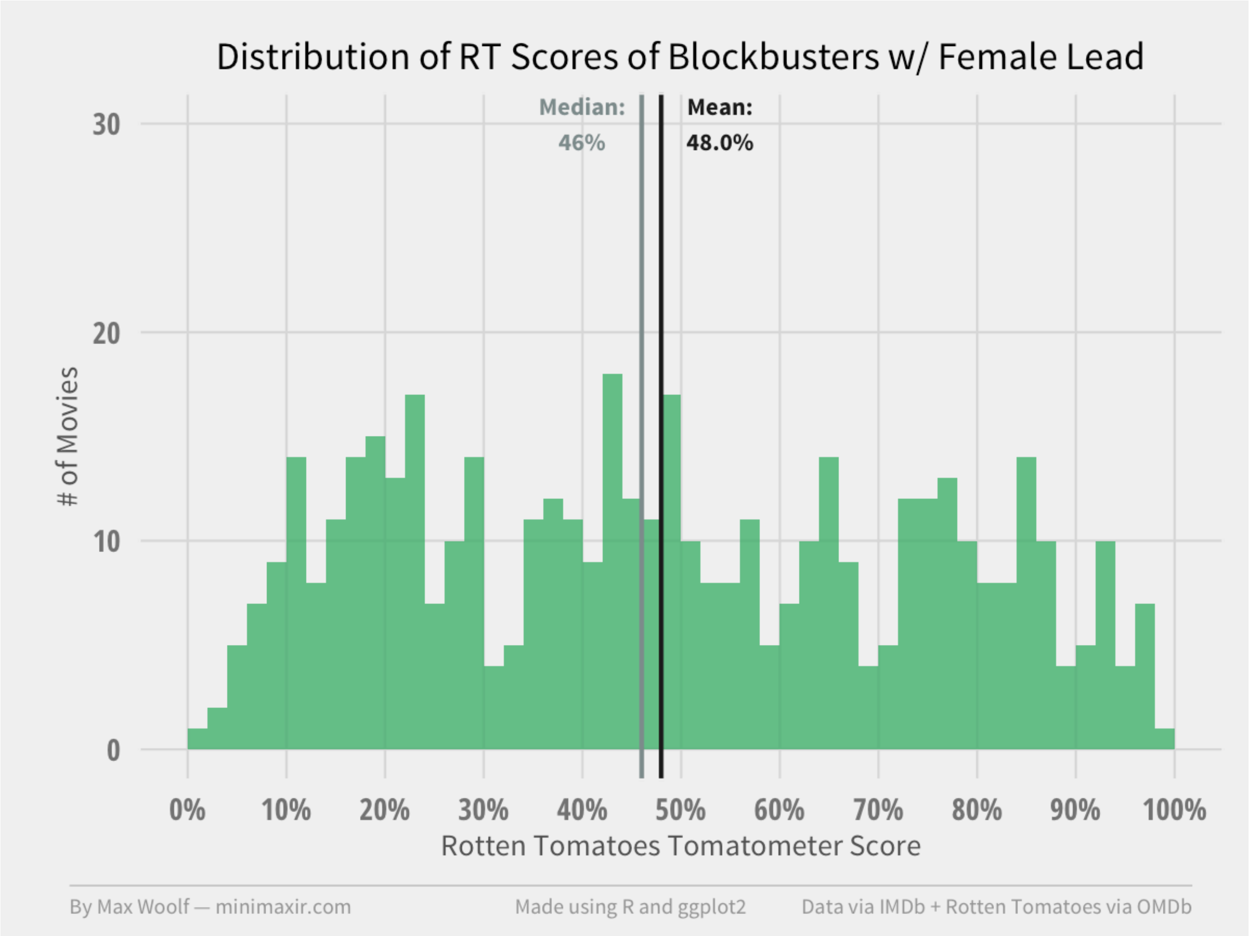
```
df_summary <- df %>%
                group_by(Gender) %>%
                summarize(mean = mean(Meter, na.rm=T), median = median(Meter, na.rm=T))


print(df_summary)
```

```
Source: local data frame [2 x 3]

  Gender     mean median
   (chr)    (dbl)  (int)
1      f 47.97859     46
2      m 49.59381     49
```

```
df_summary_m <- df_summary %>% filter(Gender=="m")

plot <- ggplot(df %>% filter(Gender=="m"), aes(x=Meter)) +
            geom_histogram(fill=color_m, bins=50, alpha=0.75) +
            fte_theme() +
            scale_x_continuous(breaks=seq(0,100, by=10), limits=c(0, 100), labels=paste0(seq(0,100, by=10),"%")) +
            geom_vline(xintercept=df_summary_m$mean, color="#1a1a1a") +
            geom_vline(xintercept=df_summary_m$median, color="#7f8c8d") +
            annotate(geom="text", label = "Mean:\n49.6%", x=df_summary_m$mean+6, y=60, color="#1a1a1a", family="Source Sans Pro Bold", size=2) +
            annotate(geom="text", label = "Median:\n49%", x=df_summary_m$median-6, y=60, color="#7f8c8d", family="Source Sans Pro Bold", size=2) +
            labs(title="Distribution of RT Scores of Blockbusters w/ Male Lead", x="Rotten Tomatoes Tomatometer Score", y="# of Movies")

max_save(plot, "movie-gender-4", "IMDb + Rotten Tomatoes via OMDb")

df_summary_f <- df_summary %>% filter(Gender=="f")

plot <- ggplot(df %>% filter(Gender=="f"), aes(x=Meter)) +
            geom_histogram(fill=color_f, bins=50, alpha=0.75) +
            fte_theme() +
            scale_x_continuous(breaks=seq(0,100, by=10), limits=c(0, 100),
                labels=paste0(seq(0,100, by=10),"%")) +
            geom_vline(xintercept=df_summary_f$mean, color="#1a1a1a") +
            geom_vline(xintercept=df_summary_f$median, color="#7f8c8d") +
            annotate(geom="text", label = "Mean:\n48.0%", x=df_summary_f$mean+6, y=30,
                color="#1a1a1a", family="Source Sans Pro Bold", size=2) +
            annotate(geom="text", label = "Median:\n46%", x=df_summary_f$median-6, y=30,
                color="#7f8c8d", family="Source Sans Pro Bold",  size=2) +
            labs(title="Distribution of RT Scores of Blockbusters w/ Female Lead", x="Rotten
                Tomatoes Tomatometer Score", y="# of Movies")

max_save(plot, "movie-gender-5", "IMDb + Rotten Tomatoes via OMDb")
```

```
Warning message:
: Removed 2 rows containing non-finite values (stat_bin).Warning message:
: Removed 2 rows containing missing values (geom_bar).Warning message:
: Removed 2 rows containing missing values (geom_bar).
```





```r
plot <- ggplot(df, aes(x=Meter, fill=Gender)) +
            geom_density(alpha=0.75) +
            fte_theme() +
            scale_x_continuous(breaks=seq(0,100, by=10), limits=c(0, 100),
                labels=paste0(seq(0,100, by=10),"%")) +
            theme(legend.title = element_blank(), legend.position="top",
                legend.direction="horizontal", legend.key.width=unit(0.5, "cm"),
                legend.key.height=unit(0.25, "cm"), legend.margin=unit(0,"cm"),
                axis.title.y=element_blank(), axis.text.y=element_blank()) +
            scale_fill_manual(labels=c("Female Lead", "Male Lead"),
                values=c(color_f,color_m)) +
            labs(title="Density Distribution of RT Scores of Blockbusters by Lead Gender",
                x="Rotten Tomatoes Tomatometer Score")

max_save(plot, "movie-gender-6", "IMDb + Rotten Tomatoes via OMDb")
```

```
Warning message:
: Removed 2 rows containing non-finite values (stat_density).
```

```r
ks_test <- ks.test(
            unlist(df %>% filter(Gender=="m") %>% select(Meter)),
            unlist(df %>% filter(Gender=="f") %>% select(Meter)))

print(ks_test)
```

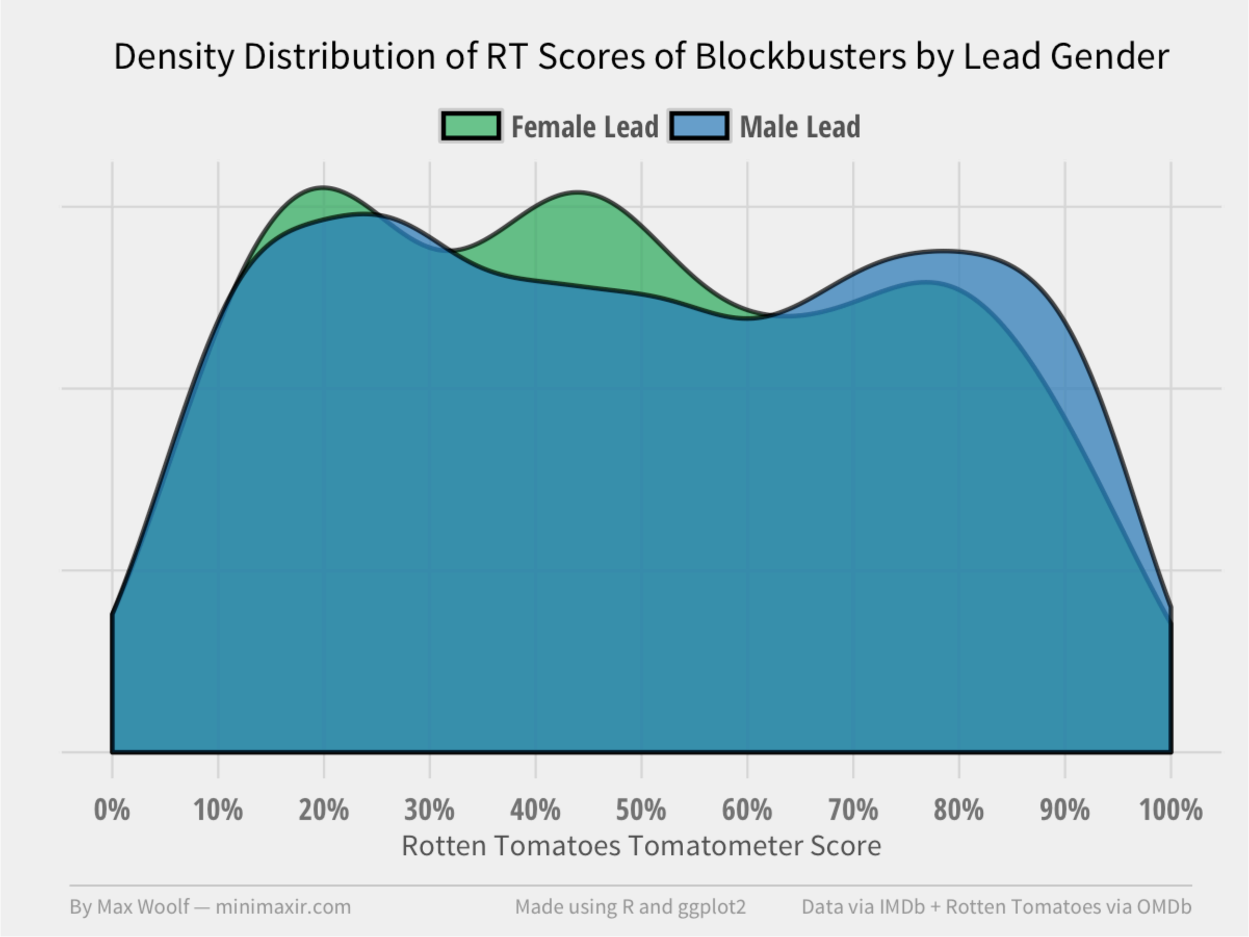

Density Distribution of RT Scores of Blockbusters by Lead Gender
Female Lead
Male Lead
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Rotten Tomatoes Tomatometer Score
By Max Woolf — minimaxir.com
Made using R and ggplot2
Data via IMDb + Rotten Tomatoes via OMDb

Figure 2:

```

wilcox_test <- wilcox.test(
            unlist(df %>% filter(Gender=="m") %>% select(Meter)),
            unlist(df %>% filter(Gender=="f") %>% select(Meter)),
            alternative="g")

print(wilcox_test)
```

```
Warning message:
In ks.test(unlist(df %>% filter(Gender == "m") %>% select(Meter)), : p-value will be
    approximate in the presence of ties


    Two-sample Kolmogorov-Smirnov test

data:  unlist(df %>% filter(Gender == "m") %>% select(Meter)) and unlist(df %>%
    filter(Gender == "f") %>% select(Meter))
D = 0.048455, p-value = 0.3684
alternative hypothesis: two-sided


    Wilcoxon rank sum test with continuity correction

data:  unlist(df %>% filter(Gender == "m") %>% select(Meter)) and unlist(df %>%
    filter(Gender == "f") %>% select(Meter))
W = 374460, p-value = 0.1326
alternative hypothesis: true location shift is greater than 0
```

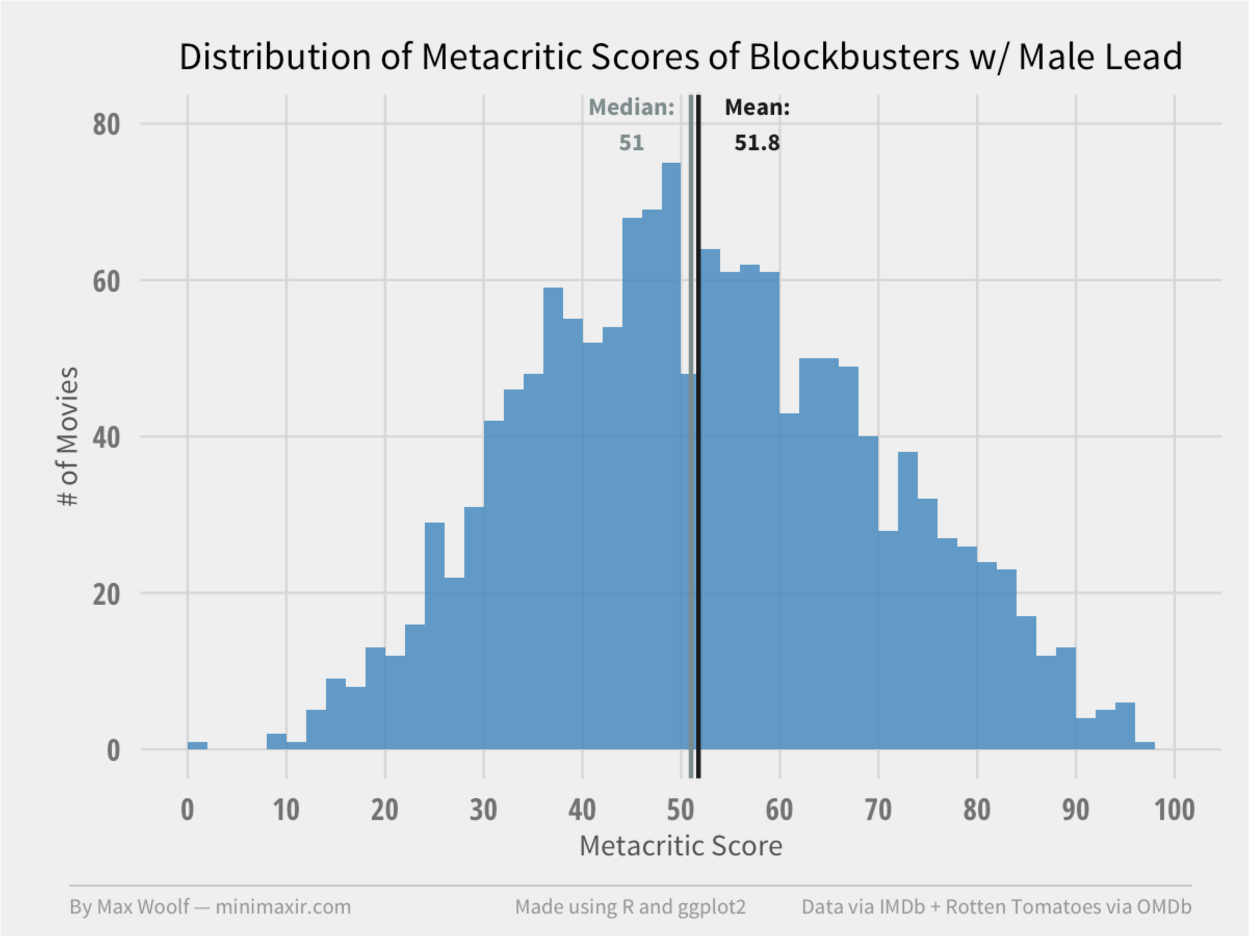
Plot Metacritic

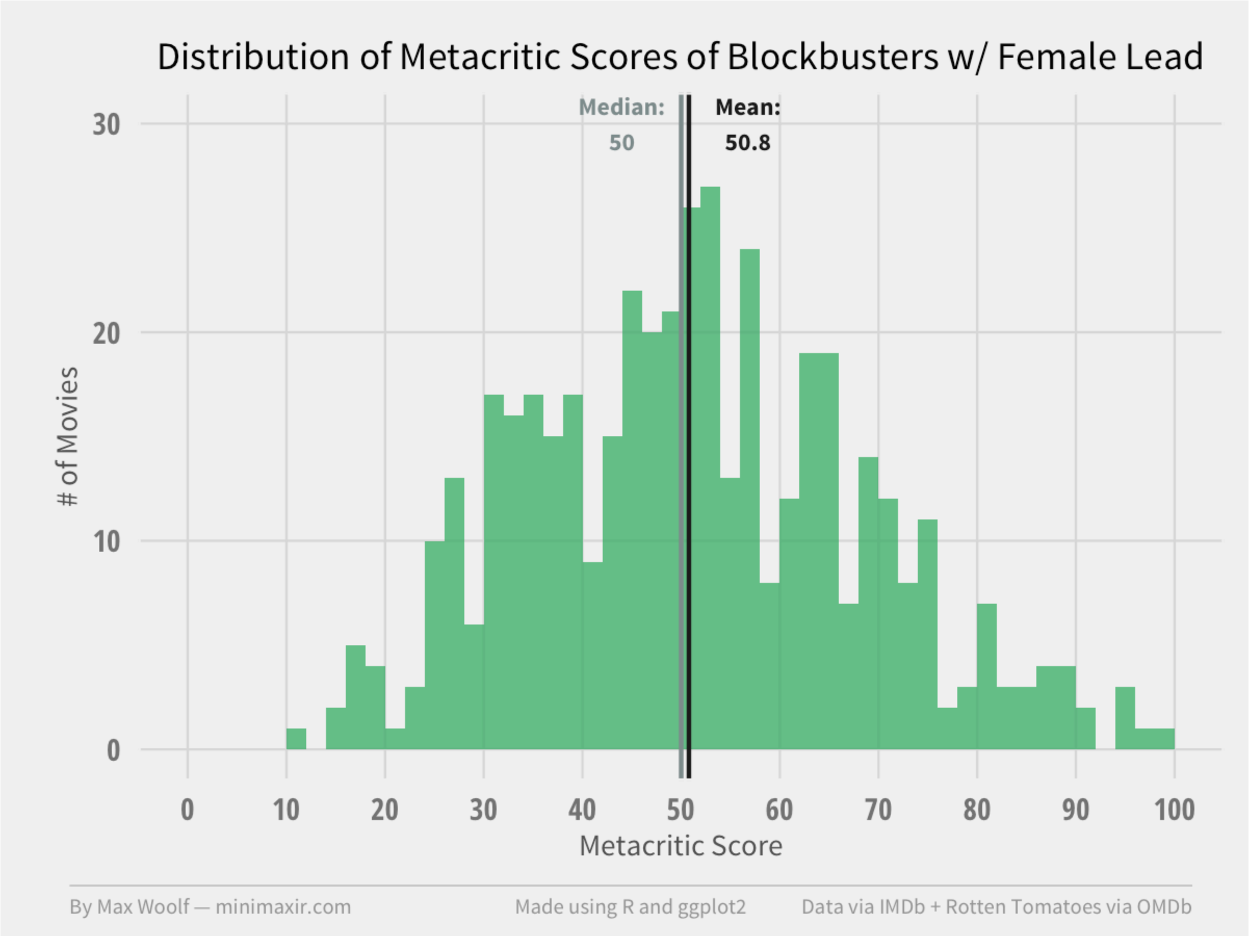
```
df_summary <- df %>%
                group_by(Gender) %>%
                summarize(mean = mean(Metacritic, na.rm=T), median = median(Metacritic,
                    na.rm=T))


print(df_summary)
```

```
Source: local data frame [2 x 3]

  Gender     mean median
   (chr)    (dbl)  (dbl)
1      f 50.78523     50
2      m 51.76032     51
```

```
df_summary_m <- df_summary %>% filter(Gender=="m")

plot <- ggplot(df %>% filter(Gender=="m"), aes(x=Metacritic)) +
            geom_histogram(fill=color_m, bins=50, alpha=0.75) +
            fte_theme() +
            scale_x_continuous(breaks=seq(0,100, by=10), limits=c(0, 100)) +
            geom_vline(xintercept=df_summary_m$mean, color="#1a1a1a") +
            geom_vline(xintercept=df_summary_m$median, color="#7f8c8d") +
            annotate(geom="text", label = "Mean:\n51.8", x=df_summary_m$mean+6, y=80,
                color="#1a1a1a", family="Source Sans Pro Bold", size=2) +
            annotate(geom="text", label = "Median:\n51", x=df_summary_m$median-6, y=80,
                color="#7f8c8d", family="Source Sans Pro Bold", size=2) +
            labs(title="Distribution of Metacritic Scores of Blockbusters w/ Male Lead",
                x="Metacritic Score", y="# of Movies")

max_save(plot, "movie-gender-7", "IMDb + Rotten Tomatoes via OMDb")

df_summary_f <- df_summary %>% filter(Gender=="f")

plot <- ggplot(df %>% filter(Gender=="f"), aes(x=Metacritic)) +
            geom_histogram(fill=color_f, bins=50, alpha=0.75) +
            fte_theme() +
            scale_x_continuous(breaks=seq(0,100, by=10), limits=c(0, 100)) +
            geom_vline(xintercept=df_summary_f$mean, color="#1a1a1a") +
            geom_vline(xintercept=df_summary_f$median, color="#7f8c8d") +
            annotate(geom="text", label = "Mean:\n50.8", x=df_summary_f$mean+6, y=30,
                color="#1a1a1a", family="Source Sans Pro Bold", size=2) +
            annotate(geom="text", label = "Median:\n50", x=df_summary_f$median-6, y=30,
                color="#7f8c8d", family="Source Sans Pro Bold",  size=2) +
            labs(title="Distribution of Metacritic Scores of Blockbusters w/ Female Lead",
                x="Metacritic Score", y="# of Movies")

max_save(plot, "movie-gender-8", "IMDb + Rotten Tomatoes via OMDb")
```

```
Warning message:
: Removed 51 rows containing non-finite values (stat_bin).Warning message:
: Removed 2 rows containing missing values (geom_bar).Warning message:
: Removed 20 rows containing non-finite values (stat_bin).Warning message:
: Removed 2 rows containing missing values (geom_bar).
```


Distribution of Metacritic Scores of Blockbusters w/ Male Lead
Median:
51
Mean:
51.8
# of Movies
0
20
40
60
80
0
10
20
30
40
50
60
70
80
90
100
Metacritic Score
By Max Woolf — minimaxir.com
Made using R and ggplot2
Data via IMDb + Rotten Tomatoes via OMDb


Distribution of Metacritic Scores of Blockbusters w/ Female Lead

By Max Woolf — minimaxir.com Made using R and ggplot2 Data via IMDb + Rotten Tomatoes via OMDb

```r
plot <- ggplot(df, aes(x=Metacritic, fill=Gender)) +
            geom_density(alpha=0.75) +
            fte_theme() +
            scale_x_continuous(breaks=seq(0,100, by=10), limits=c(0, 100)) +
            theme(legend.title = element_blank(), legend.position="top",
                legend.direction="horizontal", legend.key.width=unit(0.5, "cm"),
                legend.key.height=unit(0.25, "cm"), legend.margin=unit(0,"cm"),
                axis.title.y=element_blank(), axis.text.y=element_blank()) +
            scale_fill_manual(labels=c("Female Lead", "Male Lead"),
                values=c(color_f,color_m)) +
            labs(title="Density Distribution of Metacritic Scores of Blockbusters by Lead
                Gender", x="Metacritic Score")

max_save(plot, "movie-gender-9", "IMDb + Rotten Tomatoes via OMDb")
```

```
Warning message:
: Removed 71 rows containing non-finite values (stat_density).
```

```r
ks_test <- ks.test(
            unlist(df %>% filter(Gender=="m") %>% select(Metacritic)),
            unlist(df %>% filter(Gender=="f") %>% select(Metacritic)))

print(ks_test)

```

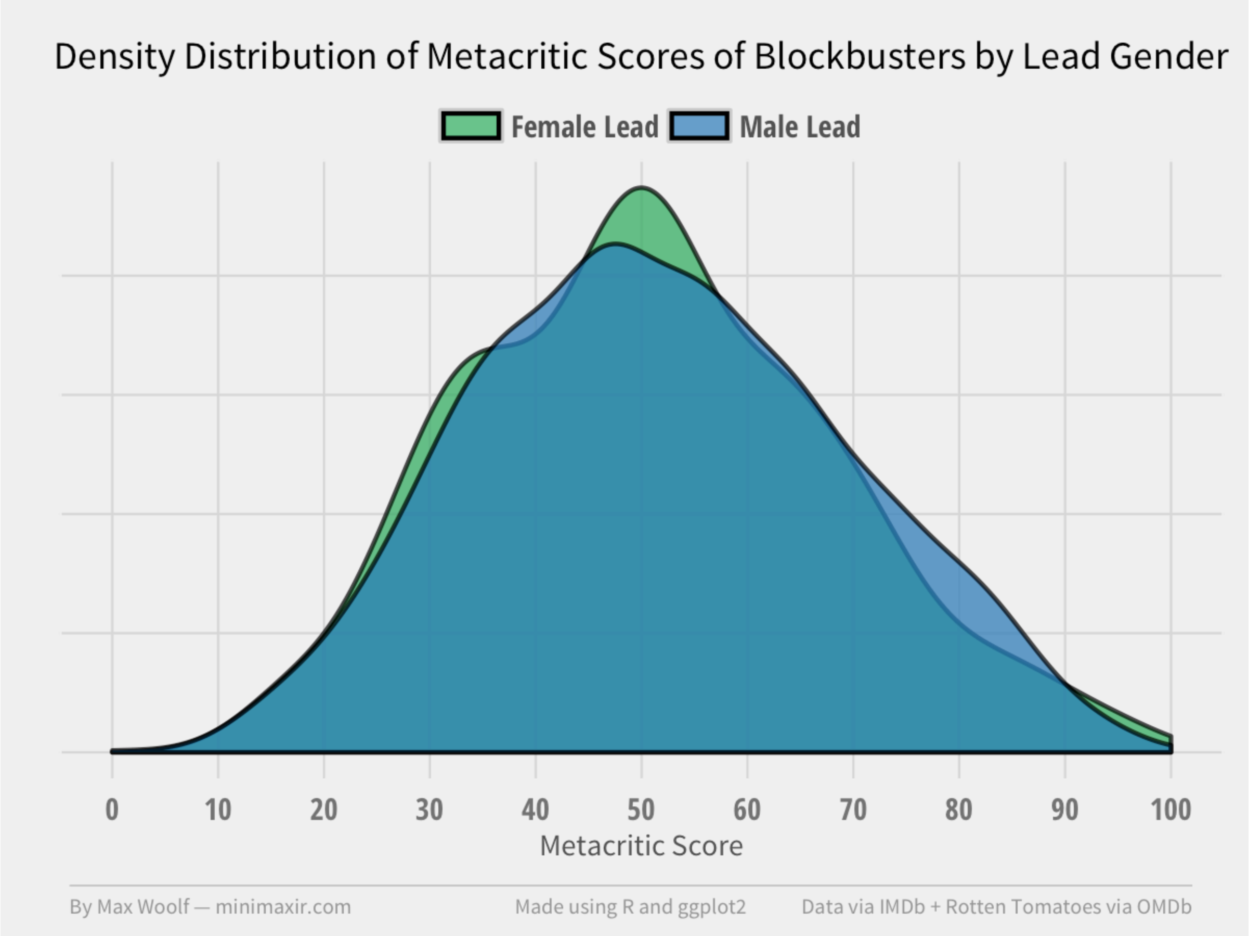

Density Distribution of Metacritic Scores of Blockbusters by Lead Gender
Female Lead
Male Lead
0
10
20
30
40
50
60
70
80
90
100
Metacritic Score
By Max Woolf — minimaxir.com
Made using R and ggplot2
Data via IMDb + Rotten Tomatoes via OMDb

Figure 3:

```
wilcox_test <- wilcox.test(
            unlist(df %>% filter(Gender=="m") %>% select(Metacritic)),
            unlist(df %>% filter(Gender=="f") %>% select(Metacritic)),
            alternative="g")

print(wilcox_test)
```

```
Warning message:
In ks.test(unlist(df %>% filter(Gender == "m") %>% select(Metacritic)), : p-value will be approximate in the presence of ties


    Two-sample Kolmogorov-Smirnov test

data:  unlist(df %>% filter(Gender == "m") %>% select(Metacritic)) and unlist(df %>% filter(Gender == "f") %>% select(Metacritic))
D = 0.046268, p-value = 0.4521
alternative hypothesis: two-sided


    Wilcoxon rank sum test with continuity correction

data:  unlist(df %>% filter(Gender == "m") %>% select(Metacritic)) and unlist(df %>% filter(Gender == "f") %>% select(Metacritic))
W = 347130, p-value = 0.1368
alternative hypothesis: true location shift is greater than 0
```

Bootstramp Resample Means

```
resampleMeans <- function(df) {
    df_new <- df %>% sample_frac(replace=T)

    summary <- df_new %>%
                group_by(Gender) %>%
                summarize(AdjBoxOffice_m = mean(AdjBoxOffice),
                            Meter_m = mean(Meter, na.rm=T),
                            Metacritic_m = mean(Metacritic, na.rm=T))

    return (summary)
}

set.seed(4)
print(resampleMeans(df))
```

```
Source: local data frame [2 x 4]

  Gender AdjBoxOffice_m  Meter_m Metacritic_m
   (chr)          (dbl)    (dbl)        (dbl)
1      f       67986152 45.93206     49.40440
2      m       82138781 49.89780     52.02617
```

Pre-allocate space per this Stack Overflow answer.

```
resampleMovieData <- function(n) {

```

```
df_resample_summary <- data.frame(Gender = character(n*2), AdjBoxOffice_m = numeric(n*2),
                  Meter_m = numeric(n*2), Metacritic_m = numeric(n*2), stringsAsFactors =
                      FALSE)

for (i in seq(1,n*2 - 1, by = 2)) {
    df_resample_summary[c(i,i+1),] <- resampleMeans(df)
}

return(tbl_df(df_resample_summary))

}

set.seed(4)
print(resampleMovieData(4))
```

```
Source: local data frame [8 x 4]

  Gender AdjBoxOffice_m  Meter_m Metacritic_m
   (chr)          (dbl)    (dbl)        (dbl)
1      f       67986152 45.93206     49.40440
2      m       82138781 49.89780     52.02617
3      f       65405109 47.98039     51.02036
4      m       80331357 49.79551     51.76354
5      f       65073479 47.72557     50.11063
6      m       78003310 48.89669     51.36024
7      f       65424463 48.83369     51.77605
8      m       80139428 49.59100     51.92642
```

```
system.time( df_boot <- resampleMovieData(10000))

print(head(df_boot))
print(nrow(df_boot)) # expect 10000 * 2
```

```
   user  system elapsed
 48.692   1.352  52.461



Source: local data frame [6 x 4]

  Gender AdjBoxOffice_m  Meter_m Metacritic_m
   (chr)          (dbl)    (dbl)        (dbl)
1      f       67708627 48.58747     51.07289
2      m       79596707 50.01286     52.30313
3      f       70793578 47.57675     50.28372
4      m       77681742 50.53171     52.40521
5      f       72614163 47.77322     50.12472
6      m       78020424 49.21093     51.50498
[1] 20000
```

```
df_boot_agg <- df_boot %>%
                group_by(Gender) %>%
                summarize(
                    AdjBoxOffice_res_m = mean(AdjBoxOffice_m),
                    AdjBoxOffice_low_ci = quantile(AdjBoxOffice_m, 0.025),
                    AdjBoxOffice_high_ci = quantile(AdjBoxOffice_m, 0.975)
                        )

print(df_boot_agg)
```

```
Source: local data frame [2 x 4]

  Gender AdjBoxOffice_res_m AdjBoxOffice_low_ci AdjBoxOffice_high_ci
   (chr)              (dbl)               (dbl)                (dbl)
1      f           65531567            59238519             72485603
2      m           79758350            75590754             84168300
```

Plot Final Bootstrap

```
df_summary_means <- df %>% group_by(Gender) %>% summarize(mean = mean(AdjBoxOffice))

plot <- ggplot(df_boot, aes(x=AdjBoxOffice_m, fill=Gender)) +
    scale_x_continuous(limits=c(5*10^7, 10^8), breaks=seq(5*10^7,9*10^7, by=10^7),
        labels=paste0("$", seq(50,90, by=10), "M")) +
    scale_y_continuous(breaks=pretty_breaks(4)) +
    geom_histogram(bins=100, alpha=0.75, position="identity") +
    geom_point(mapping=aes(x=mean, y=0), data=df_summary_means, show.legend=F,
        color="black") +
    geom_errorbarh(mapping=aes(x=AdjBoxOffice_res_m, xmin=AdjBoxOffice_low_ci,
        xmax=AdjBoxOffice_high_ci, y=0), data=df_boot_agg, show.legend=F, color="black",
        height=0) +
    fte_theme() +
    theme(legend.title = element_blank(), legend.position="top",
        legend.direction="horizontal", legend.key.width=unit(0.5, "cm"),
        legend.key.height=unit(0.25, "cm"), legend.margin=unit(0,"cm")) +
    scale_fill_manual(labels=c("Female Lead", "Male Lead"), values=c(color_f,color_m)) +
    labs(title=sprintf("Resampled Avg. B.O. Revenues by Movie Lead Gender (n = %2d)",
        nrow(df_boot)/2), x="Average Domestic Box Office Revenue for Blockbusters (2016
        Dollars)", y="# of Resampled Averages")

max_save(plot, "movie-gender-10", "IMDb + Rotten Tomatoes via OMDb")
```

```
Warning message:
: Removed 4 rows containing missing values (geom_bar).
```

Determine P-Value of Final Bootstrap

Calculate the difference between the bootstrapped means; the P-value is the proportion of values where `m - f` < 0.

```
n <- 10000

means_vector <- unlist(df_boot$AdjBoxOffice)
means_diff <- c()

for (i in seq(1,n*2 - 1, by = 2)) {
    means_diff <- c(means_diff, means_vector[i+1] - means_vector[i])
}

```

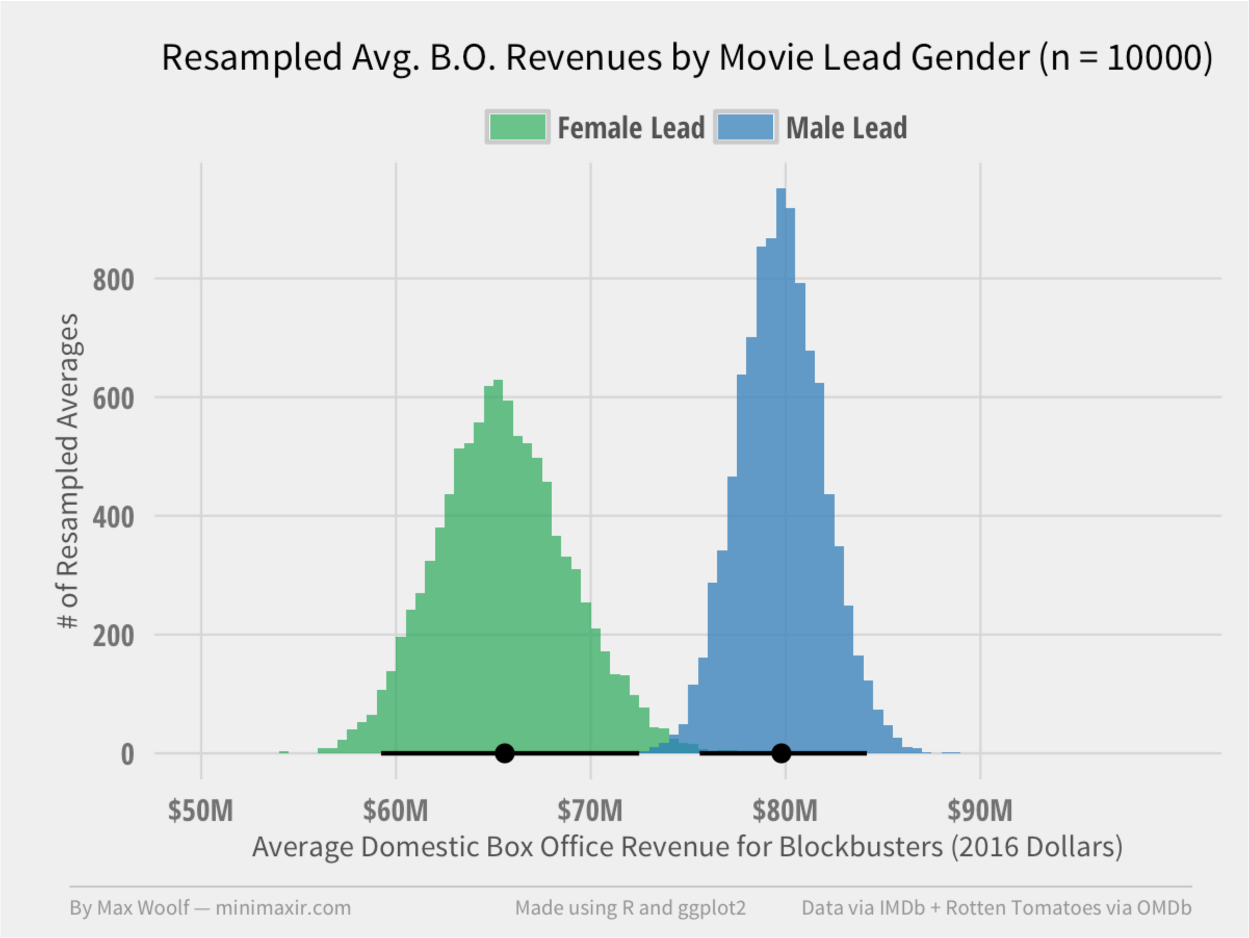

Resampled Avg. B.O. Revenues by Movie Lead Gender (n = 10000)
🟩 Female Lead
🟦 Male Lead
800
600
400
200
0
# of Resampled Averages
$50M
$60M
$70M
$80M
$90M
Average Domestic Box Office Revenue for Blockbusters (2016 Dollars)
By Max Woolf — minimaxir.com
Made using R and ggplot2
Data via IMDb + Rotten Tomatoes via OMDb

Figure 4:

```
print(means_diff[1:4])

print(sum(means_diff <= 0)/n)   # p-value of difference
```

```
[1] 11888080  6888164  5406261  8136011
[1] 2e-04
```

Bootstrap Movie!

Render each frame of the resample; composite into GIF later.

```
system("mkdir -p movie_frames")

movie_frames <- function(size) {
    df_boot_sub <- df_boot %>% head(size*2)

    df_boot_agg_sub <- df_boot_sub %>%
                group_by(Gender) %>%
                summarize(
                    AdjBoxOffice_res_m = mean(AdjBoxOffice_m),
                    AdjBoxOffice_low_ci = quantile(AdjBoxOffice_m, 0.025),
                    AdjBoxOffice_high_ci = quantile(AdjBoxOffice_m, 0.975)
                        )

    plot <- ggplot(df_boot_sub, aes(x=AdjBoxOffice_m, fill=Gender)) +
    scale_x_continuous(limits=c(5*10^7, 10^8), breaks=seq(5*10^7,9*10^7, by=10^7),
        labels=paste0("$", seq(50,90, by=10), "M")) +
    scale_y_continuous(breaks=pretty_breaks(4)) +
    geom_histogram(bins=100, alpha=0.75, position="identity") +
    geom_point(mapping=aes(x=mean, y=0), data=df_summary_means, show.legend=F,
        color="black") +
    geom_errorbarh(mapping=aes(x=AdjBoxOffice_res_m, xmin=AdjBoxOffice_low_ci,
        xmax=AdjBoxOffice_high_ci, y=0), data=df_boot_agg_sub, show.legend=F, color="black",
        height=0) +
    fte_theme() +
    theme(legend.title = element_blank(), legend.position="top",
        legend.direction="horizontal", legend.key.width=unit(0.5, "cm"),
        legend.key.height=unit(0.25, "cm"), legend.margin=unit(0,"cm")) +
    scale_fill_manual(labels=c("Female Lead", "Male Lead"), values=c(color_f,color_m)) +
    labs(title=sprintf("Resampled Avg. B.O. Revenues by Movie Lead Gender (n = %2d)", size),
        x="Average Domestic Box Office Revenue for Blockbusters (2016 Dollars)", y="# of
        Resampled Averages")

max_save(plot, sprintf("movie_frames/movie_%06d", size), "IMDb + Rotten Tomatoes via OMDb")

}
```

```
system.time( x <- lapply(seq(100,10000,100), movie_frames) )
```

```
Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
```

```
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
```

```
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).Warning message:
: Removed 4 rows containing missing values (geom_bar).




   user  system elapsed
 54.123   2.746  59.324
```

The MIT License (MIT)