DANA4810-Project

2025-11-27

```r
# Load the dataset
movies <- read.csv('Movie.csv')
movies <- head(movies, -1)
options(scipen = 999)
# Display first few rows
head(movies)
```

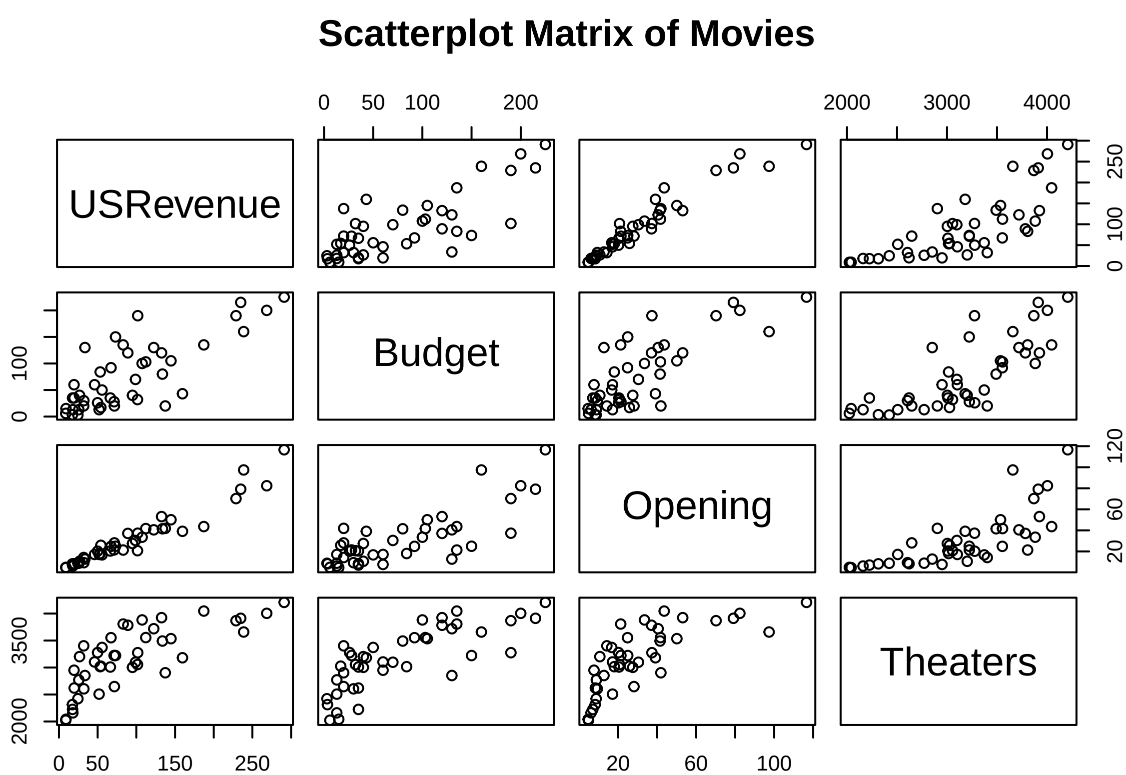
```
##                        Title USRelease           Genre Rating Sequel Budget
## 1               Man of Steel    16-Jun Action/Adventure  PG-13      0    225
## 2         Monster University    23-Jun        Animation      G      0    200
## 3           Fast & Furious 6    26-May Action/Adventure  PG-13      1    160
## 4 Oz the Great and Powerful    10-Mar Action/Adventure     PG      0    215
## 5  Star Trek: Into Darkness    19-May Action/Adventure  PG-13      1    190
## 6                 The Croods    24-Mar        Animation     PG      0    135
##   Opening USRevenue Theaters IntRevenue WorldRevenue Ratings Review Minutes
## 1   116.6     291.0     4207      377.0        668.0     7.1     55     143
## 2    82.4     268.5     4004      475.1        743.6     7.3     65     104
## 3    97.4     238.7     3658      550.0        788.7     7.1     61     130
## 4    79.1     234.9     3912      258.4        493.3     6.3     44     130
## 5    70.2     228.8     3868      238.6        467.4     7.7     72     132
## 6    43.6     187.2     4046      400.0        587.2     7.2     55      98
```

```r
# Display summary statistics
summary(movies)
```

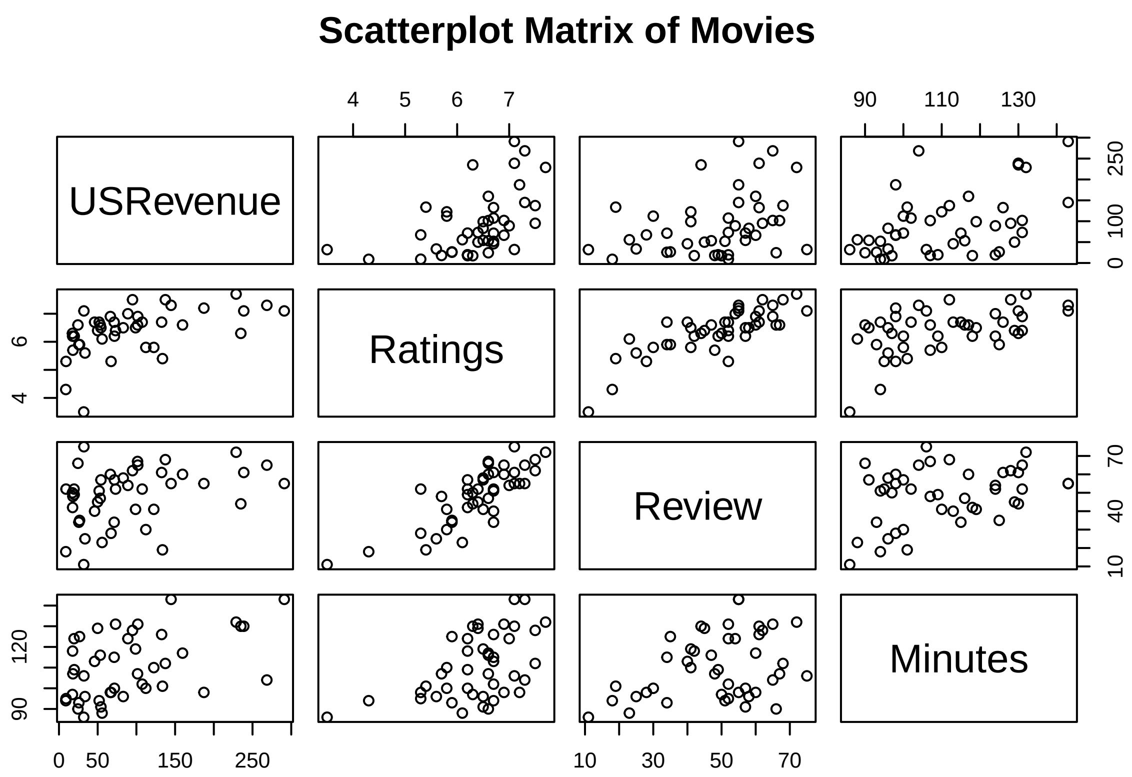
```
##     Title            USRelease            Genre              Rating
##  Length:44          Length:44          Length:44          Length:44
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      Sequel           Budget          Opening         USRevenue
##  Min.   :0.0000   Min.   :  3.00   Min.   :  4.60   Min.   :  8.80
##  1st Qu.:0.0000   1st Qu.: 24.50   1st Qu.: 12.18   1st Qu.: 32.15
##  Median :0.0000   Median : 55.00   Median : 23.10   Median : 71.45
##  Mean   :0.2045   Mean   : 77.08   Mean   : 30.69   Mean   : 91.23
##  3rd Qu.:0.0000   3rd Qu.:122.50   3rd Qu.: 40.75   3rd Qu.:125.03
##  Max.   :1.0000   Max.   :225.00   Max.   :116.60   Max.   :291.00
##     Theaters      IntRevenue      WorldRevenue       Ratings
##  Min.   :2023   Min.   :  0.20   Min.   :  9.3   Min.   :3.500
##  1st Qu.:2832   1st Qu.: 29.75   1st Qu.: 67.3   1st Qu.:6.050
##  Median :3192   Median : 66.20   Median :147.2   Median :6.500
```

```
##  Mean   :3169   Mean   :133.13   Mean   :224.4   Mean   :6.382
##  3rd Qu.:3581   3rd Qu.:214.00   3rd Qu.:326.2   3rd Qu.:6.900
##  Max.   :4207   Max.   :550.00   Max.   :788.7   Max.   :7.700
##      Review          Minutes
##  Min.   :11.00   Min.   : 86.00
##  1st Qu.:40.75   1st Qu.: 97.75
##  Median :52.00   Median :108.00
##  Mean   :48.55   Mean   :110.59
##  3rd Qu.:60.00   3rd Qu.:124.25
##  Max.   :75.00   Max.   :143.00
```

```
pairs(~USRevenue+Budget+Opening+Theaters,data = movies,
      main="Scatterplot Matrix of Movies")
```



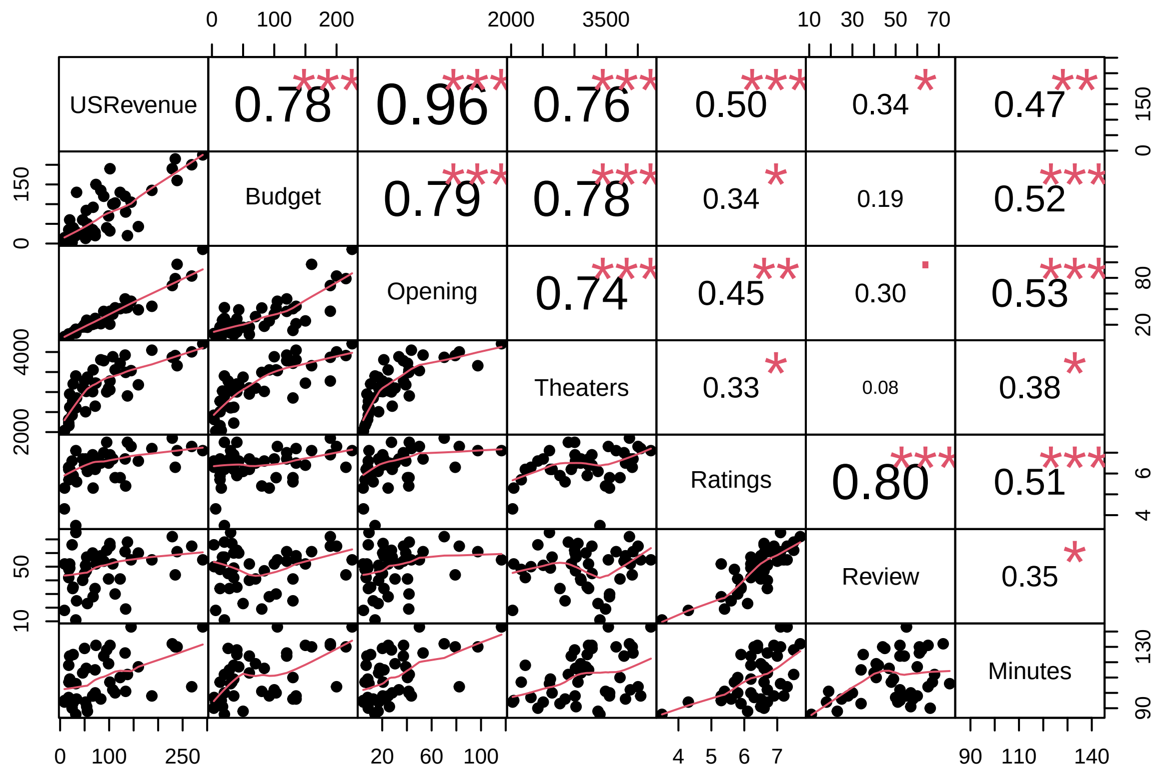

```
pairs(~USRevenue+Ratings+Review+Minutes,data = movies,
      main="Scatterplot Matrix of Movies")
```




```r
#pairs(~USRevenue+Budget+Opening+Theaters+Ratings+Review+Minutes,data = movies,
     # main="Scatterplot Matrix of Movies Complete") #to see multicollinearity
library(PerformanceAnalytics)
```

```
## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## Attaching package: 'PerformanceAnalytics'

## The following object is masked from 'package:graphics':
##
##     legend
```

```r
chart.Correlation(movies[,c("USRevenue","Budget","Opening","Theaters", "Ratings", "Review", "Minutes")]
```



#Checking for trends of categorical variables vs USRevenue

```r
aggregate(USRevenue ~ Sequel, data = movies, mean)
```

```
##   Sequel USRevenue
## 1      0  82.65429
## 2      1 124.56667
```

```r
aggregate(USRevenue ~ Genre, data = movies, mean)
```

```
##               Genre USRevenue
## 1 Action/Adventure 108.96111
## 2        Animation 161.55000
## 3           Comedy  62.86364
## 4      Crime/Drama  42.05000
## 5            Drama  70.06000
## 6           Horror  70.15000
```

```r
aggregate(USRevenue ~ Rating, data = movies, mean)
```

```
##   Rating USRevenue
## 1      G 268.50000
```

```
## 2     PG 153.15000
## 3  PG-13  94.08182
## 4      R  62.53529
```

#Side by side box plot

```r
library(dplyr)
```

```
##
## ######################### Warning from 'xts' package ##########################
## #                                                                             #
## # The dplyr lag() function breaks how base R's lag() function is supposed to  #
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or       #
## # source() into this session won't work correctly.                            #
## #                                                                             #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop           #
## # dplyr from breaking base R's lag() function.                                #
## #                                                                             #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                             #
## ###############################################################################
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:xts':
##
##     first, last
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

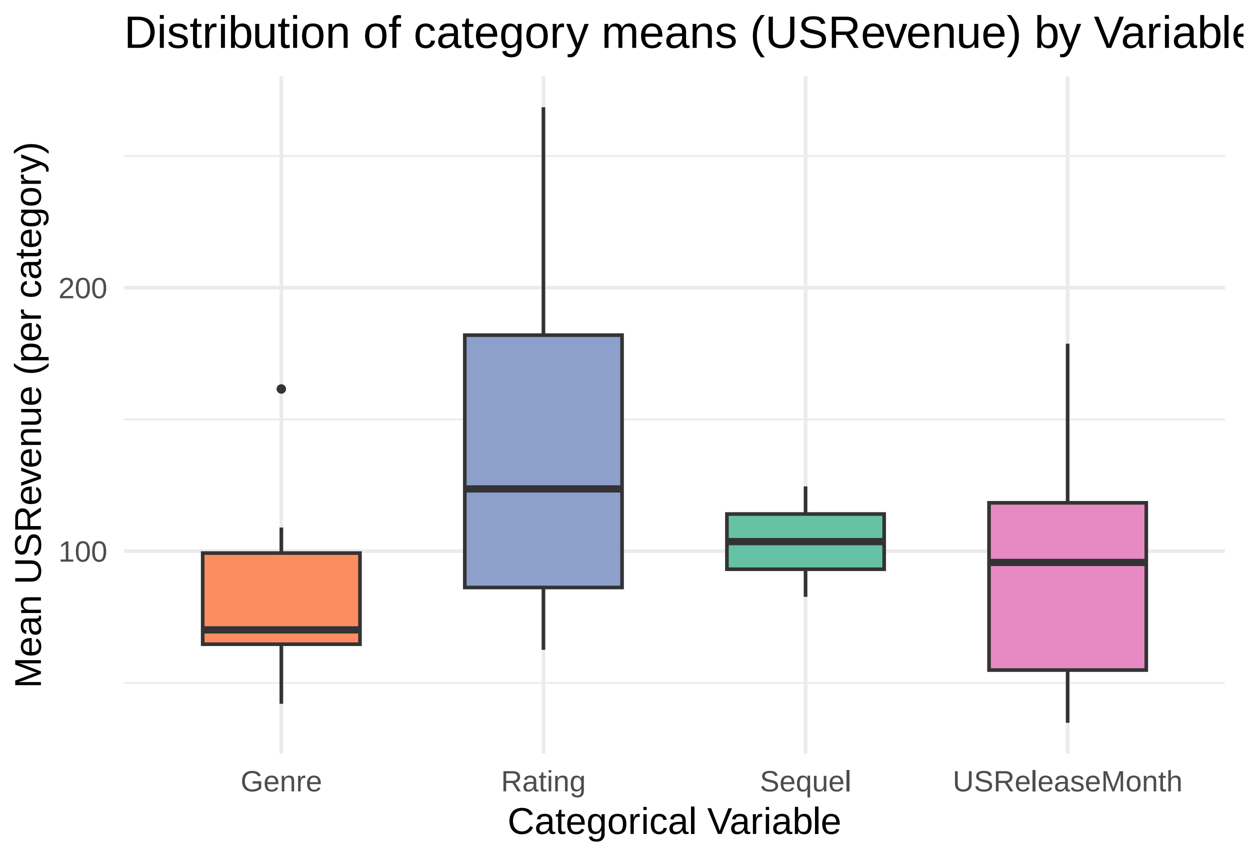
```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr)
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.5.1
```

```r
cat_vars <- c("Sequel","Genre","Rating","USReleaseMonth")
movies$USReleaseMonth <- sub("^[0-9]+-", "", movies$USRelease)

# For each variable: compute mean USRevenue for each category, then combine
mean_by_cat <- bind_rows(
  lapply(cat_vars, function(var) {
    movies %>%
```

```r
      filter(!is.na(.data[[var]])) %>%
      group_by(Category = .data[[var]]) %>%
      summarise(
        mean_USRevenue = mean(USRevenue, na.rm = TRUE),
        n = n(),
        .groups = "drop"
      ) %>%
      mutate(
        Variable = var,
        Category = as.character(Category)   # Convert to character
      )
  })
)

# Now plot boxplot of the distribution of category means for each Variable
ggplot(mean_by_cat, aes(x = Variable, y = mean_USRevenue, fill = Variable)) +
  geom_boxplot(width = 0.6, outlier.size = 1) +
  labs(title = "Distribution of category means (USRevenue) by Variable",
       x = "Categorical Variable", y = "Mean USRevenue (per category)") +
  theme_minimal(base_size = 14) +
  theme(legend.position = "none") +
  scale_fill_manual(values = c("Sequel"="#66C2A5","Genre"="#FC8D62","Rating"="#8DA0CB","USReleaseMonth"=
```



Interpretation: We can see there is a significant difference for rated PG-13 and R in regards to USRevenue, where those categories are associated with lower USRevenue. We can consider a dummy variable that accounts for this. Same with a movie being a Sequel (=1), which is associated with higher USRevenue.

```r
names(movies)
```

```
##  [1] "Title"          "USRelease"      "Genre"          "Rating"
##  [5] "Sequel"         "Budget"         "Opening"        "USRevenue"
##  [9] "Theaters"       "IntRevenue"     "WorldRevenue"   "Ratings"
## [13] "Review"         "Minutes"        "USReleaseMonth"
```

```r
Genre   <- factor(movies$Genre)
Rating <- factor(movies$Rating)
s <- ifelse(movies$Sequel == 1, 1, 0)

model55 = lm(USRevenue~Budget+Opening+Theaters+Ratings+Minutes+s+factor(Genre)+factor(Rating), data=movi
summary(model55)
```

```
##
## Call:
## lm(formula = USRevenue ~ Budget + Opening + Theaters + Ratings +
##     Minutes + s + factor(Genre) + factor(Rating), data = movies)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -22.02 -10.18   0.00   7.71  32.49
##
## Coefficients:
##                             Estimate Std. Error t value       Pr(>|t|)
## (Intercept)               -72.381544  49.022387  -1.476         0.1506
## Budget                      0.044302   0.103259   0.429         0.6711
## Opening                     2.278943   0.243749   9.350 0.000000000297 ***
## Theaters                    0.009989   0.009299   1.074         0.2916
## Ratings                    12.368468   4.938513   2.504         0.0181 *
## Minutes                     0.144467   0.334930   0.431         0.6694
## s                           7.143482   8.692724   0.822         0.4179
## factor(Genre)Animation     -1.074878  22.923535  -0.047         0.9629
## factor(Genre)Comedy        23.562711   9.731424   2.421         0.0220 *
## factor(Genre)Crime/Drama    3.473057  15.219870   0.228         0.8211
## factor(Genre)Drama          5.808265   9.937879   0.584         0.5634
## factor(Genre)Horror         7.606842  14.187448   0.536         0.5959
## factor(Rating)PG          -18.287552  20.861010  -0.877         0.3879
## factor(Rating)PG-13       -51.755002  30.174469  -1.715         0.0970 .
## factor(Rating)R           -45.397571  31.190685  -1.455         0.1563
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.31 on 29 degrees of freedom
## Multiple R-squared:  0.9668, Adjusted R-squared:  0.9507
## F-statistic: 60.26 on 14 and 29 DF,  p-value: < 0.00000000000000022
```

Interpretation: This is a full model. Opening, Ratings (numerical, not pg/r ect), and Comedy are significant. #Stepwise Regression

```r
step(lm(USRevenue ~ Genre + Rating + Sequel + Budget + Opening + Theaters + Ratings + Review + Minutes,
```

```
## Start:  AIC=258.31
## USRevenue ~ Genre + Rating + Sequel + Budget + Opening + Theaters +
##     Ratings + Review + Minutes
##
##            Df Sum of Sq     RSS    AIC
## - Budget    1      93.7  7630.3 256.85
## - Minutes   1     125.3  7661.9 257.03
## - Review    1     174.8  7711.4 257.32
## - Theaters  1     176.9  7713.5 257.33
## - Sequel    1     241.7  7778.3 257.70
## <none>                   7536.6 258.31
## - Rating    3    1182.1  8718.7 258.72
## - Genre     5    2290.3  9826.9 259.98
## - Ratings   1    1361.7  8898.3 263.62
## - Opening   1   21675.5 29212.1 315.92
##
## Step:  AIC=256.85
## USRevenue ~ Genre + Rating + Sequel + Opening + Theaters + Ratings +
##     Review + Minutes
##
##            Df Sum of Sq   RSS    AIC
## - Minutes   1       109  7739 255.47
## - Review    1       130  7760 255.59
## - Theaters  1       228  7859 256.15
## - Sequel    1       229  7859 256.15
## <none>                   7630 256.85
## - Rating    3      1220  8850 257.38
## - Genre     5      2265  9896 258.29
## + Budget    1        94  7537 258.31
## - Ratings   1      1284  8915 261.70
## - Opening   1     32283 39913 327.65
##
## Step:  AIC=255.47
## USRevenue ~ Genre + Rating + Sequel + Opening + Theaters + Ratings +
##     Review
##
##            Df Sum of Sq   RSS    AIC
## - Review    1        70  7809 253.87
## - Sequel    1       163  7902 254.39
## - Theaters  1       234  7973 254.78
## <none>                   7739 255.47
## - Rating    3      1131  8870 255.48
## - Genre     5      2180  9919 256.39
## + Minutes   1       109  7630 256.85
## + Budget    1        77  7662 257.03
## - Ratings   1      1272  9011 260.17
## - Opening   1     41034 48773 334.47
##
## Step:  AIC=253.87
## USRevenue ~ Genre + Rating + Sequel + Opening + Theaters + Ratings
##
```

```
##            Df Sum of Sq   RSS    AIC
## - Sequel    1       146  7955 252.68
## - Theaters  1       323  8132 253.65
## <none>                   7809 253.87
## - Rating    3      1155  8964 253.94
## - Genre     5      2117  9926 254.42
## + Review    1        70  7739 255.47
## + Minutes   1        49  7760 255.59
## + Budget    1        48  7761 255.60
## - Ratings   1      2206 10014 262.81
## - Opening   1     40980 48789 332.49
##
## Step:  AIC=252.68
## USRevenue ~ Genre + Rating + Opening + Theaters + Ratings
##
##            Df Sum of Sq   RSS    AIC
## - Rating    3      1009  8964 251.94
## <none>                   7955 252.68
## - Genre     5      2158 10113 253.25
## - Theaters  1       592  8546 253.84
## + Sequel    1       146  7809 253.87
## + Review    1        53  7902 254.39
## + Budget    1        47  7907 254.42
## + Minutes   1        16  7939 254.59
## - Ratings   1      2103 10057 261.00
## - Opening   1     42277 50231 331.77
##
## Step:  AIC=251.94
## USRevenue ~ Genre + Opening + Theaters + Ratings
##
##            Df Sum of Sq   RSS    AIC
## <none>                   8964 251.94
## - Theaters  1       419  9383 251.95
## + Rating    3      1009  7955 252.68
## + Review    1        91  8873 253.49
## + Budget    1        69  8895 253.60
## + Sequel    1         0  8964 253.94
## + Minutes   1         0  8964 253.94
## - Ratings   1      1841 10805 258.16
## - Genre     5      4304 13268 259.19
## - Opening   1     59978 68942 339.70

##
## Call:
## lm(formula = USRevenue ~ Genre + Opening + Theaters + Ratings,
##     data = movies)
##
## Coefficients:
##      (Intercept)    GenreAnimation       GenreComedy  GenreCrime/Drama
##        -94.64681          28.27042          19.27345           3.07602
##       GenreDrama       GenreHorror           Opening          Theaters
##          2.72053           0.50277           2.49380           0.01033
##          Ratings
##         10.76940
```

After did stepwise regression, we get the model with Genre, Opening, Theaters and Ratings as predictors.

##We will start doing models with subsets of the variables to see which ones remain significant, to eventually close in on a final model.

```
model54 = lm(USRevenue~s+factor(Genre)+factor(Rating), data=movies)
summary(model54)
```

```
##
## Call:
## lm(formula = USRevenue ~ s + factor(Genre) + factor(Rating),
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -85.347 -38.365  -8.965  21.229 201.095
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)               377.50      97.20   3.884 0.000451 ***
## s                          48.74      26.66   1.829 0.076253 .
## factor(Genre)Animation   -109.00      73.48  -1.483 0.147161
## factor(Genre)Comedy       -31.75      25.01  -1.270 0.212811
## factor(Genre)Crime/Drama  -32.12      49.98  -0.643 0.524825
## factor(Genre)Drama        -16.70      34.12  -0.489 0.627779
## factor(Genre)Horror       -11.88      36.53  -0.325 0.746931
## factor(Rating)PG         -142.60      73.48  -1.941 0.060615 .
## factor(Rating)PG-13      -287.60      99.37  -2.894 0.006596 **
## factor(Rating)R          -303.33      99.61  -3.045 0.004467 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 63.63 on 34 degrees of freedom
## Multiple R-squared:  0.4067, Adjusted R-squared:  0.2497
## F-statistic:  2.59 on 9 and 34 DF,  p-value: 0.0216
```

```
model53 = lm(USRevenue~factor(Rating), data=movies)
summary(model53)
```

```
##
## Call:
## lm(formula = USRevenue ~ factor(Rating), data = movies)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -84.98 -43.36 -11.63  36.90 196.92
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          268.50      65.40   4.105 0.000193 ***
## factor(Rating)PG    -115.35      73.12  -1.577 0.122563
## factor(Rating)PG-13 -174.42      66.87  -2.608 0.012736 *
```

```
## factor(Rating)R      -205.96      67.30  -3.060 0.003937 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65.4 on 40 degrees of freedom
## Multiple R-squared:  0.2626, Adjusted R-squared:  0.2073
## F-statistic: 4.749 on 3 and 40 DF,  p-value: 0.006326
```

```
model52 = lm(USRevenue~s, data=movies)
summary(model52)
```

```
##
## Call:
## lm(formula = USRevenue ~ s, data = movies)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -92.57 -57.03 -11.86  18.92 208.35
##
## Coefficients:
##             Estimate Std. Error t value     Pr(>|t|)
## (Intercept)    82.65      12.22   6.765 0.0000000316 ***
## s              41.91      27.02   1.551        0.128
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 72.29 on 42 degrees of freedom
## Multiple R-squared:  0.0542, Adjusted R-squared:  0.03168
## F-statistic: 2.407 on 1 and 42 DF,  p-value: 0.1283
```

```
model51 = lm(USRevenue~factor(Genre), data=movies)
summary(model51)
```

```
##
## Call:
## lm(formula = USRevenue ~ factor(Genre), data = movies)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -91.36 -52.88 -13.01  28.90 182.04
##
## Coefficients:
##                          Estimate Std. Error t value     Pr(>|t|)
## (Intercept)                108.96      16.62   6.557 0.0000000984 ***
## factor(Genre)Animation      52.59      38.97   1.349       0.1852
## factor(Genre)Comedy        -46.10      26.98  -1.708       0.0957 .
## factor(Genre)Crime/Drama   -66.91      52.55  -1.273       0.2107
## factor(Genre)Drama         -38.90      35.64  -1.091       0.2820
## factor(Genre)Horror        -38.81      38.97  -0.996       0.3256
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 70.5 on 38 degrees of freedom
```

```
## Multiple R-squared:  0.1859, Adjusted R-squared:  0.07883
## F-statistic: 1.736 on 5 and 38 DF,  p-value: 0.15
```

```r
model50 = lm(USRevenue~Opening+I(Ratings^2)+factor(Rating), data=movies)
summary(model50)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + I(Ratings^2) + factor(Rating),
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.199  -9.801  -1.351   4.650  46.316
##
## Coefficients:
##                      Estimate Std. Error t value            Pr(>|t|)
## (Intercept)           23.4040    21.9797   1.065              0.2937
## Opening                2.5597     0.1220  20.983 <0.0000000000000002 ***
## I(Ratings^2)           0.6413     0.2987   2.147              0.0382 *
## factor(Rating)PG     -12.4867    18.8030  -0.664              0.5106
## factor(Rating)PG-13 -42.7897    17.6107  -2.430              0.0199 *
## factor(Rating)R     -38.1403    18.2077  -2.095              0.0429 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.39 on 38 degrees of freedom
## Multiple R-squared:  0.956,  Adjusted R-squared:  0.9502
## F-statistic: 165.1 on 5 and 38 DF,  p-value: < 0.00000000000000022
```

```r
model49 = lm(USRevenue~Opening+Ratings+factor(Rating), data=movies)
summary(model49)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + Ratings + factor(Rating),
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.325 -10.006  -2.222   4.518  46.138
##
## Coefficients:
##                      Estimate Std. Error t value            Pr(>|t|)
## (Intercept)            6.0260    28.5790   0.211              0.8341
## Opening                2.5750     0.1217  21.160 <0.0000000000000002 ***
## Ratings                6.8894     3.5088   1.963              0.0569 .
## factor(Rating)PG     -13.1290    18.9658  -0.692              0.4930
## factor(Rating)PG-13 -43.2854    17.7600  -2.437              0.0196 *
## factor(Rating)R     -38.7172    18.3646  -2.108              0.0417 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 16.54 on 38 degrees of freedom
## Multiple R-squared:  0.9552, Adjusted R-squared:  0.9493
## F-statistic: 162.1 on 5 and 38 DF,  p-value: < 0.00000000000000022
```

```
#model50 better with Ratings^2
```

#Now add some variables back in: We will add comedy because in the original model55 with most of the variables, the factor(Genre)Comedy was the only significant one

```
c <- ifelse(movies$Genre == "Comedy", 1, 0)
model48 = lm(USRevenue~Opening+I(Ratings^2)+factor(Rating)+c, data=movies)
summary(model48)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + I(Ratings^2) + factor(Rating) +
##     c, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.218  -9.225  -1.846   8.025  34.202
##
## Coefficients:
##                     Estimate Std. Error t value             Pr(>|t|)
## (Intercept)           4.3135    21.4592   0.201              0.84179
## Opening               2.5553     0.1127  22.675 < 0.0000000000000002 ***
## I(Ratings^2)          1.0064     0.3062   3.286              0.00223 **
## factor(Rating)PG     -9.5078    17.4015  -0.546              0.58809
## factor(Rating)PG-13 -42.4343    16.2669  -2.609              0.01304 *
## factor(Rating)R     -39.5664    16.8258  -2.352              0.02413 *
## c                    16.9006     6.1546   2.746              0.00926 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.14 on 37 degrees of freedom
## Multiple R-squared:  0.9634, Adjusted R-squared:  0.9575
## F-statistic: 162.5 on 6 and 37 DF,  p-value: < 0.00000000000000022
```

#What if we change the Genre categories to be just Animation or Other, since Animation is associated with highest USRevenue of all genres.

```
movies$a <- ifelse(movies$Genre == "Animation", 1, 0)
model47 = lm(USRevenue~Opening+I(Ratings^2)+factor(Rating)+a, data=movies)
summary(model47)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + I(Ratings^2) + factor(Rating) +
##     a, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -24.227 -10.255  -1.130   4.556  46.140
##
## Coefficients:
##                      Estimate Std. Error t value            Pr(>|t|)
## (Intercept)           17.4672    30.0520   0.581              0.5646
## Opening                2.5731     0.1316  19.552 <0.0000000000000002 ***
## I(Ratings^2)           0.6189     0.3118   1.985              0.0546 .
## factor(Rating)PG     -10.6644    20.0179  -0.533              0.5974
## factor(Rating)PG-13 -36.3858    28.1541  -1.292              0.2042
## factor(Rating)R     -31.5757    28.9600  -1.090              0.2826
## a                      6.0272    20.5094   0.294              0.7705
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.59 on 37 degrees of freedom
## Multiple R-squared:  0.9561, Adjusted R-squared:  0.949
## F-statistic: 134.3 on 6 and 37 DF,  p-value: < 0.00000000000000022
```

#Animation not significant, so go back to comedy and add other variables

```r
model46 = lm(USRevenue~Budget+Opening+I(Ratings^2)+factor(Rating)+c, data=movies)
summary(model46)
```

```
##
## Call:
## lm(formula = USRevenue ~ Budget + Opening + I(Ratings^2) + factor(Rating) +
##     c, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.677  -9.247  -1.583   8.485  34.489
##
## Coefficients:
##                      Estimate Std. Error t value             Pr(>|t|)
## (Intercept)          -1.92386   22.78278  -0.084              0.93317
## Budget                0.06029    0.07162   0.842              0.40547
## Opening               2.45139    0.16742  14.642 < 0.0000000000000002 ***
## I(Ratings^2)          1.05781    0.31346   3.375              0.00178 **
## factor(Rating)PG     -9.77411   17.47330  -0.559              0.57937
## factor(Rating)PG-13 -40.57461   16.48009  -2.462              0.01874 *
## factor(Rating)R     -36.25587   17.34422  -2.090              0.04371 *
## c                    18.77029    6.56608   2.859              0.00703 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.2 on 36 degrees of freedom
## Multiple R-squared:  0.9642, Adjusted R-squared:  0.9572
## F-statistic: 138.3 on 7 and 36 DF,  p-value: < 0.00000000000000022
```

```r
model45 = lm(USRevenue~Theaters+Opening+I(Ratings^2)+factor(Rating)+c, data=movies)
summary(model45)
```

```
##
```

```
## Call:
## lm(formula = USRevenue ~ Theaters + Opening + I(Ratings^2) +
##     factor(Rating) + c, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -23.140 -11.355   0.023   6.034  32.583
##
## Coefficients:
##                        Estimate Std. Error t value             Pr(>|t|)
## (Intercept)          -31.042913  28.965907  -1.072              0.29098
## Theaters               0.011663   0.006624   1.761              0.08679 .
## Opening                2.379704   0.148194  16.058 < 0.0000000000000002 ***
## I(Ratings^2)           1.065083   0.299761   3.553              0.00109 **
## factor(Rating)PG     -14.599292  17.172997  -0.850              0.40087
## factor(Rating)PG-13 -41.412445  15.834702  -2.615              0.01294 *
## factor(Rating)R      -37.641987  16.404195  -2.295              0.02769 *
## c                     19.161577   6.123236   3.129              0.00347 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.73 on 36 degrees of freedom
## Multiple R-squared:  0.9663, Adjusted R-squared:  0.9598
## F-statistic: 147.7 on 7 and 36 DF,  p-value: < 0.00000000000000022
```

#Dummy variable for high rating, which is PG-13 or R, since those two are assoicated with lower USRevenue.

```
movies$highrate <- ifelse(movies$Rating %in% c("PG-13", "R"),1, 0)
model44 = lm(USRevenue~Budget+Opening+I(Ratings^2)+highrate+c, data=movies)
summary(model44)
```

```
##
## Call:
## lm(formula = USRevenue ~ Budget + Opening + I(Ratings^2) + highrate +
##     c, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.297 -10.918  -1.587   8.301  35.997
##
## Coefficients:
##                Estimate Std. Error t value             Pr(>|t|)
## (Intercept)    -8.83599   16.14083  -0.547             0.587284
## Budget          0.04096    0.06698   0.611             0.544566
## Opening         2.47261    0.16327  15.145 < 0.0000000000000002 ***
## I(Ratings^2)    1.07992    0.30843   3.501             0.001200 **
## highrate      -32.04749    8.04167  -3.985             0.000295 ***
## c              19.00469    6.46837   2.938             0.005587 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15 on 38 degrees of freedom
## Multiple R-squared:  0.9632, Adjusted R-squared:  0.9583
## F-statistic: 198.8 on 5 and 38 DF,  p-value: < 0.00000000000000022
```

#Start trying models with interaction

```
model43= lm(USRevenue~Budget+Theaters+Opening+I(Ratings^2)+highrate+c+Theaters*Budget, data=movies)
summary(model43)
```

```
##
## Call:
## lm(formula = USRevenue ~ Budget + Theaters + Opening + I(Ratings^2) +
##     highrate + c + Theaters * Budget, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -25.1507  -9.3279  -0.2116   8.1713  30.3583
##
## Coefficients:
##                   Estimate  Std. Error t value          Pr(>|t|)
## (Intercept)     -41.9114481  26.4586588  -1.584           0.12193
## Budget            0.4146921   0.3525072   1.176           0.24715
## Theaters          0.0120278   0.0070888   1.697           0.09838 .
## Opening           2.5550909   0.2249186  11.360 0.000000000000185 ***
## I(Ratings^2)      1.0594373   0.3058347   3.464           0.00139 **
## highrate        -34.6879863   9.6544966  -3.593           0.00097 ***
## c                20.5622829   6.4028718   3.211           0.00278 **
## Budget:Theaters  -0.0001233   0.0001050  -1.175           0.24775
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.71 on 36 degrees of freedom
## Multiple R-squared:  0.9664, Adjusted R-squared:  0.9599
## F-statistic:   148 on 7 and 36 DF,  p-value: < 0.00000000000000022
```

#Try Quadratic and interaction of Theaters.

```
model42= lm(USRevenue~I(Theaters^2)+Opening+I(Ratings^2)+highrate+c+I(Theaters^2)*Opening, data=movies)
summary(model42)
```

```
##
## Call:
## lm(formula = USRevenue ~ I(Theaters^2) + Opening + I(Ratings^2) +
##     highrate + c + I(Theaters^2) * Opening, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.3630  -8.3427   0.4639   7.0032  29.0647
##
## Coefficients:
##                             Estimate      Std. Error t value     Pr(>|t|)
## (Intercept)           -25.11889288237  19.27089153088  -1.303     0.200469
## I(Theaters^2)           0.00000202346   0.00000111162   1.820     0.076813 .
## Opening                 3.25898428876   0.49073132799   6.641 0.0000000857 ***
## I(Ratings^2)            0.96795271794   0.29968972541   3.230     0.002600 **
## highrate              -32.09017767119   8.24125638186  -3.894     0.000398 ***
## c                      19.23963494300   5.88507992669   3.269     0.002335 **
```

```
## I(Theaters^2):Opening  -0.00000005449   0.00000002953  -1.845     0.072995 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.24 on 37 degrees of freedom
## Multiple R-squared:  0.9676, Adjusted R-squared:  0.9624
## F-statistic: 184.4 on 6 and 37 DF,  p-value: < 0.00000000000000022
```

Interpretation of model42: This is the highest Adjusted R-squared yet at .9624, which is about .0034 higher than the model below. Should we go for simplicity or slightly higher adjusted R-squared?

#We found that Theaters is not significant

```r
model41 = lm(USRevenue~Opening+Ratings+highrate+c, data=movies)
summary(model41)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + Ratings + highrate + c, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.025 -11.058  -2.270   8.354  34.902
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept) -36.4260    23.6198  -1.542              0.13110
## Opening       2.5686     0.1044  24.611 < 0.0000000000000002 ***
## Ratings      11.6367     3.5400   3.287              0.00215 **
## highrate    -33.6526     7.5825  -4.438            0.0000723 ***
## c            17.5416     6.1068   2.872              0.00656 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.07 on 39 degrees of freedom
## Multiple R-squared:  0.9618, Adjusted R-squared:  0.9579
## F-statistic: 245.7 on 4 and 39 DF,  p-value: < 0.00000000000000022
```

#We curious that USRelease might affect to predict USRevenue, so we add USRelease variable to the model(only month not include date.

```r
movies$USReleaseMonth <- sub("^[0-9]+-", "", movies$USRelease)
model40 = lm(USRevenue~Opening+Ratings+highrate+c+USReleaseMonth, data=movies)
summary(model40)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + Ratings + highrate + c + USReleaseMonth,
##     data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -28.327  -9.275  -2.330  10.281  35.772
```

```
##
## Coefficients:
##                    Estimate Std. Error t value             Pr(>|t|)
## (Intercept)        -36.4097    25.1923  -1.445             0.157812
## Opening              2.4891     0.1259  19.777 < 0.0000000000000002 ***
## Ratings             10.6693     3.6600   2.915             0.006344 **
## highrate           -30.3980     7.8721  -3.861             0.000498 ***
## c                   13.8234     6.7433   2.050             0.048385 *
## USReleaseMonthFeb    3.6848     9.2814   0.397             0.693915
## USReleaseMonthJan    0.9770     8.9407   0.109             0.913645
## USReleaseMonthJul    7.7931     8.9451   0.871             0.389931
## USReleaseMonthJun   20.1585    10.8181   1.863             0.071325 .
## USReleaseMonthMar   11.2924     8.8526   1.276             0.211002
## USReleaseMonthMay    2.9504     9.9167   0.298             0.767934
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.09 on 33 degrees of freedom
## Multiple R-squared:  0.9676, Adjusted R-squared:  0.9578
## F-statistic: 98.62 on 10 and 33 DF,  p-value: < 0.00000000000000022
```

$H_0: \beta_5 = \beta_6 = \beta_7 = \beta_8 = \beta_9 = \beta_10$ H_a : At least one of β_5 to β_10 differ from 0

```r
anova(model41, model40)
```

```
## Analysis of Variance Table
##
## Model 1: USRevenue ~ Opening + Ratings + highrate + c
## Model 2: USRevenue ~ Opening + Ratings + highrate + c + USReleaseMonth
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     39 8855.0
## 2     33 7513.2  6    1341.8 0.9822 0.4529
```

Test Statistic: 0.9822 p-value: 0.4529 Decision: Since p-value is large (0.4529), we fail to reject H_0. Conclusion: At 5% level of significance, we do Not have sufficient to conclude that USRelease are significance and affect to use as a predictor.

```r
# Bin Opening weekend revenue so interaction.plot has a categorical x-axis without mutating movies
opening_bins <- cut(
      movies$Opening,
      breaks = quantile(movies$Opening, probs = seq(0, 1, 0.25), na.rm = TRUE),
      include.lowest = TRUE
)
```

##Model 41 is better NEED P-VALUE, CONCLUSION, INTERPRETATION, HO, HA

#Since the simpler model, model41, is better, we go back to this and start looking for multicollinearity

```r
library(car)
```

```
## Warning: package 'car' was built under R version 4.5.1
```

```
## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 4.5.1

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

```r
vif(model41)
```

```
##  Opening  Ratings highrate        c
## 1.337516 1.555632 1.122235 1.355081
```

```r
summary(model41)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + Ratings + highrate + c, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.025 -11.058  -2.270   8.354  34.902
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept) -36.4260    23.6198  -1.542              0.13110
## Opening       2.5686     0.1044  24.611 < 0.0000000000000002 ***
## Ratings      11.6367     3.5400   3.287              0.00215 **
## highrate    -33.6526     7.5825  -4.438            0.0000723 ***
## c            17.5416     6.1068   2.872              0.00656 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.07 on 39 degrees of freedom
## Multiple R-squared:  0.9618, Adjusted R-squared:  0.9579
## F-statistic: 245.7 on 4 and 39 DF,  p-value: < 0.00000000000000022
```

Interpretation: All VIF values were quite low, and less than 10, which is not a flag of multi-collinearity. Additionally, model41 passes the global F-test, with a p-value of 0.00000000000000022, AND all terms are significant. Looking at the coefficients for each variable, their signs all follow the trend that their variable graphs/summaries show. -highrate, which has coefficient of -33.6526, represents when a movie is rated PG-13 or R. In previous graphs, movies of those ratings have been associated with a decrease in USRevenue, which matches the sign of the coefficient in this model. -Ratings and Opening both are positive, which is represented in the original scatterplot, where Ratings and Opening have a positive relationship with USRevenue. -Comedy is a bit ambiguous and might mean this model needs further testing

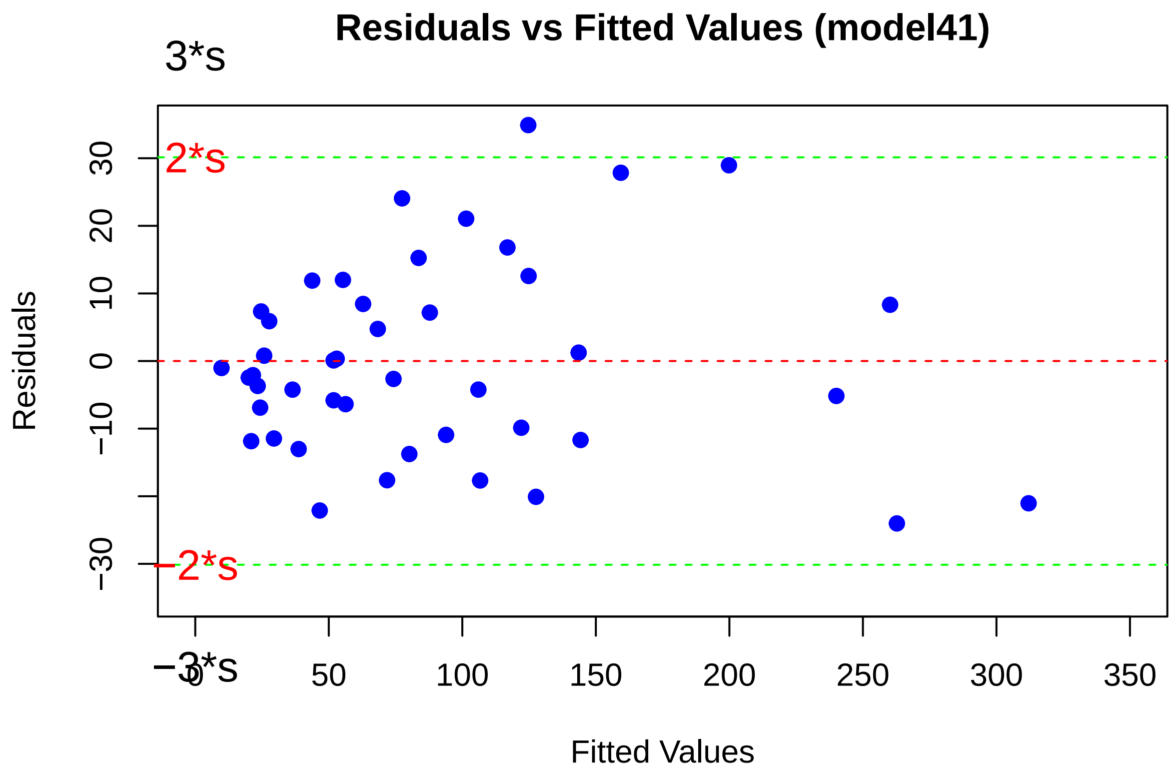
```r
# Residuals vs Fitted Values
range(fitted(model41))
```

```
## [1]   9.830128 312.042218
```

```r
range(residuals(model41))
```

```
## [1] -24.02484  34.90204
```

```r
plot(x=fitted(model41), y=residuals(model41),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (model41)", xlim=c(0, 350), ylim=c(-35, 35),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sig <- summary(model41)$sigma
abline(h=2*sig, lty="dashed", col="green")
abline(h=-2*sig, lty="dashed", col="green")
abline(h=3*sig, lty="dashed", col="black")
abline(h=-3*sig, lty="dashed", col="black")

text(x = 0,
       y = 2*sig,
       "2*s", cex=1.3, col = "red", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -2*sig,
     "-2*s", cex=1.3, col = "red", las = 1,
     xpd = TRUE)
text(x = 0,
       y = 3*sig,
       "3*s", cex=1.3, col = "black", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -3*sig,
     "-3*s", cex=1.3, col = "black", las = 1,
     xpd = TRUE)
```



Interpretation: There is heteroscedasticity here and the model will need some transformations.

##Applying ln to USRevenue (y-variable) may fix this

```
USRevenueln=log(movies$USRevenue)
model39 = lm(USRevenueln~Opening+Ratings+highrate+c, data=movies)
summary(model39)
```

```
##
## Call:
## lm(formula = USRevenueln ~ Opening + Ratings + highrate + c,
##     data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.01368 -0.31600  0.09864  0.28477  0.70777
##
## Coefficients:
##              Estimate Std. Error t value       Pr(>|t|)
## (Intercept)  1.891527   0.710827   2.661         0.0113 *
## Opening      0.025433   0.003141   8.097 0.000000000702 ***
## Ratings      0.273906   0.106534   2.571         0.0141 *
## highrate    -0.310779   0.228192  -1.362         0.1810
## c            0.072512   0.183783   0.395         0.6953
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.4535 on 39 degrees of freedom
## Multiple R-squared:  0.7735, Adjusted R-squared:  0.7503
## F-statistic:  33.3 on 4 and 39 DF,  p-value: 0.000000000004262
```

```r
vif(model39)
```

```
##  Opening  Ratings highrate        c
## 1.337516 1.555632 1.122235 1.355081
```

Interpretation: Now, highrate and c are not significant.

##Try another model, without insignificant variables

$H_0 : \beta_3 = \beta_4 = 0$ H_a : At least one of β_3 and β_4 differs from 0

```r
model38 = lm(USRevenueln~Opening+Ratings, data=movies)
summary(model38)
```

```
##
## Call:
## lm(formula = USRevenueln ~ Opening + Ratings, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9799 -0.2464  0.0530  0.3152  0.7053
##
## Coefficients:
##             Estimate Std. Error t value        Pr(>|t|)
## (Intercept) 1.672121   0.577876   2.894         0.00607 **
## Opening     0.026444   0.003038   8.705 0.0000000000728 ***
## Ratings     0.263102   0.095539   2.754         0.00874 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.453 on 41 degrees of freedom
## Multiple R-squared:  0.7624, Adjusted R-squared:  0.7509
## F-statistic:  65.8 on 2 and 41 DF,  p-value: 0.0000000000001596
```

```r
vif(model38)
```

```
##  Opening  Ratings
## 1.253941 1.253941
```

```r
anova(model38, model39)
```

```
## Analysis of Variance Table
##
## Model 1: USRevenueln ~ Opening + Ratings
## Model 2: USRevenueln ~ Opening + Ratings + highrate + c
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1     41 8.4118
## 2     39 8.0198  2   0.39202 0.9532 0.3943
```

The test statistic, F = 0.9532 and p-value = 0.3943 Decision: Since p-value is large > 0.05, we fail to reject H_0 Interpretation:** Model 38 is better. NEED HO HA, P-VALUE, DECISION, ECT FOR THIS ANOVA TEST

##We need to test the residuals for model 38

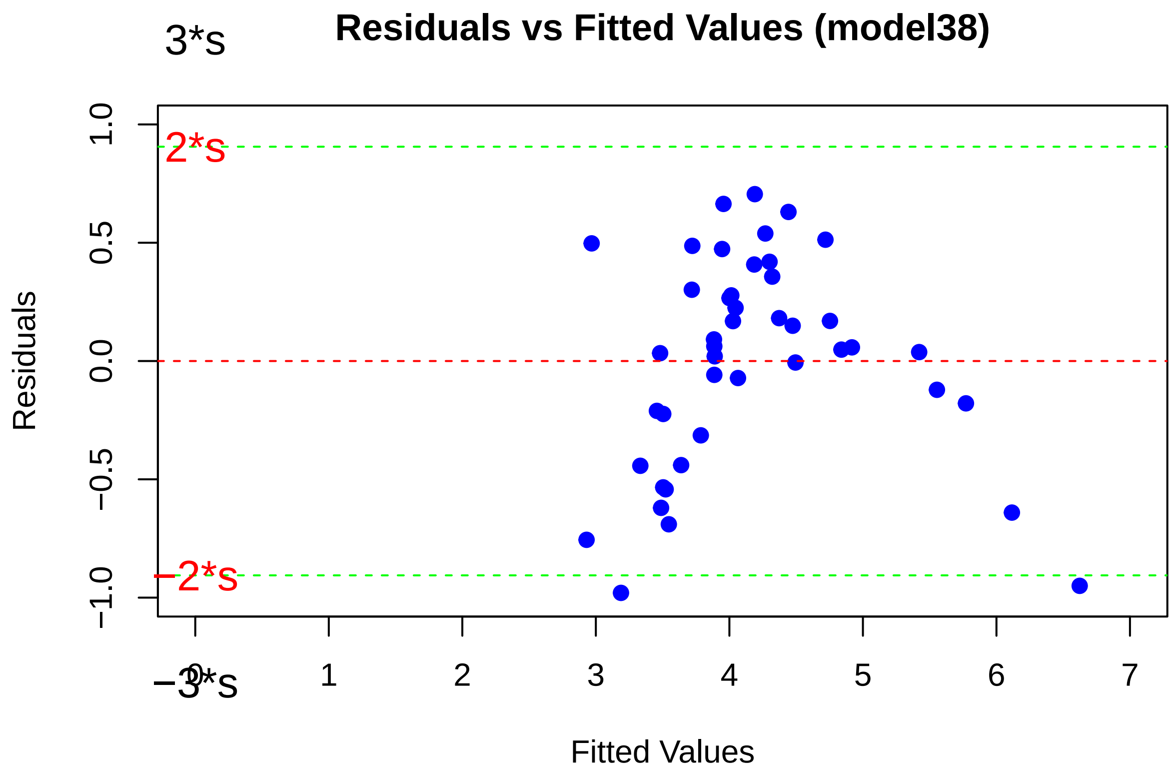
```r
# Residuals vs Fitted Values
range(fitted(model38))
```

```
## [1] 2.930388 6.623512
```

```r
range(residuals(model38))
```

```
## [1] -0.9799267  0.7053034
```

```r
plot(x=fitted(model38), y=residuals(model38),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (model38)", xlim=c(0, 7), ylim=c(-1, 1),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sig1 <- summary(model38)$sigma
abline(h=2*sig1, lty="dashed", col="green")
abline(h=-2*sig1, lty="dashed", col="green")
abline(h=3*sig1, lty="dashed", col="black")
abline(h=-3*sig1, lty="dashed", col="black")

text(x = 0,
       y = 2*sig1,
       "2*s", cex=1.3, col = "red", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -2*sig1,
     "-2*s", cex=1.3, col = "red", las = 1,
     xpd = TRUE)
text(x = 0,
       y = 3*sig1,
       "3*s", cex=1.3, col = "black", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -3*sig1,
     "-3*s", cex=1.3, col = "black", las = 1,
     xpd = TRUE)
```



#there is a curve there- maybe we need to do transformation(get rid of ln of USRevenue)

```
model37 = lm(USRevenue~Opening+Ratings+highrate, data=movies)
summary(model37)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + Ratings + highrate, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -25.127  -8.612  -2.859   6.221  48.771
##
## Coefficients:
##             Estimate Std. Error t value             Pr(>|t|)
## (Intercept)  -5.8923    22.9249  -0.257             0.798475
## Opening       2.5626     0.1134  22.596 < 0.0000000000000002 ***
## Ratings       7.1921     3.4605   2.078             0.044137 *
## highrate    -30.9454     8.1772  -3.784             0.000506 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.38 on 40 degrees of freedom
## Multiple R-squared:  0.9538, Adjusted R-squared:  0.9503
## F-statistic: 275.1 on 3 and 40 DF,  p-value: < 0.00000000000000022
```

```r
vif(model37)
```

```
##  Opening  Ratings highrate
## 1.336985 1.258426 1.104897
```

Interpretation: This model looks better, has significant values, and VIF is low.

##Lets visualize model37 residuals

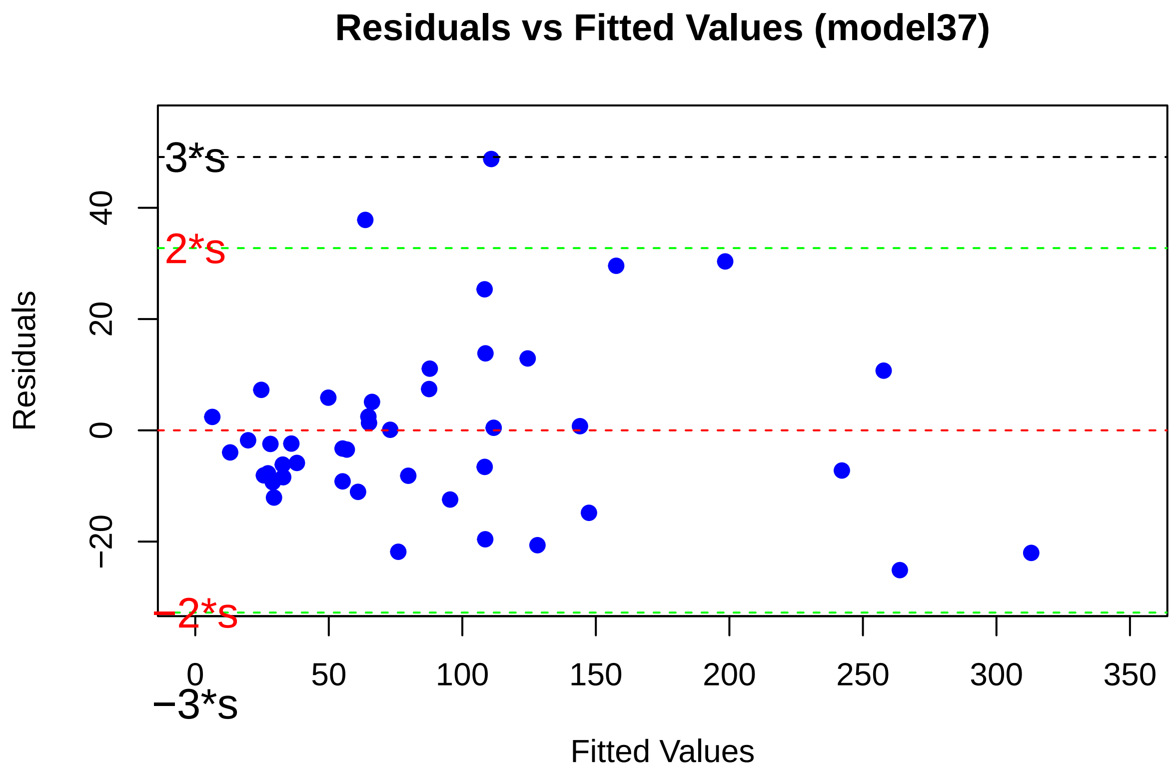
```r
# Residuals vs Fitted Values
range(fitted(model37))
```

```
## [1]   6.389032 313.029975
```

```r
range(residuals(model37))
```

```
## [1] -25.12731  48.77060
```

```r
plot(x=fitted(model37), y=residuals(model37),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (model37)", xlim=c(0, 350), ylim=c(-30, 55),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sig2 <- summary(model37)$sigma
abline(h=2*sig2, lty="dashed", col="green")
abline(h=-2*sig2, lty="dashed", col="green")
abline(h=3*sig2, lty="dashed", col="black")
abline(h=-3*sig2, lty="dashed", col="black")

text(x = 0,
       y = 2*sig2,
       "2*s", cex=1.3, col = "red", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -2*sig2,
     "-2*s", cex=1.3, col = "red", las = 1,
     xpd = TRUE)
text(x = 0,
       y = 3*sig2,
       "3*s", cex=1.3, col = "black", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -3*sig2,
     "-3*s", cex=1.3, col = "black", las = 1,
     xpd = TRUE)
```



Interpretation: Still slightly heteroscedastic.

```r
anova(model37)
```

```
## Analysis of Variance Table
##
## Response: USRevenue
##           Df Sum Sq Mean Sq  F value                Pr(>F)
## Opening    1 216043  216043 805.5011 < 0.000000000000000022 ***
## Ratings    1   1429    1429   5.3286              0.0262289 *
## highrate   1   3841    3841  14.3212              0.0005061 ***
## Residuals 40  10728     268
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

#Lets try a new model. Interaction has worked in the past with Opening and Theaters, maybe this will help

```r
model36 = lm(USRevenueln~Opening+Ratings+Opening*Theaters, data=movies)
summary(model36)
```

```
##
## Call:
## lm(formula = USRevenueln ~ Opening + Ratings + Opening * Theaters,
##     data = movies)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.62700 -0.11535  0.03111  0.11977  0.55516
##
## Coefficients:
##                      Estimate   Std. Error t value         Pr(>|t|)
## (Intercept)      -0.794551072  0.388900226  -2.043          0.04784 *
## Opening           0.108177295  0.013607126   7.950 0.00000000110374 ***
## Ratings           0.174574411  0.050605659   3.450          0.00136 **
## Theaters          0.000970183  0.000096083  10.097 0.00000000000194 ***
## Opening:Theaters -0.000023655  0.000003365  -7.030 0.00000001949281 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2314 on 39 degrees of freedom
## Multiple R-squared:  0.941,  Adjusted R-squared:  0.935
## F-statistic: 155.5 on 4 and 39 DF,  p-value: < 0.00000000000000022
```

```r
vif(model36, type="predictor")
```

```
## GVIFs computed for predictors
```

```
##              GVIF Df GVIF^(1/(2*Df)) Interacts With  Other Predictors
## Opening  1.347675  3        1.050988       Theaters           Ratings
## Ratings  1.347675  1        1.160894             --  Opening, Theaters
## Theaters 1.347675  3        1.050988        Opening           Ratings
```

Interpretation: model36 has significant terms and low VIF scores

```r
# Residuals vs Fitted Values
range(fitted(model36))
```

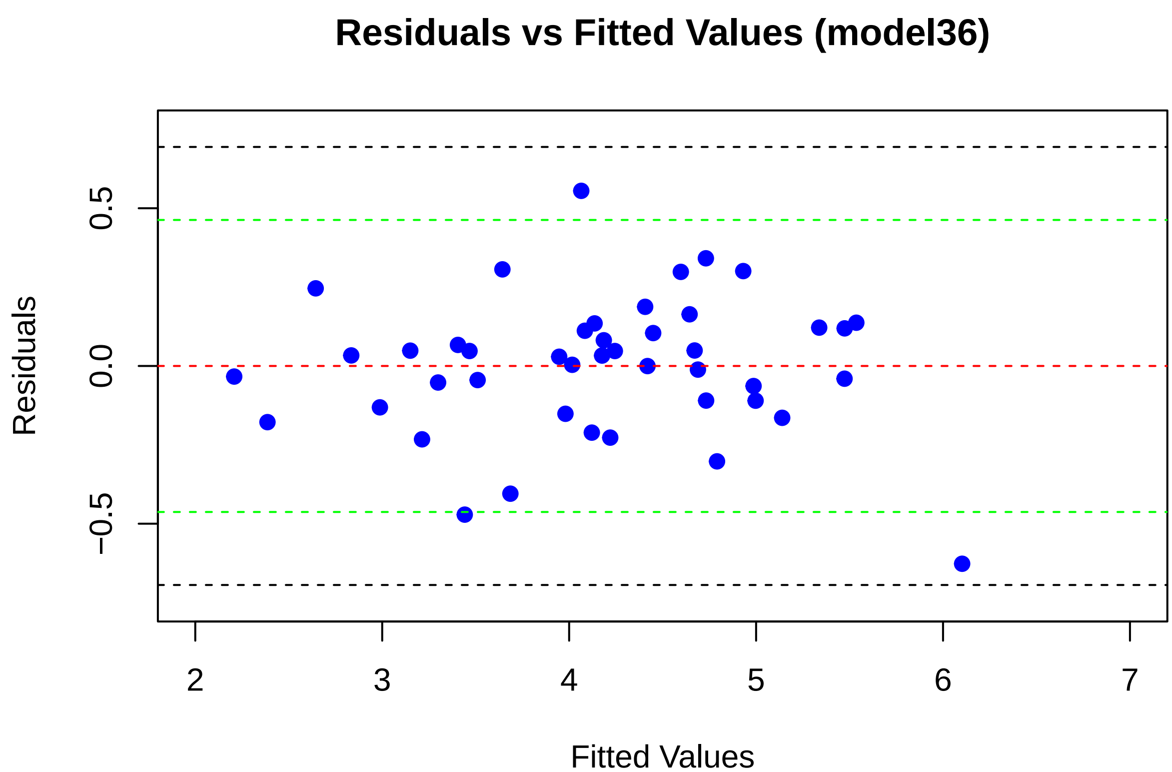
```
## [1] 2.208348 6.102203
```

```r
range(residuals(model36))
```

```
## [1] -0.6269955  0.5551636
```

```r
plot(x=fitted(model36), y=residuals(model36),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (model36)", xlim=c(2, 7), ylim=c(-.75, .75),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sig3 <- summary(model36)$sigma
abline(h=2*sig3, lty="dashed", col="green")
abline(h=-2*sig3, lty="dashed", col="green")
abline(h=3*sig3, lty="dashed", col="black")
abline(h=-3*sig3, lty="dashed", col="black")
```

```r
text(x = 0,
       y = 2*sig3,
       "2*s", cex=1.3, col = "red", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -2*sig3,
     "-2*s", cex=1.3, col = "red", las = 1,
     xpd = TRUE)
text(x = 0,
       y = 3*sig3,
       "3*s", cex=1.3, col = "black", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -3*sig3,
     "-3*s", cex=1.3, col = "black", las = 1,
     xpd = TRUE)
```




```r
#install.packages("Hmisc")
library(Hmisc)
```

```
## Warning: package 'Hmisc' was built under R version 4.5.1

##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
Z = cbind(
  movies$Opening,
  movies$Ratings,
  movies$Theaters
)
rcorr(Z, type="pearson")
```

```
##      [,1] [,2] [,3]
## [1,] 1.00 0.45 0.74
## [2,] 0.45 1.00 0.33
## [3,] 0.74 0.33 1.00
##
## n= 44
##
##
## P
##      [,1]   [,2]   [,3]
## [1,]        0.0022 0.0000
## [2,] 0.0022        0.0313
## [3,] 0.0000 0.0313
```

```
summary(residuals(model36))
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.62700 -0.11535  0.03111  0.00000  0.11977  0.55516
```

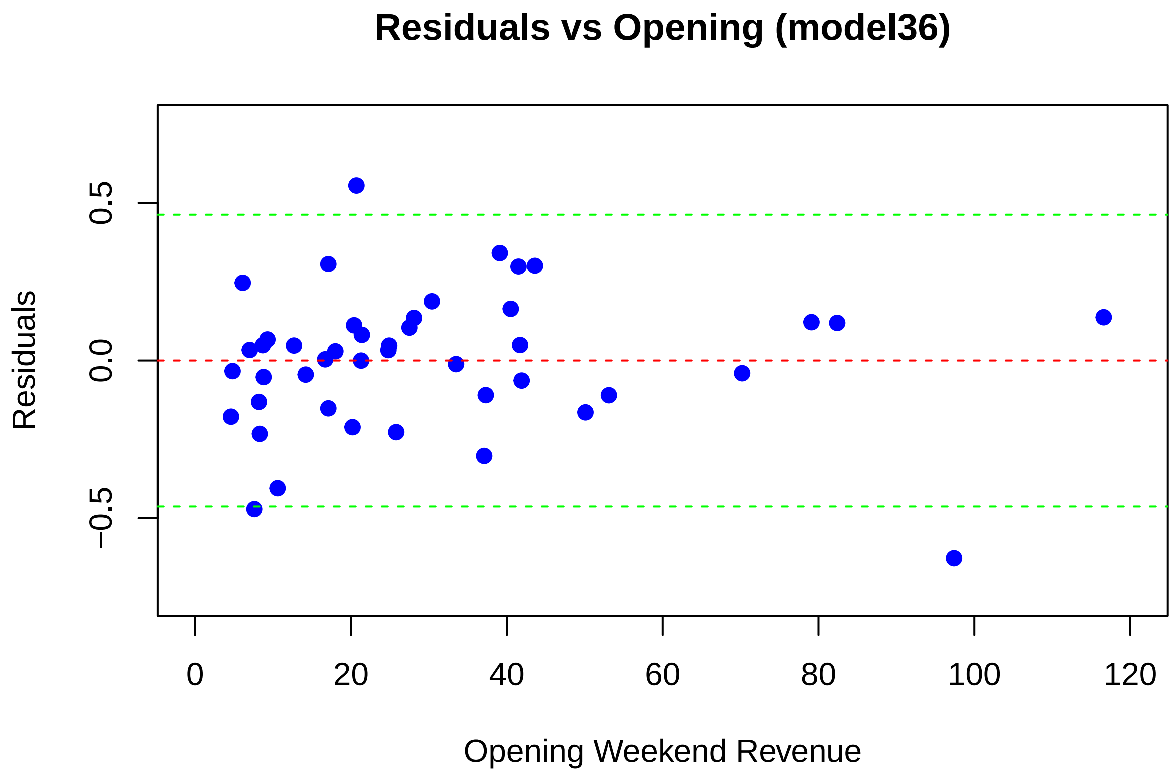
```
# Residuals vs Opening
range(movies$Opening)
```

```
## [1]   4.6 116.6
```

```
range(residuals(model36))
```

```
## [1] -0.6269955  0.5551636
```

```
plot(x=movies$Opening, y=residuals(model36),
     xlab = "Opening Weekend Revenue", ylab = "Residuals",
     main = "Residuals vs Opening (model36)", xlim = c(0, 120) , ylim = c(-.75,.75) ,
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")
abline(h=2*sig3, lty="dashed", col="green")
abline(h=-2*sig3, lty="dashed", col="green")
```



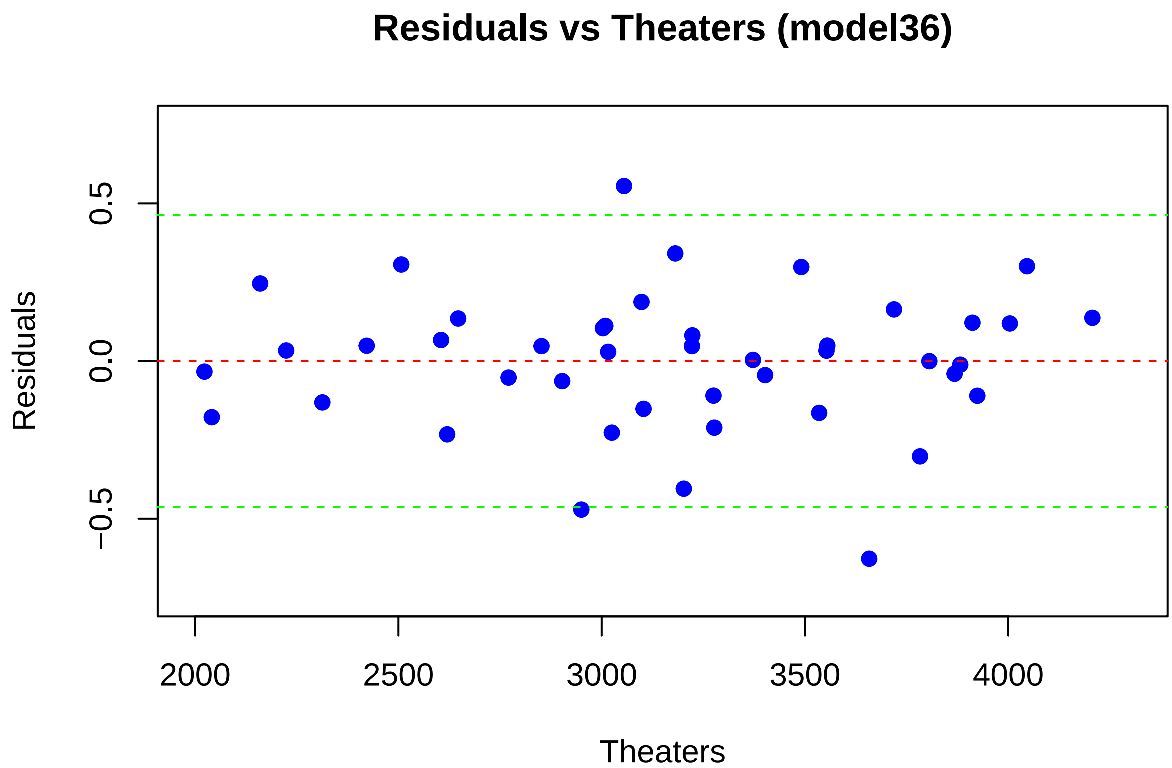

```
# Residuals vs Theaters
range(movies$Theaters)
```

```
## [1] 2023 4207
```

```
range(residuals(model36))
```

```
## [1] -0.6269955  0.5551636
```

```
plot(x=movies$Theaters, y=residuals(model36),
     xlab = "Theaters", ylab = "Residuals",
     main = "Residuals vs Theaters (model36)", xlim = c(2000, 4300) , ylim = c(-.75,.75) ,
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")
abline(h=2*sig3, lty="dashed", col="green")
abline(h=-2*sig3, lty="dashed", col="green")
```



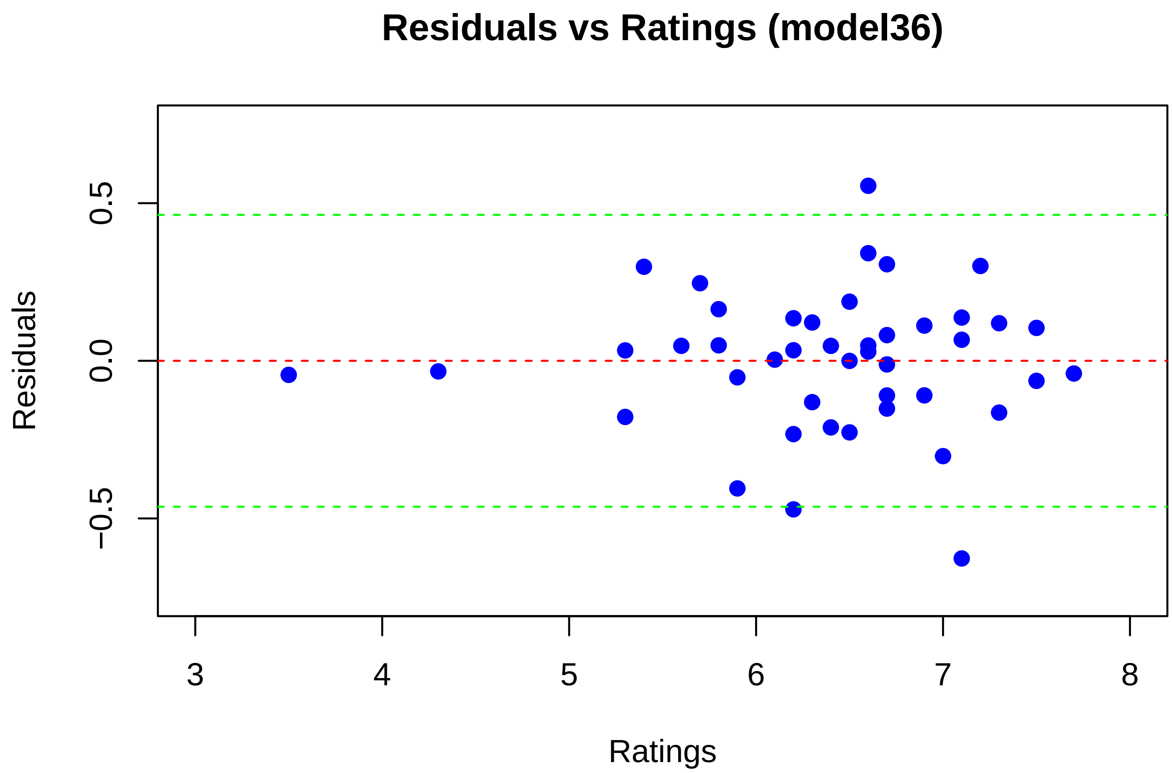

```
# Ratings vs Theaters
range(movies$Ratings)
```

```
## [1] 3.5 7.7
```

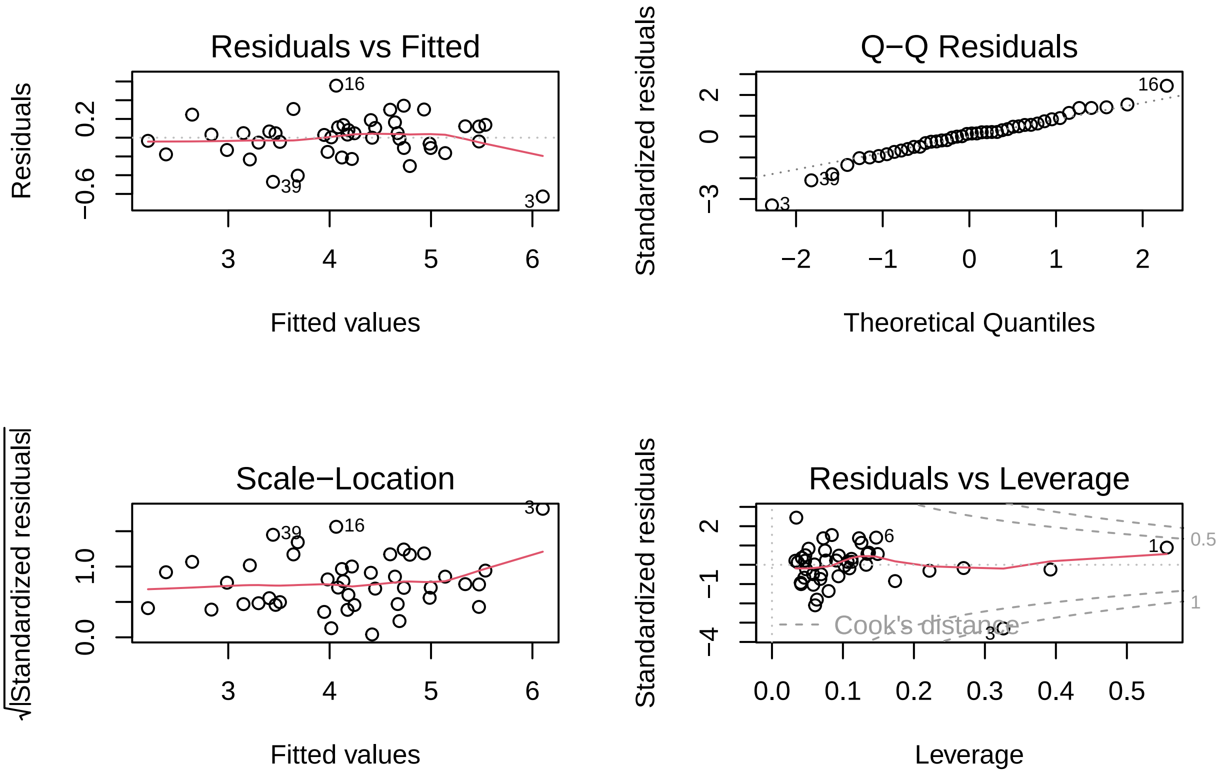
```
range(residuals(model36))
```

```
## [1] -0.6269955  0.5551636
```

```
plot(x=movies$Ratings, y=residuals(model36),
     xlab = "Ratings", ylab = "Residuals",
     main = "Residuals vs Ratings (model36)", xlim = c(3, 8) , ylim = c(-.75,.75) ,
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")
abline(h=2*sig3, lty="dashed", col="green")
abline(h=-2*sig3, lty="dashed", col="green")
```

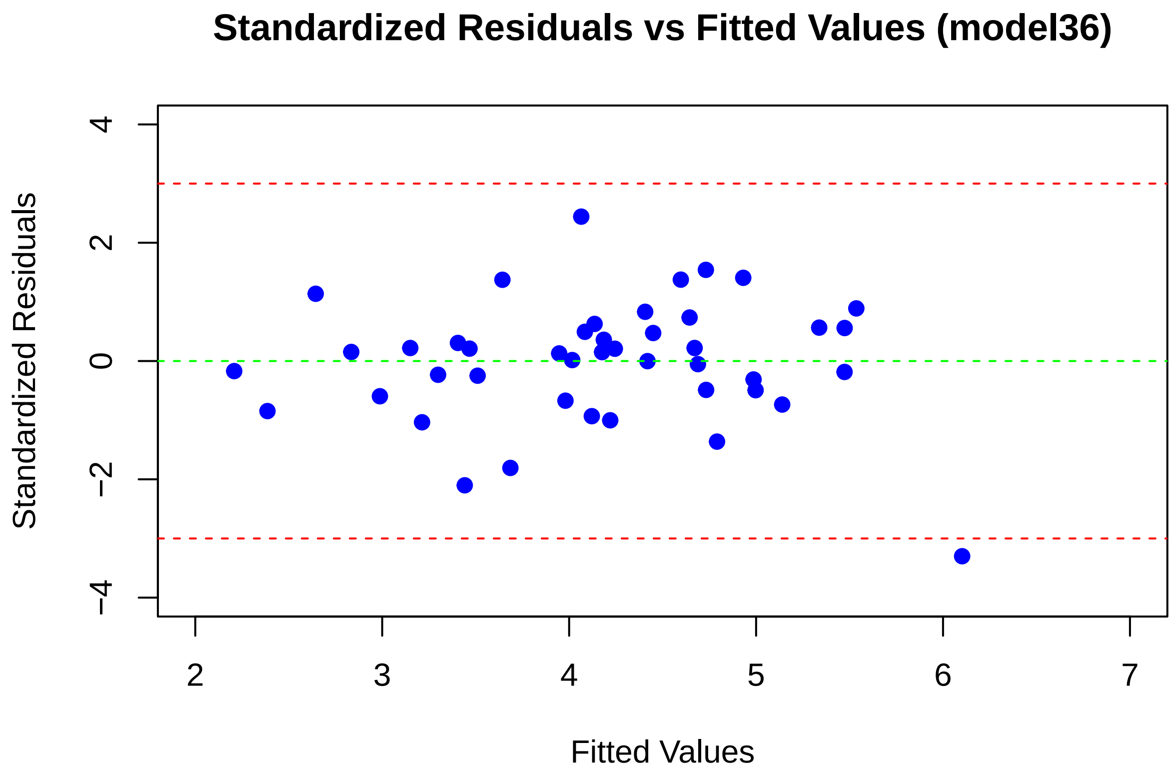



```r
# Diagnostic plots (4-panel)
par(mfrow=c(2,2))
plot(model36)
```



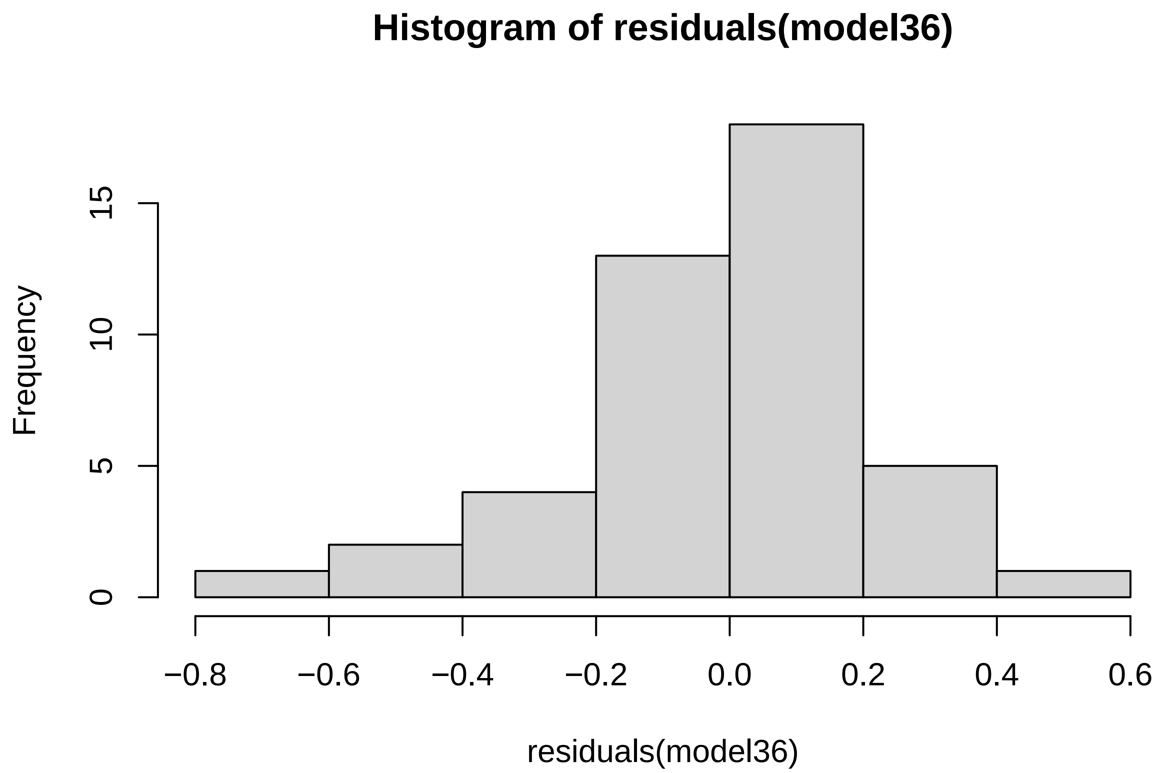

```
par(mfrow=c(1,1))
```

```
# Standardized residuals for model36
plot(y=rstandard(model36), x=fitted(model36),
     xlab = "Fitted Values", ylab = "Standardized Residuals",
     main = "Standardized Residuals vs Fitted Values (model36)",xlim = c(2,7) , ylim = c(-4,4) ,
     pch=19, col="blue")
abline(h=-3, lty="dashed", col="red")
abline(h=3, lty="dashed", col="red")
abline(h=0, lty="dashed", col="green")
identify(y=rstandard(model36), x=fitted(model36))
```

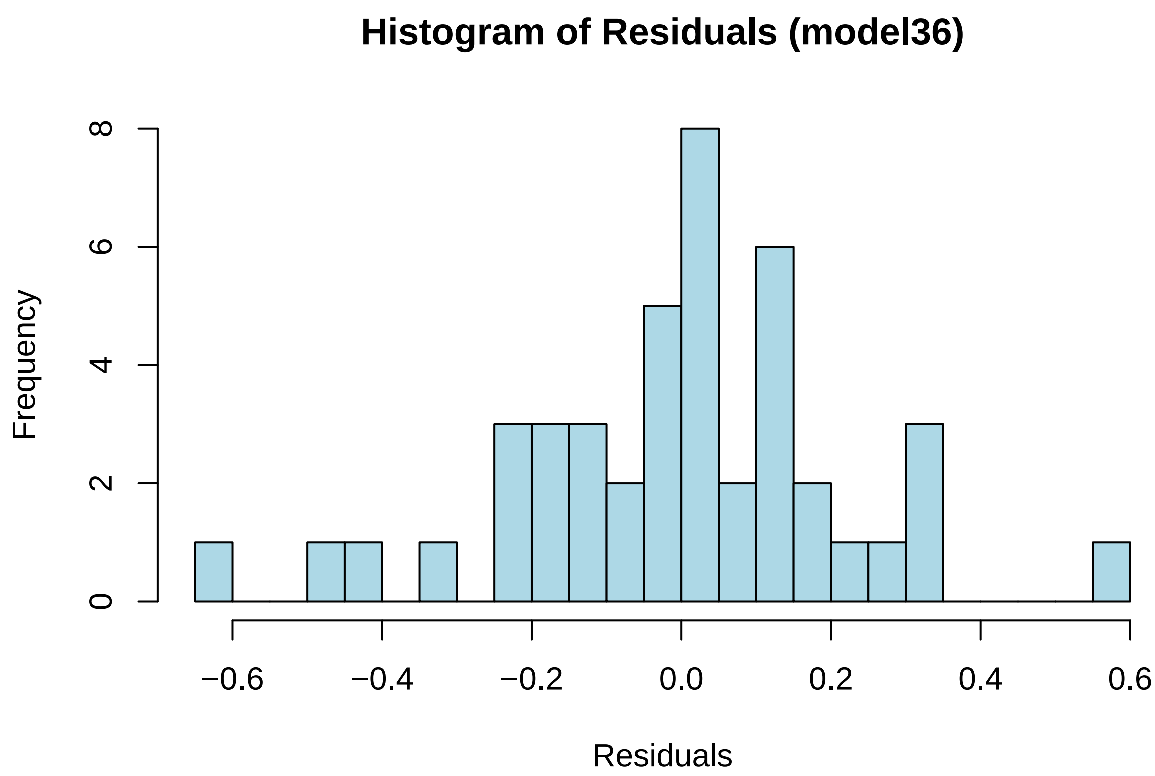



```
## integer(0)
```

```r
hist(residuals(model36))
```




```r
# Histogram of residuals
hist(residuals(model36), main="Histogram of Residuals (model36)",
     xlab="Residuals", col="lightblue", breaks=20)
```




```r
# Calculate influence measures
n <- nrow(movies)
p <- length(coef(model36))
leverage <- hatvalues(model36)
StanRes <- rstandard(model36)
cd <- cooks.distance(model36)
Rstudent <- rstudent(model36)

# Cutoffs
#cutL <- 2*p/n  # Leverage cutoff
#cutCD <- 4/(n-p-1)  # Cook's Distance cutoff

# Create influence diagnostics table
Influence_table <- data.frame(
  Observation = 1:n,
  USRevenue = movies$USRevenue,
  Leverage = round(leverage, 3),
  StanRes = round(StanRes, 3),
  CooksD = round(cd, 3),
  Rstudent = round(Rstudent, 3)
)

print(Influence_table)
```

```
##    Observation USRevenue Leverage StanRes CooksD Rstudent
## 1            1     291.0    0.556   0.889  0.198    0.887
```

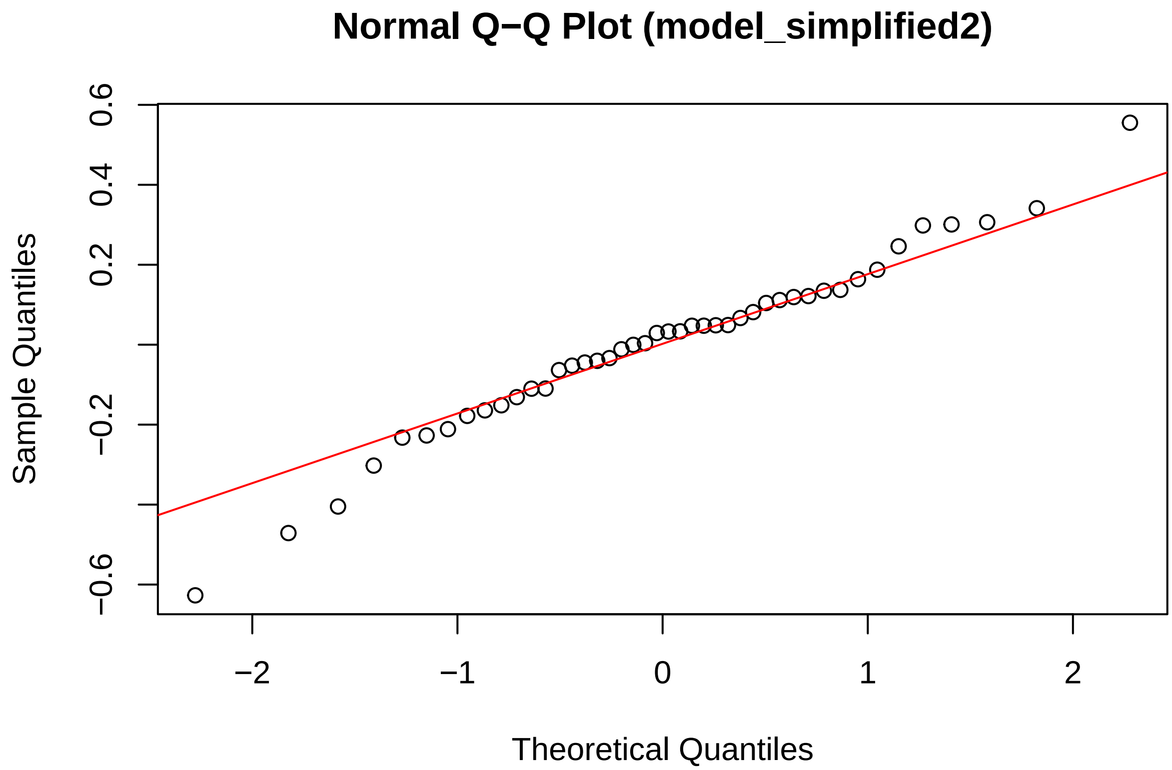
```
## 2            2     268.5    0.149   0.558  0.011    0.553
## 3            3     238.7    0.326  -3.300  1.054   -3.837
## 4            4     234.9    0.135   0.565  0.010    0.560
## 5            5     228.8    0.109  -0.185  0.001   -0.182
## 6            6     187.2    0.147   1.407  0.068    1.426
## 7            7     159.6    0.084   1.541  0.044    1.570
## 8            8     144.8    0.068  -0.736  0.008   -0.731
## 9            9     137.4    0.222  -0.312  0.006   -0.308
## 10          10     133.7    0.123   1.376  0.053    1.392
## 11          11     132.6    0.069  -0.493  0.004   -0.488
## 12          12     122.5    0.075   0.736  0.009    0.731
## 13          13     112.2    0.076   0.220  0.001    0.218
## 14          14     107.5    0.102  -0.052  0.000   -0.052
## 15          15     101.8    0.058  -0.488  0.003   -0.484
## 16          16     101.5    0.034   2.441  0.042    2.618
## 17          17      98.9    0.052   0.832  0.008    0.829
## 18          18      95.0    0.094   0.474  0.005    0.469
## 19          19      89.0    0.080  -1.362  0.032   -1.378
## 20          20      83.0    0.133  -0.002  0.000   -0.002
## 21          21      73.1    0.033   0.208  0.000    0.206
## 22          22      71.6    0.137   0.627  0.012    0.622
## 23          23      71.3    0.042   0.360  0.001    0.356
## 24          24      67.3    0.107   0.151  0.001    0.149
## 25          25      66.4    0.047   0.494  0.002    0.489
## 26          26      55.7    0.060   0.016  0.000    0.016
## 27          27      54.2    0.041  -1.002  0.009   -1.002
## 28          28      53.3    0.036   0.129  0.000    0.127
## 29          29      51.9    0.072   1.374  0.029    1.390
## 30          30      49.9    0.041  -0.932  0.007   -0.931
## 31          31      46.0    0.046  -0.670  0.004   -0.666
## 32          32      33.6    0.047   0.210  0.000    0.207
## 33          33      32.2    0.112   0.306  0.002    0.302
## 34          34      32.0    0.392  -0.248  0.008   -0.245
## 35          35      26.6    0.063  -1.807  0.044   -1.864
## 36          36      25.7    0.048  -0.232  0.001   -0.229
## 37          37      24.5    0.090   0.220  0.001    0.217
## 38          38      19.7    0.058  -1.035  0.013   -1.036
## 39          39      19.5    0.061  -2.101  0.057   -2.202
## 40          40      18.0    0.126   1.137  0.037    1.142
## 41          41      17.6    0.112   0.153  0.001    0.151
## 42          42      17.4    0.094  -0.596  0.007   -0.591
## 43          43       9.1    0.173  -0.846  0.030   -0.843
## 44          44       8.8    0.270  -0.170  0.002   -0.168
```

```r
# Show influential observations
#influential <- influence_table[leverage > cutL | cd > cutCD, ]
#if(nrow(influential) > 0) {
 # cat("\nInfluential Observations:\n")
 # print(influential)
#} else {
 # cat("\nNo influential observations detected.\n")
#}
```

##NEED CALCUATIONS OF LEVERAGE, COOKS, RSTUDENT CUTOFFS AND PARAGPHARH OF

WHICH POINTS DONT MATCH ON DATA SET

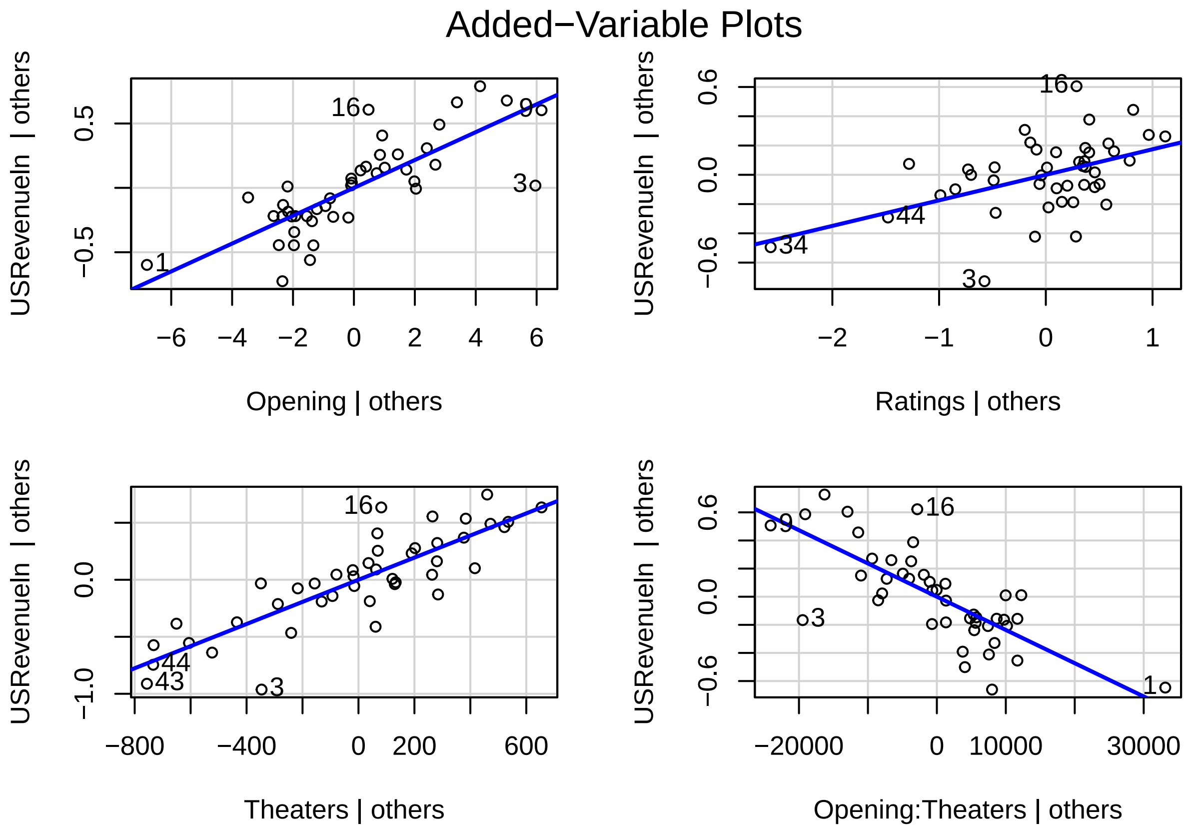
```r
# Q-Q plot for model36
qqnorm(residuals(model36), main = "Normal Q-Q Plot (model_simplified2)")
qqline(residuals(model36), col="red")
```




```r
shapiro.test(residuals(model36))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model36)
## W = 0.97508, p-value = 0.4507
```

```r
##Partial Residual plots
library(car)
avPlots(model36)
```



We also have model_complex, which is a more complicated model, but performs well. Lets test it, and compare it to model36.

```
movies$USReleaseMonth <- sub("^[0-9]+-", "", movies$USRelease)
movies$jun <- ifelse(movies$USReleaseMonth == "Jun", 1, 0)
```

```
model_complex = lm(USRevenue ~ Opening + Ratings + highrate + c + jun+ Opening*Theaters, data = movies)
summary(model_complex)
```

```
##
## Call:
## lm(formula = USRevenue ~ Opening + Ratings + highrate + c + jun +
##     Opening * Theaters, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -21.124  -8.635   0.567   6.959  36.509
##
## Coefficients:
##                     Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)      -62.4011103  30.8149232  -2.025  0.050329 .
## Opening            4.2455417   0.8445410   5.027 0.0000138 ***
## Ratings            9.0561410   3.4163551   2.651  0.011859 *
```

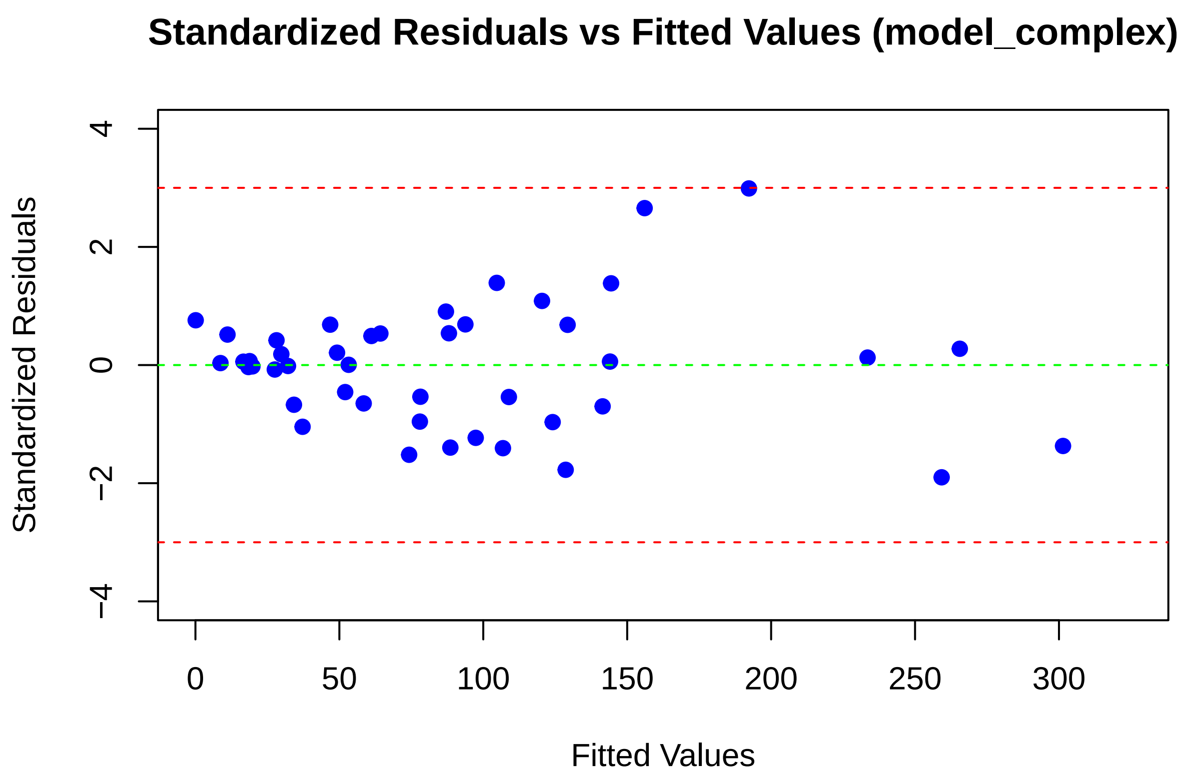
```
## highrate         -33.6007447   7.7079030  -4.359  0.000104 ***
## c                 15.3049081   5.9125200   2.589  0.013816 *
## jun               17.5447161   7.3595620   2.384  0.022523 *
## Theaters           0.0129038   0.0060914   2.118  0.041116 *
## Opening:Theaters  -0.0004765   0.0002092  -2.278  0.028778 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.56 on 36 degrees of freedom
## Multiple R-squared:  0.9715, Adjusted R-squared:  0.9659
## F-statistic: 175.2 on 7 and 36 DF,  p-value: < 0.0000000000000022
```

```
vif(model_complex, type="predictor") #takes into account interaction terms, so it doesn't raise VIF bec
```

```
## GVIFs computed for predictors
```

```
##              GVIF Df GVIF^(1/(2*Df)) Interacts With
## Opening  2.301456  3        1.149036       Theaters
## Ratings  1.790060  1        1.337931             --
## highrate 1.432752  1        1.196976             --
## c        1.569336  1        1.252731             --
## jun      1.306178  1        1.142882             --
## Theaters 2.301456  3        1.149036        Opening
##                                    Other Predictors
## Opening                   Ratings, highrate, c, jun
## Ratings        Opening, highrate, c, jun, Theaters
## highrate        Opening, Ratings, c, jun, Theaters
## c        Opening, Ratings, highrate, jun, Theaters
## jun        Opening, Ratings, highrate, c, Theaters
## Theaters                  Ratings, highrate, c, jun
```

```
# Standardized residuals for model_complex
plot(y=rstandard(model_complex), x=fitted(model_complex),
     xlab = "Fitted Values", ylab = "Standardized Residuals",
     main = "Standardized Residuals vs Fitted Values (model_complex)",xlim = c(0,325) , ylim = c(-4,4)
     pch=19, col="blue")
abline(h=-3, lty="dashed", col="red")
abline(h=3, lty="dashed", col="red")
abline(h=0, lty="dashed", col="green")
identify(y=rstandard(model_complex), x=fitted(model_complex))
```




```
## integer(0)
```

#slight heteroscacicity- let's do some transformations

```
model_complexln = lm(USRevenueln ~ Opening + Ratings + highrate+c+ jun+ Opening*Theaters, data = movies)
summary(model_complexln)
```

```
##
## Call:
## lm(formula = USRevenueln ~ Opening + Ratings + highrate + c +
##     jun + Opening * Theaters, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.49048 -0.10821 -0.02302  0.14155  0.35845
##
## Coefficients:
##                      Estimate   Std. Error t value          Pr(>|t|)
## (Intercept)      -0.935989928  0.454607731  -2.059          0.046795 *
## Opening           0.115691630  0.012459382   9.286 0.000000000043324 ***
## Ratings           0.202450907  0.050400951   4.017          0.000287 ***
## highrate         -0.183564093  0.113713485  -1.614          0.115202
## c                 0.172447628  0.087226480   1.977          0.055738 .
## jun               0.266945092  0.108574465   2.459          0.018888 *
## Theaters          0.000992786  0.000089866  11.047 0.000000000000406 ***
```

```
## Opening:Theaters -0.000025968  0.000003086  -8.414 0.000000000505580 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2 on 36 degrees of freedom
## Multiple R-squared:  0.9593, Adjusted R-squared:  0.9514
## F-statistic: 121.3 on 7 and 36 DF,  p-value: < 0.00000000000000022
```

```r
vif(model_complexln, type="predictor")
```

```
## GVIFs computed for predictors
```

```
##              GVIF Df GVIF^(1/(2*Df)) Interacts With
## Opening  2.301456  3        1.149036       Theaters
## Ratings  1.790060  1        1.337931             --
## highrate 1.432752  1        1.196976             --
## c        1.569336  1        1.252731             --
## jun      1.306178  1        1.142882             --
## Theaters 2.301456  3        1.149036        Opening
##                                 Other Predictors
## Opening                  Ratings, highrate, c, jun
## Ratings        Opening, highrate, c, jun, Theaters
## highrate        Opening, Ratings, c, jun, Theaters
## c        Opening, Ratings, highrate, jun, Theaters
## jun        Opening, Ratings, highrate, c, Theaters
## Theaters                 Ratings, highrate, c, jun
```

##Get rid of highrate and c

```r
model_complexln1 = lm(USRevenueln ~ Opening + Ratings + jun+ Opening*Theaters, data = movies)
summary(model_complexln1)
```

```
##
## Call:
## lm(formula = USRevenueln ~ Opening + Ratings + jun + Opening *
##     Theaters, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.50671 -0.10755  0.02972  0.10833  0.36477
##
## Coefficients:
##                      Estimate   Std. Error t value           Pr(>|t|)
## (Intercept)      -0.839186413  0.355819690  -2.358           0.023597 *
## Opening           0.112079002  0.012508736   8.960 0.0000000000658761 ***
## Ratings           0.165804061  0.046354788   3.577           0.000969 ***
## jun               0.323434008  0.109822242   2.945           0.005485 **
## Theaters          0.001000839  0.000088445  11.316 0.0000000000000986 ***
## Opening:Theaters -0.000025074  0.000003113  -8.054 0.0000000009688485 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.2116 on 38 degrees of freedom
## Multiple R-squared:  0.952,  Adjusted R-squared:  0.9457
## F-statistic: 150.6 on 5 and 38 DF,  p-value: < 0.0000000000000022
```

```r
vif(model_complexln1, type="predictor")
```

```
## GVIFs computed for predictors

##              GVIF Df GVIF^(1/(2*Df)) Interacts With           Other Predictors
## Opening  1.553254  3        1.076152       Theaters                Ratings, jun
## Ratings  1.353261  1        1.163297             --       Opening, jun, Theaters
## jun      1.194345  1        1.092861             --   Opening, Ratings, Theaters
## Theaters 1.553254  3        1.076152        Opening                Ratings, jun
```

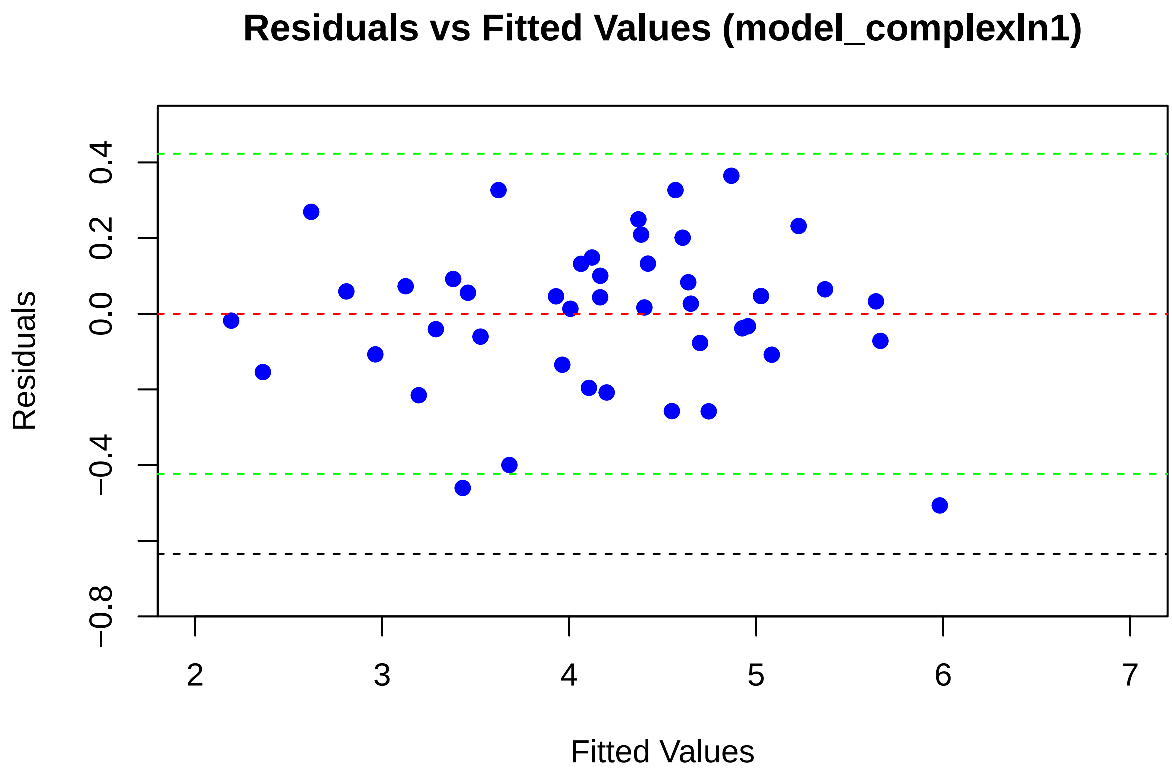
```r
# Residuals vs Fitted Values
range(fitted(model_complexln1))
```

```
## [1] 2.192966 5.981921
```

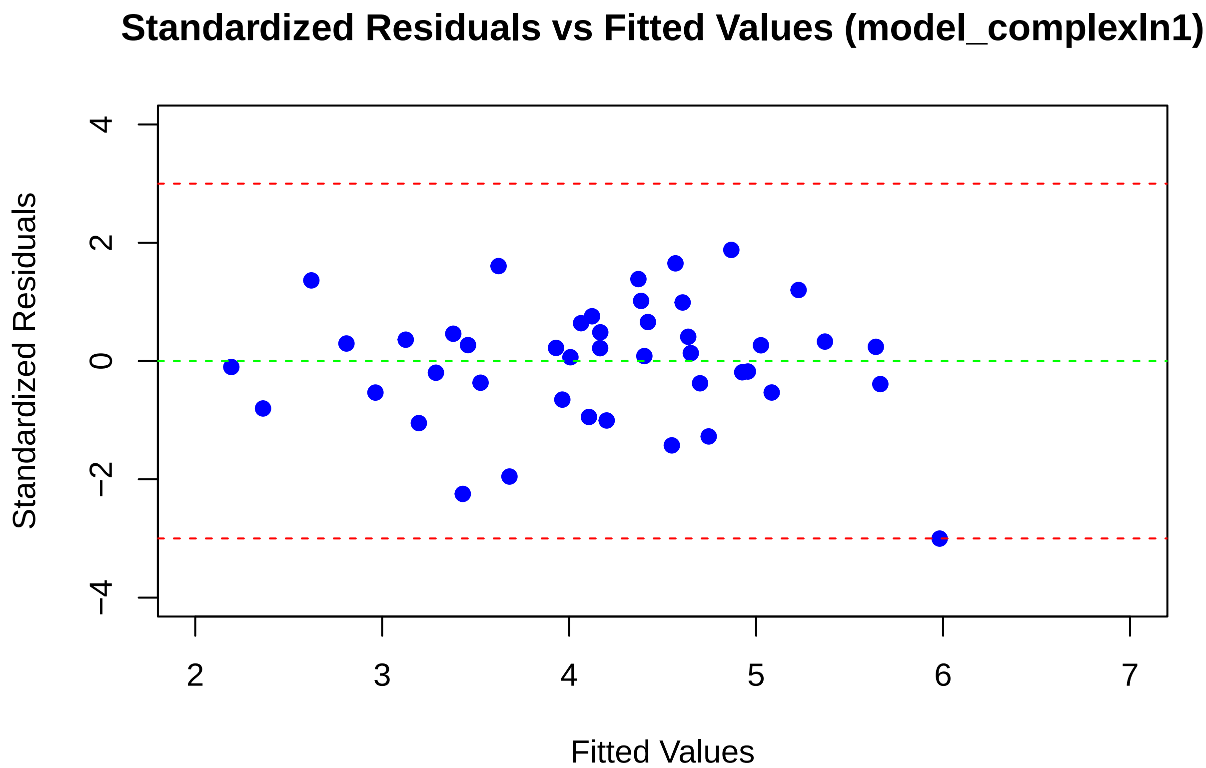
```r
range(residuals(model_complexln1))
```

```
## [1] -0.5067135  0.3647652
```

```r
plot(x=fitted(model_complexln1), y=residuals(model_complexln1),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (model_complexln1)", xlim=c(2, 7), ylim=c(-.75, .5),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sig5 <- summary(model_complexln1)$sigma
abline(h=2*sig5, lty="dashed", col="green")
abline(h=-2*sig5, lty="dashed", col="green")
abline(h=3*sig5, lty="dashed", col="black")
abline(h=-3*sig5, lty="dashed", col="black")

text(x = 0,
       y = 2*sig5,
       "2*s", cex=1.3, col = "red", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -2*sig5,
     "-2*s", cex=1.3, col = "red", las = 1,
     xpd = TRUE)
text(x = 0,
       y = 3*sig5,
       "3*s", cex=1.3, col = "black", las = 1,
       xpd = TRUE)
text(x = 0,
     y = -3*sig5,
     "-3*s", cex=1.3, col = "black", las = 1,
     xpd = TRUE)
```

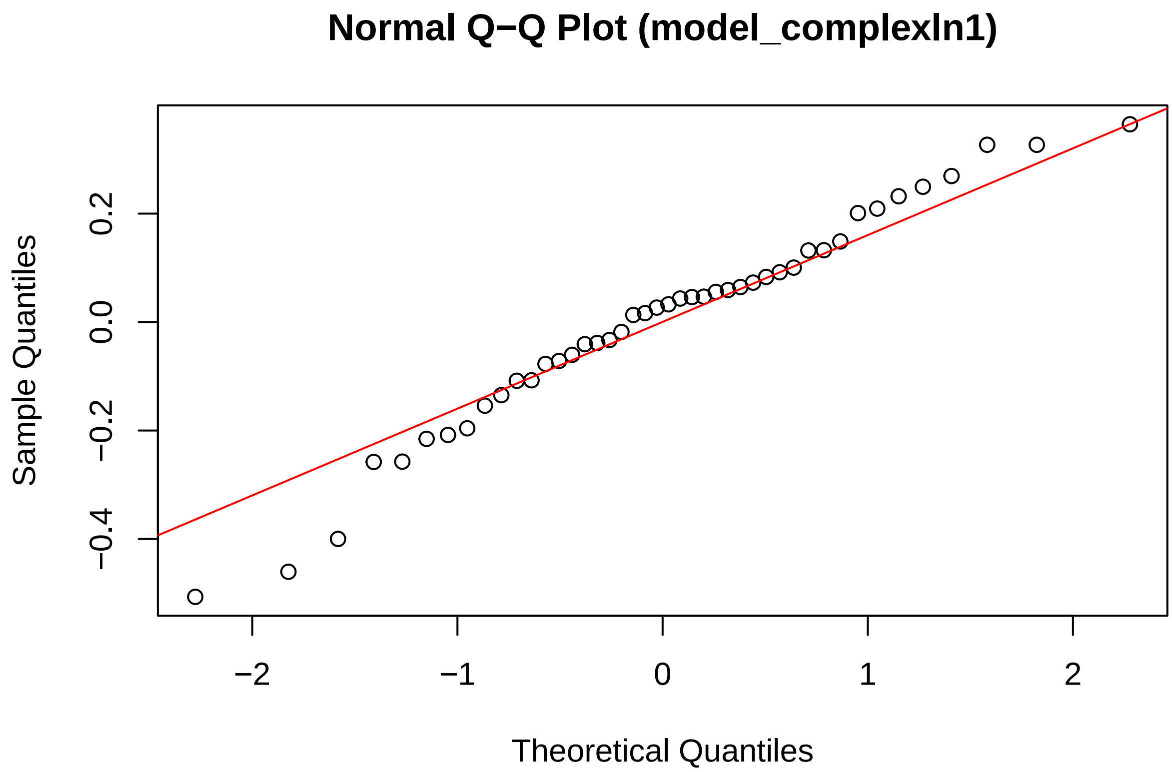



```
# Standardized residuals for model_complexln1
plot(y=rstandard(model_complexln1), x=fitted(model_complexln1),
     xlab = "Fitted Values", ylab = "Standardized Residuals",
     main = "Standardized Residuals vs Fitted Values (model_complexln1)",xlim = c(2,7) , ylim = c(-4,4)
     pch=19, col="blue")
abline(h=-3, lty="dashed", col="red")
abline(h=3, lty="dashed", col="red")
abline(h=0, lty="dashed", col="green")
identify(y=rstandard(model_complexln1), x=fitted(model_complexln1))
```

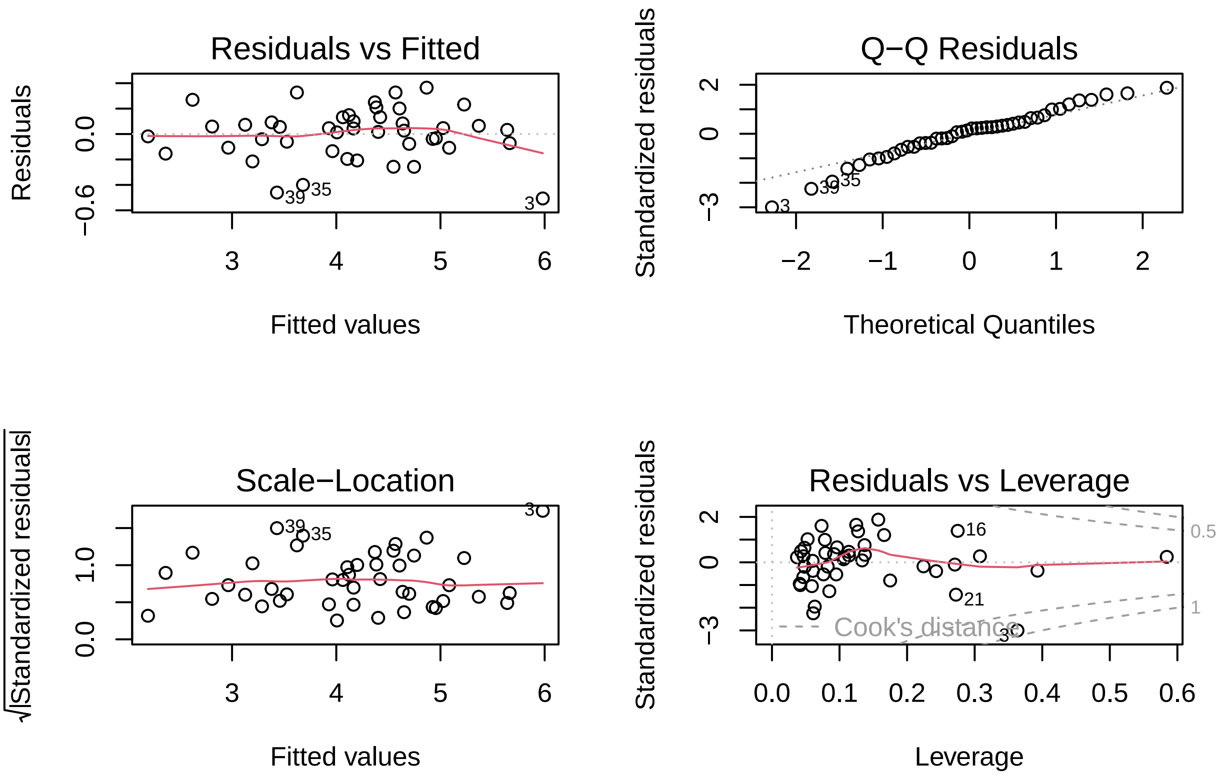



```
## integer(0)
```

```r
# Q-Q plot for model29
qqnorm(residuals(model_complexln1), main = "Normal Q-Q Plot (model_complexln1)")
qqline(residuals(model_complexln1), col="red")
```




```r
# Diagnostic plots (4-panel)
par(mfrow=c(2,2))
plot(model_complexln1)
```



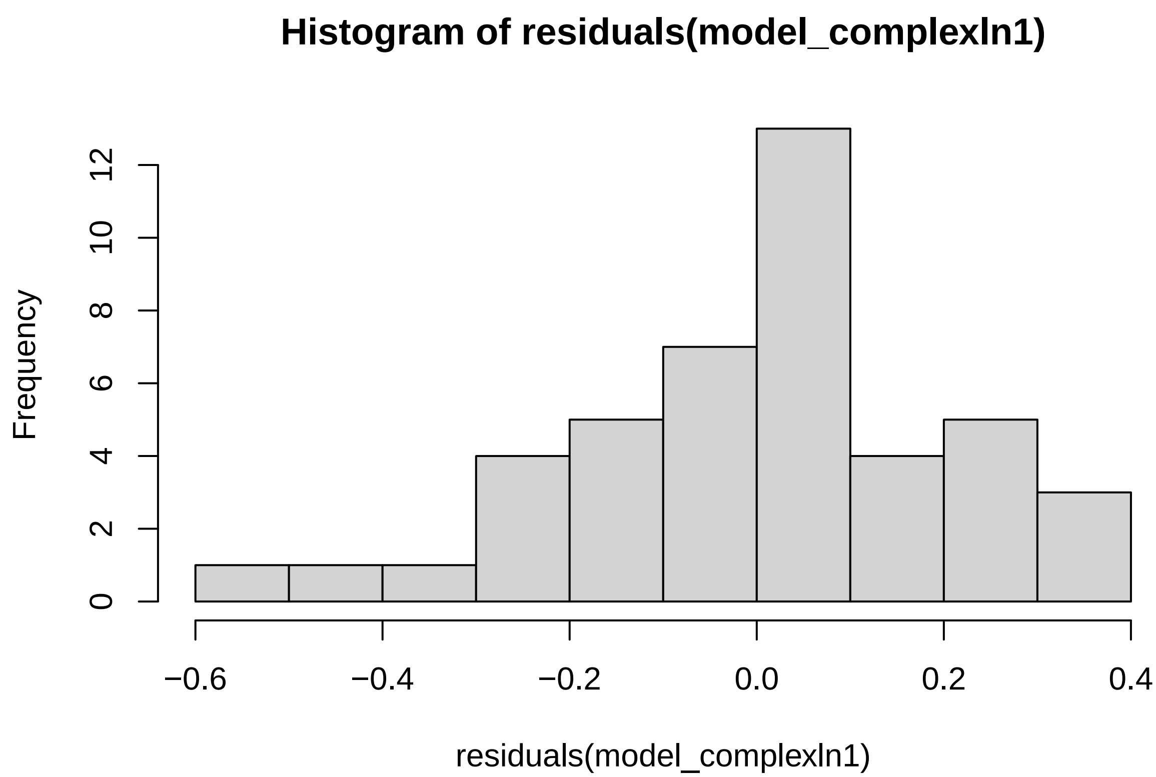

```
par(mfrow=c(1,1))
```

```
# Calculate influence measures
n <- nrow(movies)
p <- length(coef(model_complexln1))
leverage <- hatvalues(model_complexln1)
StanRes <- rstandard(model_complexln1)
cd <- cooks.distance(model_complexln1)
Rstudent <- rstudent(model_complexln1)

# Cutoffs
#cutL <- 2*p/n  # Leverage cutoff
#cutCD <- 4/(n-p-1)  # Cook's Distance cutoff

# Create influence diagnostics table
Influence_table1 <- data.frame(
  Observation = 1:n,
  USRevenue = movies$USRevenue,
  Leverage = round(leverage, 3),
  StanRes = round(StanRes, 3),
  CooksD = round(cd, 3),
  Rstudent = round(Rstudent, 3)
)

print(Influence_table1)
```

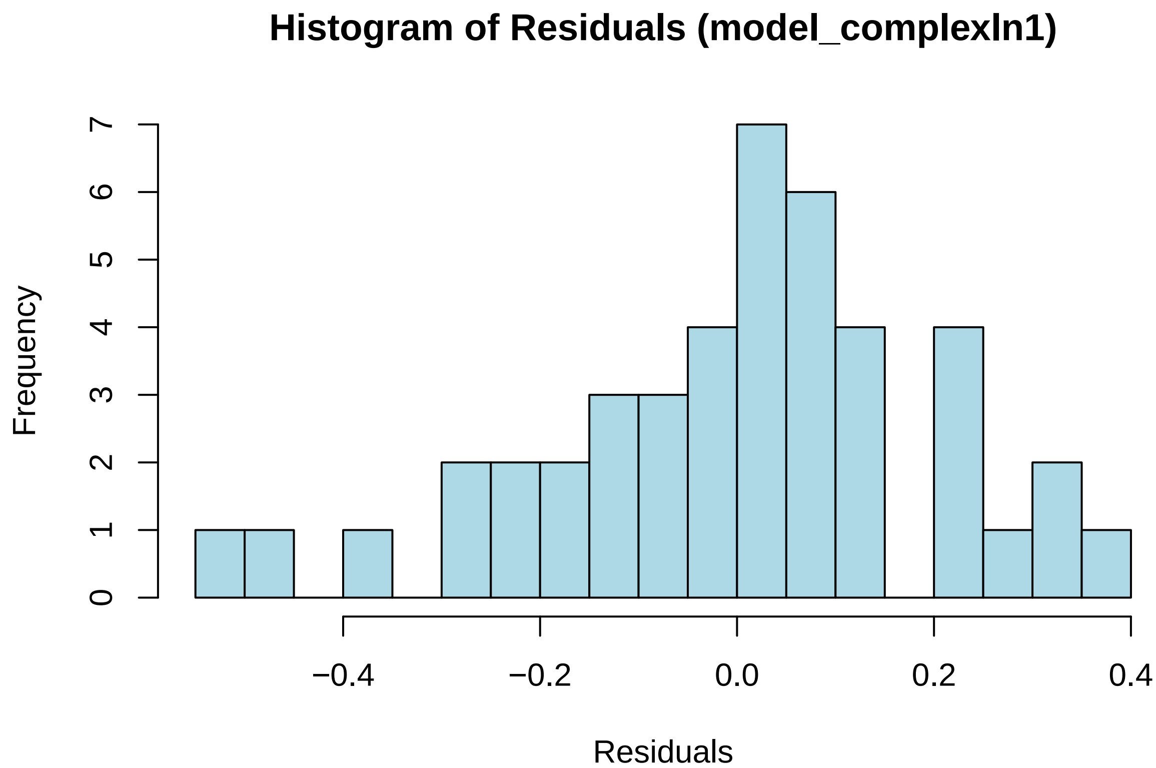
```
##    Observation USRevenue Leverage StanRes CooksD Rstudent
## 1            1     291.0    0.584   0.240  0.013    0.237
## 2            2     268.5    0.243  -0.390  0.008   -0.385
## 3            3     238.7    0.363  -3.002  0.857   -3.391
## 4            4     234.9    0.166   1.201  0.048    1.208
## 5            5     228.8    0.138   0.329  0.003    0.325
## 6            6     187.2    0.157   1.878  0.110    1.946
## 7            7     159.6    0.308   0.266  0.005    0.263
## 8            8     144.8    0.076  -0.532  0.004   -0.527
## 9            9     137.4    0.224  -0.178  0.002   -0.175
## 10          10     133.7    0.125   1.652  0.065    1.692
## 11          11     132.6    0.083  -0.190  0.001   -0.187
## 12          12     122.5    0.078   0.990  0.014    0.989
## 13          13     112.2    0.079   0.410  0.002    0.405
## 14          14     107.5    0.106   0.134  0.000    0.132
## 15          15     101.8    0.061  -0.376  0.002   -0.372
## 16          16     101.5    0.275   1.385  0.121    1.403
## 17          17      98.9    0.053   1.016  0.010    1.017
## 18          18      95.0    0.096   0.659  0.008    0.654
## 19          19      89.0    0.085  -1.275  0.025   -1.286
## 20          20      83.0    0.133   0.084  0.000    0.083
## 21          21      73.1    0.272  -1.426  0.127   -1.447
## 22          22      71.6    0.137   0.757  0.015    0.752
## 23          23      71.3    0.043   0.485  0.002    0.480
## 24          24      67.3    0.107   0.217  0.001    0.215
## 25          25      66.4    0.048   0.640  0.003    0.635
## 26          26      55.7    0.060   0.064  0.000    0.064
## 27          27      54.2    0.042  -1.005  0.007   -1.005
## 28          28      53.3    0.037   0.222  0.000    0.219
## 29          29      51.9    0.073   1.605  0.034    1.641
## 30          30      49.9    0.041  -0.946  0.006   -0.944
## 31          31      46.0    0.046  -0.652  0.003   -0.647
## 32          32      33.6    0.047   0.269  0.001    0.266
## 33          33      32.2    0.114   0.461  0.005    0.457
## 34          34      32.0    0.393  -0.367  0.015   -0.363
## 35          35      26.6    0.063  -1.953  0.043   -2.032
## 36          36      25.7    0.048  -0.197  0.000   -0.195
## 37          37      24.5    0.092   0.361  0.002    0.357
## 38          38      19.7    0.059  -1.050  0.012   -1.051
## 39          39      19.5    0.061  -2.246  0.055   -2.380
## 40          40      18.0    0.127   1.363  0.045    1.379
## 41          41      17.6    0.114   0.296  0.002    0.293
## 42          42      17.4    0.095  -0.533  0.005   -0.528
## 43          43       9.1    0.175  -0.802  0.023   -0.798
## 44          44       8.8    0.271  -0.101  0.001   -0.099
```

```r
# Show influential observations
#influential <- influence_table[leverage > cutL | cd > cutCD, ]
#if(nrow(influential) > 0) {
 # cat("\nInfluential Observations:\n")
 # print(influential)
#} else {
 # cat("\nNo influential observations detected.\n")
#}
```

```r
hist(residuals(model_complexln1))
```




```r
# Histogram of residuals
hist(residuals(model_complexln1), main="Histogram of Residuals (model_complexln1)",
     xlab="Residuals", col="lightblue", breaks=20)
```




```r
anova(model36,model_complexln1)
```

```
## Analysis of Variance Table
##
## Model 1: USRevenueln ~ Opening + Ratings + Opening * Theaters
## Model 2: USRevenueln ~ Opening + Ratings + jun + Opening * Theaters
##   Res.Df    RSS Df Sum of Sq      F   Pr(>F)
## 1     39 2.0888
## 2     38 1.7007  1   0.38817 8.6734 0.005485 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(residuals(model36))
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.62700 -0.11535  0.03111  0.00000  0.11977  0.55516
```

```r
summary(residuals(model_complexln1))
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -0.50671 -0.10755  0.02972  0.00000  0.10833  0.36477
```

```r
anova(model36)
```

```
## Analysis of Variance Table
##
## Response: USRevenueln
##                   Df  Sum Sq Mean Sq F value                Pr(>F)
## Opening            1 25.4423 25.4423 475.028 < 0.00000000000000022 ***
## Ratings            1  1.5559  1.5559  29.051        0.0000036368916 ***
## Theaters           1  3.6760  3.6760  68.634        0.0000000003965 ***
## Opening:Theaters   1  2.6470  2.6470  49.421        0.0000000194928 ***
## Residuals         39  2.0888  0.0536
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
anova(model_complexln1)
```

```
## Analysis of Variance Table
##
## Response: USRevenueln
##                   Df  Sum Sq Mean Sq  F value                Pr(>F)
## Opening            1 25.4423 25.4423 568.4918 < 0.00000000000000022 ***
## Ratings            1  1.5559  1.5559  34.7666      0.00000078990841 ***
## jun                1  0.0512  0.0512   1.1451                0.2913
## Theaters           1  3.7571  3.7571  83.9496      0.00000000003664 ***
## Opening:Theaters   1  2.9028  2.9028  64.8615      0.00000000096885 ***
## Residuals         38  1.7007  0.0448
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
beforeopening <- lm(USRevenue ~ Budget + Theaters + Ratings +c+ Budget*Theaters, data=movies)
summary(beforeopening)
```

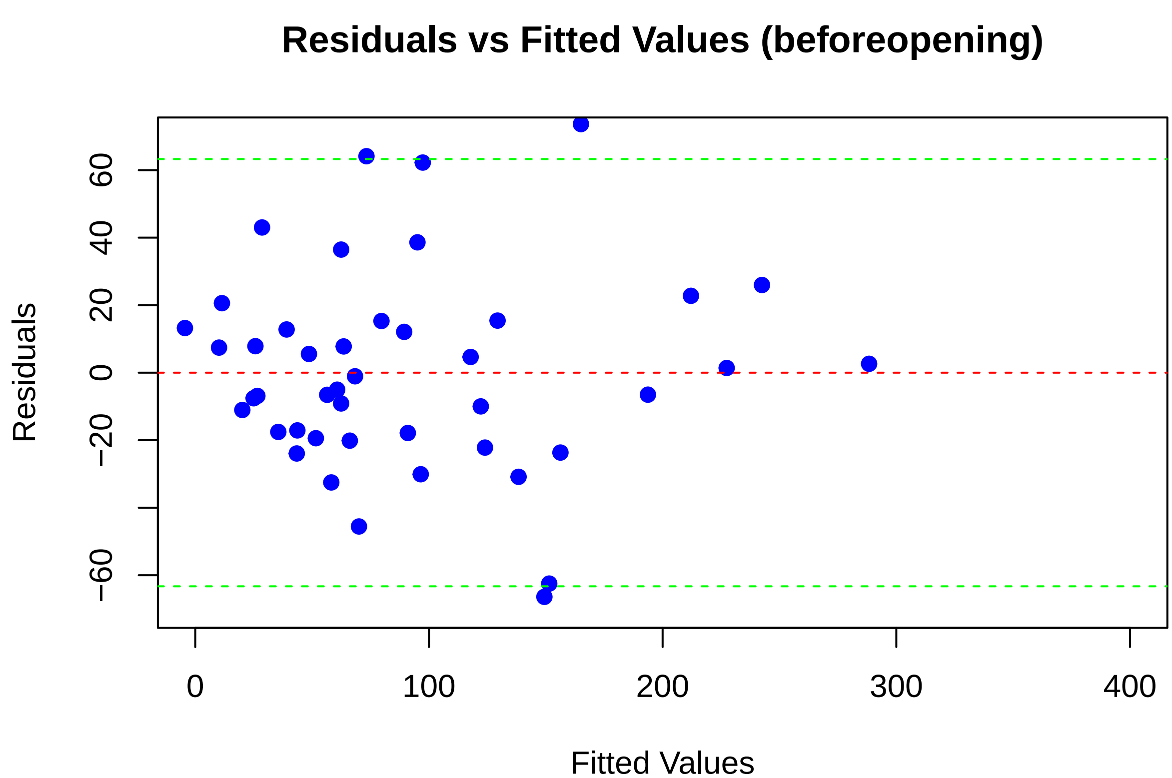
```
##
## Call:
## lm(formula = USRevenue ~ Budget + Theaters + Ratings + c + Budget *
##     Theaters, data = movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -66.410 -18.252  -3.053  13.746  73.645
##
## Coefficients:
##                     Estimate   Std. Error t value Pr(>|t|)
## (Intercept)     -196.8167555   64.1426330  -3.068 0.003957 **
## Budget            -1.6839562    0.6028927  -2.793 0.008130 **
## Theaters           0.0199483    0.0151473   1.317 0.195740
## Ratings           28.0280855    7.0980444   3.949 0.000329 ***
## c                 34.1673203   13.5828291   2.515 0.016234 *
## Budget:Theaters    0.0006140    0.0001621   3.786 0.000529 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 31.64 on 38 degrees of freedom
## Multiple R-squared:  0.836,  Adjusted R-squared:  0.8145
## F-statistic: 38.75 on 5 and 38 DF,  p-value: 0.000000000000064
```

```r
vif(beforeopening, type="predictor")
```

```
## GVIFs computed for predictors

##              GVIF Df GVIF^(1/(2*Df)) Interacts With          Other Predictors
## Budget   1.313948  3        1.046557       Theaters                Ratings, c
## Theaters 1.313948  3        1.046557         Budget                Ratings, c
## Ratings  1.418258  1        1.190906             --      Budget, Theaters, c
## c        1.520153  1        1.232945             --  Budget, Theaters, Ratings
```

```r
# Residuals vs Fitted Values
plot(x=fitted(beforeopening), y=residuals(beforeopening),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (beforeopening)", xlim=c(0, 400), ylim=c(-70, 70),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sigmal <- summary(beforeopening)$sigma
abline(h=2*sigmal, lty="dashed", col="green")
abline(h=-2*sigmal, lty="dashed", col="green")
```

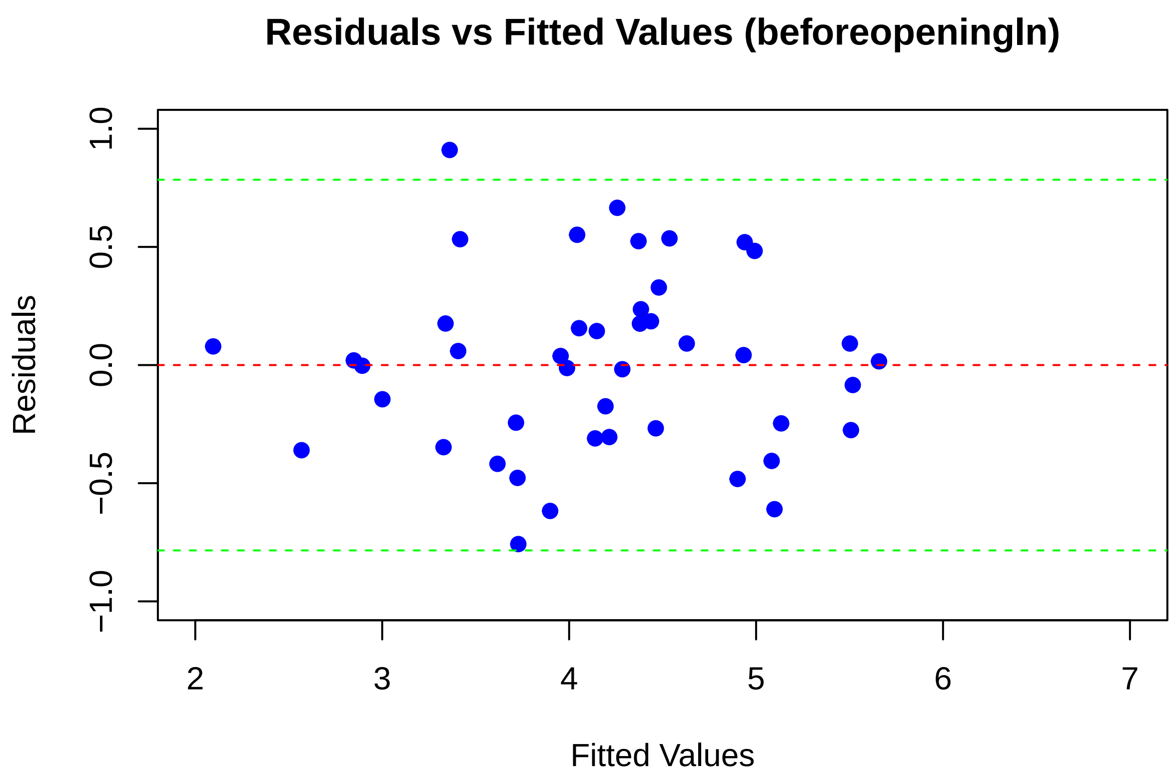


#very strong heteroscacitity (multiplicative errors)

```r
beforeopeningln <- lm(USRevenueln ~  Theaters + Ratings +c, data=movies)
summary(beforeopeningln)
```

```
##
## Call:
## lm(formula = USRevenueln ~ Theaters + Ratings + c, data = movies)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.75768 -0.28263  0.01778  0.17818  0.91025
##
## Coefficients:
##               Estimate Std. Error t value          Pr(>|t|)
## (Intercept) -2.6435564  0.6252467  -4.228          0.000133 ***
## Theaters     0.0012120  0.0001127  10.755 0.000000000000227 ***
## Ratings      0.4509917  0.0867593   5.198 0.000006283576317 ***
## c            0.3481133  0.1616249   2.154          0.037334 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3922 on 40 degrees of freedom
## Multiple R-squared:  0.8262, Adjusted R-squared:  0.8132
## F-statistic: 63.39 on 3 and 40 DF,  p-value: 0.000000000000002962
```

```r
# Residuals vs Fitted Values
plot(x=fitted(beforeopeningln), y=residuals(beforeopeningln),
     xlab = "Fitted Values", ylab = "Residuals",
     main = "Residuals vs Fitted Values (beforeopeningln)", xlim=c(2, 7), ylim=c(-1, 1),
     pch=19, col="blue")
abline(h=0, lty="dashed", col="red")

# Add +/- 2*sigma lines
sigmaln <- summary(beforeopeningln)$sigma
abline(h=2*sigmaln, lty="dashed", col="green")
abline(h=-2*sigmaln, lty="dashed", col="green")
```




```r
vif(beforeopeningln, type="predictor")
```

```
## VIFs computed for predictors
```

```
## [1] 1.177314 1.379092 1.400897
```

#AIC,BIC,R^2 for model36

```r
summary(model36)$r.squared
```

```
## [1] 0.9410104
```

```r
summary(model36)$adj.r.squared
```

```
## [1] 0.9349602
```

```r
sum(resid(model36)^2)       # RSS
```

```
## [1] 2.088827
```

```r
AIC(model36)
```

```
## [1] 2.772766
```

```r
BIC(model36)
```

```
## [1] 13.4779
```

#AIC,BIC,R^2 for model_complexln1

```r
summary(model_complexln1)$r.squared
```

```
## [1] 0.9519726
```

```r
summary(model_complexln1)$adj.r.squared
```

```
## [1] 0.9456532
```

```r
sum(resid(model_complexln1)^2)      # RSS
```

```
## [1] 1.700656
```

```r
AIC(model_complexln1)
```

```
## [1] -4.273143
```

```r
BIC(model_complexln1)
```

```
## [1] 8.216185
```

#Comparison table

```r
model_comparison <- data.frame(
  Model = c("model36", "model_complexln1"),
  R_squared = c(summary(model36)$r.squared,
                summary(model_complexln1)$r.squared),
  Adj_R_squared = c(summary(model36)$adj.r.squared,
                    summary(model_complexln1)$adj.r.squared),
  RSS = c(sum(resid(model36)^2),
          sum(resid(model_complexln1)^2)),
  AIC = c(AIC(model36), AIC(model_complexln1)),
  BIC = c(BIC(model36), BIC(model_complexln1))
)
model_comparison
```

```
##              Model R_squared Adj_R_squared      RSS       AIC       BIC
## 1          model36 0.9410104     0.9349602 2.088827  2.772766 13.477904
## 2 model_complexln1 0.9519726     0.9456532 1.700656 -4.273143  8.216185
```

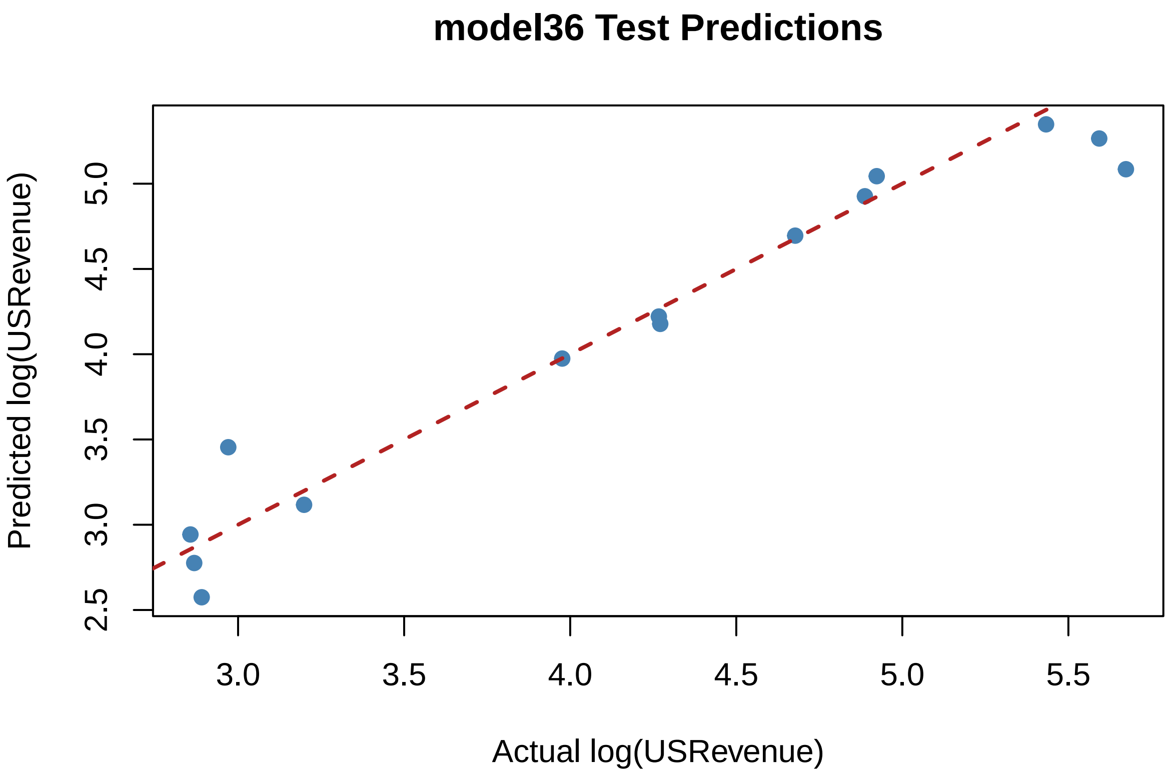
```r
## Train/Test validation
movies$USRevenueln <- log(movies$USRevenue)
summary(movies[, c("USRevenueln", "Opening", "Ratings", "Theaters")])
```

```
##   USRevenueln       Opening          Ratings         Theaters
##  Min.   :2.175   Min.   :  4.60   Min.   :3.500   Min.   :2023
##  1st Qu.:3.470   1st Qu.: 12.18   1st Qu.:6.050   1st Qu.:2832
##  Median :4.269   Median : 23.10   Median :6.500   Median :3192
##  Mean   :4.163   Mean   : 30.69   Mean   :6.382   Mean   :3169
##  3rd Qu.:4.828   3rd Qu.: 40.75   3rd Qu.:6.900   3rd Qu.:3581
##  Max.   :5.673   Max.   :116.60   Max.   :7.700   Max.   :4207
```

```r
set.seed(4810)
n_total <- nrow(movies)
train_idx <- sample(seq_len(n_total), size = floor(0.7 * n_total))
train_data <- movies[train_idx, ]
test_data <- movies[-train_idx, ]

model36_train <- lm(USRevenueln ~ Opening + Ratings + Opening * Theaters,
                                        data = train_data)
pred_test <- predict(model36_train, newdata = test_data)

results_tt <- data.frame(
    Actual = test_data$USRevenueln,
    Predicted = pred_test
)

rmse_test <- sqrt(mean((results_tt$Actual - results_tt$Predicted)^2))
rmse_test
```

```
## [1] 0.2456602
```

```r
plot(results_tt$Actual, results_tt$Predicted,
         xlab = "Actual log(USRevenue)",
         ylab = "Predicted log(USRevenue)",
         main = "model36 Test Predictions",
         pch = 19, col = "steelblue")
abline(0, 1, col = "firebrick", lwd = 2, lty = 2)
```



model36 Test Predictions
Predicted log(USRevenue)
2.5
3.0
3.5
4.0
4.5
5.0
3.0
3.5
4.0
4.5
5.0
5.5
Actual log(USRevenue)