Bivariate & Multivariate Analysis

Tomasz Wierciński & Grzegorz Meller

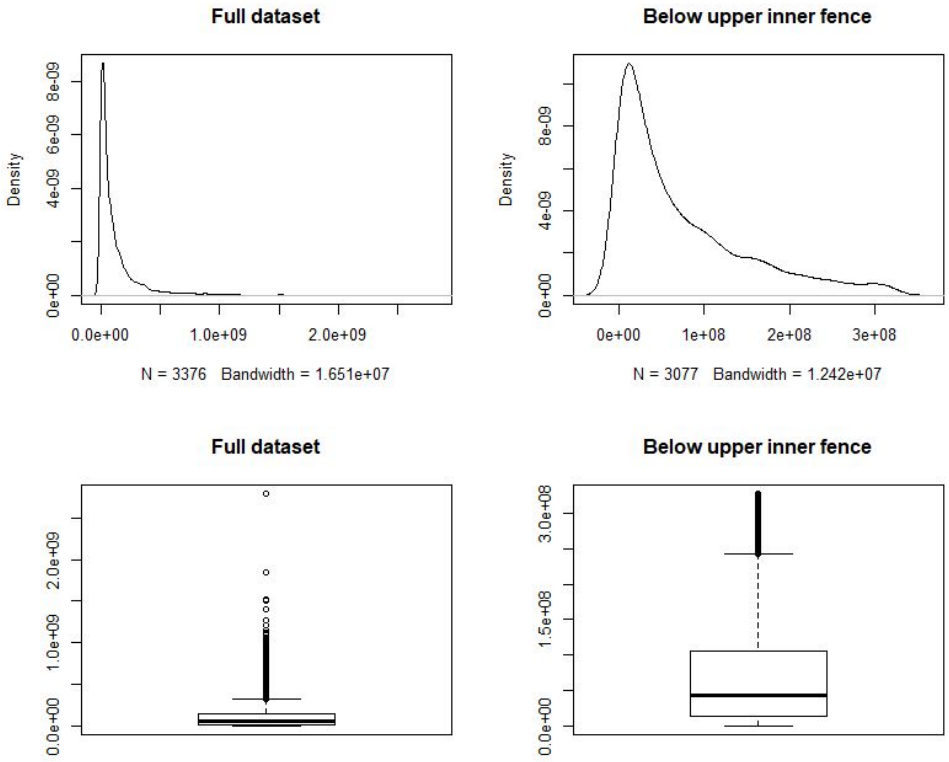
Dataset used:

[TMDB 5000 Movie Dataset]

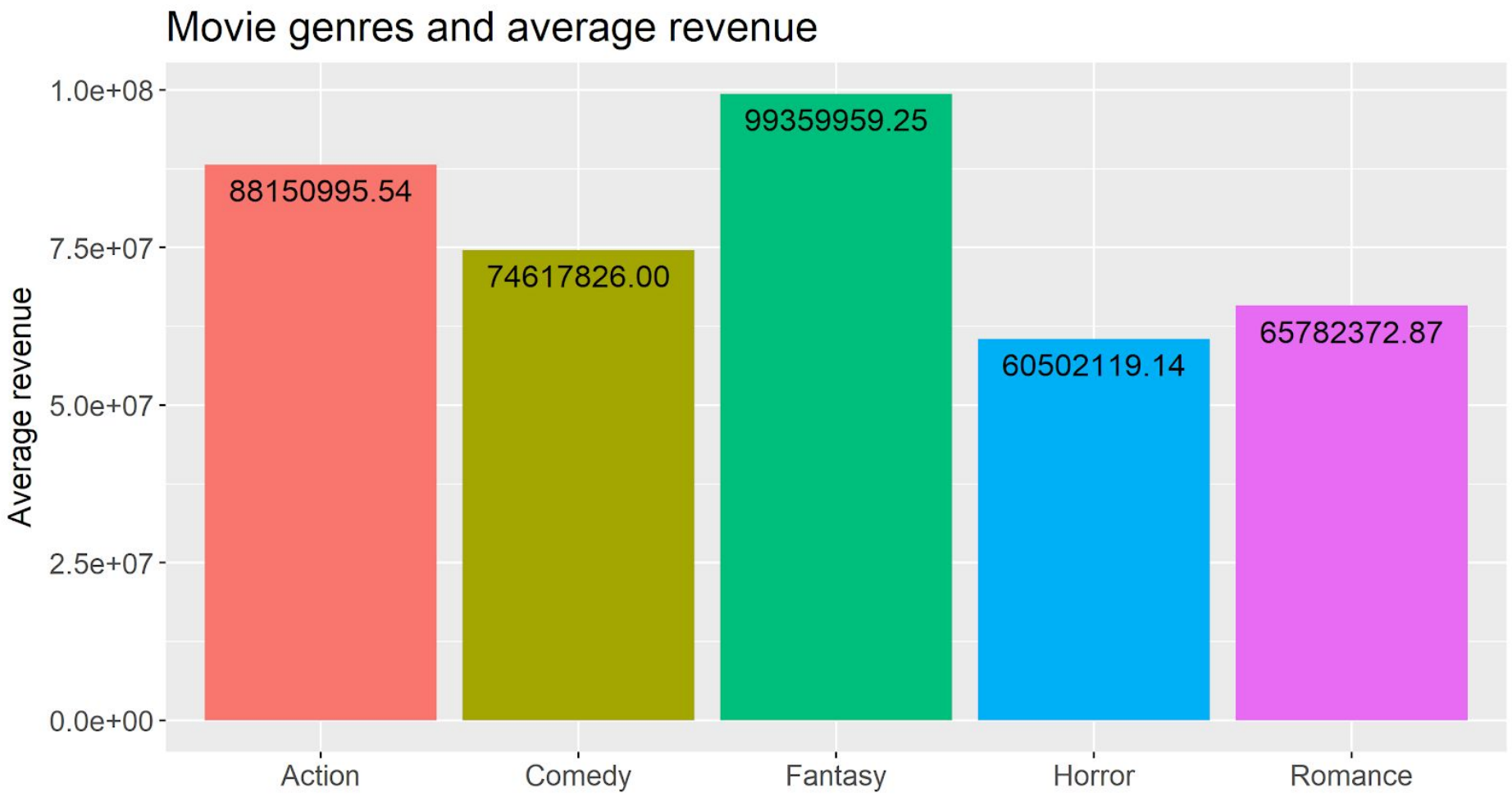
The dataset contains information on 4803 movies.
Columns:

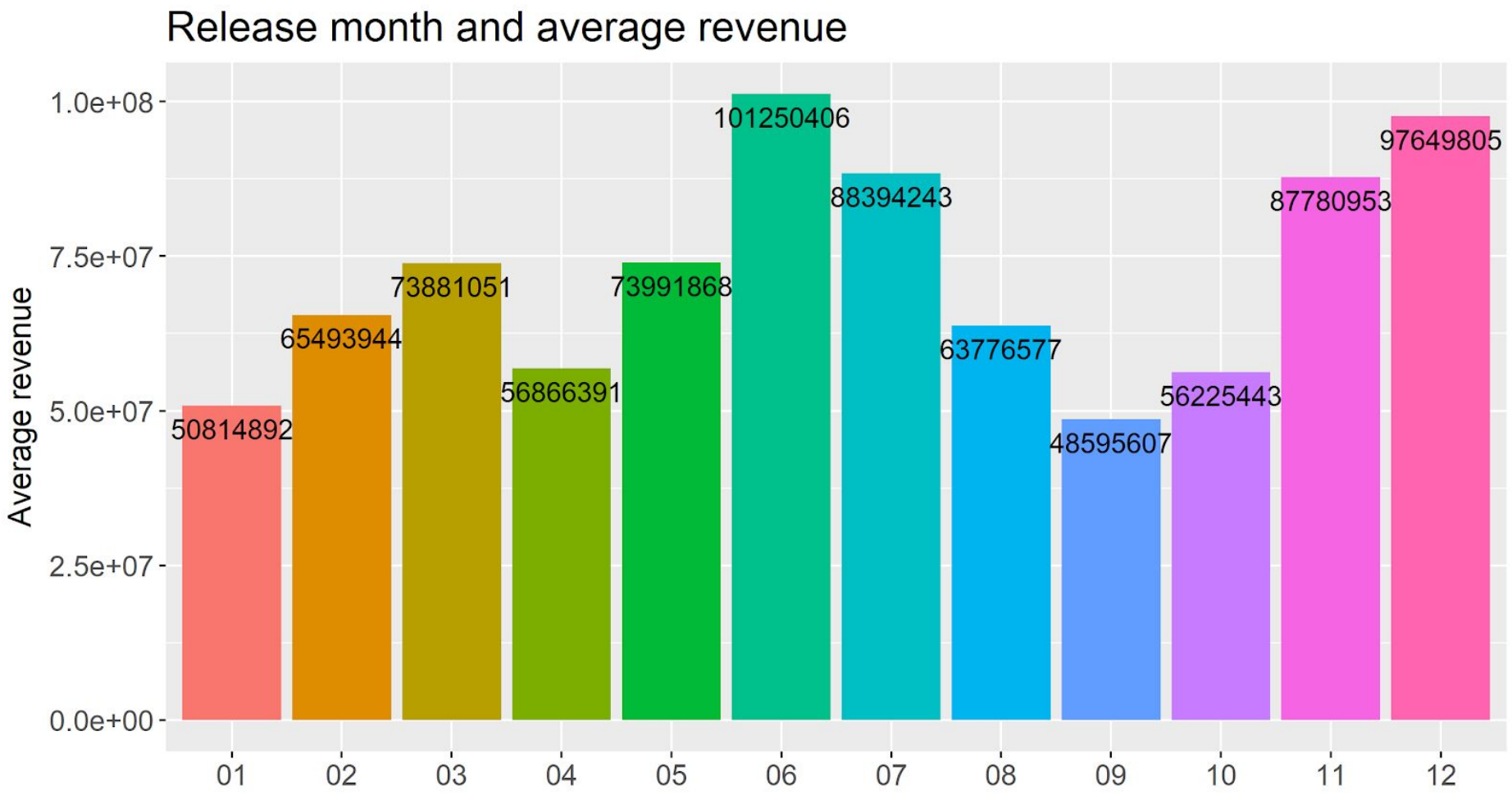
- budget
- genres
- homepage
- id
- keywords
- original_language
- original_title
- overview
- popularity
- production_companies
- production_countries
- release_date
- revenue
- runtime
- spoken_languages
- status
- tagline
- title
- vote_average
- vote_count

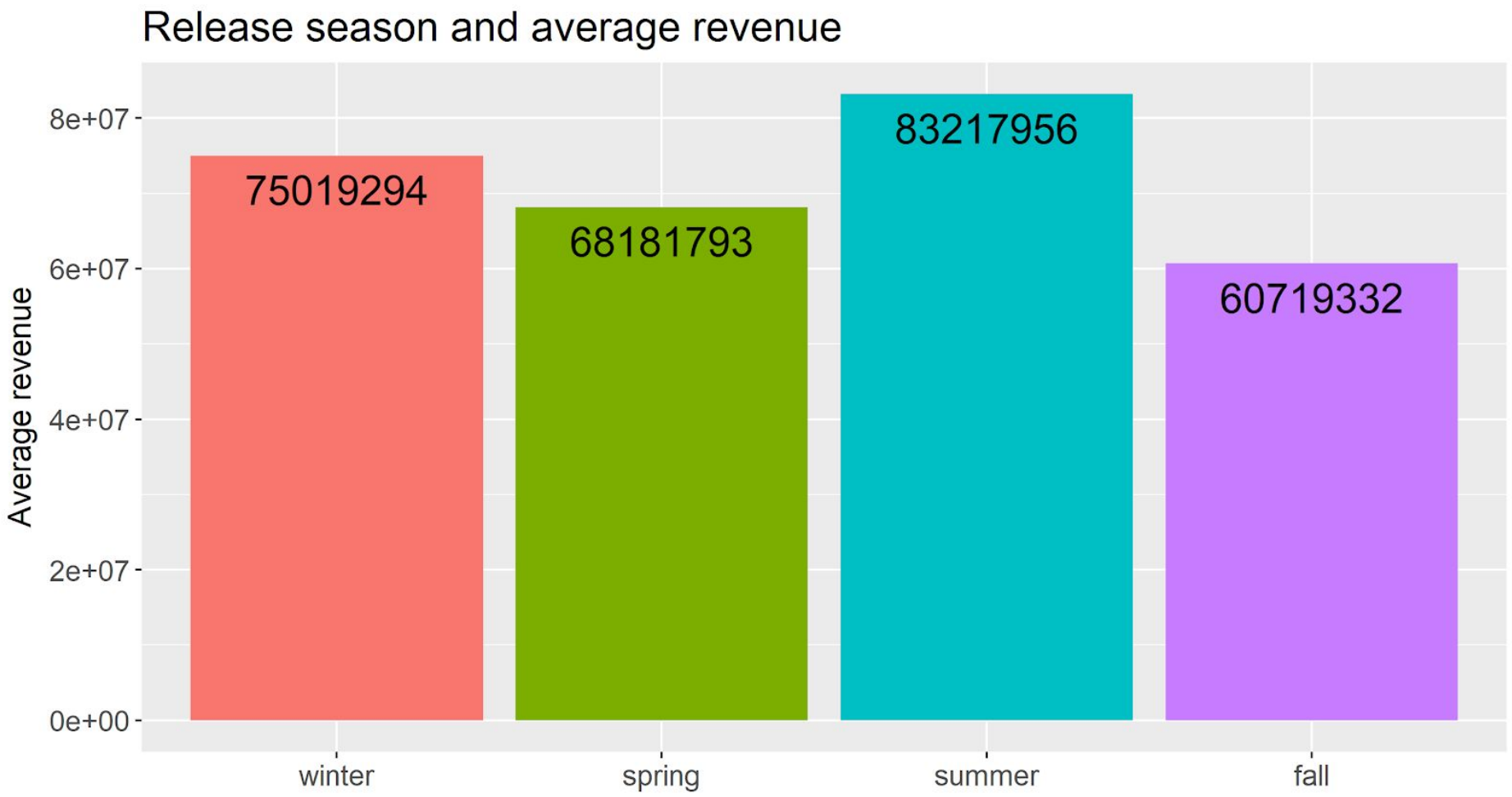
Outliers:

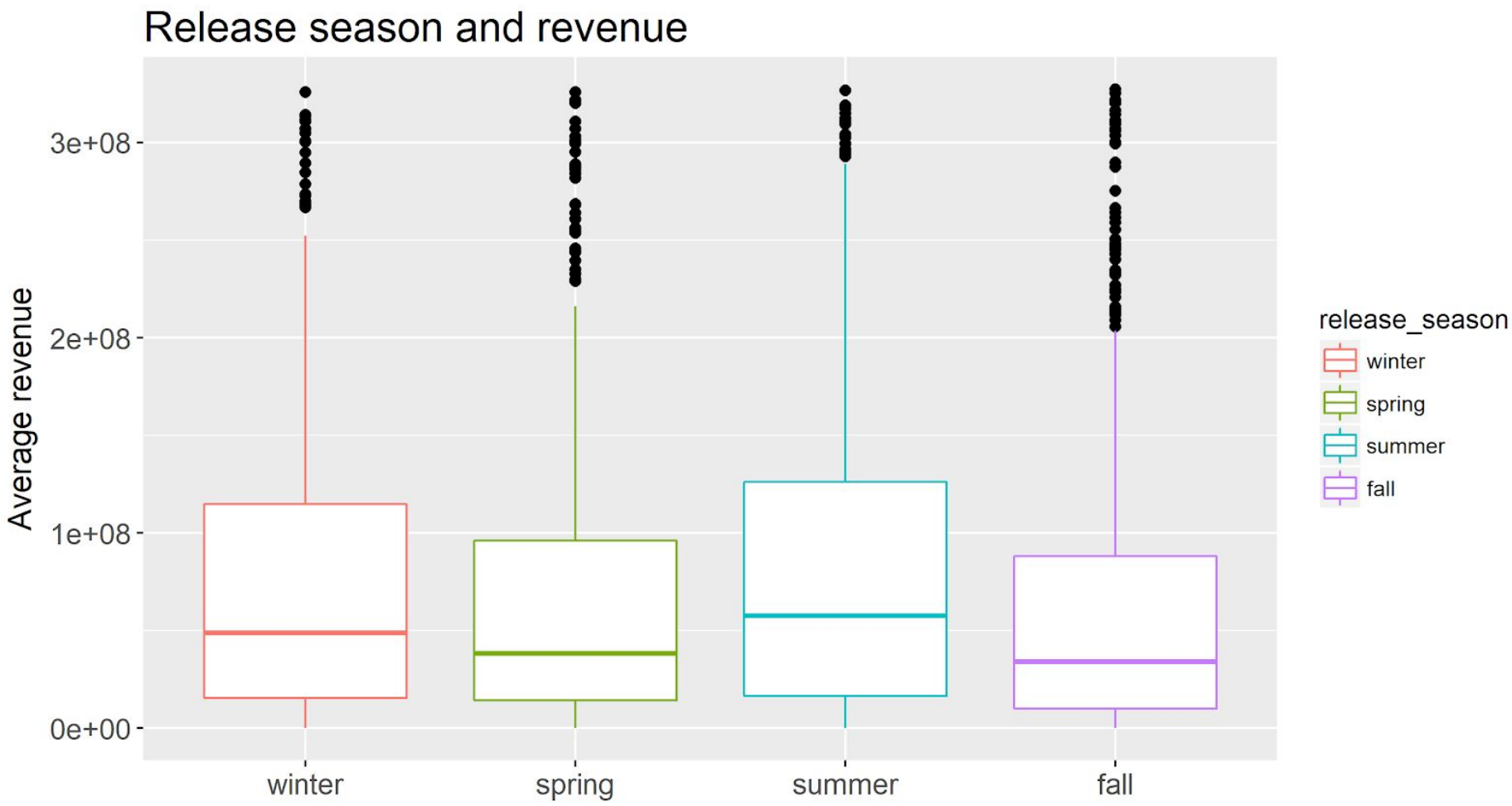


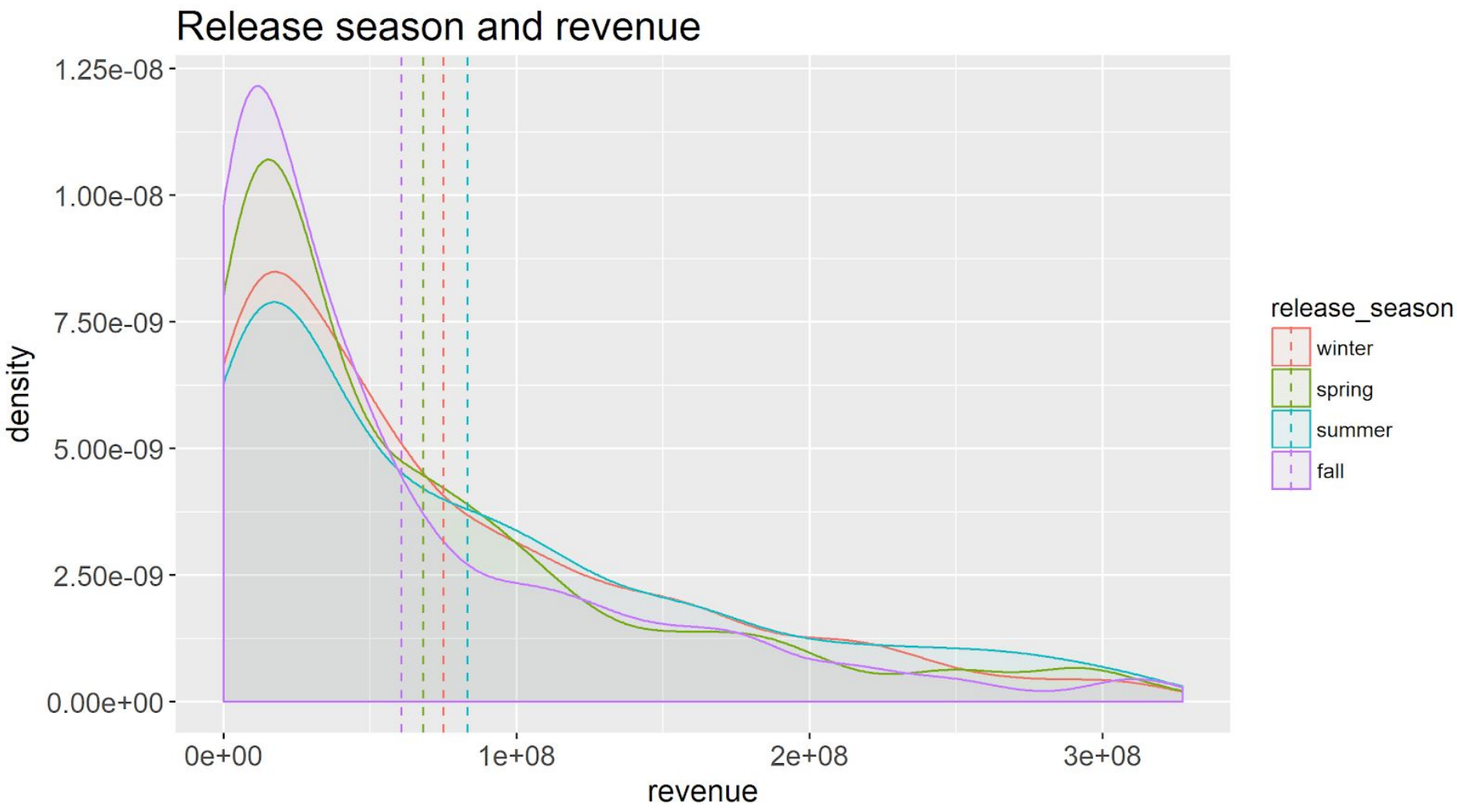
The removal of the outliers was done using a simple step by step approach with quartiles. The upper inner fence was calculated using the formula Q_3+1.5×IQ and was equal **327,383,396**. All values above were then removed.


Movie genres and average revenue
Average revenue
1.0e+08
7.5e+07
5.0e+07
2.5e+07
0.0e+00
88150995.54
74617826.00
99359959.25
60502119.14
65782372.87
Action
Comedy
Fantasy
Horror
Romance

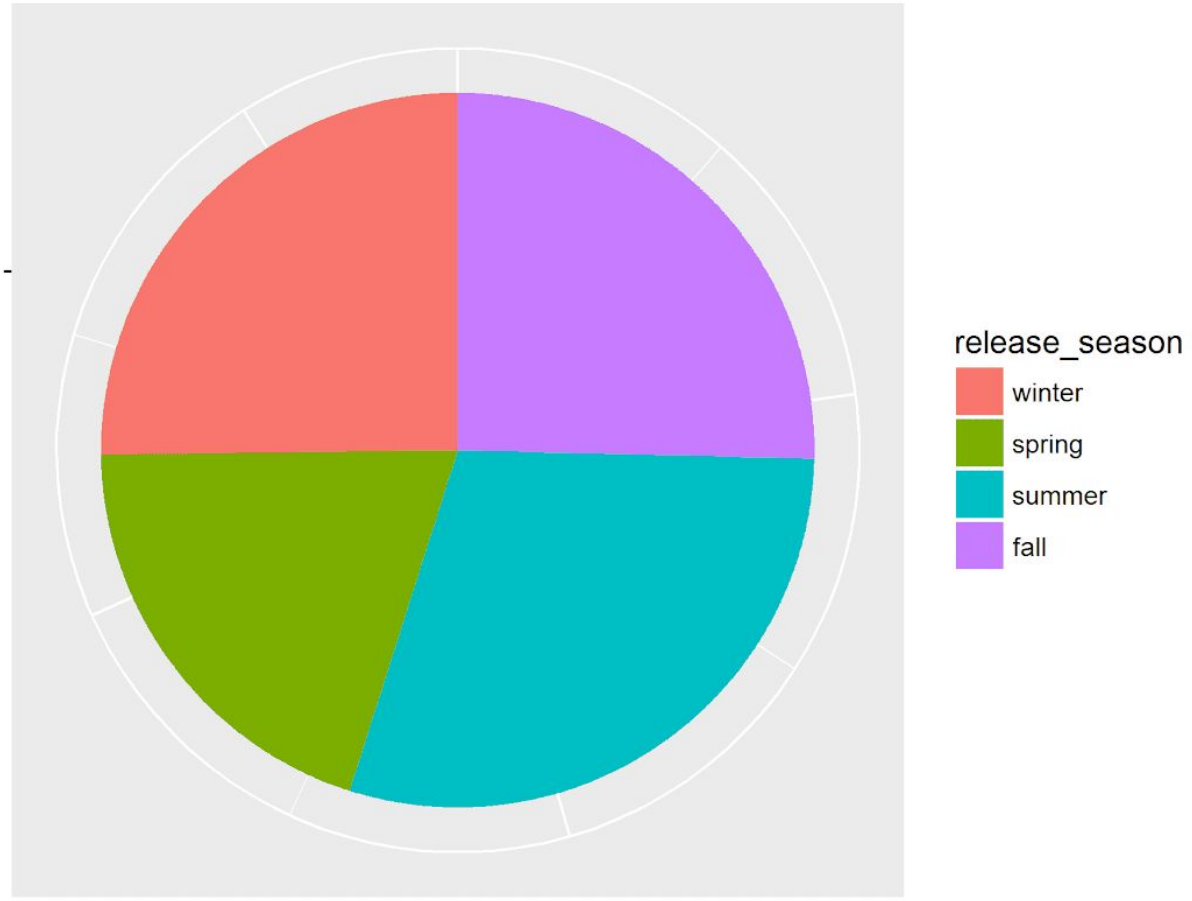

Release month and average revenue
Average revenue
1.0e+08
7.5e+07
5.0e+07
2.5e+07
0.0e+00
50814892
65493944
73881051
56866391
73991868
101250406
88394243
63776577
48595607
56225443
87780953
97649805
01
02
03
04
05
06
07
08
09
10
11
12


Release season and average revenue
Average revenue
8e+07
6e+07
4e+07
2e+07
0e+00
75019294
68181793
83217956
60719332
winter
spring
summer
fall


Release season and revenue
Average revenue
3e+08
2e+08
1e+08
0e+00
winter
spring
summer
fall
release_season
winter
spring
summer
fall


Release season and revenue
1.25e-08
1.00e-08
7.50e-09
5.00e-09
2.50e-09
0.00e+00
density
0e+00
1e+08
2e+08
3e+08
revenue
release_season
winter
spring
summer
fall

Release season

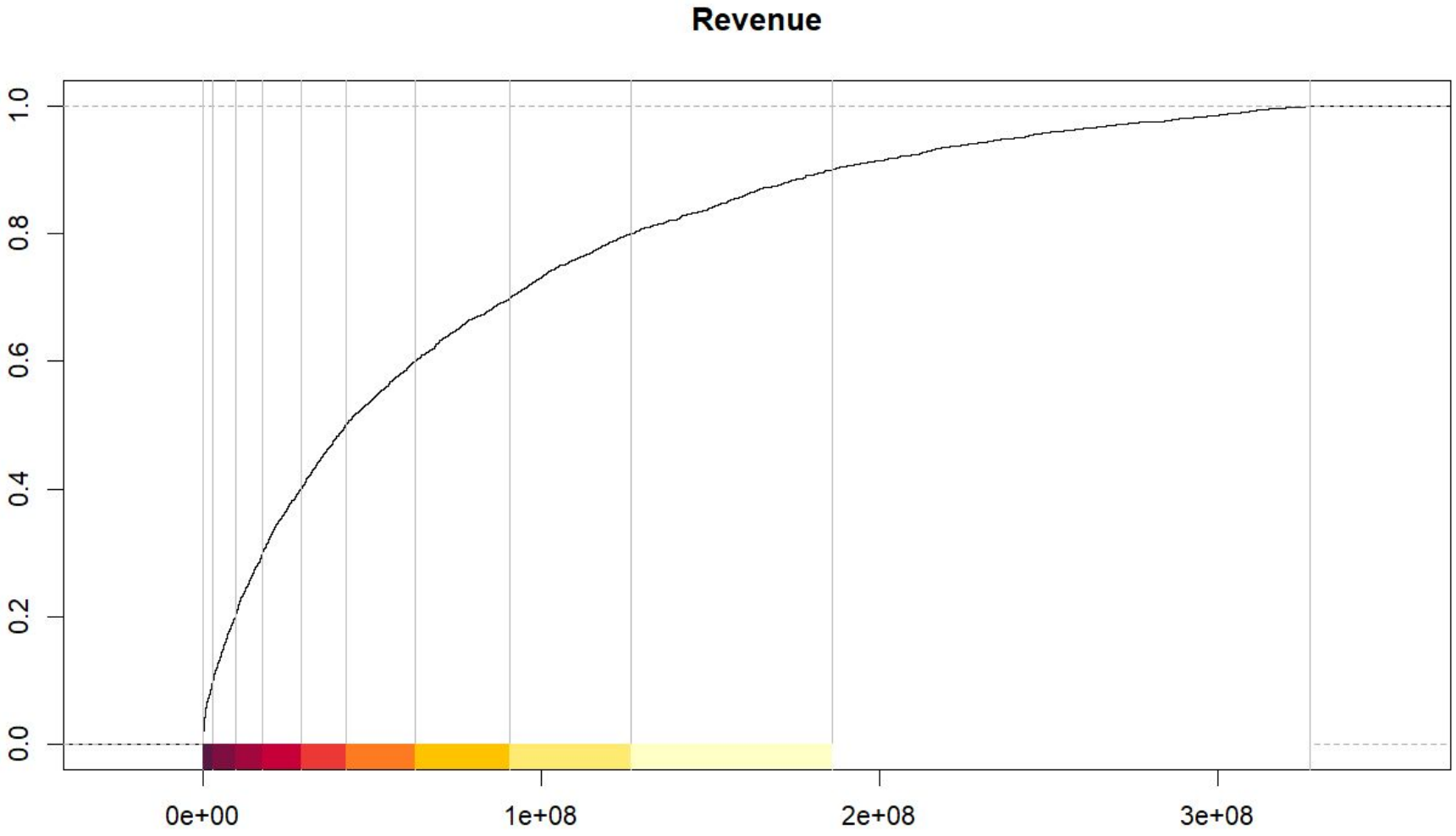


Class intervals:

Interval	[5; 2,945,457)	[2,945,457; 9,490,821)	[9,490,821; 17,560,093)	[17,560,093; 28,983,505)	[28,983,505; 42,064,105)	[42,064,105; 62,634,716)	[62,634,716; 90,709,311)	[90,709,311; 126,486,977)	[126,486,977; 185,858,754)	[185,858,754; 327,311,859]
Revenue	308	308	307	308	307	308	308	307	308	308
Fantasy	14	13	19	22	23	28	23	36	44	44
Romance	58	69	65	58	67	54	48	47	58	49
Horror	34	26	35	36	47	38	42	34	30	18
Action	50	63	84	62	54	83	90	89	96	117
Comedy	96	123	98	106	115	101	111	109	116	119

Tabular accuracy and goodness of fit:

Revenue: TAI - 0.8641989, Goodness of fit - 0.9619071
Fantasy: TAI - 0.8391908, Goodness of fit - 0.9542124
Romance: TAI - 0.8659349, Goodness of fit - 0.9621114
Horror: TAI - 0.8662442, Goodness of fit - 0.9636997
Action: TAI - 0.8512592, Goodness of fit - 0.9562399
Comedy: TAI - 0.8673474, Goodness of fit - 0.9635906


Revenue
0.0
0.2
0.4
0.6
0.8
1.0
0e+00
1e+08
2e+08
3e+08

Descriptives:

The highest and lowest values for each statistic are marked with green and red respectively
Data compiled using values below the **upper inner fence** unless otherwise specified in the **Notes** section

	Quantiles				
Variable	1st quartile	3rd quartile	1st decile	9th decile	65th percentile
Revenue	13,129,846	105,316,267	13,129,846	185,858,754	75,293,435
Fantasy	28,445,578	151,531,385	8,994,590	228,942,951	115,749,297
Romance	11,110,975	97,594,140	2,945,314	175,667,866	62,939,977
Horror	14,963,014	88,657,773	3,184,005	152,958,504	64,205,893
Action	17,727,130	134,289,033	5,946,927	216,213,165	99,984,589
Comedy	13,880,551	111,666,350	3,850,367	192,167,710	79,988,963
Notes					
Fantasy appears to have highest values out of all the genres, which means that this genre has a much higher percentage of movies with high revenue.					

	Central tendency				
Variable	Arithmetic mean	Trimmed mean	Winsorized mean	Mode	Median
Revenue	117031353	88762034	91984697	7e+06	51751835
Fantasy	233567521	200544643	210817406	1e+08	122489822
Romance	88811486	69874344	71036718	2.5e+07	38263454
Horror	65697368	55992621	57578994	1.4e+07	40458352
Action	173361611	138581243	141861001	1e+08	85490608
Comedy	104566029	84432553	86607106	7e+06	52502452
Notes					
There is quite a significant difference between the different genres and the revenue. Out of all the genres fantasy has the highest statistics which makes sense, since **40%** of the top 100 highest grossing films are fantasy films, making up approximately **40.16383%** of their total revenue. Below you can see how the statistics change once all values above the **upper inner fence** are removed (above 327383396)					
Revenue	71406929	63779729	65622345	7e+06	42064105
Fantasy	99359959	94005542	95531664	1e+08	80025159
Romance	65782373	57862462	59824905	2.5e+07	35743308
Horror	60502119	54180609	56001754	1.4e+07	40191661
Action	88150996	81523088	83273376	1e+08	61673700
Comedy	74617826	67339871	69026982	7e+06	44176209

	Variation									
Variable	Range	Interquartile range	Variance	Standard deviation	Winsorized standard deviation	Coefficient of variation	Interquartile deviation	Interquartile coefficient of variation	Median absolute deviation	Mean absolute deviation
Revenue	327,311,854 (5 - 327311859)	92,186,421	5.755377e+15	75,864,203	61,638,565	106.2421	46,093,211	109.5785	55,748,557	59,937,112
Fantasy	318,502,907 (16 - 318502923)	123,085,807	7.090752e+15	84,206,606	74,451,599	84.74903	61,542,904	76.90444	68,265,390	69,423,651
Romance	326,551,087 (7 - 326551094)	86,483,165	5.464686e+15	73,923,518	58,875,991	112.3759	43,241,583	120.9781	52,456,906	57,657,996
Horror	320,170,003 (5 - 320170008)	73,694,759	3.75759e+15	61,299,187	48,949,241	101.3174	36,847,380	91.67917	44,237,405	47,004,395
Action	325,771,419 (5 - 325771424)	116,561,904	6.949099e+15	83,361,254	71,712,776	94.56643	58,280,952	94.49887	65,150,043	67,957,033
Comedy	326,551,087 (7 - 326551094)	97,785,799	6.030264e+15	77,654,773	64,081,921	104.07	48,892,900	110.677	58,027,466	62,084,237

Notes

Standard deviation is quite large for all values, which makes sense since we are dealing with numbers on a large range. The difference between the highest and the lowest coefficient of variation as well as standard deviation isn't as large as one might think which means that the dispersion of data among all samples is quite similar.

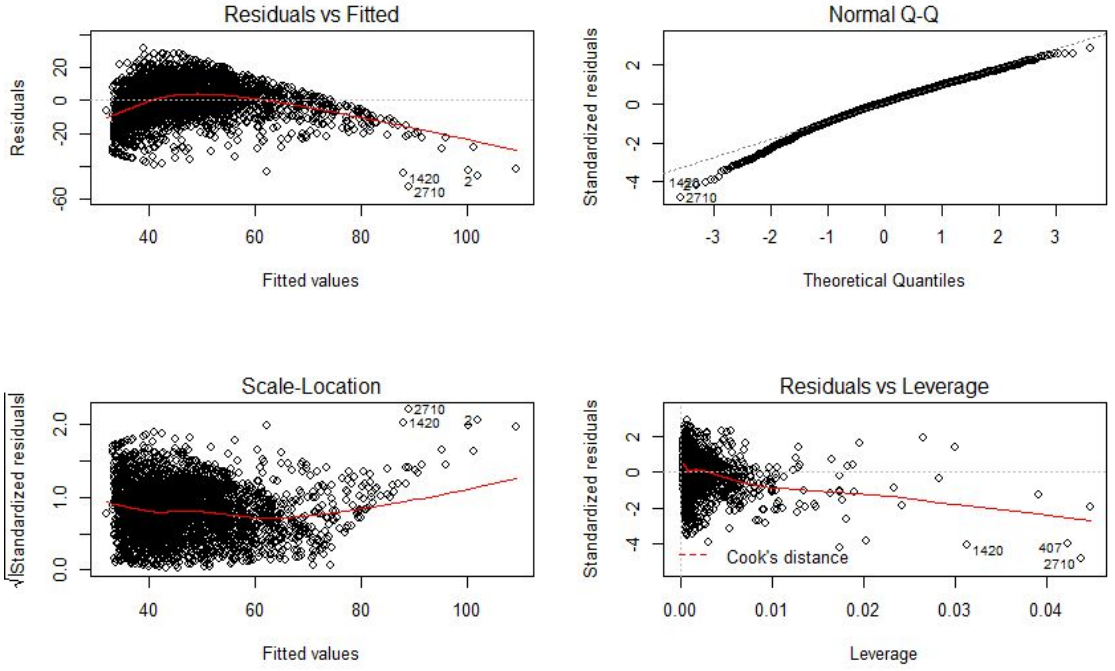
	Skewness and Kurtosis				
Variable	Pearson's skewness	Third moment skewness	Fourth moment kurtosis	Interquartile skewness	Interquartile kurtosis
Revenue	0.3867809	1.351088	1.145248	0.3722664	0.2519949
Fantasy	0.2296114	0.807973	-0.3217124	0.1618923	0.2798061
Romance	0.4063533	1.488832	1.626818	0.4303554	0.2503528
Horror	0.3313332	1.521534	2.348022	0.3153205	0.246019
Action	0.3176211	1.019757	0.1207455	0.2459531	0.2771769
Comedy	0.3920122	1.258547	0.7939823	0.3803669	0.2596304
Notes					
The revenue is skewed toward positive values. Kurtosis differs quite a lot across the samples. Romance, Horror and Comedy are leptokurtic while Fantasy is platykurtic. Action with a kurtosis very close to 0 could be considered mesokurtic.					

Regression Analysis:

Our task was to choose the best model for regression analysis for revenue of a film, so which coefficients affect the most, revenue of the movie. For this we used “step” command in R, which compared AIC to select the best model. After many comparisons we finally have chosen following coefficients for further regression analysis: budget, runtime, popularity and vote count. Before doing regression analysis we needed to transform revenue dataset, to find maximum likelihood-like approach of Box and Cox and also by rising it to the power of 0.22 (Box-Cox transformation method).

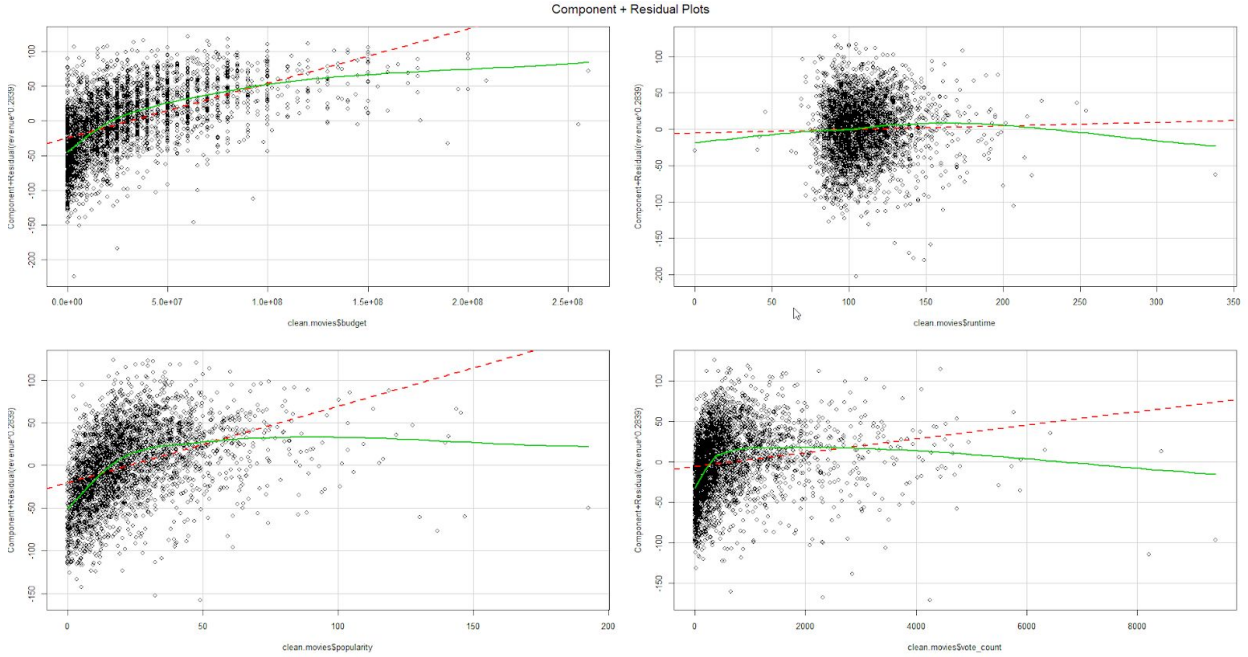
Variance inflation factor (VIF) : 1.859828
Akaike information criterion (AIC) : 23293.03

Diagnostic Plots for Linear Regression Analysis:



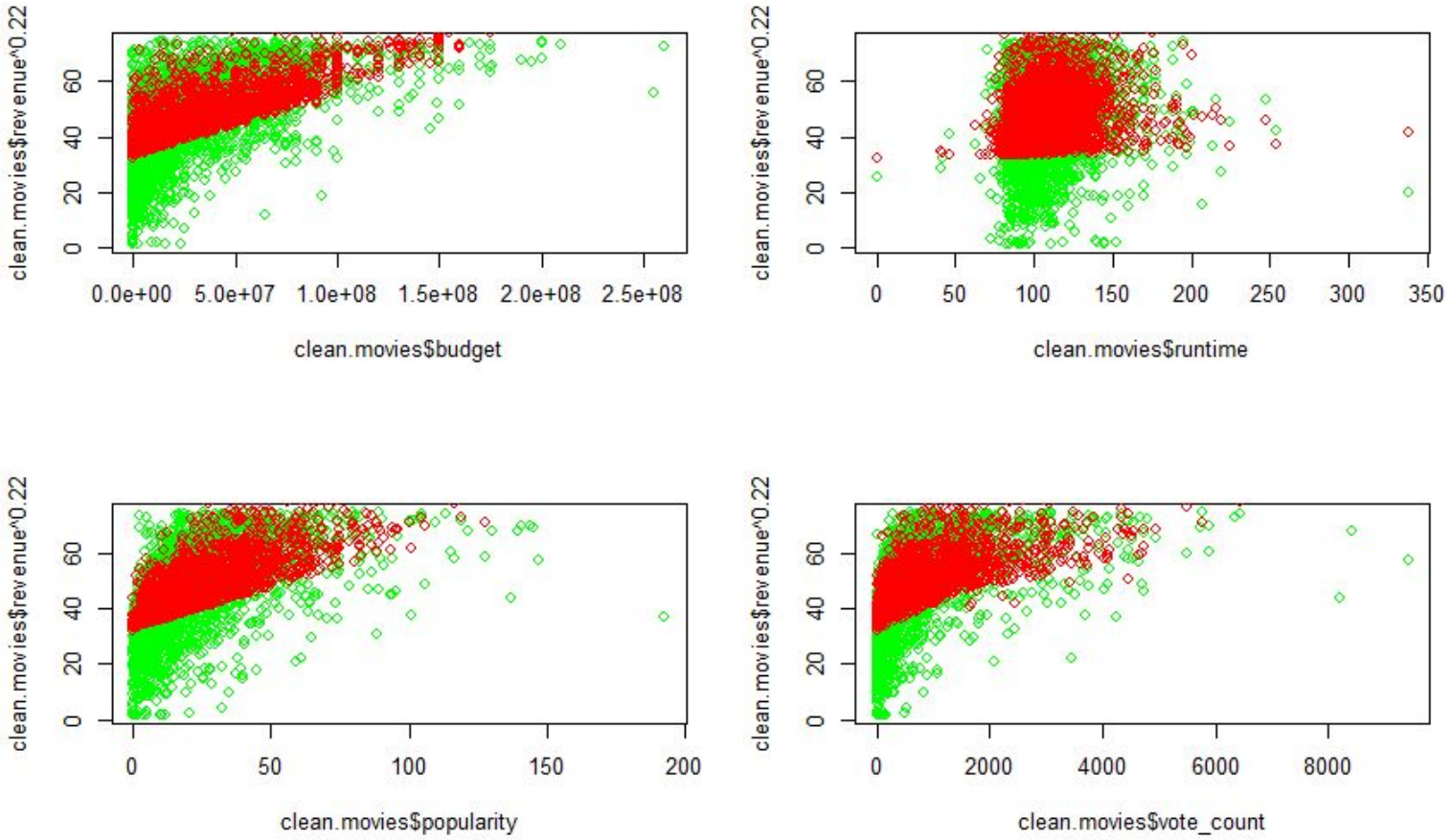
1. Residuals vs Fitted - We clearly see linear relationship between Fitted values and Residuals. So revenue of a movie is dependent form its budget, runtime, popularity and vote count.
2. Normal Q-Q - We can see that residuals in dataset are close to normal distribution.
3. Scale-Location - red line is close to horizontal. Analysed dataset is almost homoscedastic.
4. Residuals vs Leverage - We can barely see Cook's distance lines (a red dashed line) because all cases are well inside of the Cook's distance lines. So outliers are not affecting our model.

Component + Residual Plots:



We clearly see strong relationship between independent and response variable, but it's not perfectly linear, due to the influence of many other variables.

Plots presenting real data (green dots) and prediction done on our regression model (red dots):



Predict function shows tendency of our model, and comparing to real data, we clearly see that it is well fitted. For instance predict, shows that bigger budget is higher revenue, the same with popularity or vote count. So the model we have chosen during regression analysis shows well, tendency of the dataset.

Code:

```
#reading data
raw.movies <- readr::read_csv("data/tmdb_5000_movies.csv")
clean.movies <- raw.movies[raw.movies$revenue != '0',]

#remove variables
rm(raw.movies)

#top grossing films
other.top <- clean.movies[order(clean.movies$revenue, decreasing = T),c(2, 7, 13)]
other.top <- cbind(spot=1:100, other.top[1:100,])
View(other.top)

#top grossing fantasy films
other.top.fantasy <- other.top[grepl("Fantasy", other.top$genres),]

#percentage of top films
dim(other.top.fantasy)[1] / dim(other.top)[1] * 100

#percentage of total revenue
sum(other.top.fantasy$revenue) / sum(other.top$revenue) * 100

#remove variables
rm(other.top.fantasy, other.top)

#descriprives before removing outliers
```

```r
#defining functions
Mode <- function (x) {
  output <- unique(x)
  output[which.max(tabulate(match(x, output)))]
} #mode

IQR <- function (x) {
  return (quantile(x, 0.75 ) - quantile(x, 0.25 ))
} #interquartile range

CV <- function (x) {
  return ((sd(x) / mean(x)) * 100 )
} # coefficient of variation

Q <- function (x) {
  return (IQR(x) / 2 )
} #interquartile deviation

IQRKurtosis <- function (x) {
  return ((quantile(x, 0.75 ) - quantile(x, 0.25 )) / ( 2 * (quantile(x, 0.9 ) - quantile(x, 0.1 ))))
} #interquartile kurtosis / positional

IQRSkewness <- function (x) {
  return (((quantile(x, 0.75 ) - median(x)) - (median(x) - quantile(x, 0.25 )))/((quantile(x, 0.75 ) - median(x))
+ (median(x) - quantile(x, 0.25 ))))
} #interquartile skewness

IQRCV <- function (x) {
  return ((Q(x) / median(x)) * 100 )
```

```r
} #interquartile coefficient of variation

A <- function (x) {
  output <- 0
  for (i in x) {
      output <- output + (i - mean(x)) ^ 3
  }
  output <- output / length(x)
  return (output)
}

Pearson <- function (x) {
  return ((mean(x) - median(x)) / sd(x))
}

Skewness <- function (x) {
  output <- 0
  for (i in x)
  {
      output <- output + (i - mean(x)) ^ 3
  }
  return (output / (length(x) * sd(x) ^ 3 ))
}

Ku <- function (x) {
  output <- 0
  for (i in x)
  {
      output <- output + (i - mean(x)) ^ 4
```

```r
  }
  output <- output / (length(x) * sd(x) ^ 4 ) - 3
  return (output)
}

#main function
Descriptives <- function (x) {
  cat( "Arithmetic mean:" , mean(x), "\n" )
  cat( "Trimmed mean:" , mean(x, trim= 0.05 , na.rm= TRUE ), "\n" )
  cat( "Winsorized mean:" , psych::winsor.mean(x, trim = 0.1 , na.rm = TRUE ), "\n" )
  cat( "Mode:" , Mode(x), "\n" )
  cat( "Median:" , median(x), "\n" )
  cat( "1st quartile:" , quantile(x, 0.25 ), "\n" )
  cat( "3rd quartile:" , quantile(x, 0.75 ), "\n" )
  cat( "1st decile:" , quantile(x, 0.1 ), "\n" )
  cat( "9th decile:" , quantile(x, 0.9 ), "\n" )
  cat( "65th percentile:" , quantile(x, 0.65 ), "\n" )
  cat( "Minimum:" , min(x), "\n" )
  cat( "Maximum: " , max(x), "\n" )
  cat( "Range:" , range(x)[ 2 ] - range(x)[ 1 ], "\n" )
  cat( "Interquartile range:" , IQR(x), "\n" )
  cat( "Variance:" , var(x), "\n" )
  cat( "Standard deviation:" , sd(x, na.rm= T ), "\n" )
  cat( "Winsorized sd(10%):" , psych::winsor.sd(x, trim= 0.1 ), "\n" )
  cat( "Coefficient of variation:" , CV(x), "\n" )
  cat( "Interquartile deviation:" , Q(x), "\n" )
  cat( "Interquartile coefficient of variation:" , IQRCV(x), "\n" )
  cat( "Median absolute deviation:" , sum(abs(x-median(x)))/length(x), "\n" )
  cat( "Mean absolute deviation:" , sum(abs(x-mean(x)))/length(x), "\n" )
```

```
  cat( "Pearson's skewness:" , Pearson(x), "\n" )
  cat( "Third moment skewness:" , Skewness(x), "\n" )
  cat( "Fourth moment kurtosis:" , Ku(x), "\n" )
  cat( "Interquartile skewness:" , IQRSkewness(x), "\n" )
  cat( "Interquartile kurtosis:" , IQRKurtosis(x), "\n" )
}

Descriptives(clean.movies$revenue)
Descriptives(clean.movies$revenue[grepl("Fantasy", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Romance", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Horror", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Action", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Comedy", clean.movies$genres)])

#removing outliers
layout(mat=matrix(data=c(1, 2, 3, 4, 5, 6), nrow=2))

plot(density(clean.movies$revenue), main="Full dataset")
boxplot(clean.movies$revenue, main="Full dataset")

upperInnerFence <- as.numeric(quantile(clean.movies$revenue, 0.75)+1.5*IQR(clean.movies$revenue))
upperOuterFence <- as.numeric(quantile(clean.movies$revenue, 0.75)+3*IQR(clean.movies$revenue))

clean.movies <- clean.movies[clean.movies$revenue < upperOuterFence,]

plot(density(clean.movies$revenue), main="Below upper outer fence")
boxplot(clean.movies$revenue, main="Below upper outer fence")

clean.movies <- clean.movies[clean.movies$revenue < upperInnerFence,]
```

```r
plot(density(clean.movies$revenue), main="Below upper inner fence")
boxplot(clean.movies$revenue, main="Below upper inner fence")

lowerInnerFence <- as.numeric(quantile(clean.movies$revenue, 0.25)-1.5*IQR(clean.movies$revenue))
lowerrOuterFence <- as.numeric(quantile(clean.movies$revenue, 0.25)-3*IQR(clean.movies$revenue))

rm(lowerInnerFence, lowerrOuterFence, upperInnerFence, upperOuterFence)

#descriptives after removing outliers
Descriptives(clean.movies$revenue)
Descriptives(clean.movies$revenue[grepl("Fantasy", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Romance", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Horror", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Action", clean.movies$genres)])
Descriptives(clean.movies$revenue[grepl("Comedy", clean.movies$genres)])

#remove functions
rm(A, CV, IQR, IQRCV, IQRKurtosis, IQRSkewness, Ku, Mode, Pearson, Q, Skewness, Descriptives)

#extracting season
yq <- zoo::as.yearqtr(zoo::as.yearmon(clean.movies$release_date, "%Y-%m-%d") + 1/12)
clean.movies$release_season <- factor(format(yq, "%q"), levels = 1:4, labels = c("winter", "spring", "summer",
"fall"))
rm(yq)

#extracting release month
clean.movies$release_month <- substr(clean.movies$release_date, 6, 7)
```

```r
#shortening realease date to year
clean.movies$release_date <- substr(clean.movies$release_date, 1, 4)
names(clean.movies)[12] <- 'release_year'

#generating plots
genres <- c("Horror", "Romance", "Comedy", "Action", "Fantasy")
average_revenue <- rep(0, times=5)
for (i in 1:5)
{
  average_revenue[i] <- mean(clean.movies$revenue[grepl(genres[i], clean.movies$genres)])
}
rm(i)

layout(mat=matrix(c(1)))

#barplot - genres
result <- data.frame(genres=genres, average_revenue=average_revenue)
p <- ggplot2::ggplot(data=result, ggplot2::aes(x=genres, y=average_revenue, fill=genres)) +
  ggplot2::geom_bar(stat="identity") +
  ggplot2::geom_text(ggplot2::aes(label=sprintf("%0.2f", round(average_revenue, digits = 2))), vjust=1.6,
color="black", size=4.5) +
  ggplot2::theme(axis.text=ggplot2::element_text(size=12), axis.title=ggplot2::element_text(size=13),
                 plot.title = ggplot2::element_text(size=17)) +
  ggplot2::guides(fill=FALSE) +
  ggplot2::labs(title="Movie genres and average revenue", y="Average revenue", x="")
ggplot2::ggsave("plots/bar_genres.png", plot = p, device = "png",
      scale = 1, width = NA, height = NA, units = c("in", "cm", "mm"),
      dpi = 320, limitsize = TRUE)
```

```r
#barplot - release month
result <- plyr::ddply(clean.movies,~release_month,plyr::summarise,average_revenue=mean(revenue))
p <- ggplot2::ggplot(data=result, ggplot2::aes(x=release_month, y=average_revenue, fill=release_month)) +
  ggplot2::geom_bar(stat="identity") +
  ggplot2::geom_text(ggplot2::aes(label=sprintf("%0.0f", round(average_revenue, digits = 2))), vjust=1.6,
color="black", size=4) +
  ggplot2::theme(axis.text=ggplot2::element_text(size=12), axis.title=ggplot2::element_text(size=13),
                 plot.title = ggplot2::element_text(size=17)) +
  ggplot2::guides(fill=FALSE) +
  ggplot2::labs(title="Release month and average revenue", y="Average revenue", x="")
ggplot2::ggsave("plots/bar_month.png", plot = p, device = "png",
                 scale = 1, width = NA, height = NA, units = c("in", "cm", "mm"),
                 dpi = 320, limitsize = TRUE)

#barplot - release season
result <- plyr::ddply(clean.movies,~release_season,plyr::summarise,average_revenue=mean(revenue))
p <- ggplot2::ggplot(data=result, ggplot2::aes(x=release_season, y=average_revenue, fill=release_season)) +
  ggplot2::geom_bar(stat="identity") +
  ggplot2::geom_text(ggplot2::aes(label=sprintf("%0.0f", round(average_revenue, digits = 2))), vjust=1.6,
color="black", size=6) +
  ggplot2::theme(axis.text=ggplot2::element_text(size=12), axis.title=ggplot2::element_text(size=13),
                 plot.title = ggplot2::element_text(size=17)) +
  ggplot2::guides(fill=FALSE) +
  ggplot2::labs(title="Release season and average revenue", y="Average revenue", x="")
ggplot2::ggsave("plots/bar_season.png", plot = p, device = "png",
                 scale = 1, width = NA, height = NA, units = c("in", "cm", "mm"),
                 dpi = 320, limitsize = TRUE)

#density plot - seasons
```

```r
data <- data.frame(revenue=clean.movies$revenue, release_season=clean.movies$release_season)
p <- ggplot2::ggplot(data, ggplot2::aes(x=revenue, fill=release_season, color=release_season)) +
  ggplot2::geom_density(alpha=0.05) +
  ggplot2::labs(title="Release season and revenue", y="density", x="revenue") +
  ggplot2::theme(axis.text=ggplot2::element_text(size=12), axis.title=ggplot2::element_text(size=13), plot.title
= ggplot2::element_text(size=17)) +
  ggplot2::geom_vline(data=result, ggplot2::aes(xintercept=average_revenue, color=release_season),
linetype="dashed")
ggplot2::ggsave("plots/density.png", plot = p, device = "png",
                scale = 1, width = NA, height = NA, units = c("in", "cm", "mm"),
                dpi = 320, limitsize = TRUE)

#boxplot - seasons
p <- ggplot2::ggplot(data, ggplot2::aes(x=release_season, y=revenue, color=release_season)) +
  ggplot2::geom_boxplot(outlier.colour="black", outlier.shape=16, outlier.size=2, notch=F) +
  ggplot2::theme(axis.text=ggplot2::element_text(size=12), axis.title=ggplot2::element_text(size=13),
                plot.title = ggplot2::element_text(size=17)) +
  ggplot2::labs(title="Release season and revenue",
                y="Average revenue", x="")
ggplot2::ggsave("plots/box_seasons.png", plot = p, device = "png",
                scale = 1, width = NA, height = NA, units = c("in", "cm", "mm"),
                dpi = 320, limitsize = TRUE)

#pie chart
p <- ggplot2::ggplot(data, ggplot2::aes(x="", y=revenue, fill=release_season))+
  ggplot2::geom_bar(width = 1, stat = "identity") +
  ggplot2::coord_polar("y", start=0) +
  ggplot2::theme(axis.text.x=ggplot2::element_blank()) +
  ggplot2::labs(title="Release season", y="", x="")
```

```r
ggplot2::ggsave("plots/piechart.png", plot = p, device = "png",
                scale = 1, width = NA, height = NA, units = c("in", "cm", "mm"),
                dpi = 320, limitsize = TRUE)

#checking correlation
library(leaps)
library(Ecdat)
library(car)
scatterplot(genres, average_revenue)
cor(as.numeric(clean.movies$vote_count), clean.movies$revenue,method="pearson", use="complete.obs")

rm(genres, average_revenue, p, data, result)



#generate class intervals
cIntervals <- classInt::classIntervals(clean.movies$revenue, n=10)
cIntervals
classInt::jenks.tests(cIntervals)
intervals <- cIntervals$brks

#generate plot
color_scheme <- c("#581845", "#900C3F", "#C70039", "#FF5733", '#FFC300', '#FFFFAA', '#FFFFFF')
png('class_intervals.png', res=130, width=1300, height=800)
plot(cIntervals, pal=color_scheme, main="Revenue", xlab="", ylab="")
dev.off()
rm(color_scheme)

#class intervals for all genres
```

```r
genres <- c("Horror", "Romance", "Comedy", "Action", "Fantasy")
for (i in 1:5)
{
  cIntervals <- classInt::classIntervals(var=clean.movies$revenue[grepl(genres[i], clean.movies$genres)],
                                         style="fixed", fixedBreaks=intervals,
                                         n=(length(intervals) - 1))
  cat('~~~~~~~~~~~~~~~~', genres[i], '~~~~~~~~~~~~~~~\n')
  print(cIntervals)
  print(classInt::jenks.tests(cIntervals))
}
rm(i, cIntervals, genres, intervals)

#https://data.library.virginia.edu/diagnostic-plots/
pom<-lm(revenue~clean.movies$budget+clean.movies$runtime+clean.movies$popularity+clean.movies$vote_average+
            clean.movies$vote_count+clean.movies$original_language+clean.movies$production_countries+
            clean.movies$spoken_languages+clean.movies$release_year,data=clean.movies)
step(pom, direction="both")

#after step anlysis we select following coefficients:
library(Ecdat)
library(car)
powerTransform(clean.movies$revenue)
full<-lm(formula = revenue^0.22 ~ clean.movies$budget + clean.movies$runtime + clean.movies$popularity +
            clean.movies$vote_count , data = clean.movies)
par(mfrow=c(2,2))
plot(full)

summary(full)
AIC(full)
```

```r
library(fmsb)
VIF(full)


exp(full$coefficients)


crPlots(lm(formula = revenue^0.2839 ~ clean.movies$budget + clean.movies$runtime + clean.movies$popularity +
            clean.movies$vote_count , data = clean.movies))
qqPlot(full$residuals)
ncvTest(full)


full<-lm(formula = revenue^0.22 ~ budget + runtime + popularity +
            vote_count , data = clean.movies, na.action="na.exclude")

par(mfrow=c(2,2))
plot(clean.movies$revenue^0.22 ~ clean.movies$budget, col='green')
points(clean.movies$budget, predict(full), col='red')

plot(clean.movies$revenue^0.22 ~ clean.movies$runtime, col='green')
points(clean.movies$runtime, predict(full), col='red')

plot(clean.movies$revenue^0.22 ~ clean.movies$popularity, col='green')
points(clean.movies$popularity, predict(full), col='red')

plot(clean.movies$revenue^0.22 ~ clean.movies$vote_count, col='green')
points(clean.movies$vote_count, predict(full), col='red')
```