

**MİMAR SİNAN GÜZEL SANATLAR ÜNİVERSİTESİ**

**Fen Edebiyat Fakültesi**

**İstatistik Bölümü**

**IST304 Veri Analizi Dönem Projesi**

**Dersin Hocaları:**

**Prof. Dr. M. Aydın Erar**

**Arş. Gör. Zeynep Bal**

**Ad Soyad: Barışcan Bozkurt**

**Okul No: 20181101031**

**2021**

**Blockbuster Filmler 1977-2019**

**Veri Seti Tanımı**

Veri seti 1977 ile 2019 yılları arasında çıkmış 431 tane Blockbuster filmi içermektedir. Blockbuster kelimesinin tam bir Türkçe karşılığı bulunmamaktadır. Genelde sadece yüksek hasılat yapmış filmler için kullanılsa da, aslında yüksek hasılat yapması için tasarlanan ve üretilen filmlere de denmektedir. Ancak bu amaçla tasarlanan filmler her zaman istenilen sonuca da ulaşamayabilir. Yine de bu filmlere de Blockbuster film denir. Bu verideki amacım filmin türünün, uzunluğunun gibi değişkenlerin gişeyi etkileyip etkilemediği, gişe-izleyici beğenisi, aynı zamanda filme ayırılan bütçenin her zaman tatminkar bir gişe olarak geri dönüp dönmediğini gibi konuları araştırmak.

****

**Verinin İçeriği:**

Release year: Yayınlandığı yıl

Rank in year: Çıktığı yıl ki sıralaması

IMDB rating: IMDB puanı

MPAA rating: Amerikan

bütçesi

Length in min: Filmin dakika cinsinden uzunluğu

Domestic distributor: Filmin dağıtımcısı

Worldwide gross: Dünya çapında gişe

Domestic gross: ABD gişesi

Genre 1,2,3: İçerdiği türler

**Nicel Değişkenler:**

IMDB rating, Film budget, Length in min, Domestic gross, Worldwide gross

**Nitel Değişkenler:**

Release year, Rank in year, MPAA rating, Film title, Domestic distribütör, Genre

**Raporda kullanılacak veri setinin .csv dosyası:**



**Veri setinin linki:**

<https://www.kaggle.com/narmelan/top-ten-blockbusters-20191977?select=Blockbusters_2019-1977.csv>

Veri setinin ilk 6 gözlemi listelenmiştir.

head(blockbusters)

## # A tibble: 6 x 13  
## release\_year rank\_in\_year imdb\_rating mpaa\_rating film\_title film\_budget  
## <dbl> <dbl> <dbl> <chr> <chr> <dbl>  
## 1 2019 1 8.5 PG-13 Avengers:~ 356000000  
## 2 2019 2 7 PG The Lion ~ 260000000  
## 3 2019 3 7.2 PG Frozen II 150000000  
## 4 2019 4 7.6 PG-13 Spider-Ma~ 160000000  
## 5 2019 5 6.9 PG-13 Captain M~ 175000000  
## 6 2019 6 7.9 G Toy Story~ 200000000  
## # ... with 7 more variables: length\_in\_min <dbl>, domestic\_distributor <chr>,  
## # worldwide\_gross <dbl>, domestic\_gross <dbl>, genre\_1 <chr>, genre\_2 <chr>,  
## # genre\_3 <chr>

Verimizin özet bilgisine bakalım.

summary(blockbusters)

## release\_year rank\_in\_year imdb\_rating mpaa\_rating   
## Min. :1977 Min. : 1.0 Min. :3.700 Length:430   
## 1st Qu.:1987 1st Qu.: 3.0 1st Qu.:6.600 Class :character   
## Median :1998 Median : 5.5 Median :7.000 Mode :character   
## Mean :1998 Mean : 5.5 Mean :7.077   
## 3rd Qu.:2009 3rd Qu.: 8.0 3rd Qu.:7.675   
## Max. :2019 Max. :10.0 Max. :9.000   
## film\_title film\_budget length\_in\_min domestic\_distributor  
## Length:430 Min. : 325000 Min. : 60.0 Length:430   
## Class :character 1st Qu.: 22000000 1st Qu.:104.0 Class :character   
## Mode :character Median : 70750000 Median :119.0 Mode :character   
## Mean : 88928663 Mean :120.6   
## 3rd Qu.:150000000 3rd Qu.:134.0   
## Max. :356000000 Max. :201.0   
## worldwide\_gross domestic\_gross genre\_1 genre\_2   
## Min. :3.719e+07 Min. : 2721100 Length:430 Length:430   
## 1st Qu.:2.155e+08 1st Qu.:105684362 Class :character Class :character   
## Median :3.801e+08 Median :173697808 Mode :character Mode :character   
## Mean :4.994e+08 Mean :202099606   
## 3rd Qu.:7.317e+08 3rd Qu.:261092025   
## Max. :2.798e+09 Max. :936662225   
## genre\_3   
## Length:430   
## Class :character   
## Mode :character   
##   
##   
##

Satırda kayıp gözlemler:

rowSums(is.na(blockbusters))

## [1] 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
## [38] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0  
## [75] 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0  
## [112] 1 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0  
## [149] 0 1 0 0 0 0 1 0 1 1 0 1 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1  
## [186] 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 1 0 0 1 0 1 0 0 0 0 0 0 1 0  
## [223] 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1  
## [260] 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 1 0 1 1 1 0 0 0 0 0 1 1 1 1 0 1 1 0 0 0  
## [297] 1 1 0 1 1 1 0 1 0 0 0 0 0 1 0 0 1 0 1 1 1 0 0 1 1 0 0 0 0 0 1 0 0 0 1 0 1  
## [334] 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 1 0 1 0 0 1 0  
## [371] 1 0 1 1 1 1 1 0 0 1 1 1 0 1 1 0 0 0 1 0 0 0 1 1 1 1 1 0 1 0 0 1 1 0 1 1 0  
## [408] 0 1 1 1 0 0 1 1 0 1 0 1 1 0 1 1 1 0 0 0 1 1 0

Sütunda kayıp gözlemler:

colSums(is.na(blockbusters))

## release\_year rank\_in\_year imdb\_rating   
## 0 0 0   
## mpaa\_rating film\_title film\_budget   
## 0 0 0   
## length\_in\_min domestic\_distributor worldwide\_gross   
## 0 0 0   
## domestic\_gross genre\_1 genre\_2   
## 0 0 0   
## genre\_3   
## 105

Genre 3’de 100’den fazla NA var, bu yüzden datamızdan çıkartacağım.

blockbusters\_yeni<-blockbusters[,-13]

Elimizdeki veriden NA oluşturmak için önce yedekledim.

blockbusters\_miss<-blockbusters\_yeni

Length\_in\_min yani film uzunluklarından rastgele NA veri oluşturmak için:

aa<-sample(1:nrow(blockbusters\_miss),floor(nrow(blockbusters\_miss)\*0.03))  
blockbusters\_miss$length\_in\_min[aa]<-NA

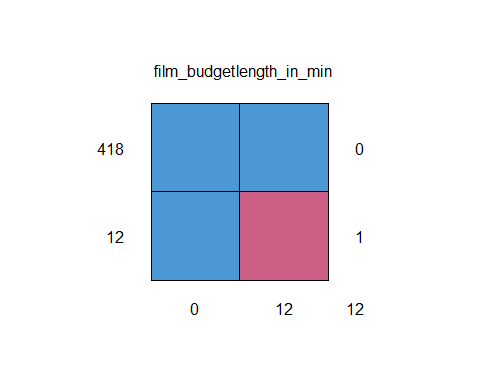
Kaç tane kayıp verimiz oluştuğuna bakmak için:

colSums(is.na(blockbusters\_miss))

## release\_year rank\_in\_year imdb\_rating   
## 0 0 0   
## mpaa\_rating film\_title film\_budget   
## 0 0 0   
## length\_in\_min domestic\_distributor worldwide\_gross   
## 12 0 0   
## domestic\_gross genre\_1 genre\_2   
## 0 0 0

Analizde fazla değişken olduğu için sadece NA olan sütun ile olmayan bir sütunu görüntüledim:

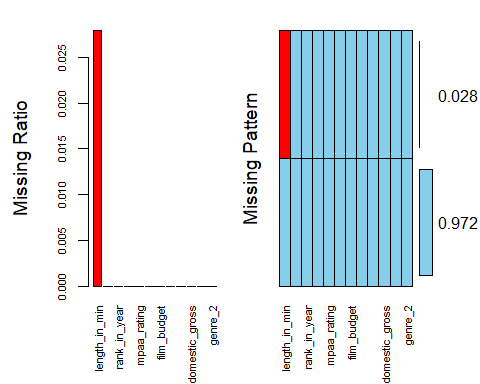
md.pattern(blockbusters\_miss[,c("film\_budget","length\_in\_min")])



## film\_budget length\_in\_min   
## 418 1 1 0  
## 12 1 0 1  
## 0 12 12

NA verilere bir başka paket ile bakalım:

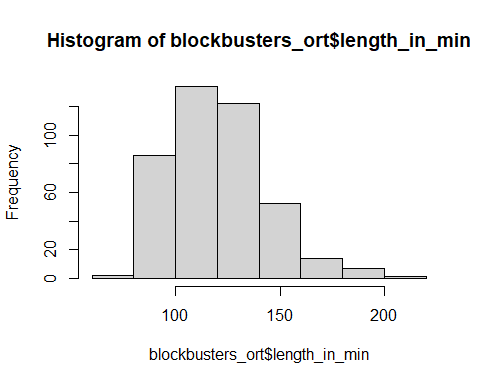
aggr(blockbusters\_miss,numbers=TRUE, sortVars=TRUE, labels=names(blockbusters\_yeni),cex.axis=.7,gap=3,ylab=c("Missing Ratio","Missing Pattern"))



##   
## Variables sorted by number of missings:   
## Variable Count  
## length\_in\_min 0.02790698  
## release\_year 0.00000000  
## rank\_in\_year 0.00000000  
## imdb\_rating 0.00000000  
## mpaa\_rating 0.00000000  
## film\_title 0.00000000  
## film\_budget 0.00000000  
## domestic\_distributor 0.00000000  
## worldwide\_gross 0.00000000  
## domestic\_gross 0.00000000  
## genre\_1 0.00000000  
## genre\_2 0.00000000

Basit doldurma yöntemi olan ortalama ile doldurma yapılabilir. Önce histogram grafiğine bakalım:

blockbusters\_ort<-blockbusters\_miss  
hist(blockbusters\_ort$length\_in\_min)



Bu kod ile de verimizi ortalaması olan 120 ile doldurduğunu göreceğiz:

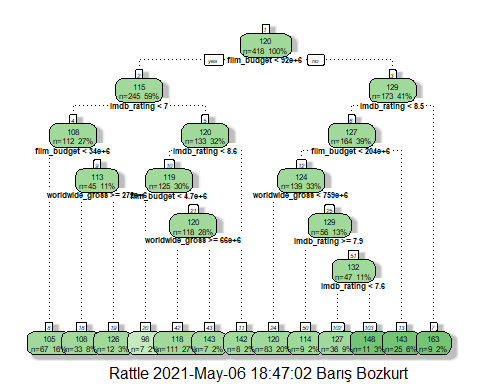
blockbusters\_ort$length\_in\_min[is.na(blockbusters\_ort$length\_in\_min)]<-mean(blockbusters\_ort$length\_in\_min,na.rm=TRUE)  
blockbusters\_ort$length\_in\_min<-as.integer(blockbusters\_ort$length\_in\_min)

Bir başka doldurma yöntemi olan karar ağacı dolduralım:

library(rpart)

## Warning: package 'rpart' was built under R version 4.0.5

data\_dt<-blockbusters\_miss  
rtree <- rpart(length\_in\_min ~ imdb\_rating + film\_budget+ worldwide\_gross, data\_dt, method="anova")  
library(rattle)  
fancyRpartPlot(rtree,cex=0.5)



Boş yerleri elde ettiğimiz verilerle doldurmak için:

data\_dt$length\_in\_min <- ifelse(is.na(data\_dt$length\_in\_min), predict(rtree,data\_dt,type="vector"),data\_dt$length\_in\_min)  
data\_dt$length\_in\_min<-as.integer(data\_dt$length\_in\_min)

Hem sayıları NA verilere atamış oldum hem de integer haline çevirdim. Ancak bu NA verileri rastgele kendim oluşturmuştum. Bu yüzden ödevin devamında verinin orjinaline kullanmaya devam edeceğim.

Verimi %80’i eğitim, %20’si test olacak şekilde böldüm.

set.seed(1234567)  
trainIndex <- sample(1:nrow(blockbusters\_yeni), size = round(0.8\*nrow(blockbusters\_yeni)), replace=FALSE)  
train<- blockbusters\_yeni[trainIndex ,]  
test <- blockbusters\_yeni[-trainIndex ,]

library("openxlsx")

## Warning: package 'openxlsx' was built under R version 4.0.5

write.xlsx(train, 'train.xlsx')  
write.xlsx(test, 'test.xlsx')

Yıl sayısı çok fazla olduğu için yılları kategorik hale getirmeye karar verdim.

train$yil\_kategorik[train$release\_year >= 1900 & train$release\_year <= 1989] <- "1990\_oncesi"

## Warning: Unknown or uninitialised column: `yil\_kategorik`.

train$yil\_kategorik[train$release\_year >= 1990 & train$release\_year <= 2004] <- "1990\_ile\_2005\_arasi"  
train$yil\_kategorik[train$release\_year >= 2005] <- "2005\_ve\_sonrasi"

Kategorik veriyi oluşturduk fakat R’da henüz factor olarak algılanmıyor. Kategorik verilerimizin factor olarak algılanması için:

train<-as.data.frame(train)  
train$mpaa\_rating<-as.factor(train$mpaa\_rating)  
train$yil\_kategorik<-as.factor(train$yil\_kategorik)  
summary(train)

## release\_year rank\_in\_year imdb\_rating mpaa\_rating film\_title   
## Min. :1977 Min. : 1.000 Min. :3.700 G : 19 Length:344   
## 1st Qu.:1987 1st Qu.: 3.000 1st Qu.:6.600 PG :112 Class :character   
## Median :1997 Median : 6.000 Median :7.000 PG-13:138 Mode :character   
## Mean :1997 Mean : 5.642 Mean :7.082 R : 75   
## 3rd Qu.:2008 3rd Qu.: 8.000 3rd Qu.:7.700   
## Max. :2019 Max. :10.000 Max. :9.000   
## film\_budget length\_in\_min domestic\_distributor worldwide\_gross   
## Min. : 325000 Min. : 60.0 Length:344 Min. :3.719e+07   
## 1st Qu.: 20000000 1st Qu.:104.0 Class :character 1st Qu.:2.003e+08   
## Median : 70000000 Median :118.0 Mode :character Median :3.729e+08   
## Mean : 86628561 Mean :120.3 Mean :4.845e+08   
## 3rd Qu.:145000000 3rd Qu.:134.0 3rd Qu.:7.260e+08   
## Max. :356000000 Max. :201.0 Max. :2.798e+09   
## domestic\_gross genre\_1 genre\_2   
## Min. : 31755742 Length:344 Length:344   
## 1st Qu.:100438485 Class :character Class :character   
## Median :167613460 Mode :character Mode :character   
## Mean :196022744   
## 3rd Qu.:261577940   
## Max. :936662225   
## yil\_kategorik  
## 1990\_ile\_2005\_arasi:120   
## 1990\_oncesi :109   
## 2005\_ve\_sonrasi :115   
##   
##   
##

Göründüğü üzere hem MPAA rating (Amerikan Sinema Filmleri Derneği Yaş Derecelendirmesi) hem de Yıl değişkenimiz artık kategorik olarak var.