Data Cleaning

Servando De La Garza

3/14/2022

Opening file and taking a look at it.

```r
movies.data <- read.csv("C:\\Users\\Servando\\Downloads\\archive (37)\\messy_IMDB_dataset.csv", sep=";")
head(movies.data)
```

```
##   IMBD.title.ID                                Original.titlÊ Release.year
## 1     tt0111161                      The Shawshank Redemption   1995-02-10
## 2     tt0068646                                 The Godfather   09 21 1972
## 3     tt0468569                               The Dark Knight  23 -07-2008
## 4     tt0071562                        The Godfather: Part II   1975-09-25
## 5     tt0110912                                  Pulp Fiction   1994-10-28
## 6     tt0167260 The Lord of the Rings: The Return of the King    22 Feb 04
##                     Genrë. Duration     Country Content.Rating
## 1                    Drama      142         USA              R
## 2             Crime, Drama      175         USA              R
## 3     Action, Crime, Drama      152          US          PG-13
## 4             Crime, Drama      220         USA              R
## 5             Crime, Drama                  USA              R
## 6 Action, Adventure, Drama      201 New Zealand          PG-13
##               Director  X        Income     Votes Score
## 1       Frank Darabont NA    $ 28815245 2.278.845   9.3
## 2 Francis Ford Coppola NA   $ 246120974 1.572.674   9.2
## 3    Christopher Nolan NA  $ 1005455211 2.241.615    9.
## 4 Francis Ford Coppola NA $ 4o8,035,783 1.098.714  9,.0
## 5    Quentin Tarantino NA   $ 222831817 1.780.147  8,9f
## 6        Peter Jackson NA  $ 1142271098 1.604.280  08.9
```

Let’s change column names of the dataframe.

```r
colnames(movies.data)[c(1:4,7,10)] <- c("ID","Title","Year","Genre","Rating","Revenue")
#Let's drop irrelevant columns
movies.data$X <- NULL
head(movies.data)
```

```
##          ID                                         Title         Year
## 1 tt0111161                      The Shawshank Redemption   1995-02-10
## 2 tt0068646                                 The Godfather   09 21 1972
## 3 tt0468569                               The Dark Knight  23 -07-2008
## 4 tt0071562                        The Godfather: Part II   1975-09-25
## 5 tt0110912                                  Pulp Fiction   1994-10-28
## 6 tt0167260 The Lord of the Rings: The Return of the King    22 Feb 04
##                      Genre Duration     Country Rating             Director
## 1                    Drama      142         USA      R       Frank Darabont
## 2             Crime, Drama      175         USA      R Francis Ford Coppola
## 3     Action, Crime, Drama      152          US  PG-13    Christopher Nolan
## 4             Crime, Drama      220         USA      R Francis Ford Coppola
## 5             Crime, Drama                  USA      R    Quentin Tarantino
## 6 Action, Adventure, Drama      201 New Zealand  PG-13        Peter Jackson
##         Revenue     Votes Score
## 1    $ 28815245 2.278.845   9.3
## 2   $ 246120974 1.572.674   9.2
## 3  $ 1005455211 2.241.615    9.
## 4 $ 4o8,035,783 1.098.714  9,.0
## 5   $ 222831817 1.780.147  8,9f
## 6  $ 1142271098 1.604.280  08.9
```

Now that column names are corrected let's fix the formatting in the year column's values.

```r
require("lubridate")
```

```
## Loading required package: lubridate

## Warning: package 'lubridate' was built under R version 4.1.3

##
## Attaching package: 'lubridate'

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

```r
#First let's check what data type we are dealing with
typeof(movies.data$Year)
```

```
## [1] "character"
```

```r
null.when.coerced <- c(2,3,6,10 ,13,16,19,46,71,84,85)
#First let's reformat the records that will convert into null values when using the as.date function
movies.data$Year[null.when.coerced]<- c("1972-09-21","2008-07-23","2004-02-22","1999-10-29","1966-12-23
                                        "2003-01-16","1976-11-18","1946-11-21","1951-03-06","1984-02-28
                                        "1976-12-24")
#Now lets extract only the years of the given dates
movies.data$Year <- year(as.Date.character(movies.data$Year))
#Here, let's get rid of the empty record in the dataframe
movies.data <- movies.data[-14,]
head(movies.data)
```

```
##          ID                                         Title Year
## 1 tt0111161                      The Shawshank Redemption 1995
## 2 tt0068646                                 The Godfather 1972
## 3 tt0468569                               The Dark Knight 2008
## 4 tt0071562                        The Godfather: Part II 1975
## 5 tt0110912                                  Pulp Fiction 1994
## 6 tt0167260 The Lord of the Rings: The Return of the King 2004
##                      Genre Duration      Country Rating             Director
## 1                    Drama      142          USA      R       Frank Darabont
## 2             Crime, Drama      175          USA      R Francis Ford Coppola
## 3     Action, Crime, Drama      152           US  PG-13    Christopher Nolan
## 4             Crime, Drama      220          USA      R Francis Ford Coppola
## 5             Crime, Drama                   USA      R    Quentin Tarantino
## 6 Action, Adventure, Drama      201  New Zealand  PG-13        Peter Jackson
##         Revenue     Votes Score
## 1    $ 28815245 2.278.845   9.3
## 2   $ 246120974 1.572.674   9.2
## 3  $ 1005455211 2.241.615    9.
## 4 $ 4o8,035,783 1.098.714  9,.0
## 5   $ 222831817 1.780.147  8,9f
## 6  $ 1142271098 1.604.280  08.9
```

Now, let's move to the duration column. We have a lot of non numeric values in that column, we'll turn them into NA's and then replace those with mean imputation.

```
#This will coerce the values into NAs if they cannot be converted to numeric
movies.data$Duration <- as.numeric(as.character(movies.data$Duration))
```

```
## Warning: NAs introduced by coercion
```

```
#Unfortunately not all values were coerced into NAs , we will have to specify the missing ones
movies.data$Duration[c(7,10)] <- NA
#Let's see how many NAs we have in the duration column
cat("There are",as.character(sum(is.na(movies.data$Duration))) ,"NA values")
```

```
## There are 7 NA values
```

```
#Simple Mean imputation in the duration column
movies.data$Duration[which(is.na(movies.data$Duration))]<-
  round(mean(movies.data$Duration, na.rm = TRUE),0)
```

Dataset it's looking better but we are still have to clean more columns. Let's start by fixing typos in the columns

```
# Let's fix the country column typos
typo.us <- c("US","US.","US ")
movies.data$Country[movies.data$Country %in% typo.us] <- "USA"
typo.nwz <- c("New Zeland","New Zesland")
movies.data$Country[movies.data$Country %in% typo.nwz] <- "New Zealand"
typo.italy <- "Italy1"
movies.data$Country[movies.data$Country %in% typo.italy] <- "Italy"
#Now, let's apply a similar logic for the rating column
erros <-c("#N/A","Approved","Not Rated","Unrated")
movies.data$Rating[movies.data$Rating %in% erros] <- NA
#This got rid of nonsensical values nicely, however we have a lot of NAs now, package Mice will come in
require("mice")
```

```
## Loading required package: mice
```

```
## Warning: package ’mice’ was built under R version 4.1.3
```

```
##
## Attaching package: ’mice’
```

```
## The following object is masked from ’package:stats’:
##
##     filter
```

```
## The following objects are masked from ’package:base’:
##
##     cbind, rbind
```

```
movies.data$Rating <- as.factor(movies.data$Rating)
imputation <- mice(movies.data,m = 5 ,method =c("","","","","","","polyreg","","","",""),)
```

```
##
##  iter imp variable
##   1   1  Rating
##   1   2  Rating
##   1   3  Rating
##   1   4  Rating
##   1   5  Rating
##   2   1  Rating
##   2   2  Rating
##   2   3  Rating
##   2   4  Rating
##   2   5  Rating
##   3   1  Rating
##   3   2  Rating
##   3   3  Rating
##   3   4  Rating
##   3   5  Rating
##   4   1  Rating
##   4   2  Rating
##   4   3  Rating
##   4   4  Rating
##   4   5  Rating
##   5   1  Rating
##   5   2  Rating
##   5   3  Rating
##   5   4  Rating
##   5   5  Rating
```

```
## Warning: Number of logged events: 8
```

```
imputation$imp$Rating
```

```
##         1     2     3     4     5
## 8      PG    PG    PG    PG     R
## 13      R    PG    PG     R     R
## 28      R     G     R     G     R
## 29      R     R PG-13     R PG-13
## 31     PG     R     R    PG     R
## 37      R     R    PG     R     G
## 41      R PG-13     R     R     R
## 42  PG-13     R     G     R     R
## 48      G     R     R     G     G
## 49      R     R PG-13    PG     R
## 57      R PG-13 PG-13 PG-13 PG-13
## 59      R PG-13 PG-13 PG-13 PG-13
## 63     PG     R    PG    PG    PG
## 64      R PG-13 PG-13     G     G
## 66      R PG-13     R PG-13     R
## 67  PG-13     R PG-13 PG-13     R
## 70      R    PG    PG    PG    PG
## 71     PG     G    PG    PG     R
## 82      R    PG     R     R    PG
## 87      R     R     R     R     R
## 90     PG     R    PG    PG    PG
## 91      R     R     R PG-13     R
## 93     PG    PG     R     R     R
## 94     PG     R     R     R    PG
## 99      R     R    PG     R     R
## 101     G     R     R    PG    PG
```

```
movies.data <- complete(imputation,3)
```

Let’s finish with the very last formatting errors

```
#Replacing dots with commas in the Votes column
movies.data$Votes <- gsub("\\.", ",", movies.data$Votes)
old <- movies.data$Revenue
old <- as.data.frame(old)
movies.data$Revenue <- as.numeric(gsub(".*?([0-9]+).*", "\\1", movies.data$Revenue))
#Row 4 dropped most numbers in the previous function due to an "o" in the string
movies.data$Revenue[4] <- 408035783
#Since we don't know if the Revenue refers to billion,millions or other units we'll leave it like that.
#Now , let's clean the scores column
movies.data$Score[c(3:6,9,12,14:16)] <- c("9.0","9.0","8.9","8.9","8.8","8.8","8.7","8.7","8.7")
```

Finally , we will store the clean dataset into a csv file

```
write.csv(movies.data,"C:\\Users\\Servando\\Documents\\Datasets\\my_csvs\\movies_clean.csv", row.names
```