University of Granada

Exploring the data of 121 patients who underwent glaucoma surgery. Conclusions.

Multivariate Statistics

Author:
Quintín Mesa Romero

1 Introduction

Glaucoma is a leading cause of irreversible blindness worldwide. It is a disease of a progressive optic neuropathy with loss of retinal neurons and their axons, which can result in blindness in case of untreatment[1][2]. In short, it is a group of diseases that kill retinal ganglion cells [2].

The strongest known risk factor is high IOP (Intraocular Pressure), but it is not the only factor responsible for glaucoma [2]. In fact, people with myopia greater than five dioptres, people aged 60 years or more, people with thin cornea, and even people with different skin type, such as africans or afro-caribbean are more likely to develop glaucoma. Of course, having family history multiplies the risk of developing the desease [3].

Given that there are about 80 million people suffering from glaucoma, and it is estimated that over 112 million individuals will have it by 2040, it is reasonable to think that there is a treatment or an operation to reverse the disease before it is too late [4]. Indeed, there is a surgery based on laser technology which is being applied to people with glaucoma[5].

From a research point of view, in relation to this surgery, we wonder whether the pre-surgery condition is related, in any way, to the long-term progression. For this purpose, we have studied a dataset with information of 121 patients who underwent glaucoma surgery using laser technology, in which variables have been meassured before and after the surgery over a three-month period at different time intervals.

Studying this relationship is crucial, primarily because it can inform better treatment strategies and patient outcomes, and clinicians can better predict which patients are more likely to benefit from laser surgery versus those who may need alternative or additional interventions, in order to improve their lives.

2 Methods and techniques

2.1 Data Collection, Preparation and Cleaning

The data from the 121 patients have been stored in an Excel file. This dataset includes clinical variables related to presurgical conditions and post-surgical outcomes after the laser-based surgery. The dataset has been processed, prepared and cleaned using the **R** statistical environment.

First of all, we have to load the library **readxl**:

```
# Load necessary libraries
```{r}
library(readxl)  # for reading excel files
```
```

Now, we are ready to load the dataset. For this purpose, we use the function **read_excel**:

```
#Load the dataset
```{r}
data_glaucoma<-read_excel("Glaucoma_DB.xlsx", sheet="DATOS")
```
```

Once the dataset is loaded, it is advisable to check the structure of the data, and for this, we use **str**:

```r
# Check the structure of the dataset
```{r}
str(data_glaucoma)
```
```

```
tibble [121 × 19] (S3: tbl_df/tbl/data.frame)
 $ OJO            : num [1:121] 0 1 0 1 0 1 0 0 0 1 ...
 $ TIPO_GLAUCOMA  : num [1:121] 0 NA 1 2 2 1 1 3 1 1 ...
 $ N_IMPACTOS     : num [1:121] 112 108 123 131 156 125 178 164 109 116
...
 $ CUADRANTES     : chr [1:121] "4" "4" "4" "4" ...
 $ ENERGIA_IMPACTO: chr [1:121] "1.5" "1.2" "1.1000000000000001" "1.5" ...
 $ ENERGIA_TOTAL  : num [1:121] 174 128 133 191 182 170 249 301 109 238
...
 $ CIRUJIA_PREVIA : num [1:121] NA 1 1 1 1 0 0 0 1 0 ...
 $ PIO_PRE_SLT    : num [1:121] 31 29 36 14 14 30 36 25 23 22 ...
 $ PIO_1_SEMANA   : num [1:121] 0 23 30 0 0 0 0 0 0 22 ...
 $ PIO_1_MES      : chr [1:121] "0" "19" "30" "21" ...
 $ PIO_3_MES      : num [1:121] 0 24 30 14 17 20 19 0 16 20 ...
 $ FARMACOS_PRE   : chr [1:121] "3" "3" "1" "1" ...
 $ FARMACOS_1_MES : num [1:121] 0 4 4 0 0 3 3 0 2 0 ...
 $ FARMACOS_3_MES : num [1:121] 0 4 4 0 0 3 3 0 2 0 ...
 $ DOLOR          : num [1:121] 0 1 1 1 1 1 1 0 0 1 ...
 $ SEXO           : num [1:121] NA 0 0 1 1 1 1 0 1 0 ...
 $ EDAD           : num [1:121] 0 56 56 49 49 74 74 65 60 82 ...
 $ PIO_NORMAL     : chr [1:121] "0" "19" "30" "21" ...
 $ PIO_NORMAL_CAT : num [1:121] 1 0 1 0 0 0 0 0 1 0 ...
```

Also, in order to not to alter the original dataset, it is advisable to make a copy and work with it in the future. We will load the variable names from the dataset too. For that, we make:

```r
# copy of the dataset
```{r}
data_glaucoma_copy <- data_glaucoma
attach(data_glaucoma_copy)

```
```

It is easy to see that there are missinng values, so we have to handle them. We can use the function **colSums** with the argument **is.na(data_glaucoma)**, which tells us the number of NA values that there are for each variable:

```r
# Check for missing values
```{r}
missing_values <- colSums(is.na(data_glaucoma))
print(missing_values)
```
```

```
            OJO   TIPO_GLAUCOMA       N_IMPACTOS       CUADRANTES 
              4               2               0               0 
ENERGIA_IMPACTO   ENERGIA_TOTAL  CIRUJIA_PREVIA     PIO_PRE_SLT 
              0               0              36               0 
   PIO_1_SEMANA       PIO_1_MES       PIO_3_MES    FARMACOS_PRE 
              0               0               0               0 
 FARMACOS_1_MES  FARMACOS_3_MES           DOLOR            SEXO 
              0               0              60              61 
           EDAD      PIO_NORMAL  PIO_NORMAL_CAT 
              0               0               0 
```

There are different options for handling **NA** data. One could be simply removing the rows with missing values, but it might lead to the loss of valuable data and even statistical power. Other option is to replace missing values with the mean or the **median**. We will used the last one; we will replace the missing data by the median of the corresponding variable. To do this, we have made a for loop in which for each variable with missing values, each of these NA values are replaced by the median of the rest of the values of the corresponding variable.

```r
# replacing missing values by the median

```{r}
for (col_name in names(missing_values[missing_values > 0])) {
    data_glaucoma_copy[[col_name]][is.na(data_glaucoma_copy[[col_name]])] <- median(data_glaucoma_copy[[col_name]], na.rm = TRUE)
}
```
```

We check the change:

```{r}
head(data_glaucoma_copy,10)
```

A tibble: 10 × 19

OJO <dbl>	TIPO_GLAUCOMA <dbl>	N_IMPACTOS <dbl>	CUADRANTES <chr>	ENERGIA_IMPACTO <chr>	ENERGIA_TOTAL <dbl>	CIRUJIA_PREVIA <dbl>	PIO_PRE_SLT <dbl>	PIO_1_SEMANA <dbl>	PIO_1_MES <chr>
0	0	112	4	1.5	174	1	31	0	0
1	4	108	4	1.2	128	1	29	23	19
0	1	123	4	1.1000000000000001	133	1	36	30	30
1	2	131	4	1.5	191	1	14	0	21
0	2	156	4	1.2	182	1	14	0	16
1	1	125	4	1.4	170	0	30	0	18
0	1	178	4	1.4	249	0	36	0	20
0	3	164	4	1.9	301	0	25	0	18
0	1	109	4	1	109	1	23	0	10
1	1	116	4	2.2000000000000002	238	0	22	22	20

1-10 of 10 rows | 1-10 of 19 columns

```{r}
colSums(is.na(data_glaucoma_copy))
```

```
            OJO   TIPO_GLAUCOMA       N_IMPACTOS      CUADRANTES ENERGIA_IMPACTO   ENERGIA_TOTAL 
              0               0               0               0               0               0 
CIRUJIA_PREVIA     PIO_PRE_SLT    PIO_1_SEMANA      PIO_1_MES       PIO_3_MES    FARMACOS_PRE 
              0               0               0               0               0               0 
FARMACOS_1_MES  FARMACOS_3_MES          DOLOR            SEXO            EDAD      PIO_NORMAL 
              0               0               0               0               0               0 
PIO_NORMAL_CAT 
              0 
```

Then, we convert the **categorical** variables of the dataset into **factor** type:

```{r}
data_glaucoma_copy$SEXO <- factor(data_glaucoma_copy$SEXO, levels = c("1", "0"),labels = c("Hombre", "Mujer"))

data_glaucoma_copy$OJO <- factor(data_glaucoma_copy$OJO, levels = c("0","1"), labels = c("Izdo", "dcho"))

data_glaucoma_copy$TIPO_GLAUCOMA <- factor(data_glaucoma_copy$TIPO_GLAUCOMA, levels = c("0","1","2", "3","4", "5", "6", "7", "8", "9", "10",
"11", "12", "13", "14","15","16"), labels = c("PIGMENTARIO", "GPPA", "GPAA", "GLAUCO PIGMENT","GPAC", "G.PSEUDOEX", "SD DISPERSION
PIGMENTARIA","GLAUCOMA CONGENITO", "GLAUCOMA POSTRABECULAR", "HTO", "CPAC", "GSAA", "GS PXE", "GCS","HTO PIGMENTARI", "GPAA?miópico",
"GPAA?"))

data_glaucoma_copy$CIRUJIA_PREVIA <- factor(data_glaucoma_copy$CIRUJIA_PREVIA, levels = c("1", "0"), labels = c("No", "Si"))

data_glaucoma_copy$DOLOR <- factor(data_glaucoma_copy$DOLOR, levels = c("0","1"), labels = c("Si", "No"))

```

Once all these changes have been applied to the data, we obtain:

```{r}
str(data_glaucoma_copy)
```

```
tibble [121 × 19] (S3: tbl_df/tbl/data.frame)
 $ OJO           : Factor w/ 2 levels "Izdo","dcho": 1 2 1 2 1 2 1 1 1 2 ...
 $ TIPO_GLAUCOMA : Factor w/ 17 levels "PIGMENTARIO",..: 1 5 2 3 3 2 2 4 2 2 ...
 $ N_IMPACTOS    : num [1:121] 112 108 123 131 156 125 178 164 109 116 ...
 $ CUADRANTES    : chr [1:121] "4" "4" "4" "4" ...
 $ ENERGIA_IMPACTO: chr [1:121] "1.5" "1.2" "1.1000000000000001" "1.5" ...
 $ ENERGIA_TOTAL : num [1:121] 174 128 133 191 182 170 249 301 109 238 ...
 $ CIRUJIA_PREVIA : Factor w/ 2 levels "No","Si": 1 1 1 1 1 2 2 2 1 2 ...
 $ PIO_PRE_SLT   : num [1:121] 31 29 36 14 14 30 36 25 23 22 ...
 $ PIO_1_SEMANA  : num [1:121] 0 23 30 0 0 0 0 0 0 22 ...
 $ PIO_1_MES     : chr [1:121] "0" "19" "30" "21" ...
 $ PIO_3_MES     : num [1:121] 0 24 30 14 17 20 19 0 16 20 ...
 $ FARMACOS_PRE  : chr [1:121] "3" "3" "1" "1" ...
 $ FARMACOS_1_MES : num [1:121] 0 4 4 0 0 3 3 0 2 0 ...
 $ FARMACOS_3_MES : num [1:121] 0 4 4 0 0 3 3 0 2 0 ...
 $ DOLOR         : Factor w/ 2 levels "Si","No": 1 2 2 2 2 2 2 1 1 2 ...
 $ SEXO          : Factor w/ 2 levels "Hombre","Mujer": 1 2 2 1 1 1 1 2 1 2 ...
 $ EDAD          : num [1:121] 0 56 56 49 49 74 74 65 60 82 ...
 $ PIO_NORMAL    : chr [1:121] "0" "19" "30" "21" ...
 $ PIO_NORMAL_CAT : num [1:121] 1 0 1 0 0 0 0 0 1 0 ...
```

A tibble: 121 × 19

OJO <fct>	TIPO_GLAUCOMA <fct>	N_IMPACT... <dbl>	CUADRANT... <chr>	ENERGIA_IMPACTO <chr>	ENERGIA_TOT... <dbl>	CIRUJIA_PREVIA <fct>	PIO_PRE_SLT <dbl>	PIO_1_SEMA... <dbl>
Izdo	PIGMENTARIO	112	4	1.5	174.0	No	31	0
dcho	GPAC	108	4	1.2	128.0	No	29	23
Izdo	GPPA	123	4	1.1000000000000001	133.0	No	36	30
dcho	GPAA	131	4	1.5	191.0	No	14	0
Izdo	GPAA	156	4	1.2	182.0	No	14	0
dcho	GPPA	125	4	1.4	170.0	Si	30	0
Izdo	GPPA	178	4	1.4	249.0	Si	36	0
Izdo	GLAUCO PIGMENT	164	4	1.9	301.0	Si	25	0
Izdo	GPPA	109	4	1	109.0	No	23	0
dcho	GPPA	116	4	2.2000000000000002	238.0	Si	22	22

Finally, since there are some quantitative variables with non-numerical values (which makes them to be considered by R as char type), I have decided to subsitute these non-numerical values by the mediane of the rest of the numeric values of the variable, instead of removing them, which would have been another solution to this problem.

The quantitative variables that R has considered are:

N_IMPACT... <dbl>	CUADRAN... <dbl>	ENERGIA_IMPAC... <dbl>	ENERGIA_TOT... <dbl>	PIO_PRE_SLT <dbl>	PIO_1_SEMA... <dbl>	PIO_1_MES <dbl>	PIO_3_MES <dbl>	FARMACOS_... <dbl>	FARMACOS_1_... <dbl>
112	4.0	1.5	174.0	31	0	0.0	0	3	0
108	4.0	1.2	128.0	29	23	19.0	24	3	4
123	4.0	1.1	133.0	36	30	30.0	30	1	4
131	4.0	1.5	191.0	14	0	21.0	14	1	0
156	4.0	1.2	182.0	14	0	16.0	17	1	0
125	4.0	1.4	170.0	30	0	18.0	20	2	3
178	4.0	1.4	249.0	36	0	20.0	19	2	3
164	4.0	1.9	301.0	25	0	18.0	0	1	0
109	4.0	1.0	109.0	23	0	10.0	16	2	2
116	4.0	2.2	238.0	22	22	20.0	20	0	0

1-10 of 121 rows | 1-10 of 14 columns

Previous 1 2 3 4 5 6 ... 13 Next

We need to do the substitution to the variables CUADRANTES, PIO_NORMAL, PIO_1_MES, FARMACOS_PRE. For that purpose, we do:

```
# CUADRANTES
```{r}
# We iidentify the numeric values of the column
es_numerico <- suppressWarnings(!is.na(as.numeric(data_glaucoma_copy$CUADRANTES)))
# We calculate the mediane of the numeric values
mediana_numericos <- median(as.numeric(data_glaucoma_copy$CUADRANTES[es_numerico]))
# Sustituimos los valores no numéricos por la mediana
data_glaucoma_copy$CUADRANTES[!es_numerico] <- mediana_numericos
# pasamos a numéricos
data_glaucoma_copy$CUADRANTES <- as.numeric(data_glaucoma_copy$CUADRANTES)
```
```

```
# PIO_1_MES
```{r}
# We iidentify the numeric values of the column
es_numerico <- suppressWarnings(!is.na(as.numeric(data_glaucoma_copy$PIO_1_MES)))
# We calculate the mediane of the numeric values
mediana_numericos <- median(as.numeric(data_glaucoma_copy$PIO_1_MES[es_numerico]))
# Sustituimos los valores no numéricos por la mediana
data_glaucoma_copy$PIO_1_MES[!es_numerico] <- mediana_numericos
# pasamos a numéricos
data_glaucoma_copy$PIO_1_MES <- as.numeric(data_glaucoma_copy$PIO_1_MES)
```
```

```
# PIO_NORMAL
```{r}
# We iidentify the numeric values of the column
es_numerico <- suppressWarnings(!is.na(as.numeric(data_glaucoma_copy$PIO_NORMAL)))
# We calculate the mediane of the numeric values
mediana_numericos <- median(as.numeric(data_glaucoma_copy$PIO_NORMAL[es_numerico]))
# Sustituimos los valores no numéricos por la mediana
data_glaucoma_copy$PIO_NORMAL[!es_numerico] <- mediana_numericos
# pasamos a numéricos
data_glaucoma_copy$PIO_NORMAL <- as.numeric(data_glaucoma_copy$PIO_NORMAL)
```
```

```
# FARMACOS_PRE
```{r}
# We iidentify the numeric values of the column
es_numerico <- suppressWarnings(!is.na(as.numeric(data_glaucoma_copy$FARMACOS_PRE)))
# We calculate the mediane of the numeric values
mediana_numericos <- median(as.numeric(data_glaucoma_copy$FARMACOS_PRE[es_numerico]))
# Sustituimos los valores no numéricos por la mediana
data_glaucoma_copy$FARMACOS_PRE[!es_numerico] <- mediana_numericos
# pasamos a numéricos
data_glaucoma_copy$FARMACOS_PRE <- as.numeric(data_glaucoma_copy$FARMACOS_PRE)
```
```

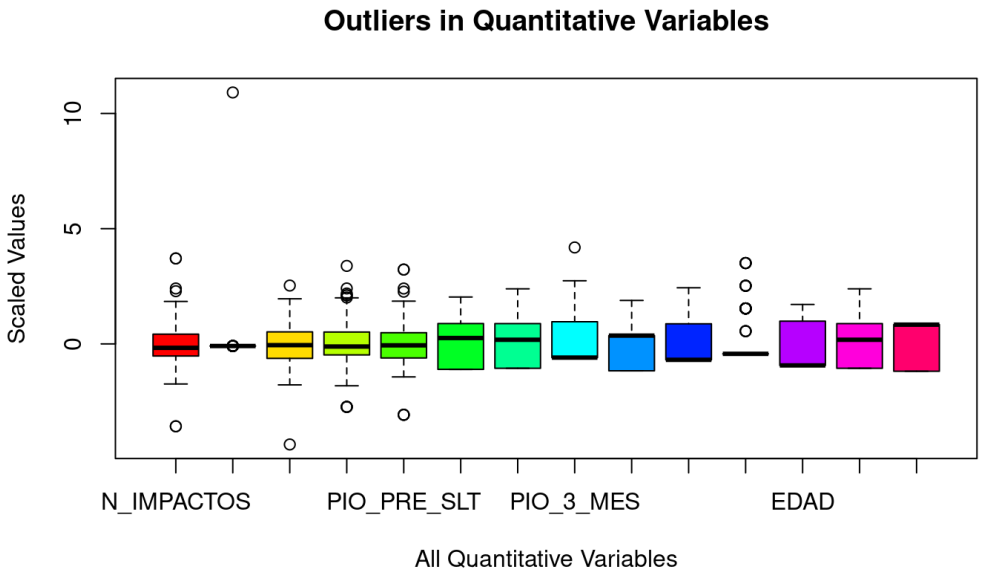
Eventually, the structure of the dataset is as follows:

```r
str(data_glaucoma_copy)
```

```
tibble [121 × 19] (S3: tbl_df/tbl/data.frame)
 $ OJO            : Factor w/ 2 levels "Izdo","dcho": 1 2 1 2 1 2 1 1 1 2 ...
 $ TIPO_GLAUCOMA  : Factor w/ 17 levels "PIGMENTARIO",..: 1 5 2 3 3 2 2 4 2 2 ...
 $ N_IMPACTOS     : num [1:121] 112 108 123 131 156 125 178 164 109 116 ...
 $ CUADRANTES     : num [1:121] 4 4 4 4 4 4 4 4 4 4 ...
 $ ENERGIA_IMPACTO: num [1:121] 1.5 1.2 1.1 1.5 1.2 1.4 1.4 1.9 1 2.2 ...
 $ ENERGIA_TOTAL  : num [1:121] 174 128 133 191 182 170 249 301 109 238 ...
 $ CIRUJIA_PREVIA : Factor w/ 2 levels "No","Si": 1 1 1 1 1 2 2 2 1 2 ...
 $ PIO_PRE_SLT    : num [1:121] 31 29 36 14 14 30 36 25 23 22 ...
 $ PIO_1_SEMANA   : num [1:121] 0 23 30 0 0 0 0 0 0 22 ...
 $ PIO_1_MES      : num [1:121] 0 19 30 21 16 18 20 18 10 20 ...
 $ PIO_3_MES      : num [1:121] 0 24 30 14 17 20 19 0 16 20 ...
 $ FARMACOS_PRE   : num [1:121] 3 3 1 1 1 2 2 1 2 0 ...
 $ FARMACOS_1_MES : num [1:121] 0 4 4 0 0 3 3 0 2 0 ...
 $ FARMACOS_3_MES : num [1:121] 0 4 4 0 0 3 3 0 2 0 ...
 $ DOLOR          : Factor w/ 2 levels "Si","No": 1 2 2 2 2 2 2 1 1 2 ...
 $ SEXO           : Factor w/ 2 levels "Hombre","Mujer": 1 2 2 1 1 1 1 2 1 2 ...
 $ EDAD           : num [1:121] 0 56 56 49 49 74 74 65 60 82 ...
 $ PIO_NORMAL     : num [1:121] 0 19 30 21 16 18 20 18 10 20 ...
 $ PIO_NORMAL_CAT : num [1:121] 1 0 1 0 0 0 0 0 1 0 ...
```

2.2 Outliers detection and treatment

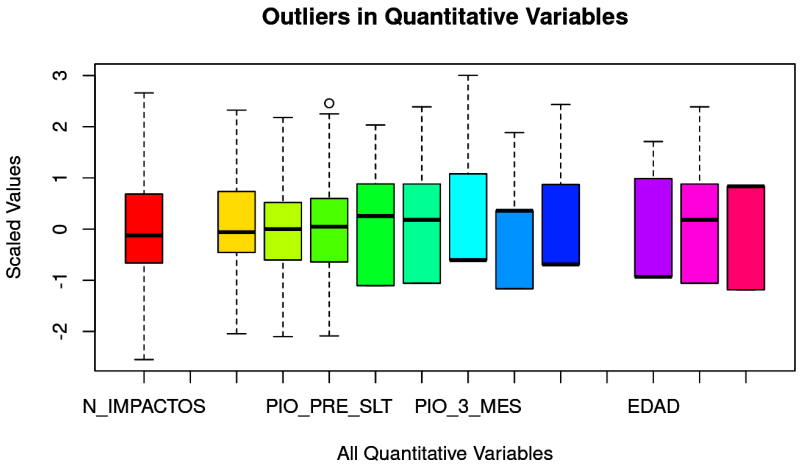
With regard to the detection of outliers, if we are studying quantitative variables, an informative way to detect them is with a boxplot graphic. However, this is not the case with categorical variables.

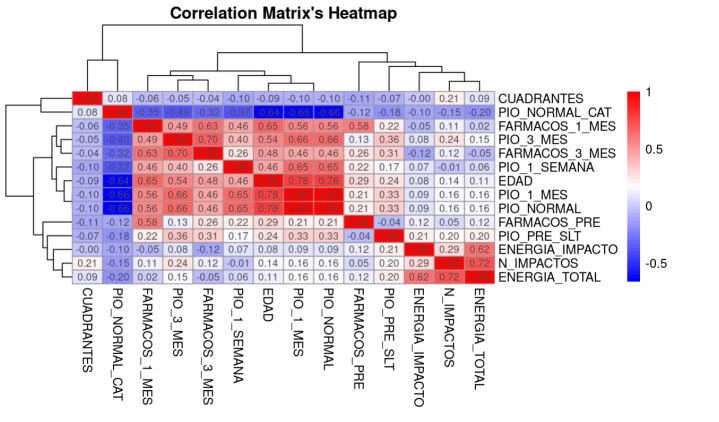


So, we need something that throws light to the outliers of all the variables. The decision we have taken about outliers is that we are going to modify them by the mean of their variables.

For the purpose of detecting the outliers, we will use the IQR (Interquartile Range), which is defined as the difference of the third and the firt quartiles ($Q_3 - Q_1$), by means of a function taken from the practice guide, which detects the outliers in each variables and subsitutes then by the mean of their variables:

```
# OUTLIERS DETECTION

```{r}

# Recursive function that modifies outliers by the mean of their variable
outlier<-function(data,na.rm=T){

  H<-1.5*IQR(data)
  data[data<quantile(data,0.25,na.rm = T)-H]<-NA
  data[data>quantile(data,0.75, na.rm = T)+H]<-NA
  data[is.na(data)]<-mean(data, na.rm = T)
  H<-1.5*IQR(data)

  if (TRUE %in% (data<quantile(data,0.25,na.rm = T)-H) |
    TRUE %in% (data>quantile(data,0.75,na.rm = T)+H))
    outlier(data)
  else
    return(data)
}

```
```

We apply that function to all the quantitative variables:

```
```{r}
# This data.frame is to preserve original data once the outliers are modified
data_glaucoma_copy_aux<-data_glaucoma_copy

# Call to outlier function for each variable identified with outliers
data_glaucoma_copy_aux$N_IMPACTOS<-outlier(data_glaucoma_copy_aux$N_IMPACTOS)
data_glaucoma_copy_aux$CUADRANTES<-outlier(data_glaucoma_copy_aux$CUADRANTES)
data_glaucoma_copy_aux$ENERGIA_IMPACTO<-outlier(data_glaucoma_copy_aux$ENERGIA_IMPACTO)
data_glaucoma_copy_aux$ENERGIA_TOTAL<-outlier(data_glaucoma_copy_aux$ENERGIA_TOTAL)
data_glaucoma_copy_aux$PIO_PRE_SLT<-outlier(data_glaucoma_copy_aux$PIO_PRE_SLT)
data_glaucoma_copy_aux$PIO_1_SEMANA<-outlier(data_glaucoma_copy_aux$PIO_1_SEMANA)
data_glaucoma_copy_aux$PIO_1_MES<-outlier(data_glaucoma_copy_aux$PIO_1_MES)
data_glaucoma_copy_aux$PIO_3_MES<-outlier(data_glaucoma_copy_aux$PIO_3_MES)
data_glaucoma_copy_aux$FARMACOS_PRE<-outlier(data_glaucoma_copy_aux$FARMACOS_PRE)
data_glaucoma_copy_aux$FARMACOS_1_MES<-outlier(data_glaucoma_copy_aux$FARMACOS_1_MES)
data_glaucoma_copy_aux$FARMACOS_3_MES<-outlier(data_glaucoma_copy_aux$FARMACOS_3_MES)
data_glaucoma_copy_aux$PIO_NORMAL<-outlier(data_glaucoma_copy_aux$PIO_NORMAL)
```
```

and obtain:



2.3 Study of Correlated Variables

Once all data from the dataset is treated, we can visualize its correlation matrix, in order to draw conclusions according to the variables.

```
# correlation matrix
```{r}
quant<-data_glaucoma_copy[sapply(data_glaucoma_copy, is.numeric)]
correlation_matrix<-cor(quant)
correlation_matrix
```
```

	N_IMPACTOS	CUADRANTES	ENERGIA_IMPACTO	ENERGIA_TOTAL	PIO_PRE_SLT	PIO_1_SEMANA	PIO_1_MES	PIO_3_MES	FARMACOS_PRE	FARMACOS_1_MES	FARMACOS_3_MES
N_IMPACTOS	1.000000000	0.209711403	0.294746776	0.71956624	0.20004545	-0.006276885	0.15731641	0.23620466	0.04948059	0.11234573	0.12246309
CUADRANTES	0.209711403	1.000000000	-0.004599065	0.08917846	-0.06846748	-0.101295757	-0.09697237	-0.05446221	-0.10693066	-0.06335843	-0.03953320
ENERGIA_IMPACTO	0.294746776	-0.004599065	1.000000000	0.62384726	0.20544432	0.071358437	0.08547159	0.08067846	0.11546577	-0.05113838	-0.12227355
ENERGIA_TOTAL	0.719566241	0.089178459	0.623847257	1.00000000	0.20412294	0.062094640	0.15568652	0.15115261	0.11581459	0.01689791	-0.04693138
PIO_PRE_SLT	0.200045446	-0.068467479	0.205444324	0.20412294	1.00000000	0.166866601	0.32948257	0.36124045	-0.04101026	0.21893095	0.31120657
PIO_1_SEMANA	-0.006276885	-0.101295757	0.071358437	0.06209464	0.16686660	1.000000000	0.64500037	0.40120415	0.22222764	0.45933005	0.26118318
PIO_1_MES	0.157316412	-0.096972372	0.085471587	0.15568652	0.32948257	0.645000371	1.00000000	0.65515472	0.21392932	0.55714365	0.45729397
PIO_3_MES	0.236204659	-0.054462206	0.080678461	0.15115261	0.36124045	0.401204147	0.65515472	1.00000000	0.12631807	0.49431869	0.69532675
FARMACOS_PRE	0.049480587	-0.106930663	0.115465774	0.11581459	-0.04101026	0.222227636	0.21392932	0.12631807	1.00000000	0.57853856	0.26271371
FARMACOS_1_MES	0.112345725	-0.063358425	-0.051138378	0.01689791	0.21893095	0.459330045	0.55714365	0.49431869	0.57853856	1.00000000	0.62940275
FARMACOS_3_MES	0.122463093	-0.039533203	-0.122273552	-0.04693138	0.31120657	0.261183176	0.45729397	0.69532675	0.26271371	0.62940275	1.00000000
EDAD	0.142094305	-0.085562361	0.079057211	0.11402745	0.23671696	0.455598165	0.77946590	0.53743268	0.28728735	0.65487613	0.48031679
PIO_NORMAL	0.157316412	-0.096972372	0.085471587	0.15568652	0.32948257	0.645000371	1.00000000	0.65515472	0.21392932	0.55714365	0.45729397
PIO_NORMAL_CAT	-0.150884079	0.076590639	-0.101654727	-0.19644010	-0.17838179	-0.373688850	-0.66240316	-0.39637864	-0.12284588	-0.35287700	-0.31692672

	EDAD	PIO_NORMAL	PIO_NORMAL_CAT
N_IMPACTOS	0.14209431	0.15731641	-0.15088408
CUADRANTES	-0.08556236	-0.09697237	0.07659064
ENERGIA_IMPACTO	0.07905721	0.08547159	-0.10165473
ENERGIA_TOTAL	0.11402745	0.15568652	-0.19644010
PIO_PRE_SLT	0.23671696	0.32948257	-0.17838179
PIO_1_SEMANA	0.45559816	0.64500037	-0.37368885
PIO_1_MES	0.77946590	1.00000000	-0.66240316
PIO_3_MES	0.53743268	0.65515472	-0.39637864
FARMACOS_PRE	0.28728735	0.21392932	-0.12284588
FARMACOS_1_MES	0.65487613	0.55714365	-0.35287700
FARMACOS_3_MES	0.48031679	0.45729397	-0.31692672
EDAD	1.00000000	0.77946590	-0.63691033
PIO_NORMAL	0.77946590	1.00000000	-0.66240316
PIO_NORMAL_CAT	-0.63691033	-0.66240316	1.00000000

In order to illustrate the information given by the correlation matrix, Here we present the heatmap of the correlation matrix, where the most related variables are shown in reddish tones:

```
# heatmap
```{r}
library(pheatmap)
# Heatmap
pheatmap(correlation_matrix,
         color = colorRampPalette(c("blue", "white", "red"))(50),
         main = "Correlation Matrix's Heatmap",
         display_numbers = TRUE)
```
```



Visually, we can infer which variables are related the most, but, we prefer to extract that information easily, so, taking advantage of the fact that R is a programming language, we do what follows:

```r
# Correlated variables

```{r}
# Establish correlation threshold
threshold <- 0.50

# Convert correlation matrix, mantaining only the correlations higher than the threshold
high_cor <- which(abs(correlation_matrix) > threshold, arr.ind = TRUE)

# Filter so that correlations with themselves (diagonal of the matrix) are not included.
high_cor <- high_cor[high_cor[, 1] != high_cor[, 2], ]

# Show the pairs of correlated variables
high_cor_results <- data.frame(
  Variable1 = rownames(correlation_matrix)[high_cor[, 1]],
  Variable2 = colnames(correlation_matrix)[high_cor[, 2]],
  Correlation = correlation_matrix[high_cor]
)

# Result
print(high_cor_results)

```
```

and obtain:

Description: df [16 × 3]

Variable1 <chr>	Variable2 <chr>	Correlation <dbl>
N_IMPACTOS	ENERGIA_TOTAL	0.7195662
ENERGIA_IMPACTO	ENERGIA_TOTAL	0.6238473
PIO_1_SEMANA	PIO_1_MES	0.6450004
PIO_1_MES	PIO_3_MES	0.6551547
PIO_1_MES	FARMACOS_1_MES	0.5571436
FARMACOS_PRE	FARMACOS_1_MES	0.5785386
PIO_3_MES	FARMACOS_3_MES	0.6953267
FARMACOS_1_MES	FARMACOS_3_MES	0.6294027
PIO_1_MES	EDAD	0.7794659
PIO_3_MES	EDAD	0.5374327
FARMACOS_1_MES	EDAD	0.6548761
PIO_1_SEMANA	PIO_NORMAL	0.6450004
PIO_1_MES	PIO_NORMAL	1.0000000
PIO_3_MES	PIO_NORMAL	0.6551547
FARMACOS_1_MES	PIO_NORMAL	0.5571436
EDAD	PIO_NORMAL	0.7794659

16 rows

We observe a certain relationship between the variables

- (N_IMPACTOS,ENERGIA_TOTAL), (ENERGIA_IMPACTO,ENERGIA_TOTAL)
- (EDAD, PIO_NORMAL), (EDAD_PIO_1_MES), (EDAD, PIO_3_MES), (EDAD, FARMACOS_1_MES)
- (PIO_1_SEMANA, PIO_1_MES), (PIO_1_MES, PIO_3_MES), (PIO_1_MES,FARMACOS_1_MES), (FARMACOS_1_MES,FARMACOS_3_MES), (FARMACOS_PRE,FARMACOS_1_MES)

Once we have all this information about the variables, we should interpret it, in order to draw some conclusions about the question we made at the beginning.

2.4 Data Analysis with R

It can be interesting to keep an eye on the correlations of the variable **EDAD** and **PIO** variables, as both are factors of risk when suffering from glaucoma. Also, studying a bit more the relation between que **PIO** variables, could give us more information about the increasing/decreasing of the Intraocular Pressure after the surgery. Also, it would be interesting to interpret the realtion between **PIO** and **FARMACOS**.

For this pursope, I have written R code, with the help of some blogs, books, tutorials, whose references I leave below, and also, I have used ChatGPT AI for making better the structure of the code and to know the meaning and use of some funtions of R which are a bit advanced compared to my R skills. I know that I could have done it easily, with the tools I have at the moment, but I preferred to investigate the R language and discover all the potential it has, because it will be beneficial for my future work.

```r
library(dplyr)

# Analyzing the effect of age on intraocular pressure (PIO)
# We convert age to categories to see the effect of age groups
data_glaucoma_copy$AGE_GROUP <- cut(data_glaucoma_copy$EDAD,
                                    breaks = c(0, 40, 50, 60, 70, Inf),
                                    labels = c("0-40", "41-50", "51-60", "61-70", "71+"),
                                    right = FALSE)

# Calculating the mean intraocular pressure for each age group
# I will use %>%, which is a functions pipe (the output of
# one function is the input of another one ), and the function group_by
pio_by_age_group <- data_glaucoma_copy %>%
  group_by(AGE_GROUP) %>%
  summarise(mean_PIO = mean(PIO_NORMAL, na.rm = TRUE))

# Results
print(pio_by_age_group)

# Identifying patients who experienced an increase in intraocular pressure after laser surgery
# Calculating the difference in PIO between 1 week, 1 month, and 3 months
data_glaucoma_copy <- data_glaucoma_copy %>%
  mutate(PIO_increase_1week_to_1month = PIO_1_MES - PIO_1_SEMANA,
         PIO_increase_1month_to_3month = PIO_3_MES - PIO_1_MES)

# We filter patients who had an increase in intraocular pressure
increased_pio_after_surgery <- data_glaucoma_copy %>%
  filter(PIO_increase_1week_to_1month > 0 | PIO_increase_1month_to_3month > 0)

# Number of patients with increased intraocular pressure
num_patients_with_increase <- nrow(increased_pio_after_surgery)
print(paste("Number of patients with increased intraocular pressure after laser surgery:", num_patients_with_increase))

# Relationship between intraocular pressure and medication
# Correlation between PIO at 1 month and medication use at 1 month
cor_PIO1M_FARMACOS1M <- cor(data_glaucoma_copy$PIO_1_MES, data_glaucoma_copy$FARMACOS_1_MES, use = "complete.obs")

# Correlation between medication use at different time points
cor_FARMACOS1M_FARMACOS3M <- cor(data_glaucoma_copy$FARMACOS_1_MES, data_glaucoma_copy$FARMACOS_3_MES, use = "complete.obs")
cor_FARMACOSPRE_FARMACOS1M <- cor(data_glaucoma_copy$FARMACOS_PRE, data_glaucoma_copy$FARMACOS_1_MES, use = "complete.obs")

print(paste("Correlation between PIO at 1 month and medication use at 1 month:", cor_PIO1M_FARMACOS1M))
print(paste("Correlation between medication use at 1 month and 3 months:", cor_FARMACOS1M_FARMACOS3M))
print(paste("Correlation between pre-operative medication use and medication use at 1 month:", cor_FARMACOSPRE_FARMACOS1M))

# Using boxplot to see distribution of PIO across age groups
ggplot(data_glaucoma_copy, aes(x = AGE_GROUP, y = PIO_NORMAL)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Intraocular Pressure by Age Group",
       x = "Age Group",
       y = "Intraocular Pressure (PIO)") +
  theme_minimal()

```

Explanation of the code above:

Since we want to analyze the effect of age on intraocular pressure, we have added a column called "AGE_GROUP", whose values are ranges of age, between 0 and more than 71. For that purpose, we have categorized the variable AGE by means of the function cut. Once we have the age groups, for each one we calculate the mean intraocular pressure, for what we use the functions group_by and summarise from the library dplyr, which enable us to calculate descriptive statistics, such us means, sums, etc., for each group, according to the chosen variables.

Now, we have the intraocular pressure of each age group, which would give us significant information.

Then, we want to identify the number of patients who experienced an increase in intraocular pressure after the surgery. So, for that motivation, we use the function mutate from the library dplyr, which enables us to add new columns to a data frame, or modify existing columns,on the basis of the values of other columns. With this function, we have added two new columns related to the increase of the PIO, from the first week to the first moth, and from the first month to the third. After that, we have calculated the number of patients who experienced the increase of PIO, by means of the function filter, from dplyr, which, given a condition, shows the data that satisfies it.

After that, we want to dig into the relationship between PIO and medication, for what we have extracted the correlation values between these variables (despite of the fact that we have just obtained them above, we wanted to use the function cor, applied to concrete variables, instead to all the quantitative, as we did before). These correlations will give us pretty significant information.

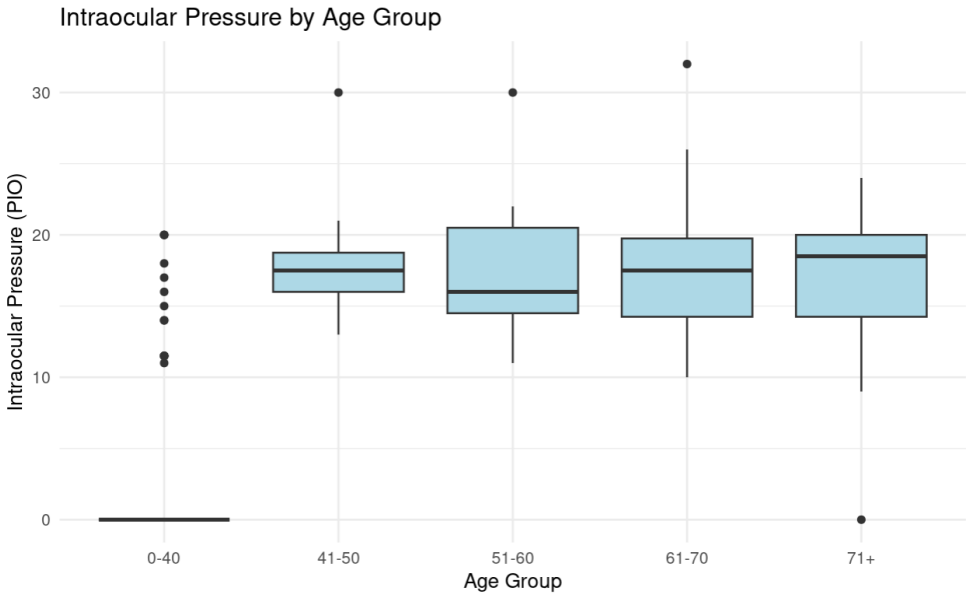
Finally, as a visual support, we have used a boxplot to see the distribution of PIO across age groups.

The result of the execution of the code is the following:

```
[1] "Number of patients with increased intraocular pressure after laser surgery: 36"
[1] "Correlation between PIO at 1 month and medication use at 1 month: 0.55714364673959"
[1] "Correlation between medication use at 1 month and 3 months: 0.629402749394311"
[1] "Correlation between pre-operative medication use and medication use at 1 month: 0.578538555128548"
```

AGE_GROUP <fctr>	mean_PIO <dbl>
0-40	3.371212
41-50	18.200000
51-60	17.727273
61-70	18.055556
71+	16.500000

5 rows



3 Results & Conclusions

Once we have analyzed the dataset, the correlation of the variables, and we have extracted significant information, we are ready to make a series of statements that will lead us to answer the question we asked ourselves in the beginning: whether the pre-surgery condition is related, in any way, to the long-term progression.

- The trend observed in mean intraocular pressure according to age groups **reinforces the idea that older patients are more likely to have higher IOP**. This suggests that preoperative conditions should be carefully evaluated in this group, as their age may increase the risk of postoperative complications.
- It has been observed that **36 patients (around 30%) experienced an increase of the IOP after the surgery**. This suggests that the surgery does not result in the expected improvement in IOP. This suggests that patients with more serious pre-existing conditions are more susceptible to experience an increase in postoperative IOP.

- The moderate **correlation between the use of preoperative medication and the use of them after 1 month** (0.58), shows that patients who required more drugs before the surgery, tend to continue to need medication after the surgery. This suggests that preoperative conditions may be a good predictor of the need for subsequent treatment and, therefore, the efficacy of surgery may be influenced by the patient's preoperative conditions.
- The **correlation between PIO and the use of medication at the firt month** (0.56), suggests that patients who present a higer IOP after the surgery, tend to need more medication. So, it is possible that patients with serious preoperative conditions need a more aggressive approach.

In view of the results obtained and analyzed, we conclude, that there is evidence that conditions prior to surgery have a significant relationship with conditions after surgery.

4 References

[1] Glaucoma https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8473801/

[2] https://eyes.arizona.edu/sites/default/files/glaucoma.pdf

[3] https://www.clinicbarcelona.org/en/assistance/diseases/glaucoma/risk-factors-and-causes

[4] https://www.glaucomapatients.org/basic/statistics/, https://glaucoma.org/articles/glaucoma-worldwide-a-growing-concern

[5] https://glaucoma.org/treatment/laser

[6] https://es.r4ds.hadley.nz/

[7] https://adv-r.hadley.nz/functions.html

[8] https://diytranscriptomics.com/Reading/files/The

[9] https://rsanchezs.gitbooks.io/rprogramming/content/chapter9/