2017 Clinic Dataset Analysis

Assisted reproduction clinics data from the CDC

Jorge Vallejo Ortega

2017 Assisted Reproduction Clinics Dataset Analysis: Practice of data analysis with R

Foreword

This work is inspired in an exercise from the course 'Software para el análisis de datos' in UOC's degree 'Bioinformática y Bioestadística'.

I am using this to learn the use of the language R, and the tools RStudio, Git and GitHub. This is not a professional level study of assisted reproduction data from clinics in the USA. If you want to have access to such studies, or even the raw data, you can get them from CDC's webpage ART's Success Rate Data.

Statement

MARKDOWN will be used for this paper, generating a Pdf report with Knitr in RStudio.

The file with the R code have to be delivered as well.

From a dataset, a statistical study must be carried out using R. The points below can be used as an outline:

1- Look for a dataset related with Biostatistics or Bioinformatics.

Must be public data. Explain source of data and include pertinent references. Justify why that specific dataset has been chosen.

2- Display the data.

Using R, display and explain the type of file that has been imported, which variables are included (type, classification,...) and anything else that seems relevant.
Include snapshots and R commands used for import and display of the data.

3- Probe questions.

Make a minimum of six questions that probe the kind of information contained in the dataset.

4- Descriptive analysis of the data.

The paper must include a parametric summary of the data and several graphic representations of said data.

5- Probability and simulation.

A minimum of three questions answering probability questions and a question corresponding a short simulation model.

6- Regression analysis.

A brief regression analysis from the variables in the dataset answering some question of interest.

7- Final assessment.

Final assessment from source data and analysis: Do we have conclusions? Would be necessary a more advanced analysis? Would be necessary more data for obtaining another kind of information?

1- Dataset

I wanted to do something with data from human assisted reproduction techniques. Those are difficult to find as public datasets, I suppose that it is due to privacy issues.

Finally, I found public datasets published by the Centers for Disease Control and Prevention (CDC) with several kinds of data from assisted reproduction clinics in the United States:
https://www.cdc.gov/art/artdata/index.html

Other datasets from previous years can be downloaded/looked up as well:
https://www.cdc.gov/art/reports/archive.html

Datasets downloading

```
# The code option eval = FALSE prevents this chunk to run. In this way, data WON'T
# be downloaded each time the .Rmd file is knitted to produce a report.
# For allowing this chunk to run, change eval to TRUE.

source_url <- "https://www.cdc.gov/art/artdata/docs/excel/FINAL-2017-Clinic-Table-Dataset.xlsx"

download.file(source_url, destfile = "../datos/FINAL-2017-Clinic-Table-Dataset.xlsx", method = "curl")
```

This is the dataset I am using for this practice study, but we could want to download datasets from previous years to do longitudinal studies:

```
# The code option eval = FALSE prevents this chunk to run. In this way, data WON'T
# be downloaded each time the .Rmd file is knitted to produce a report.
# For allowing this chunk to run, change eval to TRUE.

years <- c(1995:2015)

url_start1 <- "https://www.cdc.gov/art/excelfiles/clinic_tables_data_"
url_start2 <- "https://www.cdc.gov/art/excelfiles/"
url_start3 <- "https://www.cdc.gov/art/artdata/docs/excel/FINAL-"

file_start1 <- "clinic_tables_data_"
file_start2 <- ""
```

```
file_start3 <- "FINAL-"

url_end1 <- ".xls"
url_end2 <- "-clinic-tables-dataset.xls"
url_end3 <- "-clinic-table-dataset.xls"

folder <- "../datos/"

# This code is rough on the edges, it doesn't take into account when a file
# fails to download.

for (year in years){
  if (year <= 2012){
    url <- paste0(url_start1, year, url_end1)
    destfile <- paste0(folder, file_start1, year, url_end1)
  }
  else if (year == 2013){
    url <- paste0(url_start2, year, url_end2)
    destfile <- paste0(folder, file_start2, year, url_end2)
  }
    else if (year == 2014){
    url <- "https://www.cdc.gov/art/artdata/docs/excel/2014-Clinic-Tables-Data-Dictionary.xls"
    destfile <- paste0(folder, "2014-Clinic-Tables-Data-Dictionary.xls")
  }
  else {
    url <- paste0(url_start3, year, url_end3)
    destfile <- paste0(folder, file_start3, year, url_end3)
  }

  #cat(url, "\n", destfile, "\n\n") # Cheks that the names of url and files are rightly constructed.
  download.file(url, destfile, method = "curl")
}
```

2- Display data

The downloaded file *FINAL-2017-Clinic-Table-Dataset.xlsx* includes data from assisted reproduction treatments from 448 US clinics, collected during 2017. It is an Excel workbook containing four sheets:

National Summary Data. Aggregated data from all the clinics included in the report.
National Table Dictionary. Dictionary-table with the explanation for each variable in the previous table.
Clinic Table Data Records. Tabla with data broke down by clinic.
Clinic Table Dictionary. Dictionary-table explaining each variable from Clinic Table.

I am interested in the data broke down by clinic. Therefore, I will import the data from sheets **Clinic Table Dictionary** and **Clinic Table Data Records**.

```
library("xlsx")
clinic_dictionary <- read.xlsx("../datos/FINAL-2017-Clinic-Table-Dataset.xlsx", 4,
                               endRow = 165, encoding = "UTF-8")
# Everything is NA after row 166; row 166 is a comment.

View(clinic_dictionary)
```

```
# In Viewer, in the column Age the symbols '>=' are displayed as '=',
# but if we print the column to screen the correct symbols are displayed.

clinic_data <- read.xlsx("../datos/FINAL-2017-Clinic-Table-Dataset.xlsx", 3
                         , encoding = "UTF-8", stringsAsFactors=FALSE)
View(clinic_data)
```

Let's see how each variable has been codified:

```
str(clinic_data, list.len = length(clinic_data))
```

In the structure can be seen a problem from this data source. Percentages and ratios are coded as text, not as numbers. It will be necessary pre-processing the data from the table before we can work with them.

Why are numeric variables detected as characters? Ratios contain non-numeric symbols ("%", "/", "<"). What's the problem with the rest?

We will order alphabetically each column, see what happens.

Should we order the values in the columns, we see that there are columns with integers in which the thousands are marked with a comma (",").

```
ordenadas <- sapply(clinic_data, function(x) sort(x, na.last = TRUE))

View(ordenadas)
```

Next, we will correct those variables that should be `integer` o `numeric`.

Code adapted from a question in Stack Overflow: How to read data when some numbers contain commas as thousand separator?

```
# Vectors that refer the columns that should be integers
# and the columns that should be ratios.

integers <- c(7:11, 22:26, 37:41, 82:85, 94:99, 163)
ratios <- c(12:21, 27:36, 42:81, 86:93, 100:154)

# Pre-processing columns with integers deleting "," symbol.
clinic_data[, integers] <- lapply(clinic_data[, integers],
                                  function(x){
                                    as.integer(gsub(",", "", x))
                                  })

str(clinic_data[, integers])
```

Pre-processing the columns with ratios is a bit more complex. It is necessary to deal with three different cases:

1) Data as <1%. Since we don't have the real value, what I will do is arbitrarily chose the value 0.5%, expressed as 0.005 ratio.
2) Data including "%" symbol. Same strategy that we followed with integers; I will remove the symbol with `gsub`. Besides, I will divide the result by 100, making all numbers a ratio (parts per unit).
3) Data as fractions. Even a bit more complex. I will use regular expressions to isolate numerator and denominator, and will return the result of the division as a ratio.

```
# This function processes ratios expressed as fractions.
divide <- function(x){
  numerator <- as.numeric(gsub(" .*$", "", x))
```

```
  denominator <- as.numeric(gsub("^.*/ ", "", x))
  return (numerator / denominator)
}

# This is the main function for processing ratios from characters to numbers.
# Expect the appearance of several warnings of 'NA introduced by coercion' due to multiple # data point
numerizador <- function(dato){
    ifelse (dato == "<1%", 0.005,
            ifelse ((grepl("%", dato) == TRUE), as.numeric(gsub("%", "", dato))/100,
                    ifelse ((grepl("/", dato)==TRUE), divide(dato), as.numeric(dato))))
}
# The function 'ifelse' is needed for working with vectors of length >1.

# Pre-processing ratios:
clinic_data[, ratios] <- lapply(clinic_data[, ratios], numerizador)
# When using this expression, multiple warnings of 'NAs introduced by coercion' appear.
# I think it is due to "NA" being stored as character strings.
# I haven't been able to correct it yet. It is possible to avoid the raise of the warning,
# but I don't feel comfortable masking error warnings.

# Using chunk option warning=FALSE masks those error warnings.

str(clinic_data[, ratios], list.len = length(clinic_data[, ratios]))
```

The use of chunk option `warning=FALSE` prevents the displaying of multiple `NAs introduced by coercion` messages. I am not sure, but I suppose the cause for the warnings is that the`NA` values in the table are stored as text.

Variables: type and description

I might have allowed myself to get carried along in passion choosing a file too big. The table contains 165 variables. It is not a ridiculous number, but may be not easy enough to handle for the scope of this study.

```
str(clinic_data, list.len = length(clinic_data))
```

From those 165 variables, 17 are character strings, 25 are integers and 123 are non-integer numbers.

Character variables correspond to descriptive data as: clinic name, city of the clinic, state, name of the medical director, and the availability of several services(oocyte donation, embryo donation, oocyte cryopreservation, embryo cryopreservation, services for single women, gestational carriers, if the clinic is a member of SART (Society for Assisted Reproductive Technology), and if the clinic owns an accredited embryology laboratory). Many of these variables will work as factors.

Variables of type `integer`are:
OrderID: unique identifier for each register.
ND_NumIntentRet: number of intended oocyte retrievals (excluding donors). As in most of numerical variables in this table, this one is divided into five age categories, from less than 35 years old to more than 43 years old.
ND_NumRetrieve: number of oocyte retrievals (excluding donors). It doesn't match with intended retrievals because, sometimes, it is necessary to stop the retrieval process or a particular process is not successful.
ND_NumTrans: number of transfers (excluding donors). Transfer is the technique by with an embryo grown in vitro is transferred from cultured to the patient uterus.
Donor_NumTrans: number of transfers from donor oocytes. This category is divided into four variables depending if the oocytes are fresh of frozen, fresh or frozen embryo, and embryo from donor.
TotNumCycles: total number of cycles. In assisted reproduction, the cycles include any process in which

at least one of this conditions happen; 1) an assisted reproduction process is carried out, 2) the patient is subjected to ovarian stimulation or monitoring with the intent of having an ART procedure, or 3) frozen embryos have been thawed with the intent of transferring them to a patient.
NumResearch: number of excluded cycles for research.

Numeric variables are:
ND_IntentRetLB: Percentage of intended oocyte retrievals resulting in live births (excluding donors).
ND_IntentRetSingleLB: Percentage of intended oocyte retrievals resulting in singleton live births (excluding donors).
ND_RetrieveLB: Percentage of oocyte retrievals resulting in live births (excluding donors).
ND_RetrieveSingleLB: Percentage of oocyte retrievals resulting in singleton live births (excluding donors).
ND_TransLB: Percentage of transfers resulting in live births (excluding donors).
ND_TransSingleLB: Percentage of transfers resulting in singleton live births (excluding donors).
ND_IntentRetPerLB: Number of intended oocyte retrievals resulting in live births (excluding donors).
NewND_1IntentRetLB1: Percentage of new patients with live birth after one intended retrieval (excluding donors).
NewND_2IntentRetLB: Percentage of new patients with live birth after one or two intended retrievals (excluding donors).
NewND_AllIntentRetLB: Percentage of new patients with live birth after all intended retrievals (excluding donors).
NewND_IntentRetPerNew1: Average number of intended retrievals per new patient (excluding donors).
NewND_TransPerIntentRet1: Average number of transfers per intended retrieval (excluding donors).
Donor_TransLB: Percentage of transfers resulting in live births (only donors). This variable is divided in four variables according to the condition of the oocyte (fresh, frozen, frozen embryo, embryo from donor).
Donor_TranSingleLB: Percentage of transfers resulting in a singleton live birth (only donors). Divided in four variables according to the condition of the oocyte (fresh, frozen, frozen embryo, embryo from donor).
CycleCancel: Percentage of cycles canceled prior to retrieval or thawing.
CycleStop: Percentage of cycles canceled between retrieval and transfer or banking.
CycleFertPres: Percentage of cycles for fertility preservation.
TransCarrier: Percentage of transfers using gestational carrier.
TransFrozEmb: Percentage of transfers using frozen embryos.
TransICSI: Percentage of transfers of at least one embryo with ICSI.
TransPGT: Percentage of transfers of at least on embryo with PGT.
ReasonMale: Percentage of cycles for male factor reason.
ReasonEndo: Percentage of cycles for endometriosis reason.
ReasonTubal: Percentage of cycles for tubal factor reason.
ReasonOvul: Percentage of cycles for ovulatory dysfunction reason.
ReasonUterine: Percentage of cycles for uterine factor reason.
ReasonPGT: Percentage of cycles for PGT reason.
ReasonCarrier: Percentage of cycles for gestational carrier reason.
ReasonDOR: Percentage of cycles for diminished ovarian reserve reason.
ReasonBank: Percentage of cycles for banking reason.
ReasonPregLoss: Percentage of cycles for recurrent pregnancy loss reason.
ReasonOtherInfert: Percentage of cycles for other infertility reason.
ReasonNonInfert: Percentage of cycles for other non-infertility reason.
ReasonUnexplained: Percentage of cycles for unexplained reason.

3- Probe questions

Basic numeric summary for the total number of cycles by clinic (without taking patient age into account):

```
# Basic numeric summary
statistics <- c("average", "min.", "max.", "stdr.deviation", "C.V.", "25%", "50%", "75%")
```

```r
variables <- c("TotNumCyclesAll")
df <- clinic_data
est_vector <- c()

for (var in variables) {
  df_subset <- df[,var] # Extracts column corresponding to the variable
  var_sd <- sd(df_subset, na.rm=TRUE) # standard deviation
  var_mean <- mean(df_subset, na.rm = TRUE) # average
  var_min <- min(df_subset,na.rm = TRUE)
  var_max <- max(df_subset, na.rm = TRUE)
  var_CV <- var_sd/var_mean # coefficient of variation
  var_percentile <- quantile(df_subset, probs=c(0.25, 0.5, 0.75), names=FALSE, na.rm = TRUE) # percenti
  est_vector = c(est_vector, var_mean, var_min, var_max, var_sd, var_CV, var_percentile) # Adding all r
  }

# Transform the vector with all the results into a matrix, and that into a dataframe.
testmatrix <- matrix(data=est_vector, ncol=length(statistics), byrow = TRUE,
                     dimnames = list(variables, statistics))
testdf <- as.data.frame(testmatrix)

testdf
```

Frequency histogram displaying number of transfers from donor. Four graphs, each one for each of the starting states: fresh oocyte, frozen oocyte, frozen embryo, and donated embryo (in the other three cases it is only the oocyte which comes from a donor).

```r
par(mfrow = c(2, 2)) # generate a 2x2 array for printing the graphs.

xlab <- "Transfers by clinic"
ylab <- "Frequency"

# For better comparing the histograms, the values of the x-axis, y-axis, and the bins will be the same
htd_xlim <- c(0,  max(clinic_data[,c("Donor_NumTrans1","Donor_NumTrans2","Donor_NumTrans3","Donor_NumTra
htd_ylim <- c(0, 400)

hist(clinic_data$Donor_NumTrans1,
     main = "Fresh embryo from fresh oocyte", xlab = xlab, ylab = ylab,
     col = "blue", xlim = htd_xlim, ylim = htd_ylim,
     breaks = seq(0, max(clinic_data$Donor_NumTrans1)+5, 10))
hist(clinic_data$Donor_NumTrans2,
     main = "Fresh embryo from frozen oocyte", xlab = xlab, ylab = ylab,
     col = "tomato", xlim = htd_xlim, ylim = htd_ylim,
     breaks = seq(0, max(clinic_data$Donor_NumTrans2)+5, 10))
hist(clinic_data$Donor_NumTrans3,
     main = "Frozen embryo", xlab = xlab, ylab = ylab,
     col = "yellow", xlim = htd_xlim, ylim = htd_ylim,
     breaks = seq(0, max(clinic_data$Donor_NumTrans3)+10, 10))
hist(clinic_data$Donor_NumTrans4,
     main = "Embryo from donor", xlab = xlab, ylab = ylab,
     col = "green", xlim = htd_xlim, ylim = htd_ylim,
     breaks = seq(0, max(clinic_data$Donor_NumTrans4)+5, 10))
```

One of the variables is the state of accreditation for embryology laboratory. For this kind of data a table may be more useful than a graph:

```r
table(clinic_data$LabAccred)
```

Nevertheless, for other qualitative data we could find more interesting to display them in a graph. For example, how many clinics offer ART, by state.

```r
# Most of this code I adapted it from:
# https://stackoverflow.com/questions/10286473/rotating-x-axis-labels-in-r-for-barplot

par(mar = c(7, 4, 2, 2) + 0.2)

end_point <- 0.5 + length(unique(clinic_data$CurrentClinicState)) + length(unique(clinic_data$CurrentCli

barplot(sort(table(clinic_data$CurrentClinicState), decreasing = TRUE),
        # This adjust the maxim value in the y axis:
        ylim = c(0,5+max(table(clinic_data$CurrentClinicState))),
        xaxt = "n", # Avoids plotting the names in the x axis.
        space = 1)

text(seq(1.5,end_point,by=2), par("usr")[3]-0.25,
     srt = 60, adj= 1, xpd = TRUE,
     labels = paste(names(sort(table(clinic_data$CurrentClinicState), decreasing = TRUE))), cex=0.55)
```

Does the number of intended retrievals change with age? Let's use a box plot graphic:

```r
# To avoid problems with the logarithm of 0, let's change the value 0 by value 1.
intended_retrievals <- clinic_data[,7:11]
intended_retrievals[intended_retrievals == 0] <- 1

boxplot(intended_retrievals,
        log = "y",
        yaxt = "n",  # Don't draw the ticks in y axis.
        xlab = "Age groups",
        ylab = "Intended Retrievals by Clinic",
        main = "Intended Retrievals by Clinic and Age Group",
        names = c("<35", "35-37", "38-40", "41-42", ">=43"))

# Establishes limits for y axis and, from base 10 logarithm,
# max and min values of the dataframe.
y1 <- floor(log10(range(intended_retrievals)))
# Vector with integer values from minimum to maximum for the axis.
pow <- seq(y1[1], y1[2]+1)
# Vector with ticks' positions.
ticksat <- as.vector(sapply(pow, function(p) (1:10)*10^p))
# Drawing the axis (main ticks)
axis(2, 10^pow)
# Drawing the axis (secondary ticks)
axis(2, ticksat, labels = NA, tcl = -0.25, lwd = 0, lwd.ticks = 1)
```

I suspect that the median diminishes with age group because donor eggs are preferred for older women instead of retrieval of their own eggs.

Equally, we can wonder about the percentage of intended oocyte retrievals that results in live births. That would give a little guiding about if it is worth the effort of retrieve oocytes from patients of a relatively advanced age.

```r
# Ratio of intended retrievals resulting in live births, by age
pct_int_ret <- clinic_data[,12:16]

boxplot(pct_int_ret,
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of intended retrievals resulting in live births \nby hospital and age grou
        names = c("<35", "35-37", "38-40", "41-42", ">=43"))
```

As was suspected, this graph shows a marked descent in the ratio of intended oocyte retrieval resulting in live births when taking into account the age group of the patient.

4- Descriptive analysis

In this section we will accomplish a more thorough exploration of our variables, including a parametric summary of data, and graphic representations.

Services

Summary of services offered by the clinics.

```r
# Columns including the data for the services: 155-160 and 162.
services <- clinic_data[c(155:160, 162)]

# Variables' names:
services_names <- c("Oocyte donation",
  "Embryo donation",
  "Embryo cryopreservation",
  "Oocyte cryopreservation",
  "Services for single women",
  "Gestational carriers",
  "Accredited embryo laboratory")

colnames(services) <- services_names

lapply(services, table)
```

Results from assisted reproduction techniques

Descriptive summary of numeric variables:

```r
variables <- colnames(clinic_data[c(7:154, 163)])
statistics <- c("average", "min.", "max.", "stdr.deviation", "C.V.", "25%", "50%", "75%")
est_vector <- c()

for (var in variables) {
    df_subset <- clinic_data[,var] # Extract value column for the corresponding variable
    var_sd <- sd(df_subset, na.rm=TRUE) # standard deviation
    var_mean <- mean(df_subset, na.rm = TRUE) # average
    var_min <- min(df_subset,na.rm = TRUE)
    var_max <- max(df_subset, na.rm = TRUE)
```

```
    var_CV <- var_sd/var_mean # coefficient of variation
    var_percentile <- quantile(df_subset, probs=c(0.25, 0.5, 0.75), names=FALSE, na.rm = TRUE) # percen
    est_vector = c(est_vector, var_mean, var_min, var_max, var_sd, var_CV, var_percentile) # Adding all
  }

  # Transform the vector with all the results into a matrix, and that into a dataframe.
  testmatrix <- matrix(data=est_vector, ncol=length(statistics), byrow = TRUE,
                       dimnames = list(variables, statistics))
  testdf <- as.data.frame(testmatrix)

  testdf
```

Graphic representations

Retrievals

Egg retrieval is a procedure to collect the eggs contained in the ovarian follicles.

```
# Since I think I am going to need this code several times, I will wrap it into a function.

log_boxplot <- function(x, ...){
  # To avoid problems with the logarithm of 0, let's change the value 0 by value 1.
  x[x == 0] <- 1

  boxplot(x,
        log = "y",
        yaxt = "n",  # Do not draw ticks in y axis.
        ...) # Additional arguments to be passed to the function boxplot.

# Establishes limits for y axis and, from base 10 logarithm,
# max and min values of the dataframe.
  y1 <- floor(log10(range(x, na.rm = TRUE)))
# Vector with integer values from minimum to maximum for the axis.
  pow <- seq(y1[1], y1[2]+1)
# Vector with ticks' positions.
  ticksat <- as.vector(sapply(pow, function(p) (1:10)*10^p))
# Drawing the axis (main ticks)
  axis(2, 10^pow)
# Drawing the axis (secondary ticks)
  axis(2, ticksat, labels = NA, tcl = -0.25, lwd = 0, lwd.ticks = 1)
}
```

```
# Let's actually draw the graphs.

par(mfcol = c(3, 1))

# Egg retrievals by clinic
x <- clinic_data[,22:26]
log_boxplot(x,xlab = "Age groups", ylab = "Egg Retrievals by Clinic",
        main = "Egg Retrievals by Clinic and Age Group",
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")
```

```
# Ratio of retrievals resulting in live births
boxplot(clinic_data[c(27:31)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Retrievals Resulting in Live Births \nby Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "lightgoldenrod1")

# Ratio of retrievals resulting in singleton live births
boxplot(clinic_data[c(32:36)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Retrievals Resulting in Singleton Live Births \nby Hospital and Age Grou
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "coral")
```

I have had some problems with this image. Labels and points are more little than I wanted, but at least I have managed to include all three graphs into the same image.

Intended egg retrievals take into account even those retrievals that failed because the procedure had to be interrupted, because no egg were retrieved or due to other causes.

There are some statistics associated to the intended egg retrievals:

```
par(mfcol = c(6, 1))

# Number of intended egg retrievals per live birth
x <- clinic_data[,52:56]
log_boxplot(x,xlab = "Age groups", ylab = "Intended Egg Retrievals",
        main = "Number of Intended Egg Retrievals by Clinic and Age Group",
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")

# Ratio of new patients with live births after 1 intended egg retrieval
boxplot(clinic_data[c(57:61)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of New Patients with Live Births After One Intended Egg Retrieval by Clini
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "lightgoldenrod1")

# Ratio of new patients with live births after 1 or 2 intended egg retrievals
boxplot(clinic_data[c(62:66)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of New Patients with Live Births After One or Two Intended Egg Retrieval by
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "coral")

# Ratio of new patients with live births after all intended egg retrievals
boxplot(clinic_data[c(67:71)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of New Patients with Live Births After All Intended Egg Retrieval by Clini
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
```

```
        col = "darkseagreen")

# Average number of intended retrievals per new patient
boxplot(clinic_data[c(72:76)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Average Number of Intended Retrievals per New Patient by Clinic and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")

# Average number of transfers per intended retrievals
boxplot(clinic_data[c(77:81)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Average Number of Transfers per Intended Retrieval per New Patient by Clinic and
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "lightgoldenrod1")
```

Also in this set of variables the effect of the age group seems clear in all of them.

Transfers

Transfer is a procedure in which an embryo is transferred, from laboratory culture, to the uterus.

Like above, we will represent number of transfers, percentage of transfers resulting in live births, and those resulting in singleton live births.

```
par(mfcol = c(3, 1))

# Number of transfers
x <- clinic_data[,37:41]
log_boxplot(x,xlab = "Age groups", ylab = "Transfers by Clinic",
        main = "Number of Transfers by Clinic and Age Group",
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")

# Ratio of transfers resulting in live births
boxplot(clinic_data[c(42:46)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers Resulting in Live Births \nby Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "lightgoldenrod1")

# Ratio of transfers resulting in singleton live births
boxplot(clinic_data[c(47:51)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers Resulting in Singleton Live Births \nby Hospital and Age Grou
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "coral")
```

Look at the median value of the percentage of transfers resulting in live births for the >=43 years age group. It is zero. If we look up the value of the mean, it is a just bit better with 0.11.

All in all, even the median ratio for the youngest age group is not that great, a tad under 0.5 (0.48).

Transfers from a donor is a possibility when the own oocytes from the patient can not be used.

```
par(mfcol = c(3, 1))

# Number of transfers from donor
x <- clinic_data[,82:85]
log_boxplot(x,xlab = "Groups", ylab = "Number of transfers",
        main = "Number of Transfers from Donor by Clinic and Age Group",
        names = c("Fresh Egg", "Frozen Egg", "Frozen Embryo", "Donated Embryo"),
        col = "aliceblue")

# Ratio of transfers resulting in live births
boxplot(clinic_data[c(86:89)],
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers from Donor Resulting in Live Births by Hospital and Age Group
        names = c("Fresh Egg", "Frozen Egg", "Frozen Embryo", "Donated Embryo"),
        col = "lightgoldenrod1")

# Ratio of transfers resulting in singleton live births
boxplot(clinic_data[c(90:93)],
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers from Donor Resulting in Singleton Live Births by Hospital and
        names = c("Fresh Egg", "Frozen Egg", "Frozen Embryo", "Donated Embryo"),
        col = "coral")
```

We see than the most common source of transfer from donor is the frozen embryo. It makes sense, since it does not require to coordinate egg retrieval from the donor with patient's receptivity.

On the other hand, there does not seem to be much difference in the ratio of live births between the different sources (fresh or frozen egg, frozen embryo, and donated embryo).

Gestational carriers, frozen embryos, ICSI and PGT

A *gestational carrier* is a woman who gestates an embryo originated from the egg of another woman. The expectation is that the newborn will be given to another woman or couple. People who use gestational carriers my include women with no uterus or women who cannot carry a pregnancy due to health problems; also, gay couples.

Frozen embryos are embryos obtained by IVF (in vitro fertilization) that have been cryopreserved.

ICSI (intracytoplasmatic sperm injection) is a technique of IVF in which a sperm in injected directly into the egg.

PGT (Preimplantation Genetic Testing) is the diagnosis of a genetic condition prior to achievement of pregnancy. It is performed on embryos obtained by IVF to prevent pregnancies affected by a genetic condition or chromosomal disorder.

In the next graph we will represent the data of the ratio of transfers that use a gestational carriers, those that use frozen embryos, those in which PGT has been performed, and those in which at least one embryo has been obtained by ICSI:

```
# Ratio of transfers (gestational carriers, frozen embryos, ICSI)
boxplot(clinic_data[c(123, 129, 135, 141)],
```

```
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers Using Gestational Carriers, Frozen Embryos, PGT, or ICSI"),
        names = c("Gestational Carriers", "Frozen Embryos", "ICSI", "PGT"),
        col = "aliceblue")
```

Seen the data represented, we can say that gestational carrier transfers are a minority in most of the clinics (median 0.02), with notable exceptions. On the other hand, both the use of frozen embryos (median 0.67), and the ICSI technique (median 0.80) are normalized and very frequent. Transfers of embryos that have undergone PGT have also a low ratio (median 0.23), but not as low as gestational carriers and the distribution of ratios is wider. The relative rarity of PGT testing is probably due to it being an expensive and time-consuming technique, that usually is performed only when a possible genetic condition is suspected.

I was wondering about those two clinics with a ratio higher than 0.6 of transfers to gestational carriers.

```
carrier_clinic <- clinic_data[clinic_data$TransCarrierAll > 0.6,]

carrier_clinic$CurrentClinicName1
carrier_clinic$CurrentClinicCity
carrier_clinic$CurrentClinicState
paste((carrier_clinic$TransCarrierAll) * 100,"%", sep = "")
```

Clinics:
Cities:
States:
Pct of transfers to gestational carriers:

For anyone interested, a quick query in a search engine provides us with the websites of the clinics:
Western Fertility Institute
Reproductive Sciences Medical Center

We can break down the ratios of transfers into age groups for gestational carriers, transfers of frozen embryos, embryos obtained by ICSI, and embryos subjected to PGT.
I am assuming here that the age is that of the patient.

```
par(mfcol = c(4, 1))

# Ratio of transfers to gestational carriers
boxplot(clinic_data[c(118:122)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers to Gestational Carriers by Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "lightgoldenrod1")

# Ratio of transfers using frozen embryos
boxplot(clinic_data[c(124:128)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers Using Frozen Embryos by Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "coral")

# Ratio of transfers with embryos obtained by ICSI
boxplot(clinic_data[c(130:134)],
        xlab = "Age groups",
```

```
        ylab = "Ratio",
        main = bquote("Ratio of Transfers Using Embryos obtained by ICSI by Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "green")

# Ratio of transfers with embryos subjected to PGT
boxplot(clinic_data[c(136:140)],
        xlab = "Age groups",
        ylab = "Ratio",
        main = bquote("Ratio of Transfers Using Embryos Subjected to PGT by Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")
```

In the case of the transfers to gestational carriers, the distributions by ages doesn't seem to change much except for <35 and >=43 years old patients. It seems to be a service used slightly less often by younger patients, and somewhat more often by older patients compared with the groups covering the 35 to 42 years old range. In any case, the median of the ratio is close to 0 in all the groups.

The distribution of the ratio of the use of frozen embryos in transfer remains constant for all age groups, with their medians grazing the 0.7. It may be said the same for the use of embryos obtained by ICSI, except for the >=43 years old group, in which this technique seems slightly less common (even the median for the ratio lowers to 0.7 from around 0.8 in the rest of the groups).

When observing the ratio of transfers in which at least one of the embryos have been subjected to PGT, the distributions by age are very similar. I was expecting a rise of the median for the more aged groups, since their eggs would be more prone to experiment chromosomal abnormalities. Nevertheless, it may be that the more frequent use of donor eggs by those groups a reason to not use PGT more that the younger age groups.

Number of cycles

ART cycles start when a woman begins taking fertility drugs or having her ovaries monitored for follicle production. If eggs are produced, the cycle progresses to egg retrieval. The eggs are combined with sperm to create embryos and at least one embryo is selected for transfer. If implantation of the embryo occurs, the cycle may progress to clinical pregnancy and sometimes live birth.
Keep in mind that the woman may not necessarily be the patient, but a donor. The egg may be frozen and thawed before fertilization, or the embryo may be frozen and thawed before implantation.

```
# Total number of cycles by age group
x <- clinic_data[,94:98]
log_boxplot(x,xlab = "Age groups", ylab = "Number of cycles",
        main = "Number of Cycles by Clinic and Age Group",
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")
```

This results -in a way- have surprised me, since I was expecting more cycles for the higher age groups. It does not seem to be the case, although we don't know if the differences are significant.

Fate of the cycles

Not all cycles end successfully, and not all cycles are started to achieve pregnancy. Sometimes the cycles are cancelled and no eggs are retrieved, or the eggs are lost without being used or stored. Sometimes, the aim of the cycles is not immediately looking for a pregnancy, but storing eggs as a precaution against an estimated loss of fertility (due to illness or age).

In the dataset we have three of such cases: Cycles canceled prior to retrieval or thaw, cycles stopped between retrieval and transfer or banking, and cycles for fertility preservation.

```
# Fate of cycles
boxplot(clinic_data[c(105, 111, 117)],
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Cycles Interrupted Before Transfer"),
        names = c("Canceled", "Stopped", "Fertility preservation"),
        col = "lightgoldenrod1")
```

It is curious that cycles for fertility preservation are the less common ones, but there is a clinic in which most of the cycles are performed for such an end.

We can easily extract the name and other data of the clinic from the database:

```
preserv_clinic <- clinic_data[clinic_data$CycleFertPresAll > 0.8,]
```

Clinic:
City:
State:
Total number of cycles:
Pct of cycles for fertility preservation:

For anyone interested, a quick query in a search engine provides us with the website of the clinic: https://extendfertility.com/

Cancelled cycles by age

Cancelled cycles are cycles that have been started but where cancelled prior to the retrieval of fresh eggs or the thawing of cryopreserved eggs. Here, we want to see if the age of the patient may have any effect on the ratio of cancelled cycles.

```
# Cancelled cycles by age group
boxplot(clinic_data[c(100:104)],
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Cancelled Cycles by Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "aliceblue")
```

The median doesn't really change between the age groups; the distribution of the data points, though, keeps expanding with the age of the patients.
Also, keep in mind that the ratio is not of the total of cycles, but the cycles for each age group.

Stopped cycles by age

Stopped cycles are those cycles that started but where cancelled between retrieval and transfer (or banking). We will visualize these data to see if the age of the patient may have an effect on the ratio of stopped cycles.

```
# Stopped cycles by age group
boxplot(clinic_data[c(106:110)],
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Stopped Cycles by Hospital and Age Group"),
```

```
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "lightgoldenrod1")
```

This graphic is similar to that of cancelled cycles by age. There is no much difference in the median of the different groups, but the distribution grows wider with the age of the patient, especially that of the outliers.

Fertility preservation cycles

Fertility preservation cycles are those started with the intent to freeze all resulting eggs or embryos for 12 months or longer in order to preserve future fertility. Let's see which age group makes more use of this tool.

```
# Stopped cycles by age group
boxplot(clinic_data[c(112:116)],
        xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Fertility Preservation Cycles by Hospital and Age Group"),
        names = c("<35", "35-37", "38-40", "41-42", ">=43"),
        col = "coral")
```

This case shows an interesting trend in which the medians are slightly higher in the groups ranging 35 to 42 years old patients; the distribution of the data is quite narrow around the median in all groups; and the distribution in the group of patients older than 43 year has almost collapsed on the value of the median (0.0). Nevertheless, there is an impressive quantity of high outliers in every group (an outlier here taken as any data point more than 1.5 times the length of the box).
That trend in these distributions is more or less expected since patients of more age would prefer to achieve pregnancy as soon as possible instead of saving eggs for the future. While younger patients may be still developing their professional careers and would want to preserve good quality eggs for later in life. Also, if the cause to wanting to store the eggs is related to an illness, older women are more probable to have already had children, while younger women may not, and then would want to store healthy egg before a treatment that could sterilize them.

Cause of the cycles

In this section we will examine the data that describe the reason behind the ART cycles performed by the clinics. Most of them are directly related to infertility (male factor, endometriosis, tubal factor, and so on), while some are aimed to prevent genetic disorders (PGT), preservation of fertility (banking) or unknown.

```
# Cause of the cycles
boxplot(clinic_data[c(142:154)],
        #xlab = "Groups",
        ylab = "Ratio",
        main = bquote("Ratio of Cycles for Different Reasons by Hospital and Cause"),
        names = c("Male factor", "Endometriosis", "Tubal factor", "Ovulatory dysfunction",
                  "Uterine factor", "PGT", "Gestational carrier", "DOR", "Banking",
                  "Recurrent pregnancy loss", "Other infertility", "Other non-infertility",
                  "Unexplained"),
        col = "aliceblue",
        xaxt = "n")

cycle_labels <- c("Male factor", "Endometriosis", "Tubal factor", "Ovulatory\ndysfunction",
                  "Uterine factor", "PGT", "Gestational\ncarrier", "DOR", "Banking",
                  "Recurrent\npregnancy loss", "Other infertility", "Other\nnon-infertility",
                  "Unexplained")
```

```
cycle_end_point <- 0.5 + length(cycle_labels)

text(seq(1,cycle_end_point,by=1), par("usr")[3]-0.05,
     srt = 60, adj= 1, xpd = TRUE,
     labels = cycle_labels,
     cex=0.8)
```

The most frequent causes behind ART cycles are male factor (which encompasses all the causes dependent on the males), diminished ovarian reserve (DOR), and banking.
After that, come ovulatory dysfunction, tubal factor,infertility due to causes not specified and causes unexplained). The least frequent causes are endometriosis, uterine factor, PGT, recurrent pregnancy loss, causes not related to infertility but not specified, and -almost anecdotal- cycles for using a gestational carrier.

What stands out for me is that, even for clinics in which the ratio of transfers with the objective of using gestational carriers, the ratio of cycles committed to gestational carriers is very low:

```
cycle_carrier_table <- clinic_data[clinic_data$CurrentClinicName1 == "WESTERN FERTILITY INSTITUTE" | cli

knitr::kable(cycle_carrier_table,
             format = 'pandoc', # I have to specify this format to print the caption
             row.names = FALSE,
             col.names = c("Clinic Name", "Ratio transfers", "Ratio cycles"),
             align = c('l','c','c') , # column alignment: left, center, center
             caption = "Comparing ratio of transfers with ratio of cycles,
             for gestational carriers")
```

I don't know which is the reason for that difference.

5- Probability and simulation

5.1 Probability distribution of variable CurrentCityClinic.

Let's take a variable; for example, the name of the city (CurrentCityClinic). How many clinics are in each city that has at least one clinic?

```
# The table below fuses variables for city and state, for distinguising
# account those cities with the same name in different states.
cities_by_clinic <- sort(table(
  paste(clinic_data$CurrentClinicCity, clinic_data$CurrentClinicState, sep = ", ", collapse = NULL)),
  decreasing = TRUE)

knitr::kable(cities_by_clinic,
             format = "pandoc",
             col.names = c("City", "Clinics"),
             caption = "Number of clinics in each city")
```

If we represent it in a bar plot:

```
# Most of this code I adapted it from:
# https://stackoverflow.com/questions/10286473/rotating-x-axis-labels-in-r-for-barplot

#par(mar = c(7, 4, 2, 2) + 0.2)

#end_point <- 0.5 + length(unique(clinic_data$CurrentClinicState)) + length(unique(clinic_data$CurrentC
```

```
barplot(cities_by_clinic,
        # This adjusts the maxim value in the y axis:
        ylim = c(0,4+max(table(clinic_data$CurrentClinicCity))),
        xaxt = "n", # Avoids plotting the names in the x axis.
        space = 1,
        main = "Number of clinics by city",
        xlab = "Cities (names not represented due to lack of space)",
        ylab = "Clinics")

#text(seq(1.5,end_point,by=2), par("usr")[3]-0.25,
#     srt = 60, adj= 1, xpd = TRUE,
#     labels = paste(unique(clinic_data$CurrentClinicState)), cex=0.55)
```

We see that there are few cities with several clinics, and many cities with only one or two clinics.

We can get a contingency table:

```
clinics_frequency <- table(cities_by_clinic)

clinics_frequency
```

Which informs us that we got cities with one fertility clinic in them, and one city with clinics in it.

```
clinic_cities <- length(cities_by_clinic)
no_clinic_cities <- 19495 - clinic_cities
```

In total, we have cities with at least a fertility clinic.

According to the United States Census Bureau, there are 19,495 cities (defined as incorporated places, with some exceptions) in the country. Which means that there are cities with no fertility clinic.

Let's add this datum to the table and represent it in a bar plot. We will add counts with zero frequency as well, it will be useful later.

```
clinics_frequency_dataframe <- rbind(
  data.frame(cities_by_clinic = as.character(c(0, 9:20)),
             Freq = c(no_clinic_cities, rep(0, length(9:20)))),
  as.data.frame(clinics_frequency))

#Order clinics_frequency_dataframe by $cities_by_clinic
clinics_frequency_dataframe <- clinics_frequency_dataframe[order(as.numeric(as.character(
  (clinics_frequency_dataframe$cities_by_clinic)))), ]

barplot(height = clinics_frequency_dataframe$Freq,
        names.arg = clinics_frequency_dataframe$cities_by_clinic,
        ylim = c(0,signif(max(clinics_frequency_dataframe$Freq), digits = 1)),
        #log = "y",
        ylab = "Frequency", xlab = "Fertility clinics in a city",
        cex.names = 0.7
        )
```

If we represent it in a logarithmic scale:

```
barplot(height = (clinics_frequency_dataframe$Freq)+1, # To avoid errors converting to log
        names.arg = clinics_frequency_dataframe$cities_by_clinic,
        ylim = c(1,signif(max(clinics_frequency_dataframe$Freq), digits = 1)),
        log = "y",
```

```
        ylab = "Frequency (log10)", xlab = "Fertility clinics in a city",
        cex.names = 0.7
        )
```

We can see that the number of cities with not even one fertility clinic is many more (by two orders of magnitude) than those that have only one clinic, which are one order of magnitude more than those that have at least two clinics. This signals that having a fertility clinics is a very improbable characteristics for a randomly chosen city in the United States.

The Poisson distribution is used for frequency distribution of counts of rare but independent events. It is possible, then, that the frequency of clinics follows a binomial.

We will try to reproduce the above bar plot with a random generation for the Poisson distribution with parameter *lambda* equal to the mean count per sample.

```
# Generate a vector with the number of clinics in a city
counts <- as.numeric(as.character(clinics_frequency_dataframe$cities_by_clinic))

# Generate a vector with the number of cities for each count
freq <- clinics_frequency_dataframe$Freq

# Generate a vector with all the counts
vector_counts <- rep(counts, freq)

lambda <- mean(vector_counts)

# Generate a random vector from the poisson distribution with the same length
# that our case and lambda equal to our mean.
set.seed(92)
simulated_counts <- rpois(length(vector_counts), lambda)

table(simulated_counts)
```

The data shown in the table doesn't look like a good match for our data.

Another possibility is the **negative binomial distribution**, for cases in which the variance is much greater than the mean.
In our case:

```
var(vector_counts)/mean(vector_counts)
```

The variable is three times the mean. It would be 1 if the data were Poisson distributed.
Let's make a rough estimate of the clumping parameter k:

```
# The formulae are adapted from _The R Book_, section 7.4.7
mean(vector_counts)^2/(var(vector_counts)-mean(vector_counts))
```

A function that computes the maximum likelihood estimate of **k** from a vector of frequencies of counts:

```
#From _The R Book_, section 7.4.7
kfit <- function(x) {
  lhs <- numeric()
  rhs <- numeric()
  y <- 0:(length(x) - 1)
  j <- 0:(length(x) - 2)
  m <- sum(x * y)/sum(x)
  s2 <- (sum(x * y^2) - sum(x * y)^2/(sum(x))/(sum(x)) - 1)
  k1 <- m^2/(s2 - m)
```

```
  a <- numeric(length(x)-1)

  for (i in 1:(length(x) - 1)) a[i] <- sum(x [- c(1:i)])
  i <- 0
  for (k in seq(k1/1.2, 2*k1, 0.001)) {
    i <- i+1
    lhs[i] <- sum(x) * log(1 + m/k)
    rhs[i] <- sum(a/(k + j))
  }
  k <- seq(k1/1.2, 2*k1, 0.001)
  #plot(k, abs(lhs-rhs), xlab="k", ylab = "Difference", type = "l", col = "red")
  d <- min(abs(lhs-rhs))
  sdd <- which(abs(lhs-rhs)==d)
  k[sdd]
}
```

Trying it with the clinics count data:

```
k_for_freq <- kfit(freq)

k_for_freq
```

So, the maximum likelihood of k is this case is .

How would a negative binomial distribution with a mean of and that k value describe our count data? The expected frequencies are obtained by multiplying the probability density by the total sample size (in this case).

```
cities <- clinic_cities+no_clinic_cities
k <- k_for_freq
terms <- (0:(length(freq)-1))
negative_binomial <- cities*(1+lambda/k)^(- k)*factorial(k+ terms -1)/
  (factorial(terms)*factorial(k-1))*(lambda/(lambda+k))^(terms)
```

We will compare observed and expected frequencies using bar plot, alternating the bars for observed and expected (*from* The R Book, *section 7.4.7*).

```
#Concatenate observed and expected frequencies in an alternating sequence.
# Put the observed counts (freq) in the odd-numbered bars and the expected counts
# (negative_binomial) in the even-numbered bars.
both <- numeric(length(freq)*2)

both[1:length(both) %% 2 != 0] <- freq

both[1:length(both) %% 2 == 0] <- negative_binomial

#Create list of labels to name the bars (alternating blanks and counts)
# Produce a vector of appropiate lenght (both) containing the repeating bar labels,
# then replace the even-numbered entries with blanks.
labs <- as.character(rep(0:(length(freq)-1), each=2))

labs[1:(length(labs))%%2==0] <- " "

#Draw the barplot
barplot_legend <- function(x) {
barplot(x, col=rep(c(3,4), length(freq)), ylab = "Frequency", names=labs,
```

```
        cex.names = 0.7)

#Create a legend
legend(x = 15, y= 15000, legend=c("Observed", "Expected"), fill=(c(3,4)))
}

barplot_legend(both)
```

To the naked eye, the fit between both distributions is close. But we might want to measure the lack of fit between observed and expected distributions. At the moment, I don't know which would be the most adequate test.
Nevertheless, the assumptions of this probability distribution aren't valid. Each trial should be independent; however, it is reasonable to assume that the existence of a fertility clinic in a city depends on the city's population. Also, cities around a city that already has a fertility clinic may have less probability of having a fertility clinic.

5.2 Probability distribution of variable Donor_NumTrans2.

Let's take another variable; Donor_NumTrans2 (Number of transfers of fresh embryos from a frozen egg from donor). Which distribution follows this variable?

```
donor_trans_2 <- clinic_data$Donor_NumTrans2

#table(donor_trans_2)

hist(donor_trans_2, breaks = (-0.5:170.5), ylim = c(0, 200), xlim = c(0, 200),
     main = "Transfers of fresh embryos from a frozen egg from donor",
     xlab = "Transfers", ylab = "Frequency", col = "aliceblue")
```

This once again looks like a Poisson distribution or a negative binomial, where events (a particular kind of transfer in this case) have low probability.
Nevertheless, we may think that the number of transfers is heavily dependent on the size of the clinic. We would want to look instead at the ratio of this kind of transfers to the total number of transfers in each clinic. For each clinic, we need to divide this variable (Donor_NumTrans2) between the sum of all the variables involving transfers from donor (Donor_NumTrans1 to 4):

```
trans_donor_variables <- c("Donor_NumTrans1","Donor_NumTrans2","Donor_NumTrans3","Donor_NumTrans4")

ratio_donor_trans2 <- clinic_data$Donor_NumTrans2 / rowSums(clinic_data[,trans_donor_variables])

hist(ratio_donor_trans2, main = "Transfers of fresh embryos from a frozen egg from donor",
     xlab = "Ratio", ylab = "Frequency of ratios", col = "aliceblue")
```

Keep in mind that, this time, we are excluding those clinics with no transfers from donor whatsoever (dividing by zero produces NaN results, which are not included in the histogram).
Still, the most frequent by large is the ratio ranging from 0 to 10%.
Since the values of x (the ratio) are continuous and bounded between 0 and 1, I would say that it could be described by the *beta probability distribution*.

```
# From https://stats.stackexchange.com/questions/12232/calculating-the-parameters-of-a-beta-distributio

estBetaParams <- function(mu, var) {
  alpha <- ((1 - mu) / var - 1 / mu) * mu ^ 2
  beta <- alpha * (1 / mu - 1)
  return(params = list(alpha = alpha, beta = beta))
```

```
}

# We can obtain alpha and beta:

mu <- mean(ratio_donor_trans2, na.rm = TRUE)
var <- var(ratio_donor_trans2, na.rm = TRUE)

beta_ratio <- estBetaParams(mu, var)

hist(ratio_donor_trans2, breaks = seq(0,1,0.01), main = "Transfers of fresh embryos from a frozen egg fi
     xlab = "Ratio", ylab = "Density", col = "aliceblue", freq = FALSE)
lines(seq(0, 1, 0.01), dbeta(seq(0, 1, 0.01), beta_ratio[[1]], beta_ratio[[2]]))
```

I does not look bad. Although I wonder if we could use the negative binomial (or the gamma distribution). The relation between mean and variance:

```
var/mu
```

The mean is quite higher than the variance, which leads away from the negative binomial distribution, but away as well from the Poisson distribution (which is characterized for having the same mean and variance). Maybe the answer is in the gamma distribution.

```
# Calculate the two parameter values for the gamma distribution
rate <- mean(ratio_donor_trans2, na.rm = TRUE)
shape <- rate*mean(ratio_donor_trans2, na.rm = TRUE)

# Draw the histogram
hist(ratio_donor_trans2, breaks = seq(0,1,0.01), main = "Transfers of fresh embryos from a frozen egg fi
     xlab = "Ratio", ylab = "Density", col = "aliceblue",
     freq = FALSE)

# Draw the density function

lines(seq(0, 1, 0.01), dgamma(seq(0, 1, 0.01), shape, rate))
```

Worse, I would definitively stick to the beta distribution.

In short, **I am accepting the beta distribution as the probability distribution for this variable** (ratio of transfers of fresh embryos from frozen donor's egg). But I still have doubts about it.

5.3 Probability distribution of variable ND_IntentRetLB2.

Let's take a last variable for a probability question; ND_IntentRetLB2 (Ratio of intended retrievals resulting in live births for patients aged 35-37 years old). Which distribution follows this variable?

```
#table(clinic_data$ND_IntentRetLB2)

#max(table(clinic_data$ND_IntentRetLB2))

hist(clinic_data$ND_IntentRetLB2, breaks = seq(0,1,0.01),
     main = "Ratio of intended retrievals \nresulting in live births for patients aged 35-37 years old"
     xlab = "Ratio", ylab = "Frequency", col = "aliceblue")
```

This looks like a normal distribution skewed to the right. Maybe a gamma, then.

```r
# Draw histogram with density instead of frequency of counts
hist(clinic_data$ND_IntentRetLB2, breaks = seq(0,1,0.01),
     main = "Ratio of intended retrievals \nresulting in live births for patients aged 35-37 years old"
     xlab = "Ratio", ylab = "Density", col = "aliceblue",
     freq = FALSE)

# Draw the normal distribution
lines(seq(0,1,0.01), dnorm(seq(0,1,0.01), mean(clinic_data$ND_IntentRetLB2, na.rm = TRUE), sqrt(var(cli

# Calculate the paremeter values for the gamma distribution
rate5.3 <- mean(clinic_data$ND_IntentRetLB2, na.rm = TRUE)
shape5.3 <- rate*mean(clinic_data$ND_IntentRetLB2, na.rm = TRUE)

# Draw the density function
lines(seq(0, 1, 0.01), dgamma(seq(0, 1, 0.01), shape5.3, rate5.3), col = "red")

# Draw a legend
legend(x = 0.8, y= 3, legend=c("Normal", "Gamma"), fill=(c("blue","red")))
```

Given the fit, I would say that the variable ND_IntentRetLB2 follows a **normal distribution** with mean and variance .

5.4 A brief simulation model

Let's say that the last studied variable, ND_IntentRetLB2, effectively follows a normal distribution with the calculated mean and variance. If we were to simulate its frequency distribution, would we get and histogram like that from the real data?

```r
# Use the same population size than in real data (exclude NA)
n_NDIRLB2 <- sum(!is.na(clinic_data$ND_IntentRetLB2))

# Generate random numbers
set.seed(378992)
simulated_NDIRLB2 <- rnorm(n_NDIRLB2,
                           mean = mean(clinic_data$ND_IntentRetLB2, na.rm = TRUE),
                           sd = sqrt(var(clinic_data$ND_IntentRetLB2, na.rm = TRUE)))

# Replace negative values by zero
simulated_NDIRLB2[simulated_NDIRLB2<0] = 0

# Represent simulated data

hist(simulated_NDIRLB2, breaks = seq(0,1,0.01),
     main = "Ratio of intended retrievals \nresulting in live births for patients aged 35-37 years old
     xlab = "Ratio", ylab = "Frequency", col = "lightgoldenrod1"
     #,freq = FALSE
)

# Draw the normal distribution
#lines(seq(0,1,0.01), dnorm(seq(0,1,0.01), mean(clinic_data$ND_IntentRetLB2, na.rm = TRUE), sqrt(var(cl
```

This histogram looks way smoother than the one built with real data but, all in all, the profile is quite similar. The exception being the few counts of 0.9-1.0 that we found in the real data. I think that the problem here is

that, working with data of ratios, we loose the information about the real number of procedures performed at each clinic. That could have an effect on the distribution that best modeled the data.

6- Regression analysis

Is there a relationship between clinic size and the percentage of transfers that resulted in live births?

We can use the total number of ART cycles (variable `TotNumCyclesAll`) as a measure of the size of the clinic. Getting the number of transfers and the percentage of transfers that resulted in live birth is a bit tricky:

```
# Calculate number of living births for each group of transfers
columns_NumTrans <- c(37:41, 82:85)
columns_ratioTransLB <- c(42:46, 86:89)

NDLB1 <- clinic_data$ND_NumTrans1 * clinic_data$ND_TransLB1
NDLB2 <- clinic_data$ND_NumTrans2 * clinic_data$ND_TransLB2
NDLB3 <- clinic_data$ND_NumTrans3 * clinic_data$ND_TransLB3
NDLB4 <- clinic_data$ND_NumTrans4 * clinic_data$ND_TransLB4
NDLB5 <- clinic_data$ND_NumTrans5 * clinic_data$ND_TransLB5
DLB1 <- clinic_data$Donor_NumTrans1 * clinic_data$Donor_TransLB1
DLB2 <- clinic_data$Donor_NumTrans2 * clinic_data$Donor_TransLB2
DLB3 <- clinic_data$Donor_NumTrans3 * clinic_data$Donor_TransLB3
DLB4 <- clinic_data$Donor_NumTrans4 * clinic_data$Donor_TransLB4

# New dataframe with number of living births
living_births <- data.frame(NDLB1, NDLB2, NDLB3, NDLB4, NDLB5, DLB1, DLB2, DLB3, DLB4)

# Calculate the total number of transfers for each clinic
TotNumTransfers <- rowSums(clinic_data[,c(37:41, 82:85)])

# Total number of living births from transfer for each clinic
TotNumLB <- rowSums(living_births, na.rm = TRUE)

# Ratio of living births/transfers for each clinic
TotRatioLB <- TotNumLB/TotNumTransfers
```

Let's try a plot diagram to explore the relation between ART cycles and ratio of living births by transfer:

```
plot(clinic_data$TotNumCyclesAll, TotRatioLB, main = "Ratio of living births by transfer vs. ART cycles
```

It is difficult to make a preliminary assessment from that graph, only that most of the info is given below 2000 cycles. Data points beyond 2000 cycles doesn't seem to convey any relationship between ratio and cycles.

Let's try to carry out a regression:

If we calculate the correlation between both variables:

```
cor(clinic_data$TotNumCyclesAll, y=TotRatioLB, use = "complete")
```

There doesn't seem to be any correlation or, if there is any, it is very weak.

But let's work in the regression model:

```
RegModel <- lm(TotRatioLB~clinic_data$TotNumCyclesAll)

summary(RegModel)
```

With that data, the equation for the least squares line is (rounding to the third decimal):

Y = 1.103e-05x + 0.419

Let's add it to the graph:

```
plot(clinic_data$TotNumCyclesAll, TotRatioLB, main = "Ratio of living births by transfer vs. ART cycles

abline(RegModel, col = "red")
```

The correlation coefficient is only 0.008369, which I interpret as a very poor adjust, rending meaningless the values of the model. It may be due to a much greater variance for low number of cycles compared with the variance for high number of cycles; the model used assumes that the variables follow normal distribution and homoscedasticity.

Let's check those assumptions:

```
# Check normality with qqplot
par(mfcol = c(4,1))

qqnorm(clinic_data$TotNumCyclesAll, main="Number of cycles Q-Q Plot")
qqline(clinic_data$TotNumCyclesAll, col = "blue")

# Or with a histogram
hist(clinic_data$TotNumCyclesAll, main = "Total number of cycles", xlab = "Cycles",
     col = "aliceblue")

qqnorm(TotRatioLB)
qqline(TotRatioLB)

hist(TotRatioLB, main = "Ratios of living births per transfer", xlab = "Ratios",
     col = "lightgoldenrod1")
```

It is clear the the total number of cycles doesn't follow a normal distribution. The ratios of living births per transfer look more like it, but if we test it with Shapiro-Wilk:

```
shapiro.test(TotRatioLB)
```

With a p-value so low be reject the null hypothesis of normality. We may instead want to check correlation with a test that doesn't assume normality, like Pearson correlation. Which... we already did previously when calculated correlation with `cor()` since the default mode is "pearson".
The result was ,
which means that there is not correlation between the variables.
The significativity of that result is:

```
cor.test(clinic_data$TotNumCyclesAll, y=TotRatioLB, method = "pearson")
```

Which doesn't let as reject the alternative hypothesis.

In short, **the data doesn't show any correlation between the size of the clinic (measured in total number of ART cycles) and the ratio of living births per transfer**.
Which is good news, since it would mean that little clinics are as effective at achieving living births after transfer as big clinics are. That is probably because the techniques used are mature, well known and widely taught, and the clinics maintain high standards of operation.

7- Final assessment

I am going to focus on the conclusion of the regression analysis, which is the one that I find more interesting and, at the same time, open to refinement.

One of the caveats that I have found is that I am working in this analysis with data that are ratios, meaning that we are losing some information. For example, clinics that in the year have obtained 1 living birth from 2 transfers (1/2) have the same success ratio that clinics that obtained 50 living birth from 100 transfers (50/100). In both cases the ratio is 0.5, but one case should be more reliable than the other.
To account for this maybe I should have used a different statistical test, or I could have use number of transfers instead of number of cycles to represent the size of the clinic. Those two concepts may not be interchangeable, though.

Another problem is that we are including different variants of the same technique into a unique statistic, while those variants may be of different difficulty.
Also, we are not taking into account the age of the patient, which is known to be very important in the case of non-donor treatments. For the regression we have only used data from donor cases, but it may be true that age were relevant in these cases as well.

If we were to repeat the analysis (which we are not due to time constraints), we could differentiate by technique variant (fresh or frozen egg; fresh or frozen embryo). But it wouldn't be possible to take the age of the patient into account, since that data are not included in the source file.