

# Analisis Teoria Colas

Kevin Heberth Haquehua Apaza

## LECTURA DE DATOS

Primeramente procedemos a leer los datos en formato xlsx

```
library(readxl)
data <- read_excel("Data_tiempo_Fuente.xlsx")

library(tidyverse)

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.5
## v forcats    1.0.0      v stringr   1.5.1
## v ggplot2    3.5.1      v tibble    3.2.1
## v lubridate  1.9.3      v tidyr     1.3.1
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

options(scipen = 999)
```

##Descriptivo general con los datos

Primeramente evaluemos la fecha de donde se obtienen los datos

```
#Transformamos Fecha a tipo dato
data$Fecha <- as.Date(data$Fecha, origin = "1899-12-30")
```

Ahora evaluemos la disposicion de que fecha se tienen los datos

```
summary(data$Fecha)
```

```
##           Min.          1st Qu.          Median            Mean          3rd Qu.           Max.
## "2023-07-12" "2023-08-14" "2023-09-12" "2023-09-12" "2023-10-12" "2023-11-09"
```

De lo cual se observa que se tienen datos desde el 12 de julio del 2023 hasta el 9 de noviembre del 2023 siendo un total de 4 meses aproximadamente. y además se tienen 5013 registros en total de los cuales

```
data %>% group_by(Area) %>% summarise(Cantidad = n()) %>% mutate(Porcentaje = prop.table(Cantidad)*100)
```

```
## # A tibble: 4 x 3
##   Area                Cantidad Porcentaje
##   <chr>                <int>      <dbl>
## 1 Oftalmología         4002      80.6
## 2 INLASER - Cirugía Refractiva    558     11.2
## 3 GLAUCOMA             283       5.70
## 4 RETINA               120       2.42
```

Se observa que la mayor parte de pacientes es del área de Oftalmología, seguido de pacientes de INLASER - cirugía refractiva, Glaucoma y Retina

```
data %>% group_by(Area, Piso_Triaje) %>% summarise(Cantidad = n()) %>% mutate(Porcentaje = prop.table(C
```

```
## `summarise()` has grouped output by 'Area'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 8 x 4
## # Groups:   Area [4]
##   Area                Piso_Triaje    Cantidad Porcentaje
##   <chr>                <chr>          <int>     <dbl>
## 1 GLAUCOMA            TRIAJE 1ER PISO         54      19.1
## 2 GLAUCOMA            TRIAJE 2DO PISO        229      80.9
## 3 INLASER - Cirugía Refractiva TRIAJE 1ER PISO        279       50
## 4 INLASER - Cirugía Refractiva TRIAJE 2DO PISO        279       50
## 5 Oftalmología        TRIAJE 1ER PISO       2006     50.1
## 6 Oftalmología        TRIAJE 2DO PISO       1996     49.9
## 7 RETINA              TRIAJE 1ER PISO         29     24.2
## 8 RETINA              TRIAJE 2DO PISO         91     75.8
```

Se observa que con respecto al servicio que se tiene tanto en oftalmología e INLASER no se encuentran diferencias significativas respecto al piso y triaje, esto se observa más en áreas de GLAUCOMA y RETINA debido a que el motivo de visita es más para control y los casos atendidos al primer caso son generalmente cirugías.

```
data %>% group_by(Area, Condicion) %>% summarise(Cantidad = n()) %>% mutate(Porcentaje = prop.table(Can
```

```
## `summarise()` has grouped output by 'Area'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 8 x 4
## # Groups:   Area [4]
##   Area                Condicion Cantidad Porcentaje
##   <chr>                <chr>          <int>     <dbl>
## 1 GLAUCOMA            C              245      86.6
## 2 GLAUCOMA            N              38      13.4
## 3 INLASER - Cirugía Refractiva C             551     98.7
## 4 INLASER - Cirugía Refractiva N              7       1.25
## 5 Oftalmología        C             1964     49.1
## 6 Oftalmología        N             2038     50.9
## 7 RETINA              C              115     95.8
## 8 RETINA              N              5       4.17
```

Se observa que pacientes nuevos generalmente son en oftalmología, esto es debido que también los nuevos pasan en el siguiente día a ser continuadores, la única excepción es oftalmología general.

Tomando estos datos en cuenta se recomienda quitar estas variables (Piso Triaje y Condicion) para los analisis que se van a realizar

```
## Área de Oftalmología
```

Tomando en cuenta los estudios primeramente en la parte descriptiva se tomará en cuenta la edad del paciente, tomando esto en cuenta se tiene los siguientes resultados del área de oftalmología.

## Edad de los pacientes

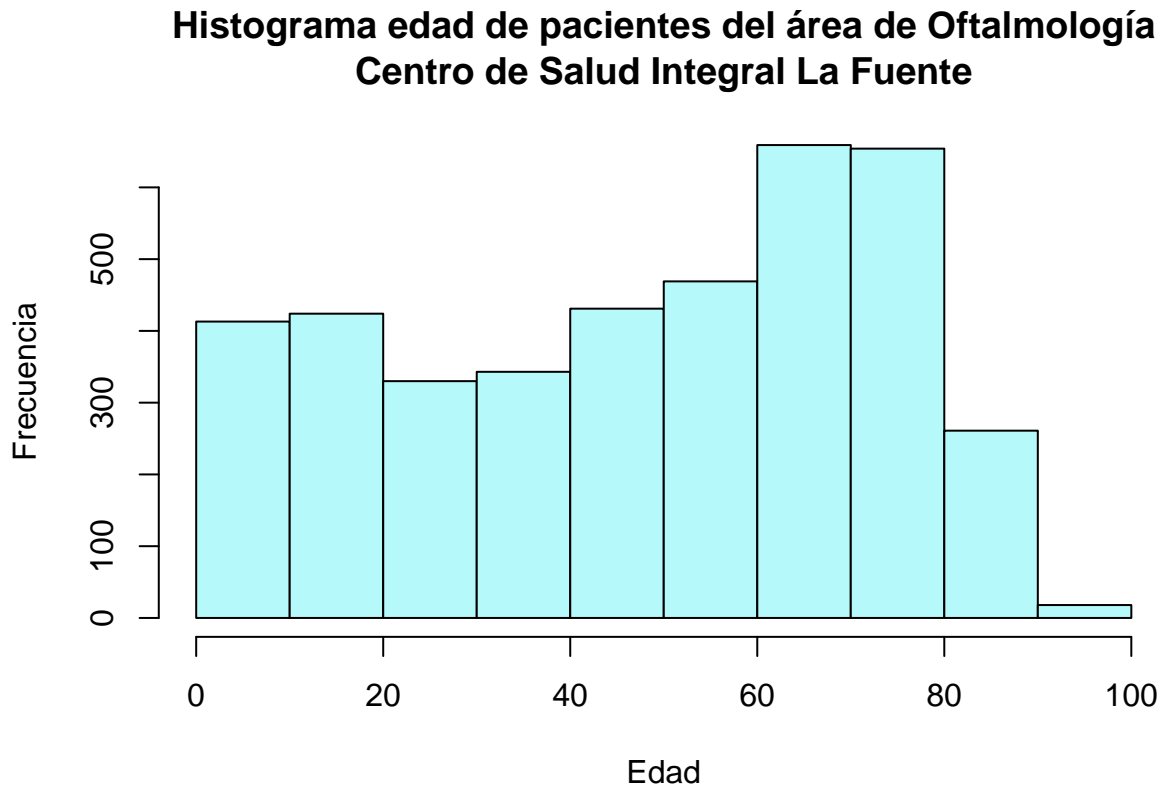
Veamos los estadísticos descriptivos de la variable edad del paciente

```
Oftalmologia <- data %>% filter(Area == "Oftalmología")
```

```
summary(Oftalmologia$Edad)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1.00   26.00   52.00   48.09   70.00   96.00
```

```
hist(Oftalmologia$Edad, main = "Histograma edad de pacientes del área de Oftalmología\nCentro de Salud Integral La Fuente")
```



Se observa que la mayor parte de los pacientes del área de oftalmología son de diversas edades, los más resaltantes se encuentran entre 60 a 80 años, seguido de los que tienen entre 5 a 10 años y así sucesivamente, hay pocos pacientes mayores de 85 años y menores a 5 años.

### Distribución de llegadas de los pacientes

Las citas son programadas desde las 7:00 am hasta las 4:00 pm

```
Oftalmologia <- Oftalmologia %>%
  mutate(Intervalo_ingreso = cut(Hora_ingreso,
    breaks = seq(floor_date(min(Hora_ingreso), "day"),
      ceiling_date(max(Hora_ingreso), "day") + hours(1), by = "30 mins"),
    labels = FALSE))
```

*# Etiquetas para los intervalos*

```
etiquetas <- seq(floor_date(min(Oftalmologia$Hora_ingreso), "day"),
  ceiling_date(max(Oftalmologia$Hora_ingreso), "day") + hours(1), by = "30 mins")
etiquetas <- paste0(format(etiquetas[-length(etiquetas)], "%H:%M"), " a ",
  format(etiquetas[-1], "%H:%M"))
```

```
tabla_frecuencias_ingreso <- Oftalmologia %>%
```

```
group_by(Intervalo_ingreso) %>%
summarise(Cantidad = n()) %>%
mutate(Tiempo = etiquetas[Intervalo_ingreso])
```

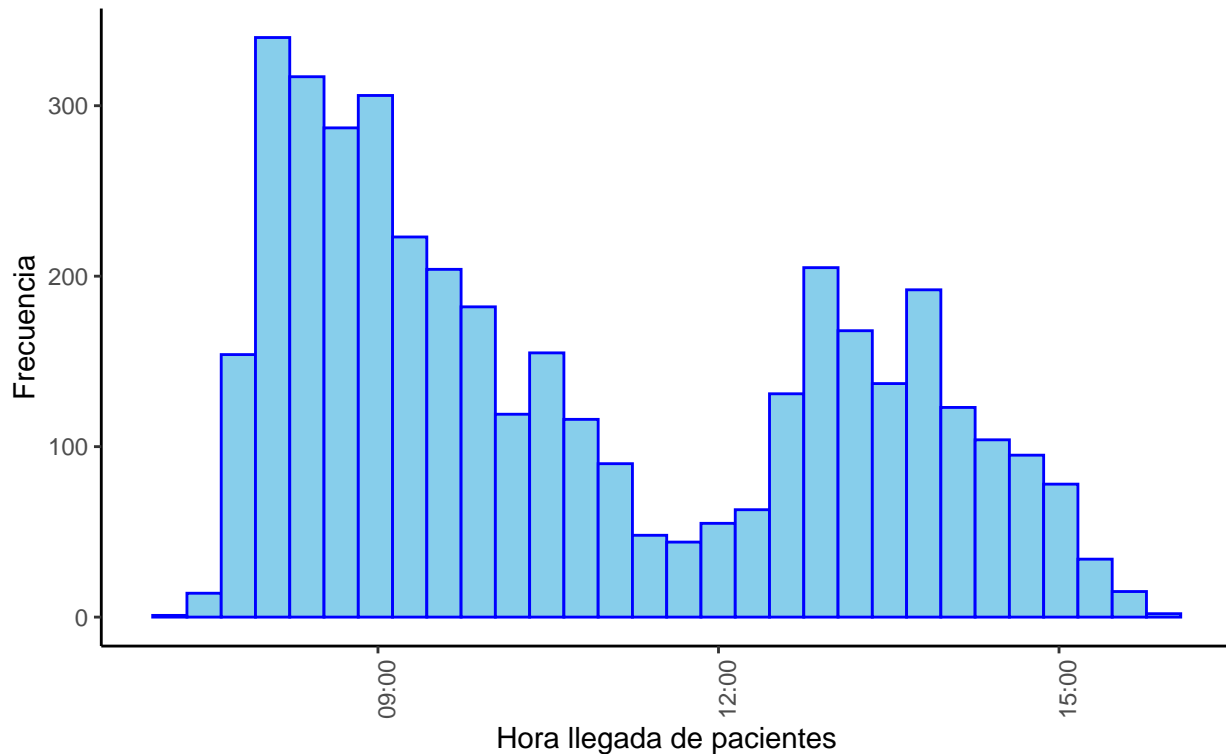
```
tabla_frecuencias_ingreso
```

```
## # A tibble: 18 x 3
##   Intervalo_ingreso Cantidad Tiempo
##           <int>      <int> <chr>
## 1             15         6 07:00 a 07:30
## 2             16        244 07:30 a 08:00
## 3             17        541 08:00 a 08:30
## 4             18        497 08:30 a 09:00
## 5             19        394 09:00 a 09:30
## 6             20        332 09:30 a 10:00
## 7             21        218 10:00 a 10:30
## 8             22        213 10:30 a 11:00
## 9             23        108 11:00 a 11:30
## 10            24         78 11:30 a 12:00
## 11            25         98 12:00 a 12:30
## 12            26        298 12:30 a 13:00
## 13            27        258 13:00 a 13:30
## 14            28        271 13:30 a 14:00
## 15            29        199 14:00 a 14:30
## 16            30        160 14:30 a 15:00
## 17            31         73 15:00 a 15:30
## 18            32         14 15:30 a 16:00
```

```
ggplot(Oftalmologia, aes(x = Hora_ingreso)) +
  geom_histogram(fill = "skyblue", color = "blue") +
  labs(title = "Distribución llegadas de pacientes programados del área de Oftalmología",
        subtitle = "Centro de Salud Integral La Fuente",
        x = "Hora llegada de pacientes",
        y = "Frecuencia") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 90))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

## Distribución llegadas de pacientes programados del área de Oftalmología Centro de Salud Integral La Fuente



Se observa que la mayor parte de los pacientes que llegan para el área de oftalmología se encuentran entre el horario de 8:00 a 9:30, después el siguiente horario con mayor presencia de pacientes es el de 12:30 a 14:00 hrs

Ahora veamos si esta distribución sigue un proceso de Poisson

```
hora_ingreso <- Oftalmologia %>% group_by(Intervalo_ingreso, Fecha) %>% summarise(Cantidad = n()) %>%
  mutate(Tiempo = etiquetas[Intervalo_ingreso])
```

```
## `summarise()` has grouped output by 'Intervalo_ingreso'. You can override using
## the `groups` argument.
```

```
table(hora_ingreso$Cantidad)
```

```
##
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 22
## 226 186 152 111  91  56  52  37  30 14 11 10  6  8  4  1  2  2  1  1
## 25
## 1
```

```
hora_ingreso
```

```
## # A tibble: 1,002 x 4
## # Groups:   Intervalo_ingreso [18]
##   Intervalo_ingreso Fecha      Cantidad Tiempo
##             <int> <date>         <int> <chr>
## 1             15 2023-08-07           1 07:00 a 07:30
## 2             15 2023-08-09           2 07:00 a 07:30
## 3             15 2023-08-10           2 07:00 a 07:30
## 4             15 2023-08-28           1 07:00 a 07:30
## 5             16 2023-07-12           2 07:30 a 08:00
```

```
## 6          16 2023-07-13      3 07:30 a 08:00
## 7          16 2023-07-14      1 07:30 a 08:00
## 8          16 2023-07-19      2 07:30 a 08:00
## 9          16 2023-07-20      2 07:30 a 08:00
## 10         16 2023-07-21      1 07:30 a 08:00
## # i 992 more rows
```

```
lambda <- mean(hora_ingreso$Cantidad)

observed <- table(hora_ingreso$Cantidad)

probs <- dpois(as.numeric(names(observed)), lambda)

expected <- probs * length(hora_ingreso)

chisq_test <- chisq.test(x = observed, p = expected / sum(expected), rescale.p = TRUE)
```

```
## Warning in chisq.test(x = observed, p = expected/sum(expected), rescale.p =
## TRUE): Chi-squared approximation may be incorrect
```

```
contrib_chisq <- (observed - expected)^2 / expected
```

```
results <- data.frame(
  Clase = as.numeric(names(observed)),
  Observado = as.numeric(observed),
  Prob_Poisson = probs,
  Esperado = round(expected, 2),
  Contrib_Chi2 = round(contrib_chisq, 2)
)
```

```
print(results)
```

##	Clase	Observado	Prob_Poisson	Esperado	Contrib_Chi2.Var1
## 1	1	226	0.073592236396888486	0.29	1
## 2	2	186	0.146964136756660507	0.59	2
## 3	3	152	0.195658840751881419	0.78	3
## 4	4	111	0.195365938295666008	0.78	4
## 5	5	91	0.156058779452945107	0.62	5
## 6	6	56	0.103883439017080290	0.42	6
## 7	7	52	0.059273099935322954	0.24	7
## 8	8	37	0.029592183874895525	0.12	8
## 9	9	30	0.013132392977082703	0.05	9
## 10	10	14	0.005245093482463569	0.02	10
## 11	11	11	0.001904451471313666	0.01	11
## 12	12	10	0.000633866832019070	0.00	12
## 13	13	6	0.000194743978331054	0.00	13
## 14	14	8	0.000055557841551246	0.00	14
## 15	15	4	0.000014793245634603	0.00	15
## 16	16	1	0.000003692775014326	0.00	16
## 17	17	2	0.000000867587507769	0.00	17
## 18	18	2	0.000000192508605350	0.00	18
## 19	19	1	0.000000040467456593	0.00	19
## 20	22	1	0.000000000279037149	0.00	22
## 21	25	1	0.000000000001288282	0.00	25
##	Contrib_Chi2.Freq				
## 1		173058.44			

```
## 2          58479.68
## 3          29217.55
## 4          15545.35
## 5          13084.46
## 6           7435.34
## 7          11301.07
## 8          11491.67
## 9          17073.26
## 10         9314.09
## 11         15861.85
## 12         39420.46
## 13         46202.52
## 14         287972.15
## 15         270385.67
## 16         67697.76
## 17        1152617.48
## 18        5194568.98
## 19        6177801.62
## 20        895938049.67
## 21       194056851185.27
```

```
print(paste("Valor lambda:", lambda))
```

```
## [1] "Valor lambda: 3.9940119760479"
```

```
print(chisq_test)
```

```
##
```

```
## Chi-squared test for given probabilities
```

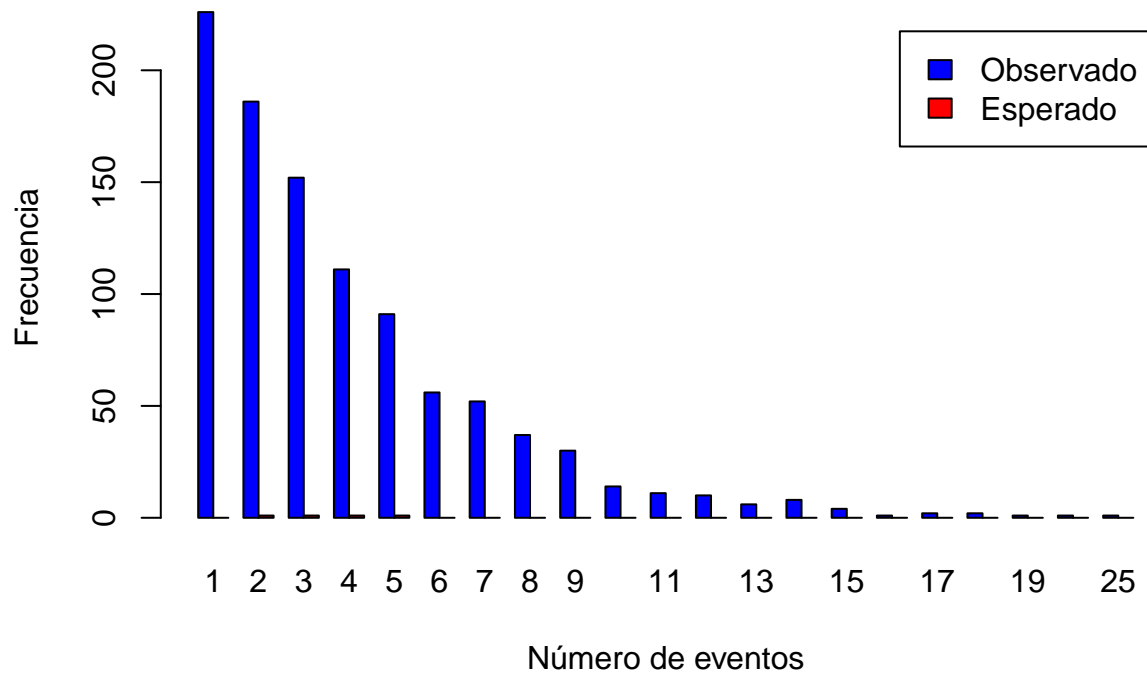
```
##
```

```
## data: observed
```

```
## X-squared = 763967089, df = 20, p-value < 0.00000000000000022
```

```
barplot(rbind(as.numeric(observed), round(expected)),
        beside = TRUE,
        col = c("blue", "red"),
        legend = c("Observado", "Esperado"),
        names.arg = names(table(hora_ingreso$Cantidad)),
        xlab = "Número de eventos",
        ylab = "Frecuencia",
        main = "Comparación de frecuencias observadas y esperadas (Poisson)")
```

## Comparación de frecuencias observadas y esperadas (Poisson)



Los datos no siguen una distribución de Poisson

```
summary(Oftalmologia$Tiempo_atencion_ingreso)
```

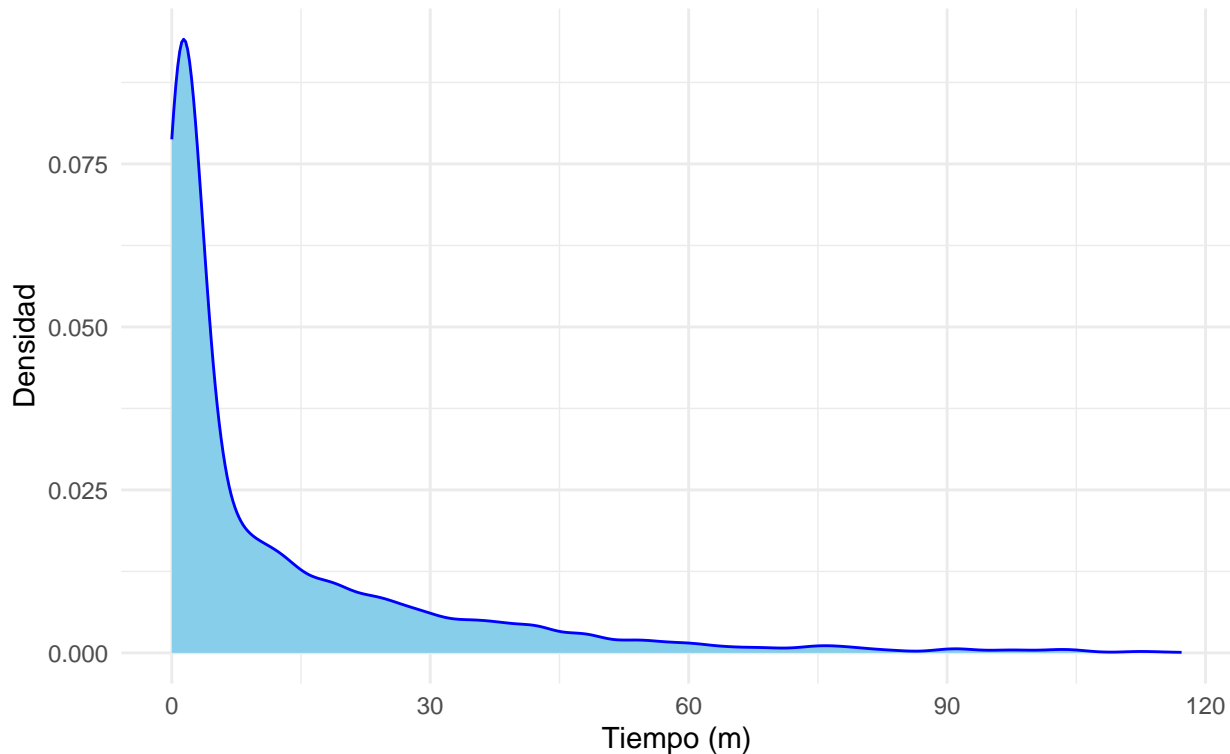
Evaluar la distribución de los tiempos de espera

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.000   1.033   3.483  12.461  17.221 117.217
```

```
ggplot(Oftalmologia, aes(x = Tiempo_atencion_ingreso)) +
  geom_density(fill = "skyblue", color = "blue") +
  labs(title = "Distribución en minutos del tiempo entre la hora programada y la hora de llegada del paciente",
        subtitle = "Centro de Salud Integral La Fuente",
        x = "Tiempo (m)",
        y = "Densidad") +
  theme_minimal()
```



## Distribución en minutos del tiempo entre la hora programada y la hora de Centro de Salud Integral La Fuente



Se observa que se ve un patron de funcion exponencial, vamos a agruparlo en intervalos

```
log(4002, base = 2)
```

```
## [1] 11.96651
```

Se deben crear 12 clases veamos el rango con el que deben sumarse cada clase

```
max(Oftalmologia$Tiempo_atencion_ingreso) - min(Oftalmologia$Tiempo_atencion_ingreso)
```

```
## [1] 117.2167
```

```
Oftalmologia$Tiempo_atencion_ingreso_clas <- ifelse(Oftalmologia$Tiempo_atencion_ingreso <= 15, "0 a 15",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 15 & Oftalmologia$Tiempo_atencion_ingreso <= 30, "15 a 30",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 30 & Oftalmologia$Tiempo_atencion_ingreso <= 45, "30 a 45",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 45 & Oftalmologia$Tiempo_atencion_ingreso <= 60, "45 a 60",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 60 & Oftalmologia$Tiempo_atencion_ingreso <= 75, "60 a 75",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 75 & Oftalmologia$Tiempo_atencion_ingreso <= 90, "75 a 90",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 90 & Oftalmologia$Tiempo_atencion_ingreso <= 105, "90 a 105",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 105 & Oftalmologia$Tiempo_atencion_ingreso <= 120, "105 a 120",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 120 & Oftalmologia$Tiempo_atencion_ingreso <= 135, "135 a 150",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 135 & Oftalmologia$Tiempo_atencion_ingreso <= 150, "150 a 165",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 150 & Oftalmologia$Tiempo_atencion_ingreso <= 165, "165 a 180",
  ifelse(Oftalmologia$Tiempo_atencion_ingreso > 165, "180 a 200"))
```

```
table(Oftalmologia$Tiempo_atencion_ingreso_clas)
```

```
##
##      0 a 15 mins 105 a 120 mins      15 a 30 mins      30 a 45 mins      45 a 60 mins
##           2907              8           550           277           132
```

```
##    60 a 75 mins    75 a 90 mins    90 a 105 mins
##              60              37              31

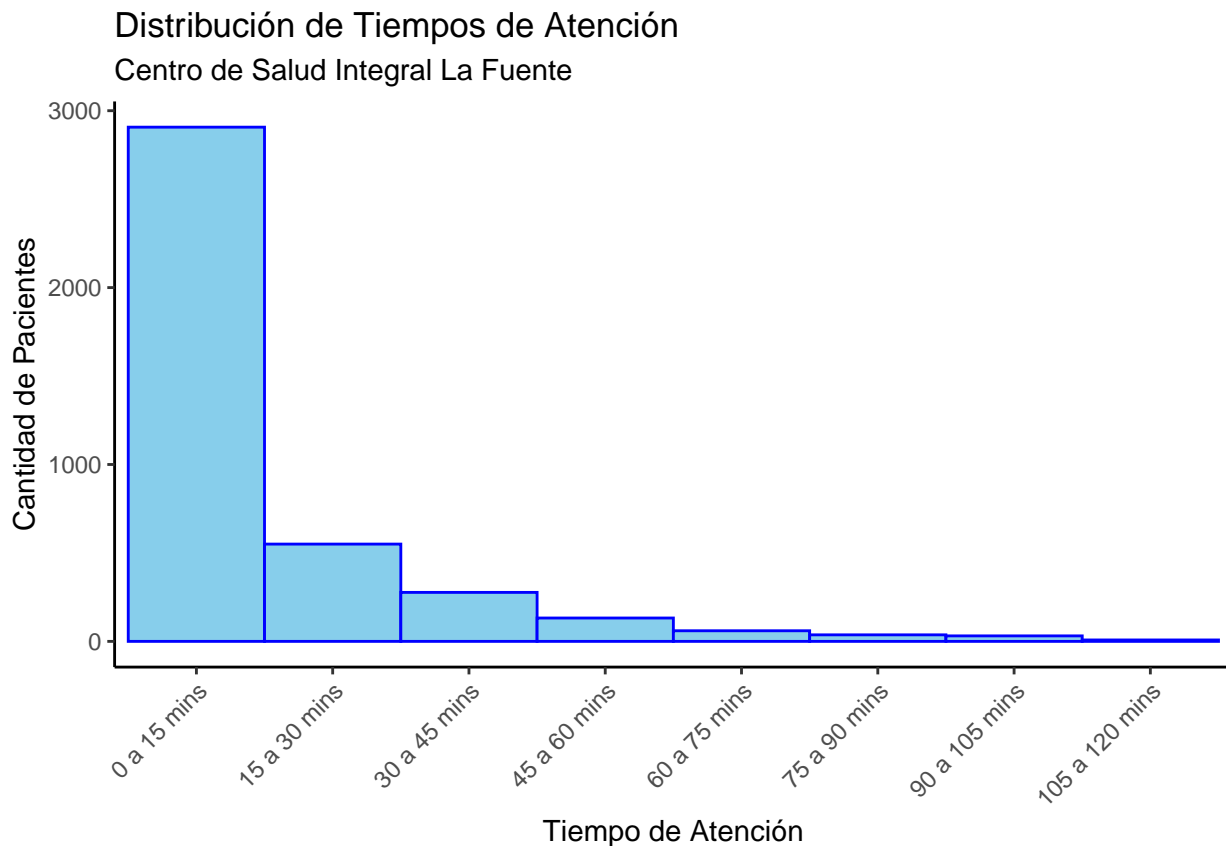
Oftalmologia$Tiempo_atencion_ingreso_clas <- factor(Oftalmologia$Tiempo_atencion_ingreso_clas,
                                                    levels = c("0 a 15 mins", "15 a 30 mins", "30 a 45 mins",
                                                            "45 a 60 mins", "60 a 75 mins", "75 a 90 mins",
                                                            "90 a 105 mins", "105 a 120 mins"))

table(Oftalmologia$Tiempo_atencion_ingreso_clas)

##
##    0 a 15 mins    15 a 30 mins    30 a 45 mins    45 a 60 mins    60 a 75 mins
##      2907         550         277         132         60
##    75 a 90 mins    90 a 105 mins    105 a 120 mins
##      37           31           8

tablaRec <- Oftalmologia %>% group_by(Tiempo_atencion_ingreso_clas) %>% summarise(Cantidad = n())

ggplot(tablaRec, aes(x = Tiempo_atencion_ingreso_clas, y = Cantidad)) +
  geom_bar(stat = "identity", fill = "skyblue", color = "blue", width = 1) +
  labs(title = "Distribución de Tiempos de Atención",
       subtitle = "Centro de Salud Integral La Fuente",
       x = "Tiempo de Atención",
       y = "Cantidad de Pacientes") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Evaluar las clases (se tiene 8 clases) entonces  $8 - 1 - 1 = 6$  grados de libertad

Plantear la prueba hipotesis

- H0: Los tiempos de espera de pacientes para su atencion siguen una distribucion exponencial
- H1: Los tiempos de espera de pacientes para su atencion no siguen una distribucion exponencial

Plantera el punto critico

```
qchisq(0.95, 6)
```

```
## [1] 12.59159
```

Calcular la media de distribucion Exponencial

```
media_exp <- 1/mean(Oftalmologia$Tiempo_atencion_ingreso)
```

```
tablaRec$LS <- c(15,30,45,60,75,90,105,120)
tablaRec
```

```
## # A tibble: 8 x 3
##   Tiempo_atencion_ingreso_clas Cantidad    LS
##   <fct>                        <int> <dbl>
## 1 0 a 15 mins                  2907    15
## 2 15 a 30 mins                 550     30
## 3 30 a 45 mins                 277     45
## 4 45 a 60 mins                 132     60
## 5 60 a 75 mins                 60     75
## 6 75 a 90 mins                 37     90
## 7 90 a 105 mins                31    105
## 8 105 a 120 mins                8    120
```

```
tablaRec$Acum_prob <- pexp(tablaRec$LS, media_exp)
tablaRec
```

```
## # A tibble: 8 x 4
##   Tiempo_atencion_ingreso_clas Cantidad    LS Acum_prob
##   <fct>                        <int> <dbl>    <dbl>
## 1 0 a 15 mins                  2907    15    0.700
## 2 15 a 30 mins                 550     30    0.910
## 3 30 a 45 mins                 277     45    0.973
## 4 45 a 60 mins                 132     60    0.992
## 5 60 a 75 mins                 60     75    0.998
## 6 75 a 90 mins                 37     90    0.999
## 7 90 a 105 mins                31    105    1.00
## 8 105 a 120 mins                8    120    1.00
```

```
tablaRec <- tablaRec %>%
  mutate(Probabilidad = Acum_prob - lag(Acum_prob, default = 0))
```

```
tablaRec
```

```
## # A tibble: 8 x 5
##   Tiempo_atencion_ingreso_clas Cantidad    LS Acum_prob Probabilidad
##   <fct>                        <int> <dbl>    <dbl>    <dbl>
## 1 0 a 15 mins                  2907    15    0.700    0.700
## 2 15 a 30 mins                 550     30    0.910    0.210
## 3 30 a 45 mins                 277     45    0.973    0.0630
## 4 45 a 60 mins                 132     60    0.992    0.0189
## 5 60 a 75 mins                 60     75    0.998    0.00567
## 6 75 a 90 mins                 37     90    0.999    0.00170
## 7 90 a 105 mins                31    105    1.00    0.000511
## 8 105 a 120 mins                8    120    1.00    0.000153
```

```
Total0ft <- sum(tablaRec$Cantidad)

tablaRec$fe <- tablaRec$Probabilidad * Total0ft
```

```
tablaRec$f0_fe <- tablaRec$Cantidad - tablaRec$fe  
tablaRec$f0_fecudad <- tablaRec$f0_fe^2  
tablaRec$f0_fe_cudad_div_fe <- tablaRec$f0_fecudad/tablaRec$fe
```

```
est_prueba <- sum(tablaRec$f0_fe_cuad_div_fe)
est_prueba
```

```
pchisq(est_prueba, 6, lower.tail = FALSE)
```

Los datos no siguen una distribucion exponencial

## lambda

lambda\_min

media\_exp

```
ro <- lambda_min/media_exp
ro
```

**Factor de utilizacion**

```
## [1] 1.65902
```

```
Lq = (lambda_min^2)/(lambda_min*(media_exp-lambda_min))
Lq
```

**Numero esperado de pacientes en la cola (Lq)**

```
## [1] -2.517404
```

```
L = lambda_min/(media_exp - lambda_min)
L
```

**Numero esperado de pacientes en el sistema (L)**

```
## [1] -2.517404
```

```
Wq = lambda_min/(media_exp*(media_exp-lambda_min))
Wq
```

**Tiempo esperado de pacientes en la cola (Wq)**

```
## [1] -31.37014
```

```
W = 1/(media_exp - lambda_min)
W
```

**Tiempo esperado de pacientes en el sistema (W)**

```
## [1] -18.90884
```