

Trabajo Práctico: Estadística Actuarial

Alumnos: Malena Irisarri, Román Landa



Rosario, Argentina
10 de Mayo de 2025

Introducción

En el sector asegurador, garantizar la solvencia financiera es fundamental para cumplir con las obligaciones frente a los asegurados y mantener la estabilidad de la compañía. En este contexto, el Margen de Solvencia Mínimo (MSM) emerge como un indicador clave que asegura la capacidad de la aseguradora para afrontar siniestros inesperados, incluso en escenarios adversos.

Para este trabajo, se analizará la sub-cartera de pólizas de seguros automotores de una compañía, compuesta por 25.615 pólizas, con el objetivo de determinar el MSM que permita alcanzar una Probabilidad de Solvencia del 99% durante el año 2024. El análisis se basará en datos históricos de siniestros de los años 2021, 2022 y 2023, considerando ajustes por inflación mediante la serie CER publicado en Banco Central de la República Argentina (BCRA). Adicionalmente, se explorarán diferentes distribuciones de probabilidad, para modelar el comportamiento de los siniestros y se probarán distintos recargos de seguridad sobre las primas puras.

Este informe presentará alternativas para el cálculo del MSM bajo el importante supuesto de que **durante el año 2024, la cantidad de pólizas y el perfil de las mismas, no cambiará**. El objetivo final es asegurar la estabilidad financiera y cumplir con los requisitos regulatorios de la Superintendencia de Seguros de la Nación (SSN).

Datos

Contamos con una base de datos que contiene 3431 cuantías pagadas durante el año 2023. A ellas les aplicamos la actualización por CER llevando todos los valores al día 01-01-2024 para poder compararlos aplicando:

$$\text{Cuantia Actualizada}_i = \text{Cuantia Original}_i \times \frac{\text{CER}_{01/01/2024}}{\text{CER}_i}$$

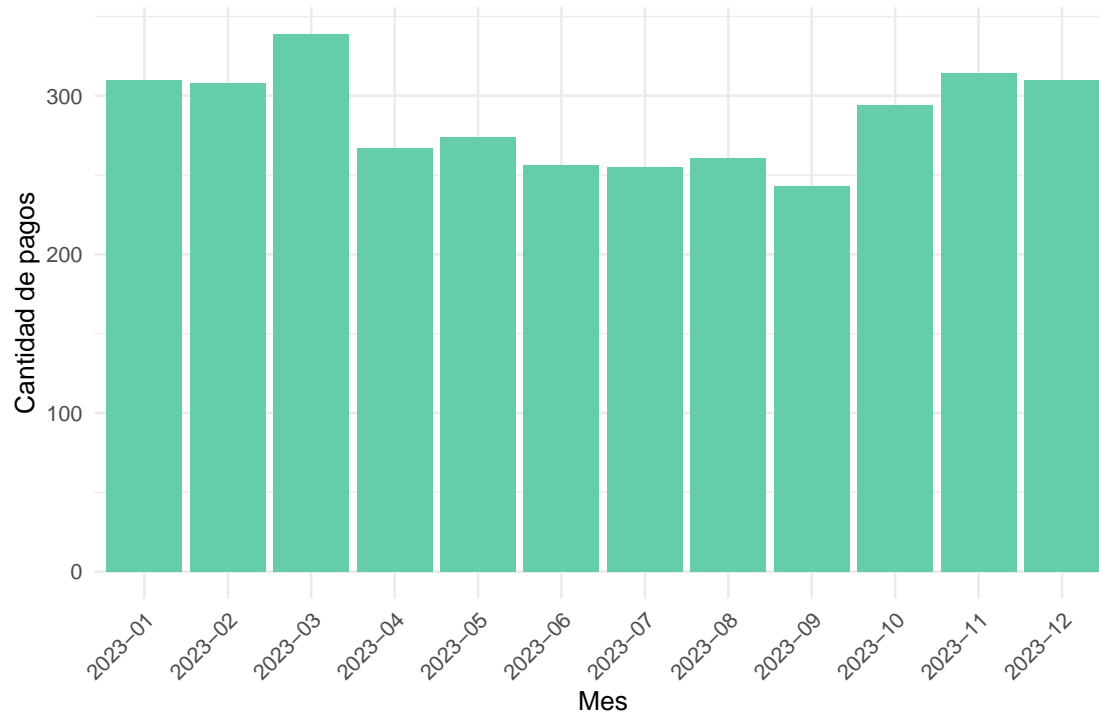
Table 1: Cuantías Pagadas en 2023.

Fecha	Cuantía	Valor CER	Cuantía Actualizada
2023-01-02	70939	73.7253	178636.7
2023-01-02	107871	73.7253	271637.9
2023-01-02	148775	73.7253	374641.3
2023-01-02	167011	73.7253	420562.7
2023-01-02	180152	73.7253	453654.0
2023-01-02	220063	73.7253	554156.8
2023-01-02	248064	73.7253	624668.2
2023-01-02	269251	73.7253	678020.7
2023-01-03	78566	73.8392	197537.6
2023-01-03	79646	73.8392	200253.1

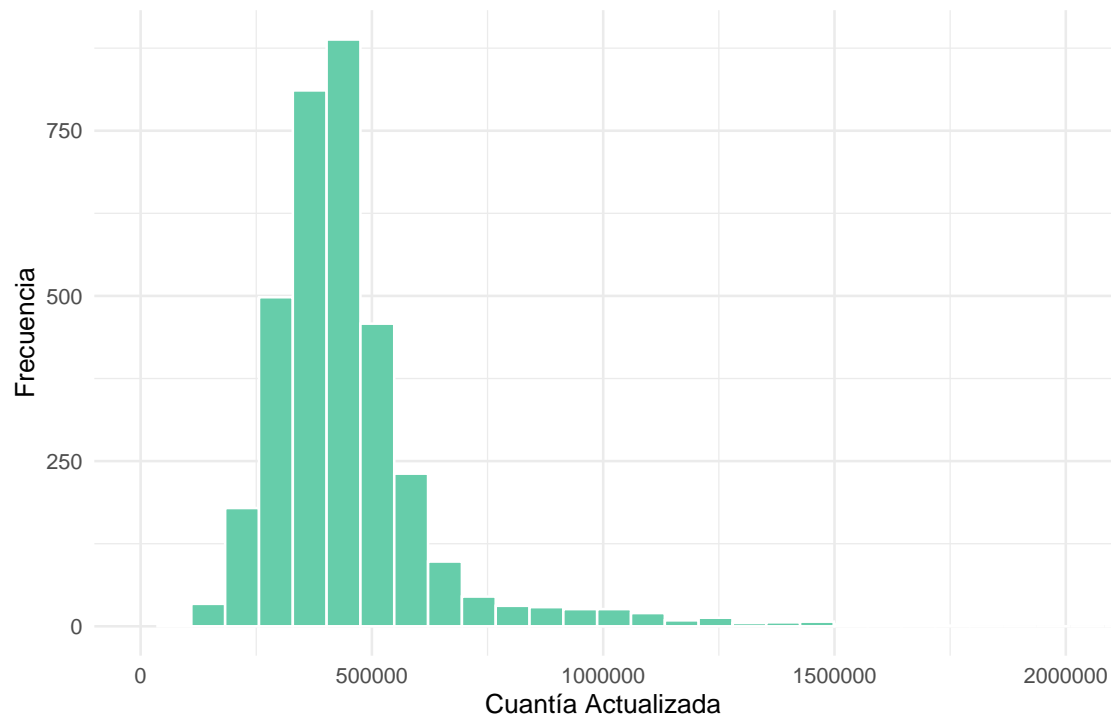
Análisis Descriptivo

Realizamos un breve análisis descriptivo para observar cuál fue la frecuencia y la severidad durante 2023.

Cantidad de cuantías pagadas por mes (frecuencia)



Distribución de las cuantías actualizadas (severidad)



En estos gráficos podemos observar que la frecuencia parece constante a lo largo del periodo de análisis y que la severidad parece distribuirse de forma asimétrica con muchos siniestros pequeños y algunos pocos de gran importe.

Además, podemos ver que al menos el 50% de los siniestros le costaron menos de \$ 417.603,40.- a la compañía y que el costo medio de los mismos fue de \$ 454.566,30.- con un desvío estandar de \$ 233.500,60.-

Finalmente, podemos mencionar que la cuantía total afrontada por la compañía durante el año 2023 fue \$954.856.697.- , lo que actualizado al día 01/01/2024 equivaldría \$1.559.617.015.-

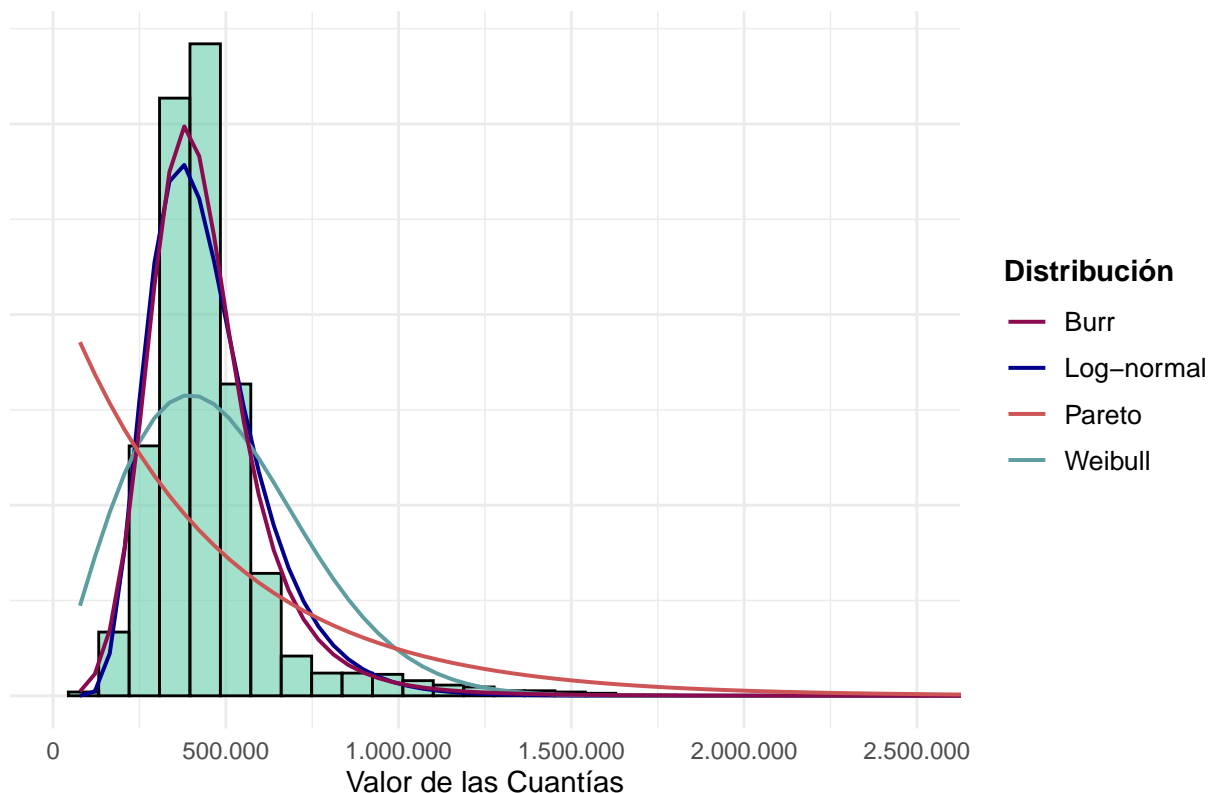
Simulaciones

Para predecir el monto esperado de siniestros y calcular el margen de solvencia, simularemos la frecuencia (número de siniestros) y la severidad (monto por siniestro), probando distintas combinaciones de distribuciones.

Comenzamos analizando qué distribución se ajusta mejor a nuestra severidad observada.

El siguiente gráfico compara los datos reales con las distribuciones candidatas (Log-Normal, Weibull, Pareto y Burr):

Ajuste de distribuciones a las cuantías



Optamos por una distribución Binomial Negativa para la frecuencia de siniestros por ser el caso más general para representar lo observado, ya que esta cubre los casos donde la variancia supera a la esperanza, situación frecuente en seguros de automóviles.

Para las cuantías individuales de los siniestros, seleccionamos las dos distribuciones de cola pesada que muestran el mejor ajuste a nuestros datos: la Log-Normal, que modela adecuadamente la asimetría positiva y la Burr que captura diversos patrones de severidad, incluyendo eventos extremos.

La estimación de parámetros para ambos modelos se hacen durante el ajuste de los mismos por el método de *máxima verosimilitud* y los resultados fueron:

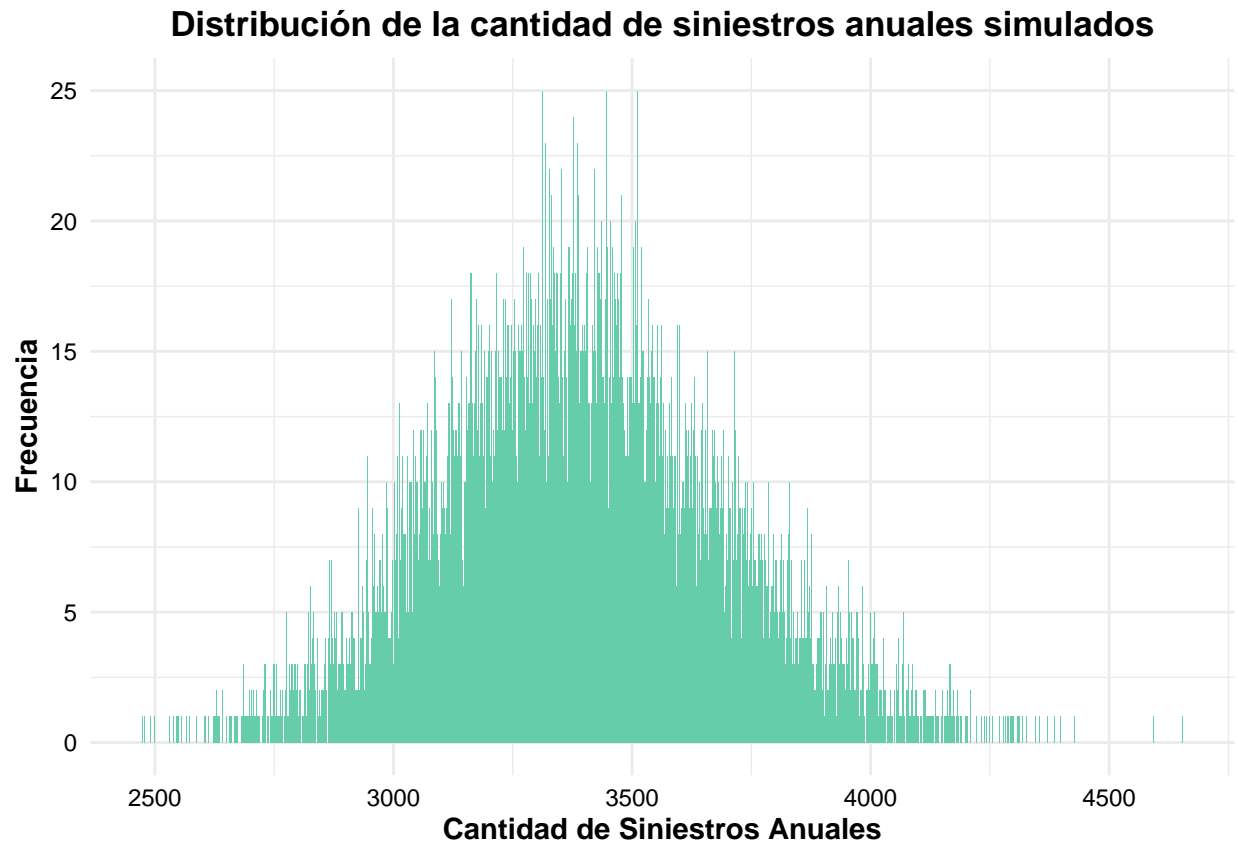
- Binomial Negativa: $\hat{\mu} = 3392,71$ y $\hat{\theta} = 143,89$
- Log Normal: $\log(\hat{\mu}) = 12,95$ y $\log(\hat{\sigma}) = 0,3617$
- Burr: $\hat{\alpha} = 1,08$, $\hat{\gamma} = 4,68$ y $\hat{\beta} = 426007,2$

Primero usamos datos históricos de los últimos 3 años: cuántas pólizas tuvimos y cuántos siniestros ocurrieron. Con eso, calculamos la tasa de siniestralidad promedio, es decir, la proporción de pólizas resultaron en un siniestro.

Luego, como para el año que viene esperamos tener una cantidad similar de pólizas, usamos esa tasa promedio para estimar cuántos siniestros podríamos tener. En lugar de usar un número fijo, se simuló la cantidad de siniestros usando una distribución **Binomial Negativa**. Para cada cantidad simulada de siniestros, generamos valores simulados de su costo individual. Y para eso usamos dos tipos distintos de distribuciones: **Burr** y **Log Normal**. Estas distribuciones reflejan bien las asimetrías de las cuantías observadas. Finalmente, para cada iteración, sumamos los costos individuales simulados para obtener un valor de la cuantía total anual a pagar.

Se realizaron 10.000 simulaciones del costo total anual. Por último, utilizando esta distribución analizamos cuál sería un escenario muy desfavorable (el peor 1% de los casos), y a partir de eso, calculamos el **Margen de Seguridad Mínimo**.

En primer lugar observamos como se distribuye la cantidad de siniestros anuales para las 10 mil simulaciones.



A partir de la simulación de la cantidad de siniestros anuales (utilizando un modelo de distribución Binomial Negativa), observamos que la frecuencia de eventos esperada para un año se sitúa alrededor de 3.500

sinistros. Los resultados de la simulación indican que es altamente probable que el número de siniestros anuales se encuentre en un rango acotado, oscilando típicamente entre 3.000 y 4.000.

Posteriormente, cada una de estas 10.000 simulaciones de la cantidad de siniestros anuales será utilizada como la base para generar la cuantía total que la compañía deberá afrontar. Para ello, modelaremos la severidad de cada siniestro (es decir, el monto individual de cada siniestro) utilizando las dos distribuciones previamente ajustadas a nuestros datos históricos: la distribución Lognormal y la distribución Burr. De esta manera, al sumar los montos individuales generados para cada simulación, se obtiene una distribución de la cuantía total anual, combinando tanto la frecuencia como la severidad de los siniestros.

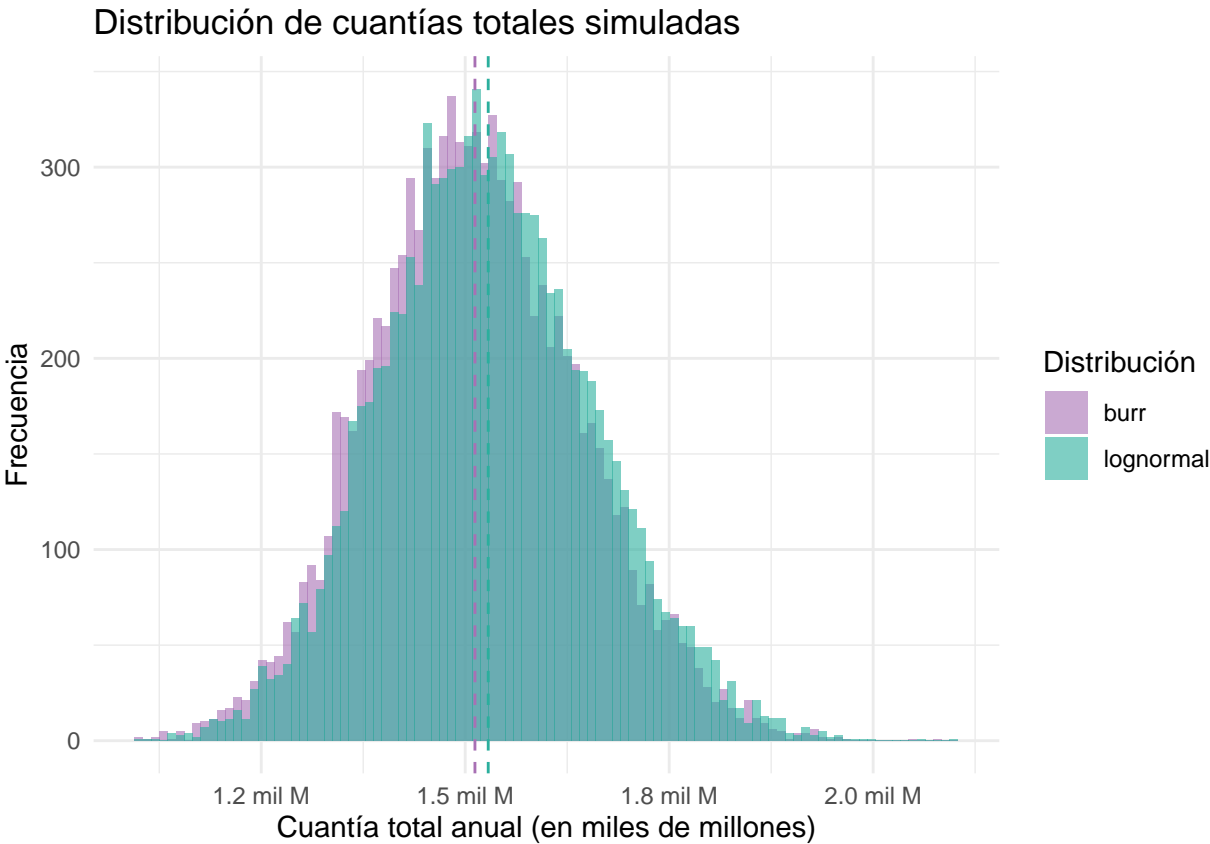


Table 2: Estadísticas descriptivas de las cuantías anuales simuladas

Distribución	Media	Desvío.estándar	Asimetría	Percentil.95	Percentil.99
Lognormal	\$1.528.183.555,18	\$129.424.984,18	0,17	\$1.749.935.942,32	\$1.841.353.093,33
Burr	\$1.511.909.266,10	\$128.306.473,14	0,17	\$1.731.241.441,29	\$1.824.398.819,32

Nota: Valores expresados en pesos argentinos al 01/01/2024

El margen de solvencia mínimo se calcula como la diferencia entre el valor que acumula una probabilidad del 99% en la distribución de la cuantía total (haciendo el supuesto de que la cuantía total se distribuye Normal Power) y las primas totales a cobrar (las cuales pueden o no estar multiplicadas por un porcentaje de recargo).

Finalmente, presentamos los resultados recargando las primas en 1%, 1,5%, 2,5%, 5% o 10%.

Resultados

A continuación presentamos los márgenes de solvencia mínimos obtenidos mediante simulación para diferentes porcentajes de recargo de seguridad, comparando los resultados bajo los dos modelos de severidad considerados: Log-Normal y Burr. La tabla resume los valores requeridos para garantizar un 99% de probabilidad de solvencia, mostrando cómo varían las necesidades de capital según el porcentaje de recargo aplicado y la distribución utilizada.

Los montos están expresados en pesos argentinos y representan el capital adicional que la compañía debería mantener para cubrir posibles desviaciones adversas en su cartera de seguros automotores durante el año 2024.

Table 3: Margen de Solvencia Mínimo según porcentaje de recargo de seguridad

Porcentaje de Recargo de Seguridad	Margen de Solvencia Mínimo Log-Normal	Margen de Solvencia Mínimo Burr
1%	\$302.082.238	\$299.790.852
1,5%	\$294.441.320	\$292.231.306
2,5%	\$279.159.484	\$277.112.213
5%	\$240.954.896	\$239.314.481
10%	\$164.545.718	\$163.719.018

Nota: Valores expresados en pesos argentinos al 01/01/2024

En conclusión, observamos que a mayor recargo aplicado, menor es el capital que la compañía necesita reservar para garantizar su solvencia en el 99% de los escenarios posibles. Esta herramienta es ampliamente utilizada en la práctica actuarial, ya que permite dimensionar adecuadamente el riesgo. Sin embargo, es importante tener en cuenta que para atraer nuevos clientes es necesario ofrecer primas competitivas en relación con el resto del mercado. La determinación final de la prima que los productores comercializan surge, entonces, del equilibrio entre este análisis técnico, la presencia de marca y el contexto macroeconómico, entre otros factores.

Anexo

Se anexa el código utilizado.

```
library(readxl)
library(dplyr)
library(fitdistrplus)
library(actuar)
library(ggplot2)
library(lubridate)
library(e1071)
library(knitr)
library(kableExtra)

cuantias_23 <- read_excel("Trabajo Final 2024 Base de Datos .xlsx",
                        col_types = c("date", "numeric"))
cer23 <- read_excel("cer23.xlsx",
                  col_types = c("date", "numeric"))

cuantias_23 <- cuantias_23 %>%
  left_join(cer23, by = "Fecha")

cuantias_23 <- cuantias_23 %>%
  mutate(cuantia24 = `Cuantia` * (cer23$ValorCER[366]/ValorCER))

tabla_cunatia <- cuantias_23
colnames(tabla_cunatia) <- c('Fecha', 'Cuantía', 'Valor CER', 'Cuantía Actualizada')
head(tabla_cunatia, 10)%>%
  kbl(caption = "Tabla 1: Cuantías Pagadas en 2023.", align = "c") %>% # Añadir un título
  kable_styling(bootstrap_options = "striped", full_width = FALSE) %>%
  row_spec(0, background = "#66CDAA")

##### DESCRIPTIVO #####

cuantias_23 <- cuantias_23 %>%
  mutate(mes = format(as.Date(Fecha), "%Y-%m"))
ggplot(cuantias_23, aes(x = mes)) +
  geom_bar(fill = "#66CDAA") +
  labs(title = "Cantidad de cuantías pagadas por mes (frecuencia)",
       x = "Mes",
       y = "Cantidad de pagos") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

#####

ggplot(cuantias_23, aes(x = cuantia24)) +
  geom_histogram(fill = "#66CDAA", bins = 60, color = "white") +
  labs(title = "Distribución de las cuantías actualizadas (severidad)",
```



```

    x = "Cuantía Actualizada",
    y = "Frecuencia") +
  theme_minimal() +
  coord_cartesian(xlim = c(0, 2000000))

datos <- na.omit(cuantias_23$cuantia24)

# Ajustes
ajuste_lnorm <- fitdist(datos, "lnorm")
ajuste_weibull <- fitdist(datos, "weibull")
ajuste_burr <- fitdist(datos, "burr", start = list(shape1 = 2, shape2 = 2, rate = 2))
ajuste_pareto <- fitdist(datos, "pareto", start = list(shape = 2, scale = min(datos)))

ggplot(data.frame(x = datos), aes(x = x)) +
  geom_histogram(aes(y = after_stat(density)), bins = 50, fill = "#66CDAA", color = "black", alpha = 0.5)

# --- Todas las líneas en tipo "solid" (continua) y tamaño 0.7 (más fino) ---
stat_function(fun = dlnorm,
  args = list(meanlog = ajuste_lnorm$estimate["meanlog"],
    sdlog = ajuste_lnorm$estimate["sdlog"]),
  aes(color = "Log-normal", linetype = "Log-normal"),
  size = 0.7) + # Más fina
stat_function(fun = dweibull,
  args = list(shape = ajuste_weibull$estimate["shape"],
    scale = ajuste_weibull$estimate["scale"]),
  aes(color = "Weibull", linetype = "Weibull"),
  size = 0.7) + # Más fina
stat_function(fun = dburr,
  args = list(shape1 = ajuste_burr$estimate["shape1"],
    shape2 = ajuste_burr$estimate["shape2"],
    rate = ajuste_burr$estimate["rate"]),
  aes(color = "Burr", linetype = "Burr"),
  size = 0.7) + # Más fina
stat_function(fun = dpareto,
  args = list(shape = ajuste_pareto$estimate["shape"],
    scale = ajuste_pareto$estimate["scale"]),
  aes(color = "Pareto", linetype = "Pareto"),
  size = 0.7) + # Más fina

# --- Personalización de colores y tipos de línea en la leyenda ---
scale_color_manual(name = "Distribución",
  values = c("Log-normal" = "darkblue",
    "Weibull" = "#5F9EA0",
    "Burr" = "#8B0A50",
    "Pareto" = "#CD5555")) +
scale_linetype_manual(name = "Distribución",
  values = c("Log-normal" = "solid", # Todas "solid"
    "Weibull" = "solid",
    "Burr" = "solid",
    "Pareto" = "solid")) +

# --- Formato del Eje X ---

```

```

scale_x_continuous(labels = scales::number_format(big.mark = ".", decimal.mark = ",", accuracy = 1))
labs(title = "Ajuste de distribuciones a las cuantías",
     x = "Valor de las Cuantías",
     y = "Densidad") +
theme_minimal() +
theme(legend.position = "right",
      legend.title = element_text(face = "bold"),
      legend.text = element_text(size = 10),
      axis.title.y = element_blank(), # Elimina el título del eje Y
      axis.text.y = element_blank(), # Elimina los números/etiquetas del eje Y
      axis.ticks.y = element_blank(), # Elimina las marcas del eje Y
      axis.line.y = element_blank() # Elimina la línea del eje Y
    )+
coord_cartesian(xlim = c(0, 2500000))

##### SIMULACION #####
# Cant siniestros
## Binomial negativa
# Datos históricos
polizas <- c(24752, 25348, 25615)
siniestros <- c(3023, 3581, 3431)

tasas <- siniestros / polizas # Tasa siniestral por año
tasa_prom <- mean(tasas) # Media de la tasa siniestral
polizas_2024 <- 25615 # Simulación para 25615 pólizas
mu <- tasa_prom * polizas_2024
var_siniestros <- var(siniestros)
size <- mu^2 / (var_siniestros - mu)

# Simular 1000 observaciones de siniestros
set.seed(1511) # Cantidad de años a simular

# Cantidad de años a simular
n_sim <- 10000

# Vectores para guardar resultados
suma_burr <- numeric(n_sim)
suma_lnorm <- numeric(n_sim)

# Simulación año por año
for (i in 1:n_sim) {
  # Simular cantidad de siniestros
  cantidad <- rbinom(1, mu = mu, size = size)

  # Simular severidades y sumar
  if (cantidad > 0) {
    burr_vals <- rburr(cantidad,
                      shape1 = ajuste_burr$estimate["shape1"],
                      shape2 = ajuste_burr$estimate["shape2"],
                      rate = ajuste_burr$estimate["rate"])

    lnorm_vals <- rlnorm(cantidad,
                       meanlog = ajuste_lnorm$estimate["meanlog"],

```

```

sdlog = ajuste_lnorm$estimate["sdlog"])

suma_burr[i] <- sum(burr_vals)
suma_lnorm[i] <- sum(lnorm_vals)
} else {
  suma_burr[i] <- 0
  suma_lnorm[i] <- 0
}
}

# Resultado final como data.frame
resultados <- data.frame(lognormal = suma_lnorm,
                          burr = suma_burr)

media <- mean(resultados$lognormal)
desvio <- sd(resultados$lognormal)
coef_asimetria <- skewness(resultados$lognormal)

z_99 <- qnorm(0.99, 0, 1)
y_99 <- z_99 + coef_asimetria/6 * (z_99^2-1)

y <- y_99* desvio + media

MSM_1porc <- y-(media*1.01)
MSM_1.5porc <- y-(media*1.015)
MSM_2.5porc <- y-(media*1.025)
MSM_5porc <- y-(media*1.05)
MSM_10porc <- y-(media*1.1)

MSM_log <- c(MSM_1porc, MSM_1.5porc, MSM_2.5porc,MSM_5porc,MSM_10porc)

media_burr <- mean(resultados$burr)
desvio_burr <- sd(resultados$burr)
coef_asimetria_burr <- skewness(resultados$burr)

z_99 <- qnorm(0.99, 0, 1)
y_99 <- z_99 + coef_asimetria_burr/6 * (z_99^2-1)

y <- y_99* desvio_burr + media_burr

MSM_1porc_burr <- y-(media_burr*1.01)
MSM_1.5porc_burr <- y-(media_burr*1.015)
MSM_2.5porc_burr <- y-(media_burr*1.025)
MSM_5porc_burr <- y-(media_burr*1.05)
MSM_10porc_burr <- y-(media_burr*1.1)
MSM_burr <- c(MSM_1porc_burr, MSM_1.5porc_burr, MSM_2.5porc_burr,MSM_5porc_burr,MSM_10porc_burr)

library(e1071) # para skewness

```

```

# Cálculo de estadísticas
stats <- resultados %>%
  summarise(
    media_log = mean(lognormal),
    sd_log = sd(lognormal),
    skew_log = skewness(lognormal),
    p50_log = quantile(lognormal, 0.50),
    p75_log = quantile(lognormal, 0.75),
    p90_log = quantile(lognormal, 0.90),
    p95_log = quantile(lognormal, 0.95),
    p99_log = quantile(lognormal, 0.99),

    media_burr = mean(burr),
    sd_burr = sd(burr),
    skew_burr = skewness(burr),
    p50_burr = quantile(burr, 0.50),
    p75_burr = quantile(burr, 0.75),
    p90_burr = quantile(burr, 0.90),
    p95_burr = quantile(burr, 0.95),
    p99_burr = quantile(burr, 0.99)
  )

# Convertir a formato largo para graficar
resultados_long <- resultados %>%
  tidyr::pivot_longer(cols = everything(), names_to = "distribucion", values_to = "cuantia_total")

library(scales) # para label_number

ggplot(resultados_long, aes(x = cuantia_total, fill = distribucion)) +
  geom_histogram(alpha = 0.6, position = "identity", bins = 100) +
  geom_vline(data = stats, aes(xintercept = media_log), color = "#2BAF9D", linetype = "dashed") +
  geom_vline(data = stats, aes(xintercept = media_burr), color = "#A870B4", linetype = "dashed") +
  labs(
    title = "Distribución de cuantías totales simuladas",
    x = "Cuantía total anual (en miles de millones)",
    y = "Frecuencia",
    fill = "Distribución"
  ) +
  scale_x_continuous(
    labels = label_number(scale = 1e-9, suffix = " mil M", accuracy = 0.1)
  ) +
  scale_fill_manual(values = c("lognormal" = "#2BAF9D", "burr" = "#A870B4")) +
  theme_minimal()

tabla1 <- data.frame(
  Distribución = c("Lognormal", "Burr"),
  Media = round(c(stats$media_log, stats$media_burr), 2),
  `Desvío estándar` = round(c(stats$sd_log, stats$sd_burr), 2),
  `Asimetría` = round(c(stats$skew_log, stats$skew_burr), 2),
  `Percentil 95` = round(c(stats$p95_log, stats$p95_burr), 2),
  `Percentil 99` = round(c(stats$p99_log, stats$p99_burr), 2)
)

```

```

# Aplicar el formato deseado a las columnas numéricas con formato monetario
columnas_monetarias <- c("Media", "Desvío.estándar", "Percentil.95", "Percentil.99")

# Usar `mutate` con `across` para aplicar la función a múltiples columnas
tabla1 <- tabla1 %>%
  mutate(across(.cols = all_of(columnas_monetarias),
    .fns = ~ paste0("$", formatC(.,
      big.mark = ".",
      decimal.mark = ",",
      format = "f",
      digits = 2)))) # Mantén 2 decimales para pesos si aplica

tabla1 <- tabla1 %>%
  mutate(across(.cols = "Asimetría",
    .fns = ~ paste0(formatC(.,
      big.mark = ".",
      decimal.mark = ",",
      format = "f",
      digits = 2)))) # Mantén 2 decimales para pesos si aplica

# Mostrar tabla en formato LaTeX
tabla1 %>%
  kbl(caption = "Estadísticas descriptivas de las cuantías anuales simuladas",
    align = "c",
    format = "latex",
    booktabs = TRUE,
    linesep = "") %>%
  kable_styling(latex_options = c("striped", "hold_position", "scale_down"),
    full_width = FALSE,
    position = "center",
    font_size = 10) %>%
  row_spec(0, background = "#66CDAA", bold = TRUE, color = "black") %>%
  column_spec(1, width = "4cm") %>%
  column_spec(2:6, width = "3cm") %>% # Asegúrate de que esto se ajuste al número de columnas
  footnote(general = "Nota: Valores expresados en pesos argentinos al 01/01/2024",
    general_title = "")

##### RESULTADOS #####

library(kableExtra)
library(knitr)
porc <- c("1%", "1,5%", "2,5%", "5%", "10%")
tabla1 <- data.frame(porc, MSM_log, MSM_burr)
# Formateo de la tabla
colnames(tabla1) <- c("Porcentaje de Recargo de Seguridad",
  "Margen de Solvencia Mínimo Log-Normal",
  "Margen de Solvencia Mínimo Burr")

# Formateo de valores numéricos con separadores de miles
tabla1$`Margen de Solvencia Mínimo Log-Normal` <- paste0("$", formatC(tabla1$`Margen de Solvencia Mínimo Log-Normal`,
  big.mark = ".",
  decimal.mark = ",",
  format = "f",
  digits = 0))

```

```

tabla1$`Margen de Solvencia Mínimo Burr` <- paste0("$", formatC(tabla1$`Margen de Solvencia Mínimo Burr`,
                                                                    big.mark = ".",
                                                                    decimal.mark = ",",
                                                                    format = "f",
                                                                    digits = 0))

# Creación de la tabla con formato profesional
tabla1 %>%
  kbl(caption = "Margen de Solvencia Mínimo según porcentaje de recargo de seguridad",
      align = "c",
      format = "latex",
      booktabs = TRUE,
      linesep = "") %>%
  kable_styling(latex_options = c("striped", "hold_position", "scale_down"),
                full_width = FALSE,
                position = "center",
                font_size = 10) %>%
  row_spec(0, background = "#66CDAA", bold = TRUE, color = "black") %>%
  column_spec(1, width = "4cm") %>%
  column_spec(2:3, width = "5cm") %>%
  footnote(general = "Nota: Valores expresados en pesos argentinos al 01/01/2024",
          general_title = "")

```