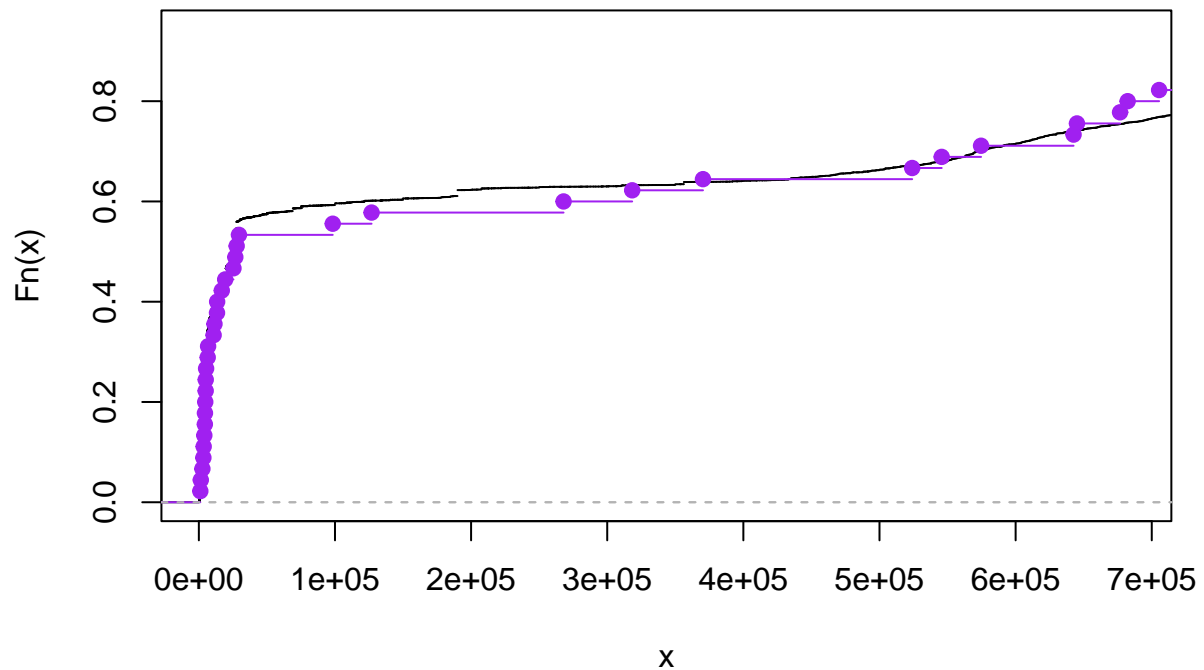


CDF empírica y teórica



Prueba de escenarios

```
#####
##Prueba de escenarios
LMR = 12000000
longitud1 = length(z)
z1 = z-LMR
length(which(z1>0))/longitud1
```

```
## [1] 0
```

```
summary(z1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## -11998915 -11994675 -11972211 -11656241 -11354945 -10208161
```

Modelando frecuencia siniestral

En el archivo anexo al presente trabajo puede obtenerse la frecuencia siniestral en el ejercicio 2025: Como insumo

```
library("readr")
library("stats")
library("tseries")
```

```
## Registered S3 method overwritten by 'quantmod':
##   method      from
##   as.zoo.data.frame zoo
```

```
library("nortest")
```

```
##
## Adjuntando el paquete: 'nortest'

## The following objects are masked from 'package:gofstest':
##
##   ad.test, cvm.test
```

```
library("FinTS")
```

```
## Cargando paquete requerido: zoo

##
## Adjuntando el paquete: 'zoo'

## The following objects are masked from 'package:base':
##
##   as.Date, as.Date.numeric
```

```
library("ZINARp")
```

```
serie_tiempo <- read_csv("C:/Users/agarciadeleon/R_Studio/Proyectos_R/TS/serie_tiempo.csv")
```

```
## Rows: 48 Columns: 1
```

```
## -- Column specification -----
## Delimiter: ","
## dbl (1): S
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
serie = ts(serie_tiempo$S, start=c(2022, 1),end = c(2025, 9), frequency=12)
```

Como insumo

```
library("readr")
library("stats")
library("tseries")
library("nortest")
library("FinTS")
library("ZINARp")
```

```
serie_tiempo <- read_csv("C:/Users/agarciadeleon/R_Studio/Proyectos_R/TS/serie_tiempo.csv")
```

```
## Rows: 48 Columns: 1
## -- Column specification -----
## Delimiter: ","
## dbl (1): S
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

serie = ts(serie_tiempo$S, start=c(2022, 1), end = c(2025, 9), frequency=12)
```

Enfoque Zinar(p)

Se usará la técnica series de tiempo generalizadas usando distribución de parte media móvil

$$X_t - \sum_{i=1}^p \alpha_i X_{t-i} \sim \text{Poisson}(\lambda)$$

Usando optimización Bayesiana pueden estimarse los parámetros del modelo asumiendo verosimilitud poisson; En nuestro caso probamos con el orden p desde 1 hasta 4 donde se consideró se tuvo un excelente ajuste.

```
set.seed(1234)
serie
```

```
##      Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 2022   0   0   0   0   0   0   0   0   0   2   0   0
## 2023   3   0   0   0   0   0   0   0   0   0   0   2
## 2024   7   2   1   0   3   8   5  11   0   5   6  15
## 2025  16  11   1  12   9  10   7   2   3
```

```
x = serie
p=4
m = estimate_zinarp(
  x,
  p,
  iter = 5000,
  thin = 2,
  burn = 0.1,
  innovation = "Poisson"
)
```

Se guardaron los parámetros del modelo

```
a1 = mean(m$alpha[,1])
a2 = mean(m$alpha[,2])
a3 = mean(m$alpha[,3])
a4 = mean(m$alpha[,4])
lambda = mean(m$lambda)
```

Se contrastó con la serie original

```

set.seed(1234)
serie1 = serie[-length(serie)]
serie2 = serie[-c(length(serie), (length(serie)-1))]
serie3 = serie[-c(length(serie), (length(serie)-1), (length(serie)-2))]
serie4 = serie[-c(length(serie), (length(serie)-1), (length(serie)-2), length(serie)-3)]
poi = rpois(length(serie1), lambda = lambda)
Proceso = a1*serie1 + a2*serie2 + a3*serie3 + a4*serie4 + poi

```

```

## Warning in a1 * serie1 + a2 * serie2: longitud de objeto mayor no es múltiplo
## de la longitud de uno menor

```

```

## Warning in a1 * serie1 + a2 * serie2 + a3 * serie3: longitud de objeto mayor no
## es múltiplo de la longitud de uno menor

```

```

## Warning in a1 * serie1 + a2 * serie2 + a3 * serie3 + a4 * serie4: longitud de
## objeto mayor no es múltiplo de la longitud de uno menor

```

```

ks.test(Proceso, serie)

```

```

##
## Exact two-sample Kolmogorov-Smirnov test
##
## data: Proceso and serie
## D = 0.22222, p-value = 0.1056
## alternative hypothesis: two-sided

```

Dando un valor de excelente ajuste.

Enfoque Proyectivo

Este enfoque no depende de supuestos probabilísticos; en álgebra lineal una proyección ortogonal es una transformación idempotente, y tal que es normal, es decir en el caso real, conmuta con su adjunto $T \circ T^* = T^* \circ T$. Adicionalmente, en este caso, el adjunto es la proyección sobre el complemento ortogonal y el vector proyección es el que satisface que la distancia de este a el complemento ortogonal respectivo es mínima. Esto se traduce en que si la distancia es el error cuadrático medio, entonces este se minimizará para el caso del estimador pronóstico. Se usa este enfoque para no depender de un ajuste paramétrico particular sino de la proyección de la historia entera hacia el futuro. Se puede observar que el pronóstico es con aproximadamente 45 periodos de historia, es decir el pronóstico depende de 45 parámetros. Afortunadamente el modelo expresado arriba solo depende de 5 lo cual como se verá más adelante, aunado a que ambos satisfacen bondad de ajuste, hace que sea más facil de manejar en términos predictivos.

```

#####Ajuste por mínimos cuadrados
autocov <- function(x, k){
  n <- length(x)
  if(k == n){
    k = 0
  }
  x_bar <- mean(x)
  sum((x[(k+1):n] - x_bar) * (x[1:(n-k)] - x_bar)) / n
}

```

```

autocor <- function(x, k){
  autocov(x,k)/autocov(x,0)
}

##SIGMA
tamano = length(serie)
serie_k = serie[1:tamano]
A = matrix(nrow = tamano, ncol = tamano)
for(i in 1:nrow(A)){
  for(j in 1:ncol(A)){
    if(j>i){
      A[i,j] = autocor(serie_k,j-i)}else{
      A[i,j] = autocor(serie_k,i-j)
    }
  }
}
tam = 12
b = matrix(rep(0, times = tamano*tam), ncol = tam)
#tendencia =rep(0, times = h)
for(h in 1:tam){
  #b = gamma(n-s-k) k:1:n
  #s horizonte prospectivo
  for(b1 in 1:tamano){
    b[b1,h] = autocor(serie_k,tamano + h - b1)
  }
  tendencia = solve(A)%*%b[,h]
  tendencia = sum(tendencia)
  serie_k = c(serie_k,tendencia)
}

#Validación
length(serie_k)

```

```
## [1] 57
```

```
length(serie)
```

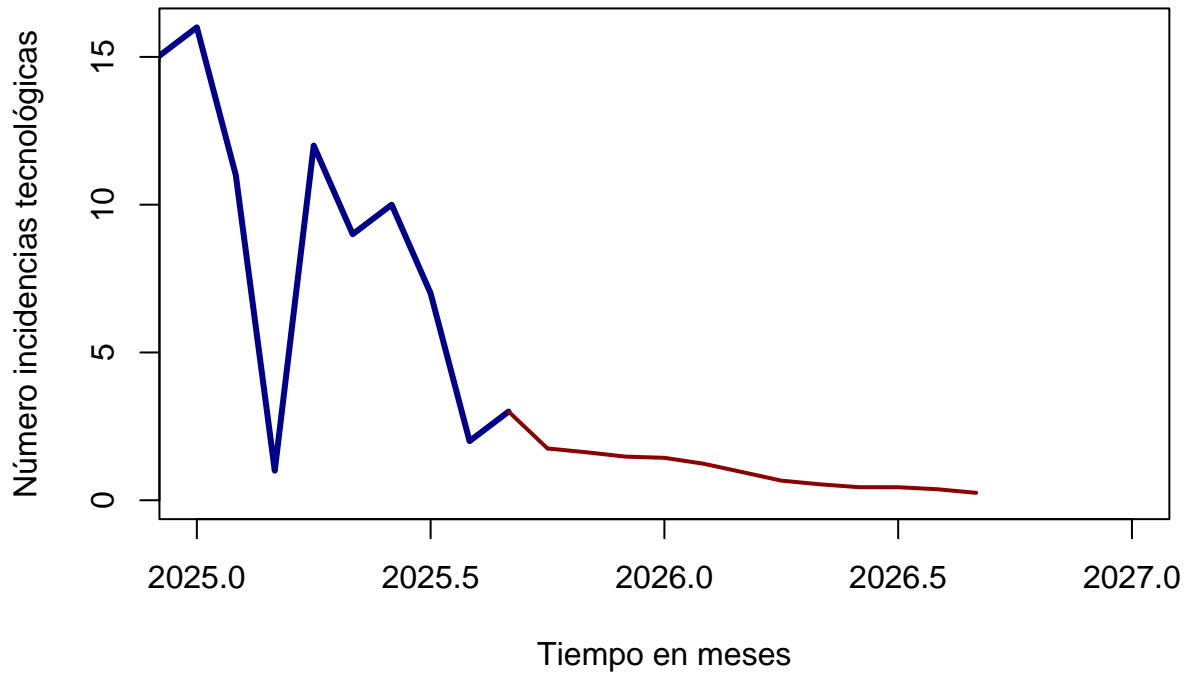
```
## [1] 45
```

```

plot(serie, col = "darkblue", lty = 1, lwd = 3, ylab = "Número incidencias tecnológicas", xlab = "Tiempo")
lines(ts(serie_k[46:57],start = c(2025,10),frequency = 12), col = "darkred", lty = 1, lwd = 2)
segments(2026-4/12, serie[length(serie)], 2026-3/12, ts(serie_k[46],start = c(2025,10),frequency = 12),col = "darkred",lty = 1,lwd = 3)

```

Serie Original y Predicciones



```
ks.test(Proceso,seriek)
```

```
##
## Exact two-sample Kolmogorov-Smirnov test
##
## data: Proceso and seriek
## D = 0.17544, p-value = 0.2667
## alternative hypothesis: two-sided
```

Nuevamente ambos presentan excelente ajuste, enfoque proyectivo y enfoque paramétrico.

```
#####
z1 = z -LMR
length(which(z1>0))/longitud1
```

```
## [1] 0
```

```
summary(z1)
```

```
##      Min.   1st Qu.   Median     Mean   3rd Qu.    Max.
## -11998915 -11994675 -11972211 -11656241 -11354945 -10208161
```

Análisis de sensibilidad