

Simulación de trayectorias de velocidad de viento con distribución arbitraria y autocorrelación de decaimiento exponencial

Proyecto de fin de Curso.

Jonatan Pablo Arenas López

1. Introducción

En los últimos años, el objetivo de reducir los costos de la producción de electricidad y el cumplimiento de los tratados internacionales sobre el cambio climático han promovido la inclusión de recursos de generación limpios y sostenibles en los sistemas eléctricos. Sin embargo, la creciente penetración de esos recursos, cuyo comportamiento es variable e intermitente, ha planteado considerables problemas para mantener la fiabilidad del suministro en los sistemas eléctricos.

Desde el punto de vista matemático, entre las características estadísticas de la velocidad del viento, las funciones de densidad de probabilidad (FDP) y la función de autocorrelación serán el eje de la presente trabajo.

En [1] se propuso un modelo de velocidad del viento basado en el proceso de Ornstein-Uhlenbeck y una transformación sin memoria llamada proceso de traducción. Para ilustrar el rendimiento de este modelo, los datos de dos lugares en Canadá reportados en [3, 4] se utilizaron para obtener los parámetros de la distribución de Weibull y la función de autocorrelación de la velocidad del viento. Además, el procedimiento anterior se utilizó para modelar la velocidad del viento de un lugar situado en el punto más al norte de Irlanda en [2].

En este documento, el enfoque presentado en [1] es desarrollado en su totalidad en el lenguaje de programación R (análisis de datos, proceso de ajuste y simulación).

2. Modelo matemático

Una ecuación diferencial estocástica (EDE) unidimensional en el intervalo de tiempo $[0, T]$ tiene la siguiente forma:

$$dX_t = \nu(t, X_t)dt + \beta(t, X_t)dW_t, \quad (1)$$

donde las funciones $\nu(t, X_t)$ y $\beta(t, X_t)$ son los llamados términos de deriva y volatilidad, respectivamente y W_t es un proceso estándar de Wiener [5, 6]. La ecuación (1) se puede escribir de forma integral de la siguiente manera:

$$X_t = X_0 + \int_0^t \nu(u, X_u)du + \int_0^t \beta(u, X_u)dW_u, \quad (2)$$

con condición inicial X_0 . La primera integral es una integral ordinaria de Riemann-Stieltjes y la segunda se define en el sentido de Itô.

2.1. Proceso Ornstein-Uhlenbeck

Sean $\alpha, \theta > 0$, el proceso Ornstein-Uhlenbeck (OU) se define como la solución de la siguiente EDE:

$$dX_t = -\alpha X_t dt + \theta dW_t. \quad (3)$$

La aplicación del lema de Itô a la función real $f(t, X_t) = X_t e^{\alpha t}$ lleva a la siguiente expresión:

$$X_t = X_0 e^{-\alpha t} + \theta \int_0^t e^{-\alpha(t-s)} dW_s. \quad (4)$$

El proceso resultante tiene las siguientes propiedades estadísticas principales [6, 7]:

- Media

$$\mathbb{E}[X_t] = \mathbb{E}[X_0] e^{-\alpha t}. \quad (5)$$

- Varianza

$$\mathbb{V}ar[X_t] = \left(\mathbb{V}[X_0] - \frac{\theta^2}{2\alpha} \right) e^{-2\alpha t} + \frac{\theta^2}{2\alpha}. \quad (6)$$

- Autocorrelación

$$\text{Aut}[X_t, X_s] = e^{-\alpha(\tau)}. \quad (7)$$

donde $\tau = t - s$ es el tiempo de retardo con $s < t$. Si $X_0 \sim \mathcal{N}(0, 1)$, las ecuaciones (5)-(6) se convierten respectivamente en $\mathbb{E}[X_t] = 0$ and $\mathbb{V}ar[X_t] = \frac{\theta^2}{2\alpha}$. Finalmente, por simplicidad el proceso de OU es adaptado a una distribución normal estándar tomando $\theta = \sqrt{2\alpha}$

2.2. Transformaciones sin memoria

Para obtener una distribución de probabilidad dada, se aplica la siguiente transformación sin memoria a (4) [8]:

$$Y_t = F^{-1}(\Phi(X_t)) \quad (8)$$

donde $\Phi(\cdot)$ es la función de distribución acumulada (FDA) de la distribución $\mathcal{N}(0, 1)$ y $F(\cdot)$ es la FDA de la distribución de probabilidad deseada. Como resultado de la transformación sin memoria (8), un proceso de traslación Y_t se obtiene con la distribución de probabilidad deseada. Debido a la transformación sin memoria, no se garantiza que la autocorrelación de Y_t sea la autocorrelación exponencial del proceso original X_t definido por (4) [1, 2]. En este trabajo la autocorrelación se analiza numéricamente.

3. Simulación

Para generar trayectorias de velocidad del viento resolvemos numéricamente la ecuación (3). El primer paso consiste en la discretización del dominio del tiempo: $t_{n+1} = t_n + \delta$ para $n \geq 0$ donde δ es el tamaño de paso. El proceso discretizado de X_t se convierte en $X_{t_{n+1}} := X_{n+1}$, entonces al implementar un esquema de Euler-Maruyama [9], la EDE (3) puede ser aproximado de la siguiente manera:

$$X_n = X_{n-1} - \alpha X_{n-1} \delta + \sqrt{2\alpha} \Delta W_n, \quad (9)$$

donde $\Delta W_n = W(t_n) - W(t_{n-1}) \sim \mathcal{N}(0, \delta)$ son incrementos aleatorios del proceso Wiener.

La transformación sin memoria dada por (8) se aplica al proceso estocástico discretizado X_n para construir las trayectorias de la velocidad del viento

$$Y_n = F^{-1}(\Phi(X_n)) \quad (10)$$

para la FDA inversa F^{-1} deseada.

4. Caso de Estudio

En esta sección, el modelo desarrollado anteriormente se utiliza para modelar la velocidad del viento de una ubicación del mundo real. Tal ubicación es Unión Hidalgo, una población ubicada en una región del Istmo de Tehuantepec, en el Estado de Oaxaca, México. El conjunto de datos está compuesto por 8760 estimaciones horarias de la velocidad media del viento correspondiente al año 2018 y pertenece el conjunto de datos de reanálisis ERA5 el cual proporciona estimaciones por hora de una gran cantidad de variables climáticas atmosféricas, terrestres y oceánicas [10].

4.1. Análisis de datos

La tabla 1 muestra las estadísticas descriptivas que incluyen la velocidad mínima (mín), velocidad máxima (máx), velocidad media, desviación estándar y mediana. Para más detalles sobre la asimetría, el grosor o la pesadez de la distribución de los datos, se proporcionan caracterizaciones adicionales, que son la asimetría y la curtosis.

La figura 1 muestra el histograma normalizado del conjunto de datos considerados. En la figura 2 se muestra la autocorrelacion del conjunto de datos considerados para retardos de tiempo de hasta 120 horas (5 días).

Tabla 1: Estadística descriptiva del conjunto de velocidades del viento.

Mín	Máx	Media	Desviación estándar	Mediana	Coefficiente de asimetría	Curtosis
[m/s]	[m/s]	[m/s]	[m/s]	[m/s]		
0.04	20.78	7.090223	3.993708	6.58	0.4583614	2.565338

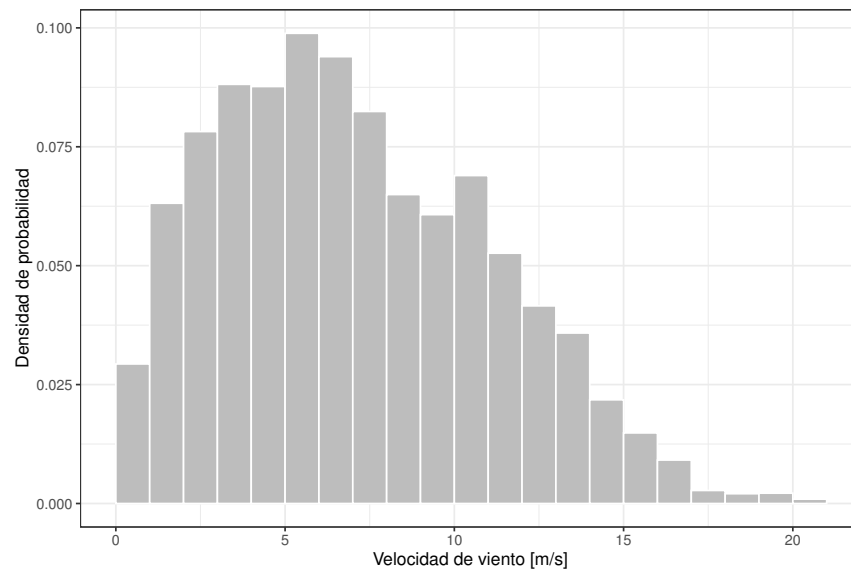


Figura 1: Histograma normalizado de los datos de la velocidad media del viento por hora.

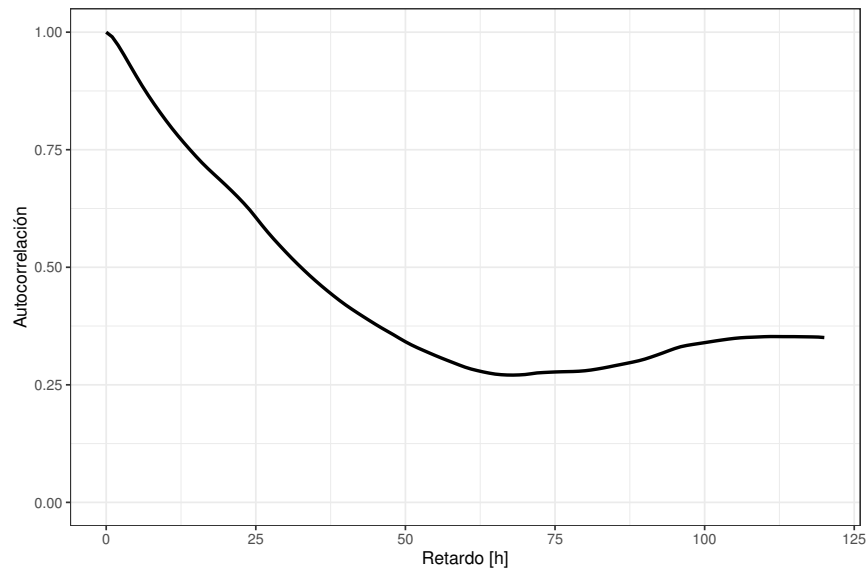


Figura 2: Autocorrelación de los datos de la velocidad media del viento por hora.

El análisis presentado en esta sección usando R fue realizado mediante el código 1.

Código 1. *DataAnalysis.R*

```
library(ggplot2)
library(dplyr)
library(moments)

dataset2018 <- read.csv("https://raw.githubusercontent.com/angeljpal/ProyectoR/main/
dataset2018.csv")

WS <- dataset2018$Speed_100m_m.s

min(WS)
max(WS)
mean(WS)
sd(WS)
median(WS)
skewness(WS)
kurtosis(WS)

dataset2018 %>% ggplot(aes(x=Speed_100m_m.s)) +
  geom_histogram(aes(y=..density..), binwidth = 1,
  fill='gray', color = 'white', alpha=1, boundary = 0) +
  xlab("Velocidad de viento [m/s]") +
  ylab("Densidad de probabilidad") +
  theme_bw()

acf.WS <- acf(WS, lag.max = 120, plot = FALSE)
```

```
FAC <- data.frame(acf.WS$acf,acf.WS$lag)

FAC %>% ggplot(aes(acf.WS.lag, acf.WS.acf)) +
  geom_line(size = 1) +
  theme_bw() +
  xlab("Retardo [h]") +
  ylab("FAC") +
  xlim(c(0,120)) +
  ylim(c(0,1))
```

4.2. Procesos de ajuste

Mediante el programa R se realiza un proceso de ajuste de los datos de velocidades de viento a un conjunto de distribuciones de probabilidad. Los resultados de pruebas estadísticas de bondad de ajuste como son Log-likelihood, Kolmogórov-Smirnov, Anderson-Darling, entre otras, además de cuatro comparaciones gráficas (FDP teórica contra FDP empírica, cuantil-cuantil, FDA teórica contra FDA empírica y probabilidad-probabilidad) muestran que el mejor desempeño para modelar el comportamiento de los datos de velocidad de viento esta dado por la distribución Weibull con coeficientes de ajuste de forma $\lambda_1 = 1.816126$ y escala $\lambda_2 = 7.962235$.

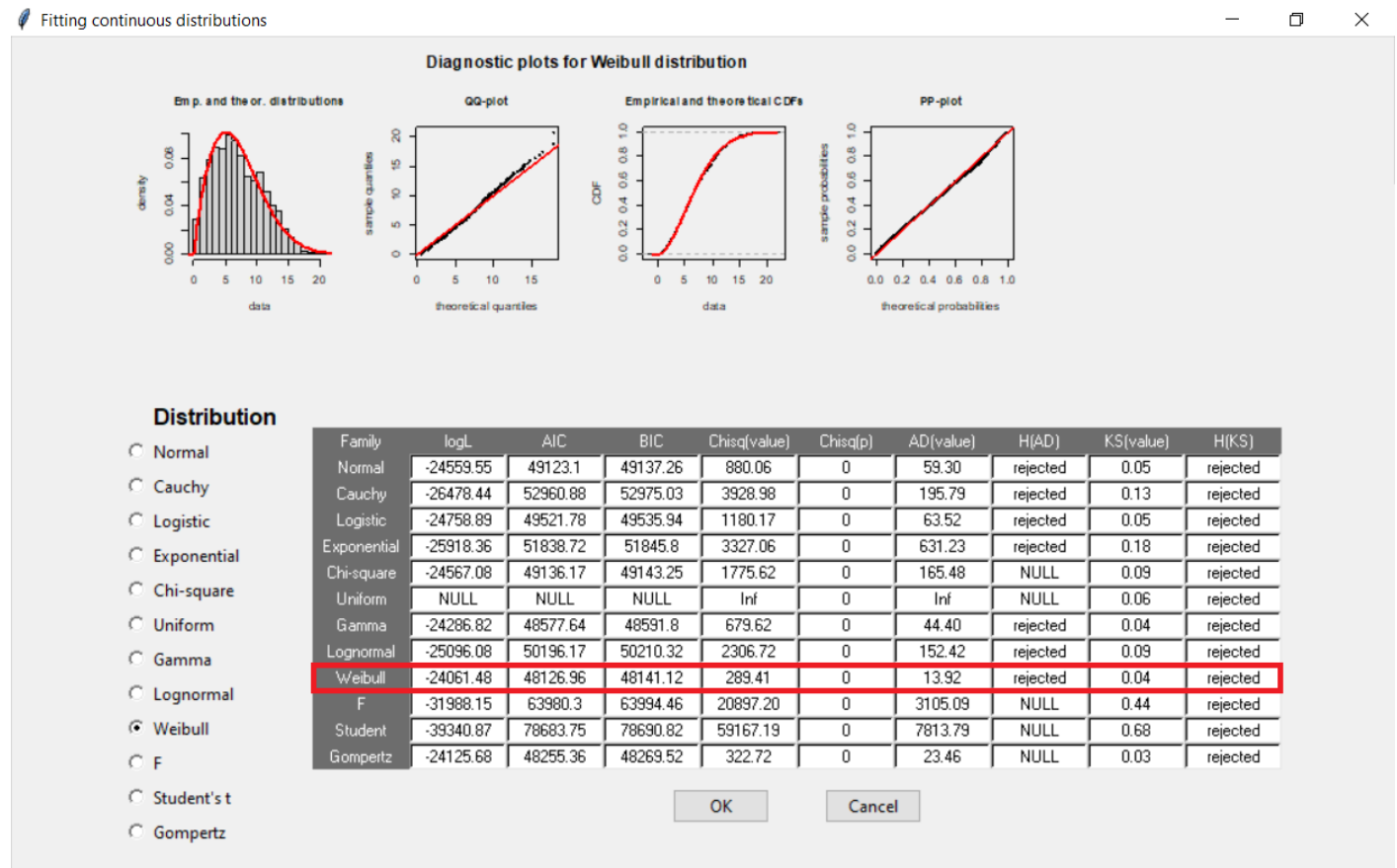


Figura 3: Proceso de ajuste de FDP a los datos de velocidad de viento con R.

La figura 4 muestra el histograma normalizado del conjunto de datos considerados y la FDP de ajuste. Puede observarse que la FDP de Weibull se ajusta bastante bien a la densidad de probabilidad empírica de los datos. La figura 5 muestra la autocorrelación del conjunto de datos considerados para retardos de hasta 120

horas y el ajuste exponencial de acuerdo a la expresión (7). Como puede observarse, la función exponencial es una buena aproximación a la autocorrelación para retardos de hasta 68 horas aproximadamente. El coeficiente de autocorrelación obtenido del ajuste exponencial es $\alpha = 0.0209$.

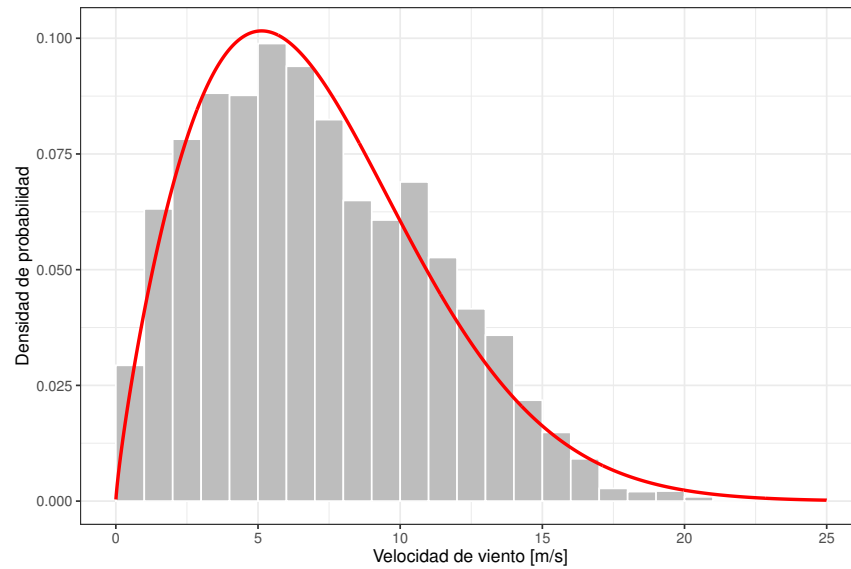


Figura 4: Histograma normalizado y la FDP Weibull de ajuste.

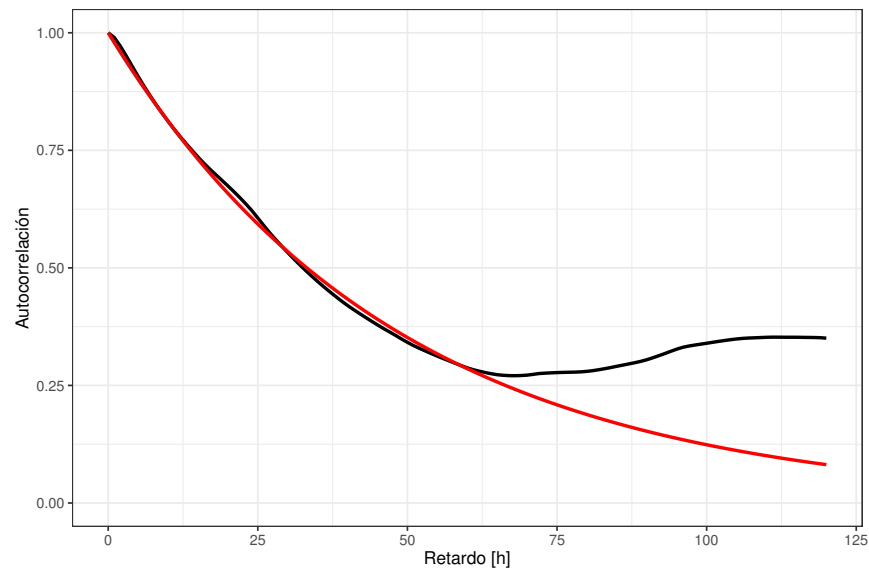


Figura 5: Autocorrelación de los datos de velocidad del viento por hora y ajuste exponencial.

El proceso de ajuste presentado en esta sección usando R fue realizado mediante el código 2.

```
Código 2. FittingProcesss.R
library(rriskDistributions)
fit.dist <- fit.cont(WS)
```

```

lambda_1 <- fit.dist$fittedParams[1]
lambda_2 <- fit.dist$fittedParams[2]

xweibull = seq(25/8760, 25, 25/8760)

yweibull <- dweibull(xweibull, shape = lambda_1, scale = lambda_2)

WEIBULL <- data.frame(xweibull, yweibull)

graphic.weibull <- data.frame(dataset2018$Speed_100m_m.s, WEIBULL)

graphic.weibull %>% ggplot(aes(x=dataset2018.Speed_100m_m.s)) +
  geom_histogram(aes(y=..density..),
    binwidth = 1, fill='gray', color = 'white', alpha=1, boundary = 0) +
  geom_line(aes(xweibull,yweibull), color = 'red', size = 1) +
  xlab("Velocidad de viento [m/s]") +
  ylab("Densidad de probabilidad") +
  theme_bw()

z <- acf.WS$acf[1:68,1,1]
x <- acf.WS$lag[1:68,1,1]
y <- log(z)
fit.exponetial <- lm(y~x-1)

alpha <- -fit.exponetial$coefficients

tau <- seq(0, 120)

FIT.FAC <- exp(-alpha*tau)

graphic.FIT.FAC <- data.frame(FAC, FIT.FAC)

graphic.FIT.FAC %>% ggplot() +
  geom_line(aes(acf.WS.lag, acf.WS.acf), size = 1) +
  geom_line(aes(acf.WS.lag, FIT.FAC), color = 'red', size = 1) +
  theme_bw() +
  xlab("Retardo [h]") +
  ylab("Autocorrelación") +
  xlim(c(0,120)) +
  ylim(c(0,1))

```

4.3. Simulación y resultados

Para ilustrar las propiedades estadísticas del modelo presentado se simularon 10000 trayectorias para un período de tiempo de 8760 h. Cabe resaltar que este proceso de simulación fue realizado en una Laptop AMD Ryzen 7 4800H 2.90 GHz y 16.0 GB RAM, con un tiempo de computo de 261.5 segundos. La figura 6 compara la FDP de la distribución de Weibull utilizada para construir el modelo (rojo) y el histograma normalizado de trayectorias simuladas (gris). En la figura 7 se muestra la comparación entre la función exponencial ajustada a la autocorrelación del conjunto de datos de la velocidad del viento (rojo) y la autocorrelación promedio de

las trayectorias simuladas por el modelo (gris). Las propiedades estadísticas de la simulaciones con los ajustes teóricos del modelo no resultan ser significativas.

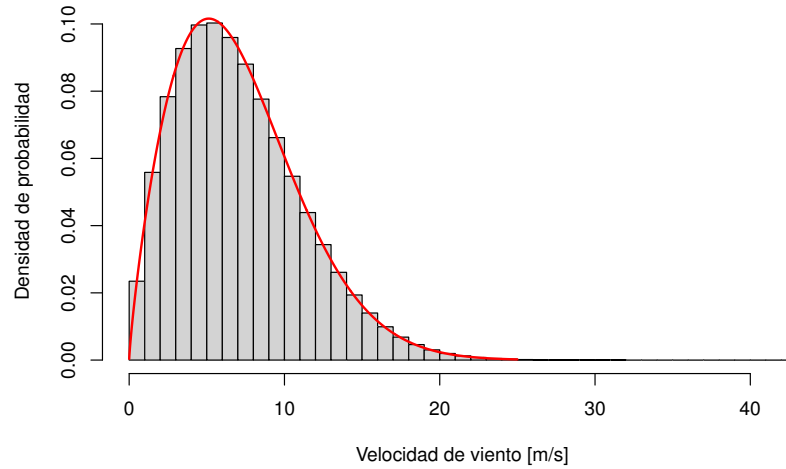


Figura 6: Densidad de probabilidad de las simulaciones de velocidad del viento y la FDP de ajuste.

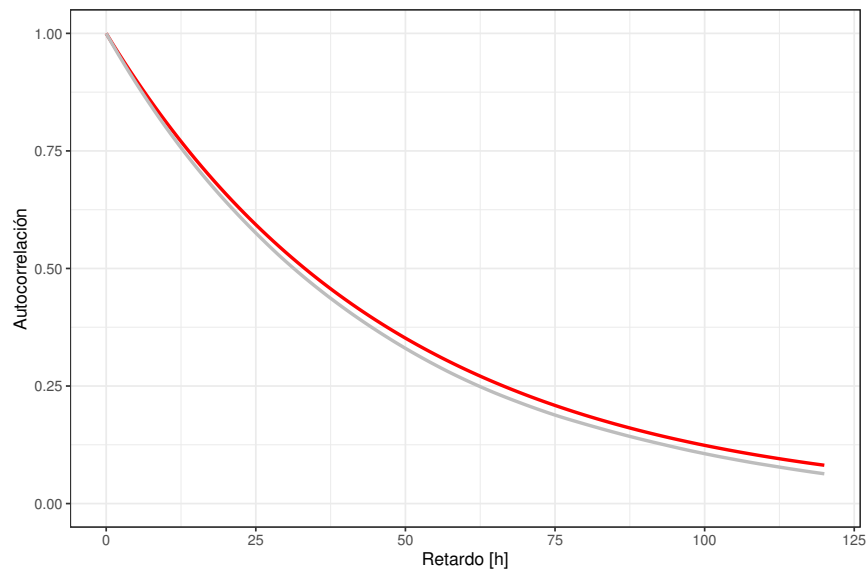


Figura 7: Autocorrelación promedio de las simulaciones de velocidad del viento y la función de ajuste.

El proceso de simulación para mostrar el despeno del modelo presentado en esta sección usando R fue realizado mediante el código 3.

Código 3. *Simulation10000.R*

```
t = 8760
tray = 10000
delta_t = 1
```



```

theta = sqrt(2*alpha)
n = t/delta_t

X=matrix(ncol = tray , nrow = n)
Y=matrix(ncol = tray , nrow = n)

system.time(
for(i in 1:tray) {
X_0 <- rnorm(1)
dW <- sqrt(delta_t)*rnorm(n)
for(j in 1:n) {
X[j,i] <- X_0 - alpha * X_0 * delta_t + theta * dW[j]
X_0 <- X[j,i]
Y[j,i] <- lambda_2 * ( - log(1 - pnorm( X[j,i] )))^(1/lambda_1)
}
}
)

hist(Y, freq=FALSE, breaks = 40, main = " ", xlab="Velocidad de viento [m/s]",
ylab="Densidad de probabilidad" )
lines(WEIBULL$xweibull, WEIBULL$yweibull, col = "red", lwd = 2)

for(i in 1:tray) {
Aux_acf <- acf(Y[,i], lag.max = 120, plot = FALSE)
if(i==1){
m_fac <- data.frame(Aux_acf$acf)
} else {
m_fac[,i] <- data.frame(Aux_acf$acf)
}
}

acf_prom <- rowMeans(m_fac[,1:121])

graphic.fac.prom <- data.frame(acf.WS$lag, FIT.FAC, acf_prom)

graphic.fac.prom %>% ggplot() +
geom_line(aes(acf.WS.lag, FIT.FAC), color = 'red', size = 1) +
geom_line(aes(acf.WS.lag, acf_prom), color = 'gray', size = 1) +
theme_bw() +
xlab("Retardo [h]") +
ylab("Autocorrelación") +
xlim(c(0,120)) +
ylim(c(0,1))

```

Finalmente, en las figuras 8 y 9 se presentan las características estadísticas de una trayectoria simulada de velocidad de viento (azul), con los ajustes teóricos del modelo (rojo) y las características del conjunto de datos reales (gris), mientras que la figura 10 presenta la trayectoria simulada para un marco de tiempo de 120 h.

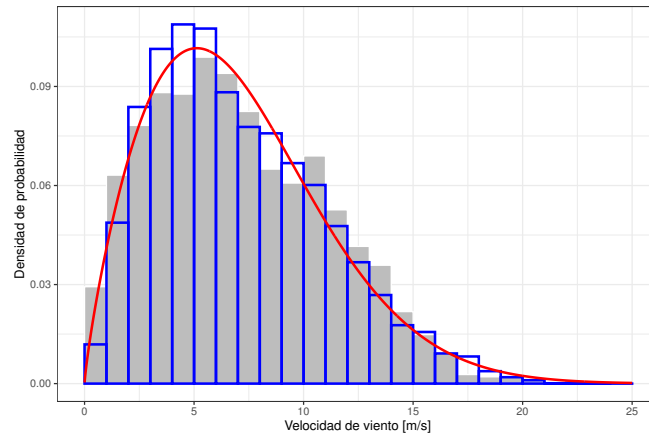


Figura 8: FDP Weibull de ajuste, histograma de conjunto de datos reales e histograma del proceso simulado.

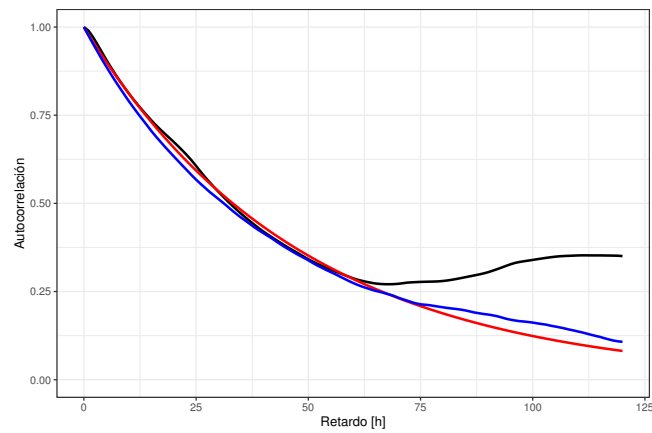


Figura 9: Autocorrelación de datos (gris), función exponencial de ajuste (rojo) y autocorrelación del proceso simulado (azul).

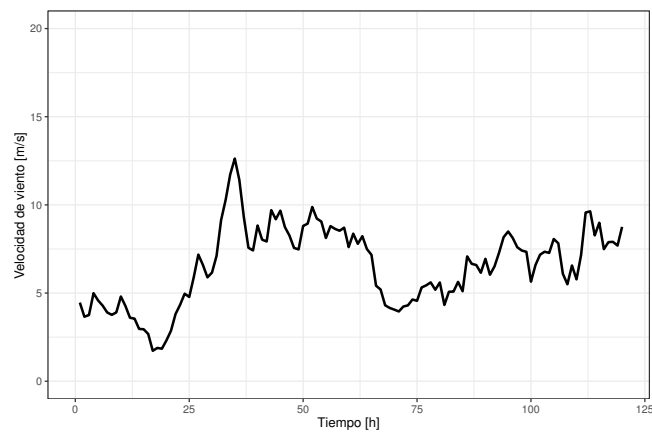


Figura 10: Trayectoria de velocidad de viento simulada.

Los resultados presentados en esta ultima sección usando R fue realizado mediante el código 4.

Código 4. *Simulation1.R*

```
set.seed(5)
t = 8760
tray = 1
delta_t = 1
theta = sqrt(2*alpha)
n = t/delta_t

X=matrix(ncol = tray , nrow = n)
Y=matrix(ncol = tray , nrow = n)

for(i in 1:tray) {
  X_0 <- rnorm(1)
  dW <- sqrt(delta_t)*rnorm(n)
  for(j in 1:n) {
    X[j,i] <- X_0 - alpha * X_0 * delta_t + theta * dW[j]
    X_0 <- X[j,i]
    Y[j,i] <- lambda_2 * ( - log(1 - pnorm( X[j,i] )))^(1/lambda_1)
  }
}

graphic.weibull.final <- data.frame(graphic.weibull, seq(1,8760), Y)

graphic.weibull.final %>% ggplot() +
  geom_histogram(aes(x=dataset2018.Speed_100m_m.s, y=..density..), binwidth = 1,
  fill='gray', color = 'white', alpha=1, boundary = 0) +
  geom_histogram(aes(x=Y, y=..density..), binwidth = 1, color = 'blue', size = 1,
  alpha=0, boundary = 0) +
  geom_line(aes(xweibull,yweibull), color = 'red', size = 1) +
  xlab("Velocidad de viento [m/s]") +
  ylab("Densidad de probabilidad") +
  theme_bw()

acf.Y <- acf(Y, lag.max = 120, plot = FALSE)

graphic.fac.final <- data.frame(graphic.FIT.FAC, acf.Y$acf)

graphic.fac.final %>% ggplot() +
  geom_line(aes(acf.WS.lag, acf.WS.acf), size = 1) +
  geom_line(aes(acf.WS.lag, FIT.FAC), color = 'red', size = 1) +
  geom_line(aes(acf.WS.lag, acf.Y.acf), color = 'blue', size = 1) +
  theme_bw() +
  xlab("Retardo [h]") +
  ylab("Autocorrelación") +
  xlim(c(0,120)) +
  ylim(c(0,1))
```

```

graphic.weibull.final %>% ggplot() +
geom_line(aes(seq.1..8760., Y), size = 1) +
theme_bw() +
xlab("Tiempo [h]") +
ylab("Velocidad de viento [m/s]") +
xlim(c(0,120)) +
ylim(c(0,20))

```

5. Conclusiones

En este trabajo se describe la construcción de un modelo para generar trayectorias de velocidad de viento capaz de reproducir las propiedades de distribución de probabilidad y autocorrelación de decaimiento exponencial observadas en una ubicación del mundo real haciendo uso del programa R. El modelo se parametriza haciendo un análisis de datos de velocidades de viento de una ubicación en particular. En este caso, el análisis de los datos para una ubicación de México, arroja que la distribución Weibull y autocorrelación de decaimiento exponencial parecen ser una buena aproximación.

Es importante hacer notar que el modelo presentado no se limita a la distribución Weibull y puede utilizarse con otras distribuciones de probabilidad.

Referencias

- [1] Zárate-Miñano R, Anghel M, Milano F. Continuous wind speed models based on stochastic differential equations. *Appl Energy*, 2013;104:42–9.
- [2] Zárate-Miñano R, Mele F, Milano F. SDE-based Wind Speed Models with Weibull Distribution and Exponential Autocorrelation. *IEEE*, 2016.
- [3] Tuller S, Brett A, The goodness of fit of the Weibull and Rayleigh distributions of observed wind speed in a topographically diverse area. *J Climatol*, 1985;5:79–84.
- [4] Brett A, Tuller S. The Autocorrelation of Hourly Wind Speed Observations. *J Appl Meteorol*, 1991;23:823–833.
- [5] Grigoriu M. *Stochastic Calculus: Applications in Science and Engineering*. Springer Science+Business Media; 2002.
- [6] Gardiner C. *Handbook of stochastic methods for Physics, Chemistry and the Natural Sciences*. 3rd ed. Springer; 2004.
- [7] Karatzas I, Steven ES. *Brownian Motion and stochastic calculus*. Springer-Verlag New York; 1998.
- [8] Grigoriu M. *Applied Non-Gaussian Processes: Examples, Theory, Simulation, Linear Random Vibration, and MATLAB Solutions*. Prentice Hall; 1995.
- [9] Kloeden E, Platen E. *Numerical Solution of Stochastic Differential Equations*. 2nd ed. Springer; 1992.
- [10] <https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-pressure-levels?tab=overview>